

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO AMAZONAS  
PRÓ-REITORIA DE PÓS-GRADUAÇÃO, PESQUISA E INOVAÇÃO – PPGI  
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO – DIPESP  
MESTRADO PROFISSIONAL EM ENSINO TECNOLÓGICO – MPET

CRISTIANO JOSÉ FERREIRA

**PROGRAMAÇÃO DE COMPUTADORES APLICADA À RESOLUÇÃO  
DE EQUAÇÕES ALGÉBRICAS E PLOTAGEM DE GRÁFICOS: UM  
ESTUDO NA LICENCIATURA EM MATEMÁTICA**

CRISTIANO JOSÉ FERREIRA

**PROGRAMAÇÃO DE COMPUTADORES APLICADA À RESOLUÇÃO  
DE EQUAÇÕES ALGÉBRICAS E PLOTAGEM DE GRÁFICOS: UM  
ESTUDO NA LICENCIATURA EM MATEMÁTICA**

Dissertação apresentada ao Mestrado Profissional em Ensino Tecnológico do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas para a obtenção do título de Mestre em Ensino Tecnológico, sob a orientação da Profa. Dra. Andréa Pereira Mendonça.

Linha de pesquisa: Recursos para o Ensino Técnico e Tecnológico

Manaus  
2018

---

F368p Ferreira, Cristiano José.

Programação de computadores aplicada à resolução de equações algébricas e plotagem de gráficos: um estudo na licenciatura em matemática. / Cristiano José Ferreira. – 2018.

148 f. : il.

Dissertação (Mestrado Profissional em Ensino Tecnológico) – Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, *Campus* Manaus Centro, 2018.

Orientadora: Profa. Dra. Andréa Pereira Mendonça.

1. Ensino tecnológico. 2. Licenciatura em Matemática. 3. Programação baseada em investigação. 4. Programação com Python. I. Mendonça, Andréa Mendonça. (Orient.) II. Instituto Federal de Educação, Ciência e Tecnologia do Amazonas. III. Título.

CDD 371.33

CRISTIANO JOSÉ FERREIRA

**PROGRAMAÇÃO DE COMPUTADORES APLICADA À RESOLUÇÃO DE  
EQUAÇÕES ALGÉBRICAS E PLOTAGEM DE GRÁFICOS: UM ESTUDO NA  
LICENCIATURA EM MATEMÁTICA**

Dissertação de mestrado, apresentada à  
Coordenação do Curso de Mestrado  
Profissional em Ensino Tecnológico do  
Instituto Federal de Educação, Ciências e  
Tecnologia do Amazonas (IFAM), Campus  
Manaus Centro, como requisito para a obtenção  
do Título de Mestre.

**BANCA EXAMINADORA**

---

Prof<sup>a</sup> Dra. Andréa Pereira Mendonça (Orientadora) - IFAM

---

Prof. Dr. Leandro Silva Galvão de Carvalho - UFAM

---

Prof<sup>a</sup> Dra. Rosa Oliveira Marins Azevedo - IFAM

Manaus  
2018

## DEDICATÓRIA

Ao meu filho, Bernardo, razão pela qual luto todos os dias. Espero que um dia eu possa compensar minhas ausências.

## AGRADECIMENTOS

Este é um momento importante e único em que posso registrar meus sinceros e singelos agradecimentos a todos que contribuíram de alguma forma para que meu objetivo fosse alcançado. Temo pela possibilidade de esquecer de citar alguém, e, se assim acontecer, antecipo meus pedidos de desculpas e reitero meus sentimentos de gratidão.

Ao Pai, que tudo sabe e me abençoou do início da minha vida até hoje. Sem Ele nada seria possível.

A minha orientadora, professora Dra. Andréa Pereira Mendonça, que, com sabedoria, dedicação, respeito, companheirismo e muito amor ao seu trabalho, fez dessa jornada uma experiência enriquecedora e inesquecível.

A minha esposa, que, ao compreender a importância deste projeto, não mediu esforços para, a sua maneira, contribuir do início ao fim.

Ao meu filho Bernardo, que, na inocência de seus quatro anos me dá forças dioturnamente e renova minhas energias com seu sorriso puro e encantador.

Aos meus pais, pelos valores e princípios que norteiam minha vida.

A todos os professores e demais servidores vinculados ao MPET, em especial à professora Dra. Rosa Marins Oliveira Azevedo, pelas valorosas contribuições nesta jornada.

À professora Dra. Elloá B. Guedes por todo apoio e disposição em contribuir com este trabalho.

A todos os meus companheiros de trabalho, em especial aos meus amigos, professora Ana Cláudia de Souza Garcia e Pedro Fabrício de Oliveira, por todo o apoio e contribuições dentro de suas áreas de formação.

A todos os meus alunos, em especial os da turma 2014 de Licenciatura em Matemática do IFAC – Campus Cruzeiro do Sul.

Ao IFAM e IFAC, pela parceria e sensibilização que facilitou o início desta jornada.

## RESUMO

Há um movimento mundial para popularizar a aprendizagem de programação entre estudantes da Educação Básica. Contudo, poucos movimentos têm sido identificados no sentido de melhorar a formação inicial de professores para utilização de programação de computadores como um recurso potencializador do ensino e da aprendizagem, tornando os futuros professores mais preparados para receber os estudantes que chegarão na escola com esse conhecimento. Dada esta problemática, apresentamos nesta dissertação uma proposta de ensino-aprendizagem, denominada PBI (Programação Baseada em Investigação) que auxilia professores de Informática que atuam em Cursos de Licenciatura em Matemática a ensinar programação aos licenciandos. Fundamentado em Investigações Matemáticas e utilizando a linguagem Python, PBI tem por objetivo empoderar os licenciandos em Matemática para utilizarem programação de computadores a fim de automatizar equações algébricas e a plotagem de gráficos. Os conhecimentos adquiridos poderão ajudá-los a explorar as potencialidades da programação para sua própria aprendizagem e também dos seus futuros alunos. Trata-se de uma pesquisa qualitativa, que se utilizou da pesquisa-ação como estratégia para seu desenvolvimento, sendo estruturada em três fases: planejamento, implementação e avaliação. Os participantes desta pesquisa foram dezoito estudantes de Licenciatura em Matemática matriculados na disciplina de Tecnologia da Informação e Comunicação Aplicada ao Ensino, do Curso de Licenciatura em Matemática do Instituto Federal do Acre (IFAC), Campus Cruzeiro do Sul. Os resultados deste estudo revelaram que a maioria dos alunos conseguiram construir programas para automatizar soluções de problemas envolvendo equações algébricas e plotagem de gráficos, no contexto de equações do primeiro e segundo grau, funções do segundo grau e elipses. Como produto desta pesquisa, disponibilizamos um Guia Didático contendo orientações e recursos para que PBI possa ser replicado por outros professores de Informática que atuam em Cursos de Licenciatura.

Palavras-chave: Licenciatura em Matemática. Investigações Matemáticas. Programação Baseada em Investigação. Programação Aplicada à Matemática. Programação com Python.

## ABSTRACT

There is a worldwide movement to popularize programming learning among students of Basic Education. However, few movements have been identified in the sense of improving initial teacher training to use computer programming as a potentiating resource for teaching and learning, making future teachers more prepared to receive students who will arrive at school with this knowledge. Given this issue, we present in this dissertation proposal for teaching and learning, named PBI (Research-Based Programming) that assists Computer teachers who work in the Graduate Courses in Mathematics, teaching the licensed programming. Based on Mathematical Investigations and using Python, PBI aims to empower the licensed in Mathematics to use computer programming in order to automate algebraic equations and plotting of graphs. The knowledge gained will help them exploit the potential of the programming for your own learning and your future students. It is a qualitative research, which used the action research as a strategy for your development, being structured in three phases: planning, implementation and evaluation. The participants of this studies were 18 Degree in Mathematics students enrolled in the discipline of Information and Communication Technology Applied to Education, the Course of Degree in Mathematics of the Federal Institute of Acre (IFAC), Campus Cruzeiro do Sul. (Acre-Brasil). The results of this study revealed that most of the students managed to build programs to automate solutions to problems involving algebraic equations and plotting of graphs, in the context of the first and second degree equations, functions of the second degree and ellipses. As a product of this research, we have a Didactic Guide containing guidelines and resources for PBI can be replicated by other Computer teachers who work in Graduate Courses.

**Keywords:** Degree in Mathematics. Mathematical Investigations. Research-Based Programming. Applied Programming to Mathematics. Programming with Python.

## LISTA DE FIGURAS

Figura 1. Gráfico de uma função quadrática .....	23
Figura 2. Representação gráfica de uma Elipse.....	23
Figura 3. Associação entre Investigação e Programação .....	34
Figura 4. Representação do ciclo da pesquisa-ação .....	37
Figura 5. Planejamento Vs Execução .....	41
Figura 6. Diagnóstico da habilidade de raciocínio e cálculos realizados à mão no contexto de funções do segundo grau .....	72
Figura 7. Resultado dos Roteiros referentes à automatização de equações algébricas Funções do segundo grau.....	74
Figura 8. Exemplo de resposta às tarefas investigativas .....	76
Figura 9. Script2R02 fornecido para os estudantes .....	77
Figura 10. Script3R02 fornecido para os estudantes .....	77
Figura 11. Scripts construídos pelo Est03 nos RTIs 01 e 02.....	79
Figura 12. Scripts construídos pelo Est03 nos RTIs 02 e 03.....	79
Figura 13. Desempenho no Roteiro Avaliativo 01 – Funções.....	81
Figura 14. Resposta do estudante Est16 a tarefa da avaliação .....	82
Figura 15. Resultado dos RTIs e Roteiro Avaliativo 01 referentes ao raciocínio e cálculo à mão – Funções do segundo grau .....	84
Figura 16. Resultado dos RTIs referentes automatização de resolução de equações algébricas – Elipses .....	85
Figura 17. Praça.....	86
Figura 18. Programa do estudante Est07 para responder aos itens i), ii) e iii) .....	86
Figura 19. Resultado do Roteiro Avaliativo 02 referente à automatização de resolução de equações algébrica – Elipses .....	88
Figura 20. Representação da praça (a) e script de apoio (b) .....	89
Figura 21. Representação de solução implementada pelo estudante Est07 em resposta à tarefa do roteiro avaliativo 2.....	89
Figura 22. Diagnóstico da construção e identificação de gráficos à mão no contexto de funções do segundo grau e elipses .....	91
Figura 23. Resultado do roteiro 03 quanto a automatização de plotagem de gráficos no contexto de funções do segundo grau.....	93

Figura 24. Trecho de código utilizado como resposta a tarefa do roteiro pelo estudante Est15 .....	94
Figura 25. Resultado da execução do programa descrito na Figura 23, plotagem do gráfico de $f(x) = 2x^2 + 3x - 1$ .....	95
Figura 26. Resultados da avaliação 1 referente à plotagem de gráficos no contexto de funções do segundo grau.....	97
Figura 27. Trecho de código utilizado como resposta a tarefa do roteiro pelo estudante Est12 .....	98
Figura 28. Resultado da execução do programa descrito na Figura 26, Est12.....	99
Figura 29. Resultado da automatização de plotagem de gráficos no contexto de elipses roteiro 6 .....	100
Figura 30. Trecho do Script1R06, dado com exemplo, que plota uma elipse.....	101
Figura 31. Resposta do estudante Est02 .....	101
Figura 32. Resposta do estudante Est12 .....	102
Figura 33. Elipse plotada utilizando o programa do estudante Est12 .....	103
Figura 34. Resultado da Roteiro Avaliativo 2 .....	104

## LISTA DE QUADROS

Quadro 1. Momentos da Investigação e Atividades Desenvolvidas .....	25
Quadro 2. Papéis do professor frente a exploração e investigação .....	26
Quadro 3. Momentos da investigação Matemática vs Momentos de PBI.....	31
Quadro 4. Estrutura dos Roteiros de Tarefas Investigativas (RTI). .....	32
Quadro 5. Fragmento de uma Tarefa Investigativa .....	32
Quadro 6. Conteúdos dos RTIs .....	43
Quadro 7. Conteúdos de Matemática .....	45
Quadro 8. Conteúdos de Programação .....	45
Quadro 9. Organização macro dos Roteiros de Tarefas Investigativas Planejados.....	47
Quadro 10. Rubrica avaliativa planejada.....	51
Quadro 11. Dinâmica utilizada na aplicação dos Roteiros.....	54
Quadro 12. Comparativo entre o planejado e o executado no RTI03 .....	58
Quadro 13. Rubrica avaliativa aplicada.....	64
Quadro 14. Técnicas e instrumentos de coletas de dados na pesquisa ação.....	67
Quadro 15. Exemplo de uma resposta correta para a atividade da Figura 7 .....	78

## SUMÁRIO

<b>INTRODUÇÃO</b> .....	12
<b>1 FUNDAMENTAÇÃO TEÓRICA</b> .....	17
1.1 Formação de Professores de Matemática e a Adoção das Tecnologias .....	17
1.2 Programação como Recurso para o Ensino e Aprendizagem.....	19
1.2.1 Python.....	21
1.3 Investigações Matemáticas .....	23
1.3.1 Momentos da Investigação Matemática .....	24
1.3.2 O papel do professor em uma Investigação Matemática.....	26
1.4 Trabalhos Relacionados.....	27
<b>2 PROGRAMAÇÃO BASEADA EM INVESTIGAÇÃO</b> .....	30
2.1 Visão Geral de PBI.....	30
2.2 Momentos da Investigação Matemática em PBI.....	31
2.3 Roteiros de Tarefas Investigativas.....	31
2.4 Dinâmica para o Ensino de Programação.....	33
2.5 Recursos de Apoio.....	35
2.6 Estudo Autônomo .....	35
<b>3 PROCEDIMENTOS METODOLÓGICOS</b> .....	36
3.1 Visão Geral da Pesquisa-Ação .....	36
3.2 Contexto de Investigação e Participantes da Pesquisa .....	38
3.3 O Desenvolvimento da Pesquisa-Ação .....	39
3.3.1 Planejamento da Pesquisa Ação .....	42
3.3.1.1 Detalhamento.....	42
3.3.1.2 Planejamento da Carga Horária.....	44
3.3.1.3 Conteúdos de Matemática .....	44
3.3.1.4 Conteúdos de Programação .....	45
3.3.1.5 Atividades de Aprendizagem.....	46
3.3.1.6 Perspectiva de ensino e aprendizagem .....	49
3.3.1.7 Recursos de Apoio.....	49
3.3.1.8 Apoio ao Estudo autônomo .....	49

3.3.1.9 Avaliação .....	50
3.3.2 Implementação das Ações .....	53
3.3.3 Avaliação das Ações.....	63
3.4 Técnicas e Instrumentos de coleta de dados .....	67
3.5 Perspectiva da Análise.....	68
<b>4 ANÁLISE DOS RESULTADOS .....</b>	<b>69</b>
4.1 Síntese dos Resultados .....	69
4.2 Detalhamento dos Resultados da Primeira Questão de Pesquisa .....	71
4.3 Detalhamento dos Resultados da Segunda Questão de Pesquisa .....	90
4.4 Detalhamento dos Resultados da Terceira Questão de Pesquisa.....	105
<b>CONSIDERAÇÕES FINAIS.....</b>	<b>107</b>
<b>REFERÊNCIAS .....</b>	<b>109</b>
APÊNDICE A – Roteiros de Tarefas Investigativas planejados .....	114
APÊNDICE B – Roteiros de Tarefas Investigativas aplicados .....	123
APÊNDICE C – Avaliação Diagnóstica .....	132
APÊNDICE D – Avaliação Planejada Roteiros 1,2 e 3 .....	135
APÊNDICE E – Avaliação Executada dos Roteiros 1, 2 e 3 .....	137
APÊNDICE F – Questionário Socioacadêmico .....	140
APÊNDICE G – Termo de Consentimento Livre e Esclarecido.....	143
APENDICE H – Questionário de Experiências e Melhorais .....	146

## INTRODUÇÃO

De forma simples, programação pode ser compreendida como a atividade de instruir a máquina quanto às ações que ela deve executar. A programação, nos últimos anos, vem sendo apontada como uma extensão da habilidade de escrita (RESNICK, 2013), pois com ela é possível “escrever” diferentes artefatos digitais, tais como, histórias animadas, jogos, simulações, entre outros, desencadeando um movimento mundial em favor da aprendizagem de programação por estudantes da Educação Básica.

A ideia de tornar a programação acessível para crianças e de utilizá-la como um recurso para que os estudantes desenvolvam o raciocínio lógico e solucionem problemas com o uso do computador tem origem com o trabalho de Seymour Papert, na década 60 e, mais recentemente, tem crescido o movimento para a popularização da programação na Educação Básica (BARCELOS; SILVEIRA, 2012; BENITTI et al., 2009; BLIKSTEIN, 2008; PAPERT, 1980; WOLFRAM, 2016).

Nesse contexto, aprender a programar não significa que o estudante terá que ser um programador. O foco não é a programação como fim, mas como *meio* para favorecer a aprendizagem, isto é, para que os estudantes sejam capazes de utilizar o computador como meio para expressar suas ideias, automatizar soluções para problemas em conteúdos relacionados à Biologia, Física ou Matemática, por exemplo, e criarem artefatos de *software* de acordo com suas necessidades.

Para incentivar e facilitar a aprendizagem de programação, especialmente por estudantes da Educação Básica, há um conjunto de plataformas virtuais, tais como “Programaê<sup>1</sup>”, “Code.org<sup>2</sup>” e “Khan Academy<sup>3</sup>” que disponibilizam cursos e materiais gratuitos. Há também um movimento global para popularização da programação, denominado *Hour of Code*<sup>4</sup> (Hora do Código), que contempla milhares de estudantes em mais de 180 países. Há também escolas destinadas a este fim, como por exemplo a *Happycode*<sup>5</sup> e outras escolas que vêm incluindo a programação em seu currículo, como o Colégio Visconde de Porto Seguro<sup>6</sup>. Há ainda Secretarias de Educação que estão implantando clubes de programação, promovendo concursos na área, engajando estudantes e corroborando com o movimento mundial de incentivo à popularização da programação. Como exemplo dessas iniciativas,

---

<sup>1</sup> <http://goo.gl/2QzrJV>

<sup>2</sup> <https://code.org/>

<sup>3</sup> <https://pt.khanacademy.org/>

<sup>4</sup> <http://programae.org.br/horadocodigo/>, <https://hourofcode.com/br>

<sup>5</sup> <http://www.happycode.com.br/escolaparceira/>

<sup>6</sup> <https://www.portoseguro.org.br/>

citamos a Secretaria Municipal de Educação de Manaus, com o projeto PROCURUMIM<sup>7</sup>, e a Secretaria Estadual de Educação de Pernambuco, com o projeto PERNAMBUCODERS<sup>8</sup>.

À medida que cresce o número de estudantes que despertam interesse pela programação, aumenta também a necessidade de empoderar os professores para a adoção e emprego da programação nas áreas de conhecimento nas quais atuam como docentes. Especialmente, essa capacitação deve ser incluída na formação inicial de professores, preparando-os para atender a uma geração de estudantes que chegarão à escola com conhecimento prévio de programação ou demandando por esses conhecimentos.

Em se tratando de formação inicial de professores, atualmente, é possível identificar na maioria dos Currículos dos Cursos de Licenciatura, pelo menos, uma disciplina de Informática<sup>9</sup>. Contudo, essa disciplina é, muitas vezes, voltada ao ensino de Informática Básica (utilização do pacote de *software* de escritório<sup>10</sup> e da Internet) ou para discussões sobre a aplicação de tecnologias no ensino, negligenciando as demandas atuais de formações de professores para uso das tecnologias, tal como, a programação de computadores.

A necessidade de melhorar a formação de professores, no que diz respeito à adoção das tecnologias no ensino, especialmente na formação inicial, é evidente na literatura. Gatti e Nunes (2009), por exemplo, fizeram uma análise de currículos e ementas referentes a 31 cursos de licenciaturas em todo o país e os resultados de sua pesquisa apontam que, apesar de apenas um currículo não possuir ao menos uma disciplina que aborda conceitos de computação, o uso da informática para a educação é referenciado claramente em apenas 29% dos cursos. Observam ainda que as ementas que fazem referência ao uso de tecnologia para o ensino a abordam mais como uma discussão sobre a utilização dessas tecnologias e possuem poucas diretrizes para utilização prática.

Ainda sobre essa questão, Parente et al. (2015) afirmam que os professores saem da graduação com muita bagagem teórica, mas não sabem como aplicá-la no ensino de seus alunos. Em sua formação, pouca ênfase é dada na instrumentalização do professor.

Indicadores relacionados às Tecnologias da Informação e Comunicação nas escolas, referente ao ano de 2014, publicados pelo Centro Regional de Estudos para o Desenvolvimento

---

<sup>7</sup> <http://semed.manaus.am.gov.br/alunos-da-rede-municipal-participam-do-primeiro-concurso-de-programacao-e-robotica/>

<sup>8</sup> <http://www.pe.gov.br/mobile/blog/2016/10/27/pernambuco-sai-na-frente-e-implanta-clubes-de-programacao-nas-escolas-da-rede-estadual/>

<sup>9</sup> Esta disciplina apresenta-se com diferentes denominações: “Informática na Educação”, “Informática Aplicada a Educação”, “Tecnologia de Informação e Comunicação Aplicada ao Ensino”, dentre outras.

<sup>10</sup> Expressão que remete ao conjunto de *software* voltado para auxiliar nas tarefas de escritório, tais como, editores de textos, de planilhas eletrônicas e de apresentação no formato de slides.

da Sociedade da Informação (Cetic.br<sup>11</sup>), apontam que, de um total de 1702 professores que possuem graduação, 61% não cursaram nenhuma disciplina que promovesse suporte à utilização de recursos computacionais e internet para o ensino. Em outro indicador, tendo como amostra professores da Rede Pública Estadual, em todo o Brasil, apontam que 65% dos docentes utilizam as tecnologias para projetar informações, por meio de slides, imagens, textos etc.

Embora a adoção de tecnologias para comunicar informações (por meio de slides, textos, páginas de internet) seja relevante, pois ajuda o professor a elaborar seus instrumentos de exposição de conteúdo, controle de notas etc., a formação do professor não deve se restringir à adoção destas tecnologias. A literatura tem evidenciado que, para a formação dos estudantes do Século XXI, é fundamental que os professores tenham condições de incluir na sala de aula tecnologias que favoreçam a resolução de problemas e a construção do conhecimento por meio do computador e, nesse sentido, destacamos a importância da programação (BLIKSTEIN 2008; GATTI, 2009; PAPERT 2008).

Especialmente no contexto do ensino-aprendizagem de Matemática, foco deste trabalho, a programação de computadores oportuniza, por exemplo, a criação de programas para automatizar equações algébricas e plotagem de gráficos, auxiliando professores e alunos na resolução de uma variedade de problemas.

O domínio de equações algébricas e plotagem de gráficos na Matemática são conhecimentos considerados de bastante relevância, haja vista o fato de eles permearem o currículo do Ensino Fundamental e Médio, sendo também priorizados por instrumentos nacionais e internacionais de avaliação, tais como Prova Brasil<sup>12</sup> e PISA (*Programme for International Student Assessment*)<sup>13</sup>. Além disso, esses conhecimentos prosseguem presentes no currículo de Cursos de Superiores de Matemática.

Assim, consideramos necessária a ressignificação de propósito das disciplinas que tratam sobre Informática no currículo dos Cursos de Licenciatura, para que estas possam incluir o ensino de programação para os licenciandos.

Porém, há nesse contexto, desafios quanto à *forma* de ensinar programação em cursos de Licenciatura. Primeiro, não é adequado que os professores de Informática usem as mesmas estratégias de ensino de programação utilizadas nos cursos de Computação, isto porque o foco não é formar programadores, mas sim pessoas que tenham autonomia para construir programas

---

<sup>11</sup> <http://goo.gl/9egWo0>

<sup>12</sup> <http://portal.inep.gov.br/web/saeb/aneb-e-anresc>

<sup>13</sup> <http://portal.inep.gov.br/pisa-programa-internacional-de-avaliacao-de-alunos>

como recursos para o ensino e aprendizagem. Segundo, há ainda que se considerar as especificidades da área de conhecimento dos Cursos de Licenciatura.

Quando um aluno se defronta com um problema, ele precisa acionar, por exemplo, sua capacidade de raciocínio lógico, de interpretação das informações e de estabelecimento de um plano de solução (POZO,1998). Esses são processos gerais, aplicáveis a todas as áreas do conhecimento. Contudo, é necessário considerar questões específicas, como, por exemplo, os tipos de problemas, a forma de expressar soluções e, ainda, processos específicos para resolução de problemas.

No domínio da Matemática, por exemplo, Ponte e Oliveira (1998) apresentam Investigações Matemáticas como uma estratégia didático-metodológica para o desenvolvimento do pensamento matemático e da capacidade dos estudantes para trabalhar de forma autônoma na resolução de tarefas investigativas. As tarefas investigativas, por sua vez, designam situações que promovem o desenvolvimento de processos matemáticos, tais como procurar regularidades, formular, testar, justificar e provar conjecturas, refletir e generalizar. As tarefas investigativas são, portanto, aquelas que motivam o desenvolvimento da investigação pelos alunos e, por conseguinte, o desenvolvimento do pensamento matemático.

Dados esses desafios e considerando a Licenciatura em Matemática como foco deste trabalho, destacamos o nosso *problema de pesquisa: como podemos ensinar programação de computadores para os licenciandos em Matemática, no contexto de disciplinas de Informática na Licenciatura?*

Tendo como foco este problema de pesquisa, nosso *objetivo* foi o de desenvolver uma proposta de ensino-aprendizagem para favorecer a aprendizagem de programação para automatização de resolução de equações algébricas e plotagem de gráficos, tendo como público alvo licenciandos de Matemática. Esta proposta baseou-se em Investigações Matemáticas e na adoção da linguagem de programação Python, sendo denominada Programação Baseada em Investigações (PBI). Segundo PBI, os licenciandos adquirem habilidades de programação de forma progressiva, por meio de exemplos, a partir de atividades de investigação, tomando como elemento motivador da aprendizagem as tarefas investigativas, guiados por Roteiros de Tarefas Investigativas (RTIs).

No escopo deste trabalho, as tarefas investigativas focaram no estudo de álgebra e plotagem de gráficos, estando limitados aos conteúdos de equações do primeiro e segundo grau, plotagem de gráficos de funções quadráticas e das elipses. Estes assuntos, por constarem na aprendizagem prévia dos estudantes desde o ensino médio, facilitam a aprendizagem de programação e sua aplicação na Matemática. Com os conhecimentos adquiridos, os

licenciandos poderão, no futuro e de forma autônoma, aplicá-los em outros conteúdos da Matemática.

Para a condução da pesquisa, adotamos, como percurso metodológico, a pesquisa ação, vislumbrando indicativo de respostas para as seguintes *questões de pesquisa*: (1) Em que aspectos PBI melhoram as habilidades dos alunos para automatização de resolução de equações algébricas?; (2) Em que aspectos PBI melhoram as habilidades dos alunos para automatização da plotagem de gráficos?; e, (3) Quais as limitações desta aplicação de PBI?

A pesquisa teve como participantes 18 estudantes do Curso de Licenciatura em Matemática do Instituto Federal do Acre - IFAC, campus Cruzeiro do Sul, do 7º período, regularmente matriculados na disciplina de Tecnologias da Informação e Comunicação Aplicadas ao Ensino, sendo PBI implementada em sala de aula no primeiro semestre de 2017.

Em termos de *produto*, esta pesquisa resultou em um Guia Didático contendo orientações e recursos de apoio (indicação de materiais didáticos, planejamento de ensino, tarefas investigativas, exemplos de códigos etc.) para incentivar o ensino de programação com adoção de PBI por outros docentes de Informática em Cursos de Licenciatura em Matemática.

A fim de apresentar o trabalho realizado, organizamos este documento em três capítulos. No Capítulo 1, descrevemos os conceitos que fundamentam a nossa pesquisa. Dedicamos o Capítulo 2 para descrever Programação Baseada em Investigação. O percurso metodológico adotado é apresentado no Capítulo 3. Em seguida, no Capítulo 4, discutimos e apresentamos os resultados da análise dos dados, seguidos das considerações finais.

## 1 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo apresentamos os aspectos teóricos em que fundamentam nossa pesquisa. Iniciamos com uma apresentação das diretrizes curriculares para o ensino de Matemática, referenciadas pelos pareceres do Conselho Nacional de Educação, Parâmetros Curriculares Nacionais e Base Nacional Curricular Comum. Posteriormente, apresentamos a programação como recurso para o ensino e aprendizagem, tendo como aporte pesquisadores como Papert (1980), Resnick (2013), Bers (2014), Blikstein (2008). Logo em seguida, discutimos sobre investigações matemáticas, pautadas principalmente estudos realizados por Ponte et al. (2016), Braumann (2002) e Serrazina (2002).

### 1.1 Formação de Professores de Matemática e a Adoção das Tecnologias

A Resolução nº 2, de 1º de julho de 2015 (BRASIL, 2015), do Conselho Nacional de Educação, institui as Diretrizes Curriculares Nacionais para a formação inicial em nível superior e para formação continuada de professores. Segundo essa Resolução, as instituições que ofertam cursos de formação inicial e continuada para licenciaturas devem garantir ao egresso habilidades para o uso competente das tecnologias, destinadas ao aprimoramento da prática pedagógica, possibilitando o enriquecimento da mediação da aprendizagem.

De forma complementar, o Parecer CNE/CES nº 1.302, aprovado em 6 de novembro de 2001 (BRASIL, 2002b) que trata sobre as Diretrizes Curriculares Nacionais para os Cursos de Matemática, Bacharelado e Licenciatura, já orientava que o currículo das Licenciaturas deve oportunizar ao licenciando a familiaridade com o uso do computador como instrumento de trabalho, incentivando-se sua utilização para o ensino de Matemática, em especial para a formulação e solução de problemas.

Em consonância com estas legislações, destacamos o que evidencia os Parâmetros Curriculares Nacionais - PCNs (BRASIL, 2002) sobre a necessidade de o professor dominar a tecnologia como um recurso que favoreça a aprendizagem do estudante. Demanda que também está presente na Base Nacional Curricular Comum<sup>14</sup> – BNCC (BRASIL, 2018), quando ressalta a importância da utilização das tecnologias, inclusive fazendo referência ao desenvolvimento do pensamento computacional. A BNCC (BRASIL, 2018, p. 518) propõe que:

[...]os estudantes utilizem tecnologias, como calculadoras e planilhas eletrônicas, desde os anos iniciais do Ensino Fundamental. Tal valorização possibilita que, ao chegarem aos anos finais, eles possam ser estimulados a desenvolver o pensamento computacional, por meio da interpretação e da elaboração de fluxogramas e algoritmos.

---

<sup>14</sup> Documento ainda em discussão até o fechamento deste trabalho.

Em conjunto, essas orientações reforçam a necessidade premente de capacitar os professores para adoção das Tecnologias, em especial, aquelas que impulsionam a resolução de problemas com o uso do computador.

Em contrapartida a essa necessidade, há um conjunto de autores que apontam que os currículos das Licenciaturas ainda dão pouca ênfase a instrumentalização do professor, sobretudo quando se trata da adoção das TICs (GATTI; NUNES, 2009; PARENTE et al, 2015).

Com respeito aos conteúdos no Ensino de Matemática, tanto os PCNs quanto a BNCC ressaltam que o estudo de equações permeia toda a educação básica e transcende para além da Matemática, tornando essa habilidade essencial para o estudante, que se depara constantemente com situações que exigem o domínio de tal habilidade para resolvê-las, como na trigonometria, geometria analítica e o estudo das funções, por exemplo. No domínio da Química, Física e Biologia, as equações algébricas estão presentes e enaltecem a interdisciplinaridade dessas áreas e a necessidade de o estudante ser capaz de lidar com a diversidade de conteúdos e problemas durante sua aprendizagem (BRASIL, 1999; BRASIL, 2002; BRASIL, 2018).

Quanto aos gráficos, os PCNs (BRASIL, 2002) e a BNCC (BRASIL, 2018) destacam sua utilização na trigonometria, estudo de funções e na geometria analítica, por exemplo. O estudante deve ser capaz de converter as representações algébricas e gráficas, sendo essa habilidade vital para a análise e interpretação das variáveis matemáticas envolvidas. A habilidade de construir gráficos é exigida também em outras áreas do conhecimento, como a Física e a Química, sendo seu domínio requerido para a síntese de conclusões obtidas no desenvolvimento das atividades discentes.

Mesmo diante dessas recomendações dos documentos oficiais, as resoluções de equações algébricas são muitas vezes massificadas na forma de exercícios, para que os alunos resolvam à mão, muitas vezes focando no procedimento do cálculo e não na compreensão de seu significado ou na resolução de problemas mais significativos para os alunos. Do mesmo modo, os gráficos, normalmente, são desenhados à mão, muitas vezes, de forma imprecisa e dispendiosa em termos de tempo.

Conforme destaca Blikstein (2008), o mundo atual exige mais da formação dos estudantes e isto inclui saber usar o computador como instrumento de aumento do poder cognitivo e operacional humano. Destaca ainda que não se trata de usar apenas lápis e papel, mas de saber utilizar o computador para resolver problemas de forma mais rápida e eficiente.

Todas estas considerações contribuem para a perspectiva da programação de computadores como um recurso que possibilita ao licenciando explorar o computador para automatizar procedimentos rotineiros e repetitivos, além de possibilitar maior precisão e

agilidade na construção de gráficos, potencializando sua própria aprendizagem em Matemática e, futuramente, como docente, o ensino de Matemática, conforme preconizado pelas regulamentações de ensino. As potencialidades da programação serão discutidas na seção seguinte.

## 1.2 Programação como Recurso para o Ensino e Aprendizagem

A programação tem se mostrado de interesse para além das áreas do conhecimento, que necessitam dessa habilidade na formação do profissional (cursos de Engenharia, Mecânica, Computação, Eletrônica, entre outros). A aprendizagem de programação, fora do escopo da profissionalização, já era defendida por Seymour Papert<sup>15</sup> e continua sendo discutida por outros pesquisadores atualmente (BLIKSTEIN, 2008; RESNICK, 2013).

O matemático Seymour Papert, pesquisador do Instituto de Tecnologia de Massachusetts e um dos pioneiros da Inteligência Artificial, na década de 60, ao perceber que a computação estava mudando a forma das pessoas trabalharem, projetou sua utilização para além dos fins científicos e militares da época. Papert, inspirado pelas ideias de Jean Piaget, vislumbrou o uso do computador pelas crianças e o seu uso para favorecer a aprendizagem e construção de conhecimentos. Assim, criou a linguagem de programação *Logo* e, posteriormente, a Robótica Educacional, sendo pioneiro no uso de programação pelas crianças (PAPERT, 1980).

Por acreditar que a tecnologia deve ser utilizada para além da comunicação de conteúdo, fundamentado nas ideias do Construtivismo de Jean Piaget, Papert cunhou o termo *Construcionismo*. O *Construcionismo* diz respeito a uma filosofia de educação na qual o sujeito é ativo na construção do conhecimento e utiliza o computador como um instrumento para a aprendizagem. A aprendizagem se torna mais significativa quando a criança, através do computador, constrói seus artefatos, levando-as a refletir acerca do processo de construção, a depuração de erros e testes de possíveis soluções. A aprendizagem se torna desafiadora e prazerosa (PAPERT, 2008).

A aprendizagem de programação pelas crianças tem ganhado força e notoriedade, sendo que diversos pesquisadores defendem a inserção da programação na Educação Básica. Mitchel Resnick, professor do Instituto de Tecnologia de Massachusetts (MIT), criador da linguagem de programação *Scratch*<sup>16</sup>, concebido para pessoas entre 8 e 16 anos de idade, aponta que, apesar das tecnologias estarem presentes no cotidiano de todos, ao jogar vídeo *game*, trocar

---

<sup>15</sup> <http://www.papert.org/>

<sup>16</sup> <https://scratch.mit.edu>

mensagens pela internet utilizando o *smartphone*, por exemplo, os jovens mostram apenas familiaridade com a tecnologia.

Contudo, é necessário que os jovens possam ir além e que sejam capazes de expressar suas ideias e exercitar suas habilidades de criação. Considerando um mundo permeado por tecnologias, essas habilidades são possíveis com conhecimento de programação, pois, por meio da programação, os jovens podem criar diferentes artefatos digitais e ainda compartilhá-los com o mundo (RESNICK, 2013).

Na mesma vertente que Mitchel Resnick, Marina Umashi Bers lidera o projeto do *Scratch Junior*<sup>17</sup>, idealizado para crianças entre 5 e 7 anos, que ainda estão na etapa de alfabetização. Por meio do *Scratch Junior*, as crianças conseguem expressar suas ideias (na forma de jogos, estórias animadas, entre outros), aguçando sua criatividade de forma interativa e divertida, ao mesmo tempo em que desenvolvem habilidades importantes, concomitante à alfabetização (BERS, 2014).

A programação possibilita que os estudantes desenvolvam estratégias para a resolução de problemas em diferentes perspectivas, contribui com o aprimoramento do raciocínio lógico e capacidade de abstração, permitindo-os visualizar, implementar, testar e depurar diversas possibilidades para a resolução de problemas. Tais características compõem o que Wing (2006) denomina de *pensamento computacional*.

Jeannette Wing define *pensamento computacional* como os processos de pensamento necessários na formulação, interpretação e solução de problemas, utilizando recursos computacionais para resolvê-los<sup>18</sup>. A autora afirma que essa é uma habilidade crucial e deverá ser utilizada por todos no mundo no Século XXI (WING, 2006).

Blikstein (2008), por sua vez, afirma que pensamento computacional é saber utilizar o computador como uma ferramenta para potencializar o poder cognitivo e operacional humano, utilizando os recursos computacionais em prol da produtividade, criatividade e inventividade. O autor divide o pensar computacionalmente em duas etapas: identificar as tarefas cognitivas que podem ser executadas pelo computador e programar o computador para realizar essas tarefas. Tanto Blikstein (2008) como Wing (2006) concordam que o pensamento computacional é uma das habilidades mais importantes do Século XXI, necessária para todos os indivíduos e pode ser desenvolvida através da programação.

Ao programar, além de desenvolver essas habilidades, o estudante terá oportunidade de projetar suas próprias soluções para os diferentes problemas propostos no âmbito escolar,

---

<sup>17</sup> <https://www.scratchjr.org/>

<sup>18</sup> Tradução nossa.

identificando e corrigindo erros, trabalhando em equipe, buscando alternativas criativas para a solução dos problemas apresentados. Assim, o estudante não estará aprendendo a programar, mas programando para aprender (RESNICK, 2013). Sua inserção na Educação Básica e seus benefícios no aprendizado de Matemática também são defendidos em trabalhos como de Wolfram (2016), Nunes (2011), Benitti et al. (2009) e Barcelos; Silveira (2012).

Atualmente, há um conjunto de iniciativas para promover a aprendizagem de programação por crianças e jovens em todo o mundo. Algumas dessas iniciativas são vislumbradas por meio de plataformas gratuitas para que os estudantes aprendam mais facilmente programação. Dentre elas, citamos Programaê!<sup>19</sup>, Microsoft Virtual Academy<sup>20</sup>, EdX<sup>21</sup>, CodeAcademy<sup>22</sup>, Coursera<sup>23</sup> e a Pycursos<sup>24</sup>, que buscam aproximar jovens da programação utilizando temáticas atraentes para crianças e adolescentes, como o Python para Zumbis<sup>25</sup>, por exemplo.

Essas plataformas disponibilizam cursos e materiais que permitem a aprendizagem autônoma de programação, tornando-a acessível às pessoas em todo o mundo. Utilizando essas plataformas, os estudantes desenvolvem a lógica de programação e também aprendem uma ou mais linguagens de programação. Assim, por meio da linguagem de programação, construímos os comandos que o computador deve executar. Há inúmeras linguagens de programação disponíveis, cada qual com suas especificidades e aplicabilidades, *C*, *Perl*, *Java*, *PHP*, *C#*, *Ruby*, *Python* são alguns exemplos.

Em nosso trabalho de pesquisa, adotaremos a linguagem de programação *Python*, cujas características são descritas a seguir.

### 1.2.1 Python

*Python* é uma linguagem de alto nível, lançada por Guido van Rossum, em 1991, possuindo mais de um milhão de usuários em diversos países. A linguagem de programação *Python* combina fatores como a facilidade de aprendizado, uma extensa biblioteca e funcionalidades pré-construídas, rapidez de desenvolvimento, flexibilidade, portabilidade e é baseada em *software* livre, não havendo custo para sua utilização. Trabalhos realizados pela

---

<sup>19</sup> Informações completas através do endereço <http://goo.gl/2QzrJV>

<sup>20</sup> <https://mva.microsoft.com/>

<sup>21</sup> [https://www.edx.org/course/introduction-computer-science-harvardx-cs50x#.VNYLwvnF\\_0c](https://www.edx.org/course/introduction-computer-science-harvardx-cs50x#.VNYLwvnF_0c)

<sup>22</sup> <https://www.codecademy.com/>

<sup>23</sup> <https://www.codeavengers.com/>

<sup>24</sup> <https://www.pycursos.com>

<sup>25</sup> <https://www.pycursos.com/python-para-zumbis/>

*Python Foundation*<sup>26</sup> e por inúmeros colaboradores têm proporcionado o avanço, diversificação e crescimento da linguagem (COELHO, 2007; LUTZ, 2013; MENEZES, 2010).

Diante desse cenário, há várias adesões à aprendizagem de programação com *Python*. Instituições de ensino do mundo todo vêm utilizando a linguagem *Python* no ensino de lógica de programação, bem como sua aplicabilidade nas áreas de exatas. No cenário internacional, o *Massachusetts Institute of Technology*<sup>27</sup> (MIT), *Stanford University*<sup>28</sup>, *Oxford University*<sup>29</sup>, *University of London*<sup>30</sup> são exemplos de instituições que adotam a linguagem de programação Python e oferecem cursos em diversas modalidades. No Brasil, a Universidade de São Paulo (USP), Universidade Federal do Espírito Santo e a Universidade Federal de Caxias do Sul são exemplos de instituições de ensino que adotam a linguagem de programação Python no ensino e aprendizagem de programação, segundo a comunidade Python-Brasil<sup>31</sup>, em conjunto com a Universidade Federal de Campina Grande (UFCG) e a Universidade Federal de Pernambuco<sup>32</sup>, entre outras instituições no país.

A linguagem de programação *Python* possui um conjunto de bibliotecas, pacotes e recursos que facilitam a realização de cálculos matemáticos, manipulações algébricas e plotagem de gráficos. No contexto deste trabalho, utilizaremos o pacote *Numpy*<sup>33</sup>, que permite trabalhar com arranjos, vetores e matrizes de  $N$  dimensões, transformando *Python* em uma ferramenta de programação numérica que combina eficiência e facilidade de uso. Com todos esses recursos, *Python* facilita a construção de artefatos para automatização de soluções na área da Matemática, Engenharia e Biologia, por exemplo (COELHO, 2007; IDRIS, 2015; LUTZ, 2013; SAHA, 2015).

A linguagem *Python* possui ainda bibliotecas específicas para a plotagem de variados tipos de gráficos. A *Matplotlib*<sup>34</sup>, que também será adotada em nosso trabalho, possibilita a plotagem de gráficos, como barras, colunas, histogramas, entre outros, e que podem ser exportados como figuras em vários formatos. As Figuras 1 e 2 ilustram alguns tipos de gráficos gerados com o *Matplotlib*.

---

<sup>26</sup> <https://www.python.org/psf/>

<sup>27</sup> <https://goo.gl/alwsnU>

<sup>28</sup> <http://stanford.edu/~schmit/cme193/>

<sup>29</sup> <http://www.lsidtc.ox.ac.uk/core-modules/index.html>

<sup>30</sup> <http://www.city.ac.uk/courses/short-courses/introduction-to-programming-with-python>

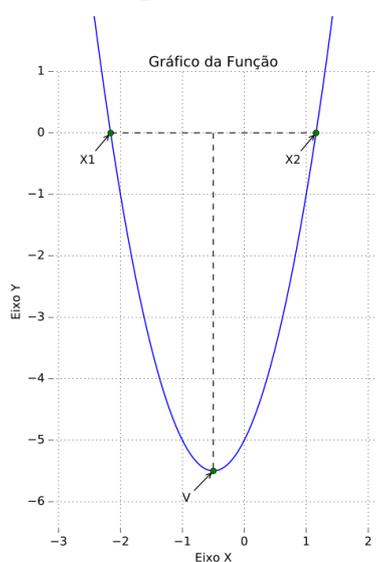
<sup>31</sup> [http://wiki.python.org.br/PythonNoBrasil#Universidade\\_de\\_S.2BAOM-o\\_Paulo](http://wiki.python.org.br/PythonNoBrasil#Universidade_de_S.2BAOM-o_Paulo)

<sup>32</sup> <https://goo.gl/HPz0tV>

<sup>33</sup> <http://www.numpy.org/>

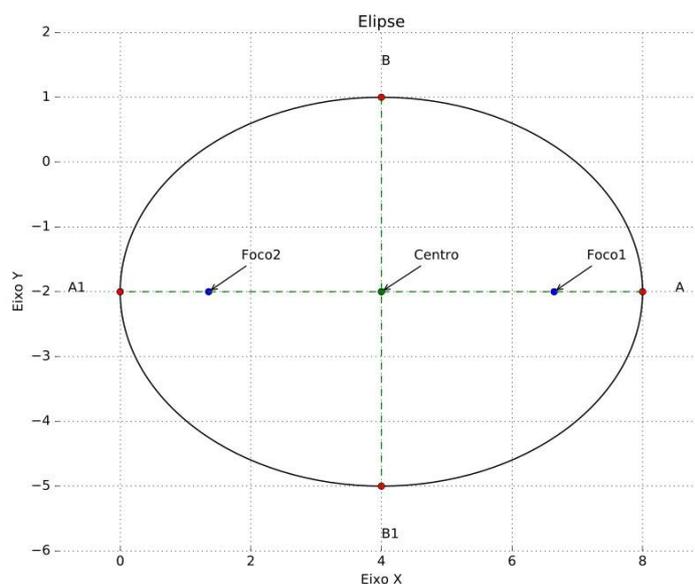
<sup>34</sup> <http://matplotlib.org/>

Figura 1. Gráfico de uma função quadrática



Fonte: Elaboração própria

Figura 2. Representação gráfica de uma Elipse



Fonte: Elaboração própria

Dado o potencial da programação para o ensino e aprendizagem, destacaremos na próxima seção considerações sobre sua aprendizagem baseada na estratégia didático-metodológica denominada Investigações Matemáticas.

### 1.3 Investigações Matemáticas

Investigar é a busca por conhecer algo que ainda não se sabe. O dicionário Houaiss<sup>35</sup> define o termo como a busca metódica e consciente por descobrir algo e ainda sugere o vocábulo pesquisar como acepção para a palavra investigar. Ponte (2003) atribui ao termo investigar a procura de respostas fundamentadas e rigorosas para as nossas próprias questões. Para os matemáticos, a procura por padrões, a descoberta de relações e propriedades entre objetos matemáticos conhecidos ou não dão sentido ao termo investigar (PONTE et al., 2016).

Investigações Matemáticas vêm sendo amplamente discutidas por pesquisadores, como Ponte et al. (1998) e Serrazina et al. (2002), e podem ser definida como uma estratégia didático-metodológica para o desenvolvimento do pensamento matemático e da capacidade dos estudantes para trabalhar de forma autônoma na resolução de tarefas investigativas. As tarefas investigativas, por sua vez, designam situações que promovem o desenvolvimento de processos matemáticos, tais como procurar regularidades, formular, testar, justificar e provar conjecturas, refletir e generalizar.

<sup>35</sup> <http://www.iah.com.br/sp/index.php>

Os autores corroboram com a afirmação de que essa estratégia pode favorecer a aprendizagem e é um veemente processo de construção do conhecimento. Braumann (2002, p. 5) afirma:

Aprender Matemática não é simplesmente compreender a Matemática já feita, mas ser capaz de fazer investigação de natureza matemática (ao nível adequado a cada grau de ensino). Só assim se pode verdadeiramente perceber o que é a Matemática e a sua utilidade na compreensão do mundo e na intervenção sobre o mundo. Só assim se pode realmente dominar os conhecimentos adquiridos. Só assim se pode ser inundado pela paixão 'detectivesca' indispensável à verdadeira fruição da Matemática. Aprender Matemática sem forte intervenção da sua faceta investigativa é como tentar aprender a andar de bicicleta vendo os outros andar e recebendo informação sobre como o conseguem. Isso não chega. Para verdadeiramente aprender é preciso montar a bicicleta e andar, fazendo erros e aprendendo com eles.

Nesse sentido, a BNCC salienta a importância de o estudante ser protagonista na construção do seu conhecimento (BRASIL, 2018). O processo de ensino e aprendizagem é inviabilizado com a ausência da prática, os professores não estão ensinando e os estudantes não estão aprendendo, ocasionando o desestímulo no estudante, levando ao desinteresse. Dessa forma, os professores devem trabalhar com situações desafiadoras, que estimulam o raciocínio e a dedução Matemática, por exemplo. Por isso, é necessário um olhar cuidadoso nos primeiros anos da educação básica e principalmente na formação dos professores (BRAUMANN, 2002, p. 21).

Assim, Ponte et al. (1998) destacam que a aprendizagem Matemática deve oportunizar aos estudantes o envolvimento com atividades genuinamente matemáticas, possibilitando a experimentação, de forma com que o foco esteja na criação do saber e não somente no seu fim.

Quanto ao currículo, pesquisas apontam que Investigações Matemáticas estão presentes e vêm sendo enfatizadas em diversos países, tais como Estados Unidos, Inglaterra, França, Portugal e Brasil (PONTE et al., 2016). No Brasil, a BNCC destaca como processos e práticas de investigação o envolvimento dos estudantes já nos anos iniciais da educação básica, com atividades que os levem a comparar, classificar, questionar, buscar informações, registrar dados e comunicar os resultados. Tais atividades devem proporcionar aos estudantes a melhor apropriação e compreensão dos conhecimentos envolvidos (BRASIL, 2018). Na seção seguinte, abordaremos os momentos da investigação Matemática.

### 1.3.1 Momentos da Investigação Matemática

Na investigação matemática, é defendido que o estudante aprecie a jornada e não a chegada, o que engaja os estudantes na aprendizagem, fazendo-os descobrir as respostas por meio de suas próprias atividades.

As tarefas investigativas são os elementos motivadores desse processo de investigação. O conceito de tarefa investigativa pretende trazer para a sala de aula o espírito da atividade matemática genuína, constituindo, por isso, uma poderosa metáfora educativa. O aluno é chamado a agir como um matemático, não só na formulação de questões e conjecturas e na realização de provas e refutações, mas também na apresentação dos seus resultados e na sua discussão e argumentação com os colegas e o professor (PONTE, 2003).

Assim, durante o processo de investigação o conhecimento é construído e desenvolvido em quatro momentos, tido como principais, que podem ocorrer separadamente ou simultaneamente. No primeiro momento, ocorre o reconhecimento, exploração inicial da situação e a formulação de questões. As conjecturas são formuladas em seguida. Os testes e o esmero das conjecturas são realizados no terceiro momento, seguidos da argumentação, demonstração e avaliação do trabalho realizado, no quarto e último momento (PONTE et al., 2016). O Quadro 1 expõe algumas atividades possíveis nos quatro momentos.

Quadro 1. Momentos da Investigação e Atividades Desenvolvidas

1- Exploração e formulação de questões	Reconhecer uma situação problemática; Explorar a situação problemática; Formular Questões;
2- Conjecturas	Organizar dados; Formular conjecturas (e fazer afirmações sobre uma conjectura);
3- Testes e reformulação	Realizar testes; Refinar uma conjectura;
4- Justificação e Avaliação	Justificar uma conjectura; Avaliar o raciocínio ou o resultado do raciocínio

Fonte: Adaptado de Ponte et al. (2016).

Ainda, conforme o autor, a interação entre os matemáticos pode ocorrer em todos os momentos, mas deve ser obrigatória no momento final, objetivando a divulgação, confirmação e validação dos resultados. É neste momento que os estudantes expõem e discutem os resultados, frente a uma atividade de investigação.

Um professor, ao se fundamentar em investigações matemáticas, altera a sua forma de dar aulas e de propor suas tarefas que, nesse contexto, são tarefas investigativas. Assume então o papel de mediador, estimulando a autonomia dos estudantes diante das atividades. Em um ambiente educacional no qual a investigação está sendo introduzida, é prudente que o professor esclareça a dinâmica de aprendizagem e o papel ao qual ele se propõe a desempenhar, ajudando o estudante a compreender o que significa investigar e como fazê-la, estando disponível para

dirimir dúvidas e intervir sempre que necessário (PONTE et al., 2016). Na seção seguinte, apresentamos os papéis que o professor deve assumir, frente a exploração e investigação.

### 1.3.2 O papel do professor em uma Investigação Matemática

Em atividades de investigação, Ponte et al. (1998a) ressaltam que o professor exerce papel importante para os estudantes. No início da atividade, o professor deve envolver os estudantes no trabalho, propondo-lhes a execução de uma tarefa. Compete ao professor, averiguar o envolvimento dos estudantes com a investigação proposta, certificando que estão executando as tarefas propostas, formulando questões, propondo, testando conjecturas e justificando-as. A manutenção do diálogo entre o professor e os estudantes é tão importante quanto o estímulo que deve ser dado, pelo professor, para que os estudantes se comuniquem, à medida em que avançam na investigação, tornando o ambiente de sala de aula mais propício para o aprendizado. Na fase final de realização das atividades, o professor deve se atentar às conclusões chegadas pelos estudantes, bem como as justificativas para elas (PONTE et al., 1998a).

Ao propor atividades de exploração e investigação, o professor, segundo pesquisas realizadas por Ponte et al. (1998a), deve assumir um conjunto de seis papéis que visam envolver os alunos nas tarefas. À luz do autor, citamos cada um deles no do Quadro 2.

Quadro 2. Papéis do professor frente a exploração e investigação

Desafiar os estudantes	Formular questões que desafiam os estudantes e despertam neles o interesse em resolvê-las.
Avaliar o progresso dos estudantes	Verificar se os estudantes entenderam a tarefa proposta, se estão formulando questões, conjecturas e testando-as.
Raciocinar matematicamente	Proporcionar aos estudantes a experiência de observar o professor investigar, formulando questões, testando conjecturas.
Apoiar o trabalho dos estudantes	Fazendo perguntas adequadas, o professor tende a estimular o raciocínio.
Fornecer e recordar informação	Proporcionar informação útil aos estudantes, ajudando-os a relembrar os compreender assuntos matemáticos.
Promover a reflexão dos estudantes	É importante que os estudantes consigam relacionar o trabalho que estão desenvolvendo com seus conhecimentos já adquiridos.

Fonte: Adaptado de Ponte et al. (1998a, p.11-21)

Os meios que o professor pode adotar para atingir seus objetivos, quanto à aprendizagem dos estudantes, são sintetizados em três papéis fundamentais que perpassam a lógica do desenvolvimento de qualquer atividade, segundo Ponte et al. (1998a): desafiar, apoiar e avaliar. Assim, segundo o autor, o professor envolve os estudantes em trabalho investigativo, desafiando-os com situações motivadoras, ao mesmo tempo em que os apoia, por meio de perguntas, comentários ou sugestões. Por último, avalia os resultados, buscando identificar e coletar informações sobre os progressos e dificuldades de seus estudantes, para que então possa tomar a decisão de prosseguir ou alterar aspectos relacionados a sua prática, na continuidade do trabalho.

#### 1.4 Trabalhos Relacionados

Nesta seção, apresentamos alguns trabalhos que tratam do ensino de programação de computadores para licenciandos em Matemática.

Neto (2015) desenvolveu um trabalho com estudantes do 2º ano do curso de Licenciatura em Matemática da Universidade do Estado do Pará, no qual utilizava programação em blocos, com utilização do Scratch, a fim de promover a construção de instrumentos matemáticos e didáticos, com tecnologia digital incorporada, juntamente com uma estratégia baseada na criação de situações didáticas. Como artefatos, os estudantes desenvolveram o que Neto (2015) denominou de Calculadora de Fração, Calculadora Falante e Calculadora com interface robótica, promovendo a associação de programação de computadores e Matemática. Em suas considerações, Neto (2015) defende que a programação de computadores pode empoderar professores de Matemática em formação inicial, fazendo com que os mesmos passem de expectadores para criadores de artefatos tecnológicos, por meio da programação. Apesar de algumas limitações apresentadas pelo autor, o mesmo conclui que os estudantes conseguiram realizar as implementações solicitadas, de maneira engajada, demonstrando o senso de propriedade, criatividade e fluência na tecnologia utilizada.

O trabalho de Afini et.al (2013) apresenta uma intervenção pedagógica para licenciandos em Matemática, que aborda conteúdos de geometria plana focados no ensino de pavimentação do plano. Um dos objetivos do trabalho, segundo o autor, era o de incentivar a utilização de recursos computacionais no processo de ensino e aprendizagem de geometria, além de prover aos licenciandos em Matemática, o domínio de ferramentas computacionais que possam favorecer a planejamento de suas aulas e o ensino/aprendizagem de conceitos matemáticos.

Nesse trabalho, o autor utilizou a linguagem LOGO e em sua intervenção pedagógica, utilizou o *Software SuperLogo*. O autor relata que foram aplicadas atividades investigativas sobre a temática pavimentação periódica e aperiódica, que envolve conceitos de polígonos e ângulos. Seguindo uma estrutura de mosaicos, os estudantes utilizaram a programação de computadores para construir pavimentações, aplicando conceitos de geometria plana.

Nossa pesquisa se aproxima a de Neto (2015), à medida que ambas possuem como focos potencializar licenciandos em Matemática para o uso de programação de computadores, na perspectiva da construção do conhecimento. Quanto ao trabalho de Afini et.al (2013), corroboramos a afirmação do autor de que é necessário preparar licenciandos em Matemática para o uso de recursos computacionais, principalmente àqueles em que os estudantes desenvolvem seus próprios artefatos para resolver problemas matemáticos.

A utilização de programação de computadores, na formação inicial docente, mostrou-se também na dissertação de Mestrado de Curci (2017). O autor, por meio da linguagem de programação em blocos e utilizando o *Software Scratch*, propõe a criação de objetos de aprendizagem por licenciandos em Matemática para o estudo de assuntos relacionados à geometria. A fluência tecnológica é apontada por Curci (2017) como uma necessidade à formação docente. Um dos resultados de sua pesquisa aponta que os estudantes possuem familiaridade com o uso de computadores e dispositivos móveis, bem como a operacionalização de aplicativos. Porém, esse mesmo cenário não é percebido, quanto à habilidade de programação de computadores, já que os estudantes demonstram maiores dificuldades.

Quando o processo de ensino e aprendizagem considera o estudante como protagonista de sua aprendizagem, utilizando recursos computacionais que possibilitam a construção do conhecimento, tanto Curci (2017), Neto (2015) e Afini et.al (2013) afirmam que a aprendizagem se torna mais dinâmica, pois, além de possibilitar inovações no ensino, a construção de artefatos proporciona aprendizado com os erros, por meio de exploração e testes nos artefatos desenvolvidos pelos estudantes.

Eloy et.al (2017) propõem, em sua pesquisa, uma revisão sistemática da literatura que aborda a utilização da programação com *Scratch* em trabalhos publicados entre os anos de 2012 e 2016. Para tanto, o autor focou suas buscas em veículos de publicação que possuem mais de cinco trabalhos relacionados ao tema. No total, foram identificados pelo autor 104 artigos, distribuídos entre Workshop de Informática na Escola (WIE), 49 artigos; Congresso Brasileiro de Informática na Educação (CBIE), 34 artigos; Simpósio Brasileiro de Informática na Educação (SBIE), 29 artigos; Revista Novas Tecnologias na Educação (RENTE), 15 artigos;

Revista Brasileira de Informática na Educação (RBIE), 8 artigos; e Jornada de Atualização em Informática na Educação (JAIE), 6 artigos.

Após aplicados os critérios de exclusão, Eloy et.al (2017) utilizaram uma amostra de 53 artigos. Um dos resultados obtidos no trabalho do autor, e que cabe destaque para a nossa análise, mostra que os objetivos das pesquisas selecionadas estão focados no ensino e aprendizagem de programação, computação ou pensamento computacional, o que se aproxima de nossa pesquisa. Quanto ao público alvo identificado pelo autor, refere-se, na sua maioria, 65%, a estudantes da educação básica, 22% a estudantes de nível superior, e um pouco mais de 5% refere-se a professores, sendo o restante, um pouco mais de 6%, classificado como público variado. Apesar de ser notória a preocupação em desenvolver habilidades de programação de computadores nos estudantes da Educação Básica, a pesquisa de Eloy et.al (2017) evidencia poucos trabalhos apontando para a formação de professores, preocupação destacada em nossa pesquisa.

A associação entre programação e Matemática, focadas tanto no ensino quanto na aprendizagem, vem sendo apresentada em outros trabalhos (SANTOS, 2015; SOUZA; YONEZAWA, 2016; PINTO, 2010) e incentivada por diferentes comunidades internacionais de professores, tais como, MathAndCoding<sup>36</sup>, Bootstrap<sup>37</sup> e Computer based Math<sup>38</sup>.

Por fim, cabe destacar dois aspectos diferenciais do nosso trabalho em relação aos anteriormente mencionados: o primeiro diz respeito ao fato do nosso trabalho promover a aprendizagem de programação baseando-se em investigações matemáticas. Até o momento, pela pesquisa realizada, não detectamos trabalhos com esta característica; o segundo relaciona-se ao produto derivado de nossa pesquisa, isto é, do Guia Didático que fornece orientações para outros professores de Informática sobre como aplicar PBI (Programação Baseada em Investigação). Este é um aspecto bastante relevante, porque, muitas vezes, há experiências bem sucedidas de ensino cujo material produzido não permite orientar a condução por terceiros. No próximo Capítulo, discutiremos aspectos relacionados à PBI.

---

<sup>36</sup> <http://www.mathandcoding.org/>

<sup>37</sup> <http://www.bootstrapworld.org/>

<sup>38</sup> <https://www.computerbasedmath.org/>

## 2 PROGRAMAÇÃO BASEADA EM INVESTIGAÇÃO

Neste Capítulo, apresentamos uma visão geral de PBI, seguida pelos Momentos da Investigação Matemática em PBI. Na sequência, discorremos sobre os Roteiro de Tarefas Investigativas e a dinâmica para o ensino de programação.

### 2.1 Visão Geral de PBI

Para que estudantes de licenciaturas aprendam programação de computadores, entendemos que a dinâmica do ensino não deve ser a mesma utilizada em cursos de computação. O intuito não é formar programadores, mas empoderar futuros professores de licenciatura, para a adoção de programação de computadores em sua prática de ensino e na construção de artefatos que automatizem a resolução de problemas matemáticos.

Neste sentido, Programação Baseada em Investigação – PBI é uma proposta de ensino-aprendizagem de programação de computadores voltada para estudantes de licenciatura em Matemática. PBI associa conteúdos de programação de computadores e da Matemática, pautados em investigações matemáticas. Segundo PBI, de forma incremental, os licenciandos adquirem habilidades de programação por meio de investigação, guiados por Roteiros de Tarefas Investigativas – RTIs.

No que diz respeito à programação de computadores, para a aplicação de PBI, faz-se necessária a adoção de uma linguagem de programação, que, no contexto desta pesquisa, optamos pela linguagem *Python*, porém é possível a utilização de outras linguagens de programação disponíveis. Quanto aos conteúdos de Matemática, esta pesquisa foi instanciada para a automatização de resolução de equações algébricas e plotagem de gráficos no contexto de equações de primeiro e segundo grau e das elipses, mas PBI pode ser focada em outros conteúdos matemáticos.

Com aporte em Investigações Matemáticas, abordadas na Seção 1.3, valemo-nos dos momentos da Investigação Matemática para subsidiar o planejamento e a construção das tarefas investigativas focadas no ensino-aprendizagem de programação de computadores. Na seção seguinte, discutiremos aspectos relacionados à associação dos momentos da Investigação Matemática com PBI, focados no ensino de programação de computadores.

## 2.2 Momentos da Investigação Matemática em PBI

Ao retomarmos o Quadro 1, nos deparamos com os momentos da investigação Matemática, nos quais, por meio de tarefas investigativas, os estudantes: (i) Exploram e formulam questões; (ii) Conjecturam; (iii) realizam testes e reformulações; (iv) Justificam e avaliam. Em PBI, as tarefas investigativas, guiadas por roteiros, provocam os momentos da investigação Matemática, que, neste contexto, chamaremos de momentos de PBI. O Quadro 3 apresenta um paralelo destes momentos da investigação Matemática em PBI.

Quadro 3. Momentos da investigação Matemática vs Momentos de PBI.

Instigação Matemática	PBI
1- Exploração e formulação de questões	1- Resolução das Tarefas propostas nos Roteiros de Tarefas Investigativas; Explorar os recursos de apoio (Códigos, vídeos e tutorias) e com base em exemplos iniciar as tarefas de programação
2- Conjecturas	2 - Realização das Tarefas propostas nos RTIs, questionando-se a partir da exploração dos códigos dados como exemplos;
3- Testes e reformulação	3 - Realizar testes nos códigos fornecidos e incrementalmente construir novos códigos a partir dos exemplos dados.
4- Justificação e Avaliação	4 - Responder o porquê das respostas às tarefas do roteiro; Socializar as respostas e os caminhos percorridos para encontrá-las.

Fonte: Elaboração Própria.

Para sistematizar as tarefas investigativas e nortear o ensino/aprendizagem dos estudantes, nos pautamos em roteiros, que no contexto de PBI são denominados Roteiros de Tarefas Investigativas – RTIs. Na próxima seção, discutiremos aspectos relacionados aos RTIs.

## 2.3 Roteiros de Tarefas Investigativas

Denominam-se Roteiros de Tarefas Investigativa o instrumento norteador para a aplicação de PBI e conseqüentemente para a aprendizagem do estudante. Por meio das tarefas investigativas, organizadas em roteiros, o estudante é conduzido ao processo de descoberta e aprendizagem, sendo sujeito autônomo e ativo na construção do seu conhecimento.

Para responder as tarefas investigativas propostas nos roteiros, os estudantes vivenciam os momentos de PBI elucidados no Quadro 3, e ora possuem à disposição um conjunto de recursos de apoio, tais como vídeos, tutoriais e trechos de código que auxiliam na aprendizagem autônoma e na retirada de dúvidas referentes aos conteúdos abordados nos roteiros; e ora possuem o apoio do professor. Apresentamos, no Quadro 4, a estrutura geral de um RTI e dos elementos que o compõem.

Quadro 4. Estrutura dos Roteiros de Tarefas Investigativas (RTI).

<b>Título</b>	Apresenta o tema de estudo do roteiro.
<b>Descrição e Informações Gerais</b>	Apresenta orientações de estudo e conduta para a resolução do roteiro.
<b>Resultados Pretendidos da Aprendizagem.</b>	Descrevem os objetivos da aprendizagem que devem ser alcançados pelos estudantes.
<b>Conteúdos Abordados</b>	Descrevem os conteúdos de Matemática e de programação que devem ser estudados para o desenvolvimento das tarefas investigativas.
<b>Recursos de Apoio</b>	Apresenta links para recursos de estudo (vídeos, textos, códigos), a fim de auxiliar o estudo autônomo.
<b>Tarefas Investigativas</b>	Apresentação de tarefas que devem ser resolvidas, adotando-se o processo de investigação e fazendo uso dos recursos de apoio.
<b>Referência dos Recursos</b>	Descrição de fontes bibliográficas para estudo e indicação das fontes dos demais recursos utilizados no roteiro.

Fonte: Elaboração própria

Especificamente, no que diz respeito às Tarefas Investigativas, os roteiros apresentam enunciados de problemas e, para levar os estudantes à investigação, é proposto um conjunto de questionamentos e orientações, de forma com que possam analisar as perguntas, utilizar os recursos de apoio, buscando respostas para irem construindo e testando suas soluções. Como exemplo, apresentamos, no Quadro 5, um fragmento de uma tarefa investigativa.

Quadro 5. Fragmento de uma Tarefa Investigativa

1.0)	Vamos começar com o seguinte problema.
	<p>Considere um número, cujo seu quadrado aumentado de seu dobro é igual a 15. Qual é esse número?</p>
	<p>a) Analisando o problema proposto, extraia do mesmo a equação que o representa, descrevendo-a em papel.</p> <p>b) Resolva a equação que você montou e responda o que se pede no problema.</p>

Caso tenha alguma dúvida quanto aos passos para resolução de equações do 2º grau, assista ao **vídeo 02** e/ou leia o **tutorial 06**. Agora, utilizando o Wing IDE, abra e execute o script1R02. Leia todas as linhas de código e seus respectivos comentários, isto o ajudará no entendimento do funcionamento do script.

1.1) Após a análise do script, responda as seguintes perguntas:

- a) Os resultados gerados pelo script são os mesmos que você obteve a partir da resolução manual?
- b) Você consegue perceber a fórmula de bhaskara presente no script? Se sim, em quais linhas de código?
- c) Faça as alterações necessárias no script1R02 para que o usuário possa informar os valores das variáveis da equação, considerando a entrada como uma float.
- d) Salve o script na sua pasta de respostas com um nome significativo e que identifique a questão.

Fonte: Elaboração própria

Como pode ser observado no Quadro 5, inicialmente, o estudante se depara com situações de Matemática e de programação, sendo desafiado a resolvê-las à mão e, posteriormente, utilizando programação. Tanto os assuntos relacionados à Matemática quanto à programação iniciam com um grau de dificuldade pequeno, que é ampliado paulatinamente, à medida que o estudante avança pelos roteiros

Segundo PBI, a dinâmica para o ensino de programação de computadores é diferente das que comumente são utilizadas nos cursos de Computação. A dinâmica da aprendizagem será abordada na seção seguinte.

#### 2.4 Dinâmica para o Ensino de Programação

Para aprender programação, segundo PBI, o estudante tem à disposição um conjunto de recursos de apoio, tais como vídeos, tutorias e scripts. Conduzidos pelas tarefas investigativas, os estudantes exploram os recursos de apoio e alteram as linhas de códigos presentes nos scripts para construir, de maneira incremental, seus próprios artefatos, conforme demonstrado na Figura 3.

Figura 3. Associação entre Investigação e Programação.



Fonte: Elaboração própria

Como representado na Figura 3, o estudante se depara com uma tarefa investigativa que o conduz à exploração de um determinado script. Nessa tarefa, o estudante faz alterações no script e o executa com o intuito de visualizar os resultados obtidos. Numa segunda tarefa, é solicitado que o estudante construa um artefato que resolva uma tarefa similar à anterior, e, para a resolução desta, ele poderá utilizar um dos scripts fornecidos, fazendo alterações e incrementando o mesmo. Na terceira tarefa, o estudante recebe um script com inúmeras lacunas nas linhas de código e cabe a ele completá-las e corrigi-las para a que o artefato resolva o que fora solicitado na tarefa.

Por meio das tarefas investigativas presentes nos roteiros, o estudante é questionado sobre como ele resolveu e o porquê de ter resolvido daquela forma. Além disso, as tarefas solicitam que o estudante realize testes em seus artefatos e apresente, de forma escrita, os resultados obtidos e o porquê deles. Dessa forma, de maneira incremental, o estudante constrói, baseado em exemplos, artefatos que solucionem as tarefas dos roteiros e que possibilitam o desenvolvimento da habilidade de programação.

Em PBI, da mesma forma como o abordado na Seção 1.3.2, o professor assume papéis importantes na condução das aulas, desafiando os estudantes por meio de tarefas investigativas, apoiando o estudante, ora fornecendo recursos que facilitem o estudo autônomo (vídeos, tutoriais e *scripts*), ora dirimindo dúvidas, individual ou coletiva, mantendo sempre uma postura investigativa. Ao ser questionado pelos estudantes, o professor deve estimular o raciocínio deles, que poderá ser feito por meio de questionamentos que apontem possíveis caminhos para a solução da atividade requerida por eles, fazendo que os mesmos descubram e construam suas próprias respostas às tarefas investigativas. É importante que o professor “pense em voz alta”, demonstrando possibilidades de organização das ideias e do raciocínio empregado na busca por respostas.

Após a conclusão de um ou mais roteiros, é importante que o professor avalie coletivamente as respostas dos estudantes, proporcionando momentos de interação e compartilhamento dos caminhos percorridos pelos alunos na busca de suas respostas. Na seção seguinte, abordamos aspectos relacionados aos recursos de apoio.

## 2.5 Recursos de Apoio

Em PBI, deve ser disponibilizado um conjunto de artefatos de apoio, como recursos digitais, textos no formato de tutoriais e códigos/*scripts*. Os recursos digitais se referem a um conjunto de vídeos que buscam dirimir dúvidas a respeito dos conteúdos estudados. Com o mesmo objetivo, os textos em formato de tutoriais abordam conceitos e exemplos para auxiliar na aprendizagem e ou revisão de conteúdos de Matemática e programação.

Quanto aos códigos<sup>39</sup>/*scripts*, estes foram elaborados para possibilitar as atividades de exploração e investigação. Assim, baseado em um código dado pelo professor, os alunos tinham o mínimo necessário de linhas de programação para auxiliá-los na resolução das atividades e construção de seus próprios códigos. Portanto, códigos e trechos de códigos que combinam conteúdos de matemática e de programação estavam à disposição do estudante durante todo o processo de ensino.

## 2.6 Estudo Autônomo

Segundo PBI, os estudantes são sujeitos ativos na construção do seu conhecimento e, guiados por roteiros, desenvolvem habilidades de programação de maneira autônoma. Além das horas dedicadas em sala de aula, é necessário que os estudantes dediquem horas para o estudo extraclasse, aprimorando os conhecimentos adquiridos no decorrer da resolução das tarefas investigativas propostas nos roteiros.

No Capítulo seguinte, apresentamos os procedimentos metodológicos adotados em nosso estudo, bem como o detalhamento de nossa aplicação de PBI.

---

<sup>39</sup> No contexto deste trabalho, código e programa são utilizados como sinônimos.

### 3 PROCEDIMENTOS METODOLÓGICOS

O estudo que realizamos é de caráter qualitativo e se pautou na pesquisa-ação como estratégia metodológica, considerando as definições de David Tripp (2005). Nesse sentido, estruturamos o capítulo de tal modo a apresentar, inicialmente, uma visão geral sobre pesquisa-ação, seguida do desenvolvimento de suas fases.

#### 3.1 Visão Geral da Pesquisa-Ação

A adoção de pesquisa-ação, neste trabalho, deveu-se ao fato dela corroborar nossas intenções de pesquisa, uma vez que buscamos promover ações que envolviam a melhoria de um contexto educacional, partindo de problemas reais, em um ambiente no qual exercemos a nossa prática. Destarte, na pesquisa-ação, é possível promover alterações na prática, a partir de um diagnóstico ou de ações que são realizadas em conjunto com os participantes da pesquisa, já que seus *feedbacks* retroalimentam o planejamento para uma nova ação. Ainda, a pesquisa-ação contribui para reflexões sobre a prática e podem proporcionar mudanças e melhorias no contexto pessoal e social (ESTEBAN, 2010).

Segundo Thiollent (2011, p. 20):

[...] a pesquisa-ação é um tipo de pesquisa social com base empírica que é concebida e realizada em estreita associação com uma ação ou com a resolução de um problema coletivo e no qual os pesquisadores e os participantes representativos da situação ou do problema estão envolvidos de modo cooperativo ou participativo.

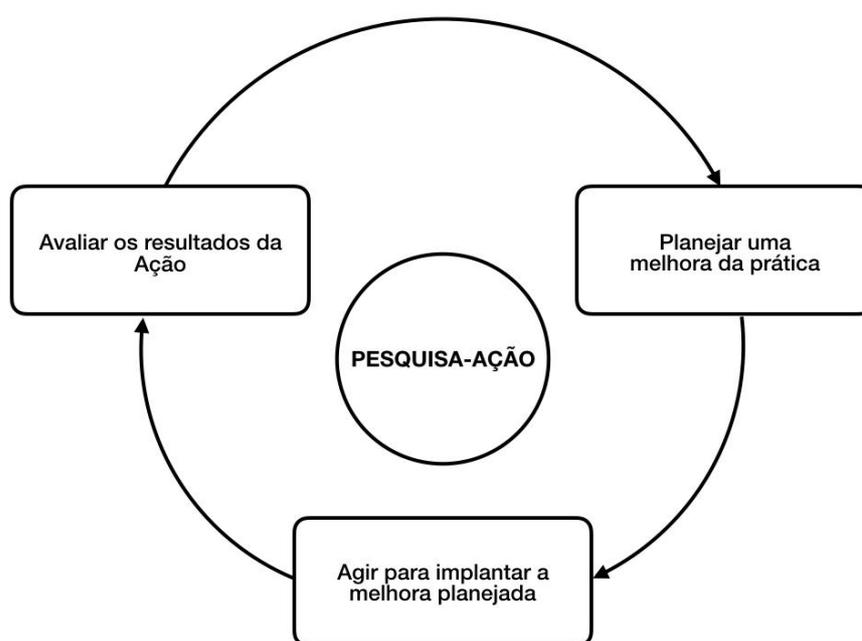
Tripp (2005, p. 447) define a pesquisa-ação como sendo “[...] uma forma de investigação-ação que utiliza técnicas de pesquisa consagradas para informar a ação que se decide tomar para melhorar a prática[...]”. Na pesquisa-ação, a pesquisa exerce a função de diagnosticar uma situação para, então, iniciar uma ação, acompanhando-a e observando-a, de forma com que lhe sejam conferidos sentidos, para, assim, avaliá-la e desencadear novas ações.

A pesquisa-ação, naturalmente, é participativa e o aprimoramento da prática, em pesquisa-ação, ocorre a partir do cumprimento, de forma cíclica, de etapas como planejamento, implementação, monitoramento e avaliação das ações realizadas (vide Figura 4), proporcionando aprendizado contínuo sobre a prática e a própria investigação. Cabe ressaltar que, no contexto desta pesquisa, a etapa de monitoramento foi realizada no decorrer da implementação, adaptando o ciclo da pesquisa ação definido por Tripp (2005). No que diz respeito à etapa de planejamento, esta preocupa-se em sistematizar as ações que deverão ser realizadas para solucionar problemas identificados durante a prática. Na etapa de

implementação, colocamos em prática as ações planejadas. Quanto à etapa da avaliação, busca-se avaliar os resultados produzidos no momento da ação, o que poderá conduzir ao replanejamento, reiniciando o ciclo.

Dessa forma, o objetivo da pesquisa-ação é proporcionar a mudança, considerando a prática e o saber como fontes dela. Com este olhar, a mudança não é advinda da aplicação de uma teoria, mas sim como produto de renovados ciclos de ação e reflexão (ANDALOUSSI, 2004).

Figura 4. Representação do ciclo da pesquisa-ação



Fonte: Adaptado de Tripp (2005)

Como dito, no âmbito deste trabalho, valemo-nos das definições e dos ciclos definidos por Tripp (2005), ilustrado na Figura 3, para o desenvolvimento da pesquisa. A pesquisa-ação inicia com um reconhecimento, feito por meio de uma análise situacional, que visa proporcionar uma leitura macro do contexto em que ela está inserida, incluindo as práticas utilizadas e os participantes envolvidos. Assim, ao projetar as mudanças que deverão ocorrer para melhora da prática, o reconhecimento segue as etapas previstas no ciclo da pesquisa-ação, planejando a ação e a avaliação, entendendo os resultados da avaliação, para que, então, seja possível replanejar adequadamente uma mudança da prática (TRIPP, 2005).

No seu desenvolvimento, os ciclos da investigação são executados de forma contínua, porém, não necessariamente ocorrem na mesma sequência, ou são repetidos de maneira corrente. Durante o desenvolvimento da pesquisa-ação, o ciclo deve ser permeado pela reflexão,

possibilitando uma melhor identificação do que se deve melhorar. Portanto Dessa forma, a reflexão tende a potencializar a eficácia no planejamento e implementação, terminando o ciclo com uma reflexão sobre os resultados (TRIPP, 2005; ESTEBAN, 2010).

### 3.2 Contexto de Investigação e Participantes da Pesquisa

Nossa pesquisa foi realizada no Instituto Federal de Educação, Ciência e Tecnologia do Acre (IFAC) – Campus Cruzeiro do Sul, no período de março a junho do ano de 2017. O Instituto Federal do Acre integra a Rede Federal de Educação Profissional, Científica e Tecnológica, composta por 38 institutos no país, contando com mais de 400 unidades organizadas, oferecendo ensino gratuito, do ensino médio à pós-graduação.

O IFAC foi implantado recentemente, em julho de 2010, com oferta de cursos com ênfase em Recursos Naturais e Ambiente, Saúde, Segurança do Trabalho e Licenciaturas. Atualmente, o IFAC conta com cinco campus situados nas diferentes regiões do Estado do Acre. Os campi de Rio Branco e Xapuri encontram-se localizados na região do Alto Acre; de Sena Madureira, na região do Alto Purus; de Tarauacá na região de Tarauacá/Envira; e o de Cruzeiro do Sul, na região do Vale do Juruá, extremo ocidental do Brasil, região que apresenta uma diversidade de riquezas naturais e, até pouco tempo, o acesso era somente por via fluvial e aérea.

Ainda no ano de 2010, o campus Cruzeiro do Sul iniciou as atividades de ensino, a princípio com dois cursos técnicos na modalidade subsequente, sendo um em Agropecuária e outro em Meio Ambiente. O curso superior de Licenciatura em Matemática foi implantado no primeiro semestre de 2011, juntamente com o curso superior de Tecnologia em Agroecologia. Atualmente o campus possui quatro cursos superiores, sendo duas licenciaturas (Matemática e Física) e dois cursos de tecnologia (Agroecologia e Processos Escolares). A unidade também oferta dois cursos técnicos integrados ao ensino médio (Agropecuária e Meio Ambiente) e três cursos na modalidade subsequente (Aqüicultura, Recursos Pesqueiros e Zootecnia).

A pesquisa teve como participantes o pesquisador, que também é professor da disciplina de TICs, e os estudantes. Quanto ao pesquisador, o mesmo possui formação inicial na área de computação e atua como docente desde o ano de 2007. No que diz respeito aos demais participantes, são 18 (dezoito) estudantes do Curso de Licenciatura em Matemática, do 7º período, que estavam regularmente matriculados na disciplina Tecnologia da Informação e Comunicação Aplicada ao Ensino - TICs. No contexto desta pesquisa, utilizamos a carga horária total da disciplina de TICs para a aplicação de PBI. Para a realização da pesquisa, tivemos à disposição uma sala de aula e dois laboratórios de informática do campus Cruzeiro do Sul. A unidade dispõe de dois laboratórios, sendo um com trinta e cinco computadores e

outro com trinta e dois, todos com acesso à internet. Ambos os laboratórios foram utilizados na aplicação da pesquisa, nos quais foram providenciados a instalação dos recursos de *software* necessários, *Python* (3.6.0), editor *WingIdle101* (6.0.3-1) e *Anaconda* (4.3.1), que instala automaticamente bibliotecas e pacotes como a *Matplotlib*, *Numpy* e *Sympy*.

Para caracterizar os participantes da pesquisa, fizemos uso dos dados coletados por meio da aplicação de um questionário socioacadêmico. Dos 18 (dezoitos) participantes de nossa pesquisa, 61,1% eram do sexo masculino e 38,9% do sexo feminino. No que diz respeito à faixa etária, 66,7% dos licenciandos tinham entre 19 (dezenove) e 25 (vinte e cinco) anos, 11,1%, de 26 (vinte e seis) a 32 (trinta e dois) anos e os outros 22,2% possuíam mais de 32 (trinta e dois) anos. A maioria dos estudantes, 61,1%, afirmou possuir microcomputador em casa, mas apenas 44,4% possuíam acesso à internet por meio de provedores de banda larga ou de conexão via rádio. Quanto à programação de computadores, 38,9% dos estudantes afirmaram que tinham tido algum contato com programação, por meio de cursos *on-line* e/ou oficinas. Quando questionados sobre o estudo de elipses, no contexto da Geometria Analítica, 61,1% dos estudantes afirmaram desconhecer o assunto, 5,6% dos estudantes afirmaram ter estudado apenas no ensino médio e outros 27,8% dos estudantes, no ensino médio e na graduação.

Para preservar a identidade dos participantes, os mesmos, quando citados, serão identificados por um codinome, como Est01.

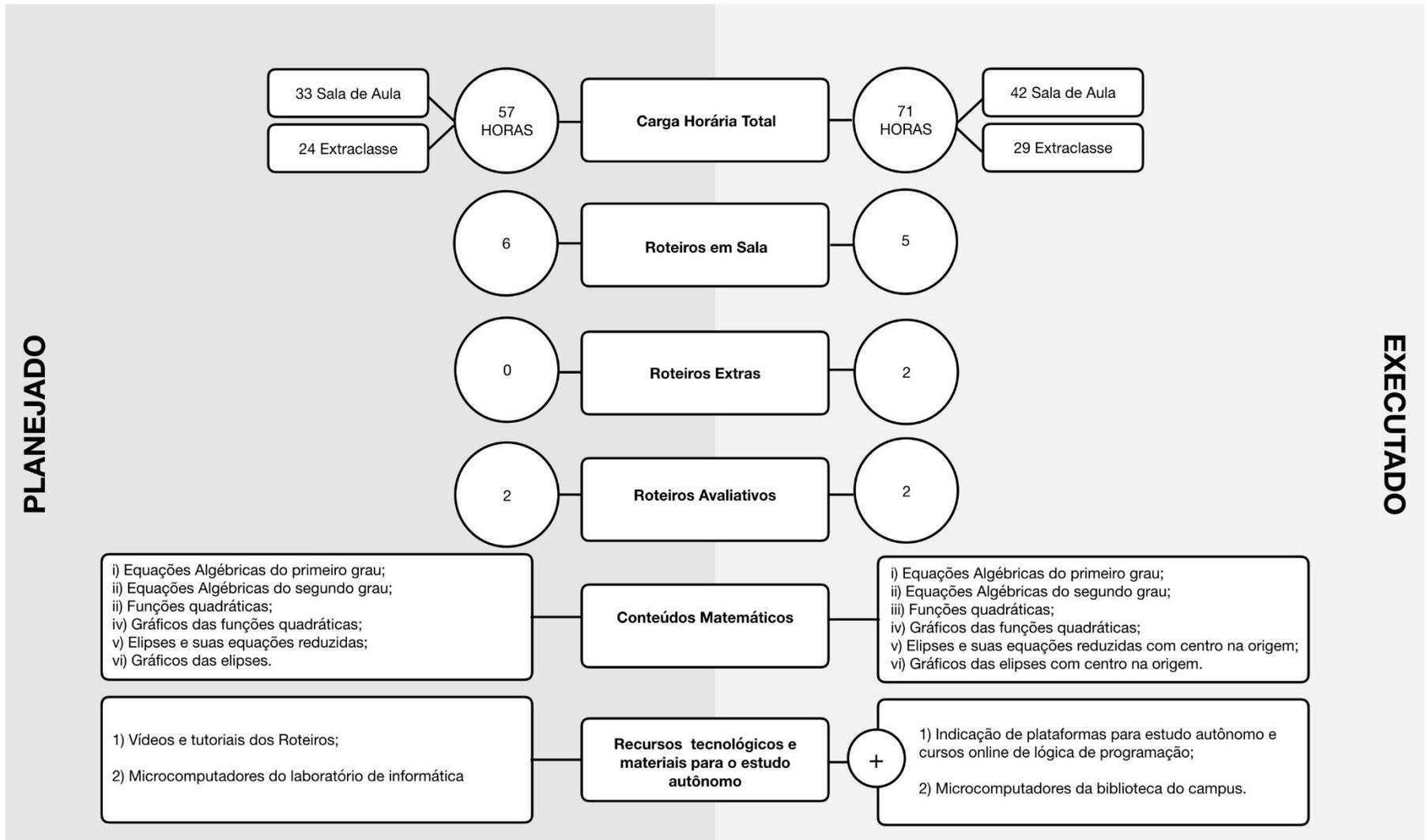
### 3.3 O Desenvolvimento da Pesquisa-Ação

Antes de descrevermos o que ocorreu em cada etapa da pesquisa-ação, cabe esclarecer alguns aspectos. Embora comumente na pesquisa-ação o planejamento seja realizado de maneira coletiva com os participantes da pesquisa, optamos neste trabalho por elaborar uma proposta de planejamento de ensino sem a participação inicial dos licenciandos de Matemática. Isto porque conhecíamos previamente o currículo do curso de Licenciatura do IFAC e a realidade dos participantes da pesquisa. O conhecimento prévio foi oportunizado pela experiência como docente no referido curso em mais de uma disciplina do currículo. Assim, sabíamos previamente da ausência de ementas que favorecessem o que estava sendo proposto em nosso trabalho.

Além disso, para ampliar nosso conhecimento sobre os participantes, aplicamos um questionário socioacadêmico para identificar, por exemplo, aspectos relacionados à disponibilidade de recursos tecnológicos para o estudo fora do ambiente escolar. Também realizamos uma avaliação diagnóstica, a fim de identificar os conhecimentos prévios dos estudantes quanto aos assuntos de programação de computadores e de matemática.

Na Figura 5, ilustramos o que foi proposto no planejamento e o executado, demonstrando, assim, as principais alterações efetuadas no planejamento inicial.

Figura 5. Planejamento Vs Execução



Fonte: Elaboração própria

O detalhamento do que levou a estas modificações é descrito na seção de implementação. A seguir, apresentaremos todo o planejamento prévio da pesquisa.

### 3.3.1 Planejamento da Pesquisa-Ação

Como mencionamos anteriormente, o planejamento ocorreu de forma prévia, baseado em nossos conhecimentos sobre os estudantes, associados ao fato do tempo de docência na disciplina e do diagnóstico prévio. Parte do planejamento prévio seguiu coerentemente, outros foram modificados em virtude da aplicação e os *feedbacks* produzidos, tanto pelos estudantes, quanto por nossas observações, pontuadas no decorrer da aplicação da pesquisa.

A estratégia do planejamento e replanejamento foram mantidas durante toda a aplicação da pesquisa, a fim de melhorar a prática. Na etapa de planejamento realizamos um conjunto de atividades, tais como: definição dos resultados pretendidos de aprendizagem, seleção e organização de conteúdos de Matemática e Programação, seleção e elaboração de um conjunto de recursos de apoio a aprendizagem (por exemplo, vídeos, textos, códigos) e a elaboração das tarefas investigativas.

Em conjunto com esse planejamento de ensino, foram elaborados instrumentos que tratam da relação do estudante com a pesquisa, como o Termo de Consentimento Livre e Esclarecido (Apêndice G), Questionário Socioacadêmico (Apêndice F), e uma Avaliação Diagnóstica (Apêndice C). Estes instrumentos, principalmente o Questionário Socioacadêmico e a Avaliação Diagnóstica, foram utilizados para o reconhecimento situacional e para nortear o planejamento. O detalhamento do planejamento inicial de todo o processo de ensino será apresentado nas seções seguintes.

#### 3.3.1.1 Detalhamento

O processo de ensino levou em consideração a conjugação de conteúdos de Matemática e de conteúdos de programação. A forma de administrar esses conteúdos obedece a uma dinâmica que é, na perspectiva de investigação, onde as atividades de ensino e aprendizagem foram guiadas por roteiros. Estes roteiros foram concebidos tomando como referências Investigações Matemáticas.

Nosso planejamento levou em consideração os seguintes pressupostos pedagógicos:

- i) Adotar a programação como uma possibilidade de otimização dos recursos computacionais, proporcionando ao estudante uma experiência de aprendizagem

pautada na construção do conhecimento, utilizando o computador para atividades que excedem a operacionalização de aplicativos;

- ii) Desenvolver habilidades de programação que permitam ao estudante sua utilização na resolução de equações algébricas e plotagem de gráficos, vislumbrando suas potencialidades para a aprendizagem em Matemática e também para o ensino, em sua futura atuação profissional;
- iii) Adotar roteiros de estudos como um instrumento norteador para a aprendizagem dos estudantes, os quais denominamos Roteiros de Tarefas Investigativas (RTI), que estão pautados em Investigações Matemáticas.

Como já dito, os Roteiros de Tarefas Investigativas (RTI) são instrumentos que apresentam tarefas aos licenciandos e os conduzem no processo de descoberta, por meio de perguntas, no qual, ora eles têm que estudar de forma autônoma, com o suporte de recursos de estudo (vídeos, textos, exemplos de códigos), ora eles devem interagir com o professor. Os conteúdos abordados em cada um dos roteiros estão sumarizados no Quadro 6.

Quadro 6. Conteúdos dos RTIs

<p><b>RTI 01: Automatização de Resolução de Equações Algébricas do 1º Grau.</b>  <b>Conteúdos:</b> <u>Matemática</u> - Equações Algébricas do primeiro grau. <u>Programação</u> - Conceito; Variáveis; Atribuição de valores às variáveis; Estrutura sequencial e de seleção simples (IF); Automatização de cálculos para equações do primeiro grau com pacote Numpy.</p>
<p><b>RTI 02: Automatização de Resolução de Equações Algébricas do 2º Grau.</b>  <b>Conteúdos:</b> <u>Matemática</u> - Equações Algébricas do segundo grau. <u>Programação</u> - Estrutura de seleção simples (IF); Automatização de cálculos para equações do segundo grau com utilização do pacote Numpy.</p>
<p><b>RTI 03: Automatização de Plotagem de Gráficos de Função do 2º Grau.</b>  <b>Conteúdos:</b> <u>Matemática</u> - Funções quadráticas; Obtenção de raízes e vértice da função; Plotagem do gráfico da função. <u>Programação</u> - Estrutura sequencial; Estrutura de decisão encadeada (IF, ELIF, ELSE); Automatização de cálculos para funções quadráticas com utilização do pacote Numpy; Plotagem do gráfico da função quadrática com Matplotlib.</p>
<p><b>RTI 04: A Elipse e suas equações reduzidas</b>  <b>Conteúdos:</b> <u>Matemática</u> - As Cônicas/Elipses; Equações reduzidas da Elipse; Cálculo das equações reduzidas da Elipse.</p>
<p><b>RTI 05: Automatização e Obtenção dos elementos da Elipse.</b>  <b>Conteúdos:</b> <u>Matemática</u> - Cálculo para obtenção dos elementos da Elipse. <u>Programação</u> - Automatização de cálculos para obtenção de elementos da Elipse com Python e Numpy.</p>
<p><b>RTI 06: Plotagem do gráfico da Elipse.</b>  <b>Conteúdos:</b> <u>Matemática</u> - Esboçar o gráfico da Elipse; Elementos da Elipse no gráfico. <u>Programação</u> - Utilização de recursos da biblioteca Matplotlib e comandos da linguagem Python para plotagem de gráficos e marcações de elementos da Elipse</p>

Fonte: Elaboração própria

Os RTIs são apenas uma parte do processo de ensino-aprendizagem, os quais são constituídos também por conteúdos de Matemática, conteúdos de programação, atividades de ensino, atividades de aprendizagem, recursos de apoio e avaliação. Na seção seguinte, discutiremos sobre cada uma delas. Logo após, será feita a apresentação da execução da proposta de ensino.

### 3.3.1.2 Planejamento da Carga Horária

Nossa proposta foi, inicialmente, composta por seis roteiros, previstos para serem executados em 34 (trinta e quatro) horas aulas a serem cumpridas na disciplina de Tecnologia da Informação e Comunicação Aplicadas ao Ensino, que compõe a grade curricular da Licenciatura em Matemática, disposta no 7º período do curso. Essas aulas foram administradas em 42 (quarentas e duas) aulas de cinquenta minutos cada.

### 2.3.1.3 Conteúdos de Matemática

Nesta proposta de ensino, primamos por abordar tanto conteúdos na parte de álgebra quanto na parte de geometria que já fizeram parte do ensino médio dos estudantes e destacamos nos PCNS (BRASIL, 2002) e na BNCC (BRASIL, 2018), pela sua relevância na Educação Básica.

Para seleção dos conteúdos de Matemática e elaboração dos roteiros, analisamos um conjunto de livros que são comumente adotados no ensino médio (DANTE, 2009; FARAGO et al., 2012; IEZZI et al., 2013; SMOLE; DINIZ, 2010, 2010a) e na graduação, considerando o curso de Licenciatura em Matemática (CAMARGO; BOULOS, 2005; LEITHOLD, 1994; WINTERLE, 2014). Quanto à organização dos conteúdos, iniciamos com a álgebra, aplicada na resolução de equações do primeiro grau para determinação do valor da variável. Na sequência, abordamos equações do segundo grau, determinando e verificando o valor de *Delta* e da incógnita. O próximo assunto envolvia as funções quadráticas ou funções do segundo grau, focando na obtenção dos zeros ou raízes da função e coordenadas do vértice, por exemplo. Na sequência, os gráficos das funções quadráticas e sua construção.

No campo da geometria, os conteúdos abordavam o estudo de elipses, no âmbito da geometria analítica, cobrindo a forma geométrica da elipse e seus elementos, como focos, centro e eixos. Na sequência, a ênfase é nos cálculos algébricos para obtenção das equações reduzidas da elipse e obtenção dos valores como as coordenadas dos focos, centro e dimensões dos eixos. Posteriormente, a construção do gráfico da elipse e marcação de seus elementos principais, a partir da equação reduzida. A síntese dos conteúdos é apresentada no Quadro 7.

Quadro 7. Conteúdos de Matemática.

- |                                                                                                                                                                                                                                                                                                                  |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>i) Equações Algébricas do primeiro grau;</li> <li>ii) Equações Algébricas do segundo grau;</li> <li>iii) Funções quadráticas;</li> <li>iv) Gráficos das funções quadráticas;</li> <li>v) Elipses e suas equações reduzidas;</li> <li>vi) Gráficos das Elipses.</li> </ul> |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Fonte: Elaboração própria

Aliados aos conteúdos matemáticos, havia também os conteúdos de programação, conforme será apresentado na próxima seção.

### 3.3.1.4 Conteúdos de Programação

No escopo deste trabalho, utilizamos a linguagem de programação *Python* juntamente com pacotes e bibliotecas adicionais, como o *Numpy* e a *Matplotlib*. A abordagem de programação utilizada nesta pesquisa não foca no ensino-aprendizagem de programação por exploração, baseada em exemplos de códigos. Assim, os conteúdos relacionados à programação foram selecionados para introduzir os estudantes de licenciatura no universo da programação, focados nas habilidades que buscamos desenvolver e/ou potencializar neles.

Assim, trabalhamos conteúdos relacionados às variáveis, como manipulação, atribuição de valores, operações matemáticas básicas, funções de entrada de dados, como *input*, e de saída como *print*, por exemplo. Tratamos também conceitos relacionados à estrutura sequencial de comandos e estruturas de decisão, como *if*, *elif*, *else* por exemplo e listas. Funções como *math.sqrt()*, que possibilita a extração de raízes e *np.arange()*, para criação de um arranjo a partir de um intervalo específico.

No que se refere à plotagem dos gráficos, exploramos comandos e funções pertencentes à biblioteca *matplotlib*, utilizados também para customização dos mesmos. Dentre esses comandos, citamos o *plt.plot()* e *Ellipse()*, para plotagem de gráficos, além do *ax.annotate()*, *ax.spines*, para customização do gráfico e configuração do ambiente de plotagem. Os conteúdos estão sintetizados e apresentados no Quadro 8.

Quadro 8. Conteúdos de Programação

- |                                                                                                                                                                                                                                      |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>i) Conceito, Manipulação e impressão de variáveis;</li> <li>ii) Atribuição de valores às variáveis pelo próprio usuário;</li> <li>iii) Estrutura sequencial de comandos em Python;</li> </ul> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

- iv) Estrutura de seleção simples em Python;
- v) Estrutura de decisão encadeada em Python;
- vi) Funções e comandos do pacote *numpy* para resolução de equações;
- vii) Automatização, plotagem e customização de gráficos utilizando a biblioteca *Matplotlib*.

Fonte: Elaboração própria

A introdução desses conteúdos aos estudantes de Licenciatura em Matemática possibilitar que os mesmos vislumbrem aplicações de programação para além da automatização de resolução de equações e plotagem de gráficos. Ao estudarem os comandos e estruturas em *Python*, os estudantes são conduzidos a conhecimentos necessários para o atendimento de demandas específicas em sua área de formação, como, por exemplo a plotagem de triângulos e quadrados. Com o intuito de viabilizar a assimilação desses conteúdos, descrevemos na seção seguinte as atividades de aprendizagem propostas neste trabalho.

#### 3.3.1.5 Atividades de Aprendizagem

As atividades de aprendizagem foram conduzidas por meio de Roteiros de Tarefas Investigativas (RTI), construídos e administrados em uma perspectiva de exploração e investigação. O objetivo dos RTIs era orientar o estudante no processo de aprender por investigação e exploração, conduzindo para a construção de programas aplicados ao contexto da Matemática.

Ainda, por meio de atividades desafiadoras e que incentivam o raciocínio, os roteiros convidam o estudante à leitura, interpretação e organização das ideias, habilidades requeridas para toda a vida. Destarte, acreditamos que podemos favorecer a autonomia do estudante por meio de RTI, pois os mesmos colocam os estudantes em uma posição de sujeitos ativos na construção do seu conhecimento. Nesse sentido, apresentamos no Quadro 9 uma visão geral dos roteiros planejados, de acordo com o elucidado na Seção 2.3 e no Apêndice A disponibilizamos um dos roteiros<sup>40</sup> planejados na íntegra.

Durante a execução dos roteiros, o pesquisador exerce papel importante, detalhados nas atividades de ensino apresentadas na seção seguinte.

---

<sup>40</sup> Está disponível no Apêndice o RTI03, sendo que todos os outros podem ser acessados no endereço: <http://bit.do/esxru>

Quadro 9. Organização macro dos Roteiros de Tarefas Investigativas Planejados

Resultados Pretendidos da Aprendizagem (Principais)	Número e Título do RTI	Conteúdos Abordados				
		Matemática	Programação			
i) Aplicar os comandos básicos da linguagem Python para automatizar soluções de equações algébricas, de primeiro grau.	01 – Automatização de Resolução de Equações Algébricas do 1º Grau	i) Equações Algébricas do primeiro grau – estrutura e resolução de problemas;	i) Conceito, Manipulação e impressão de variáveis; ii) Atribuição de valores às variáveis pelo próprio usuário; iii) Estrutura sequencial de comandos em Python; iv) Estrutura de seleção simples em Python (IF); v) Automatização de cálculos para resolução de equações do primeiro grau com utilização do pacote <i>numpy</i> .	Recursos de Apoio: multimídia, tutoriais, scripts.	Tarefas Investigativas	Avaliação 01 referente aos RTIs 01, 02 e 03
i) Aplicar os comandos básicos da linguagem Python para automatizar soluções de equações algébricas, de segundo grau.	02 – Automatização de Resolução de Equações Algébricas do 2º Grau	i) Equações Algébricas do segundo grau – estrutura e resolução de problemas;	i) Estrutura de seleção simples em Python (IF); ii) Automatização de cálculos para resolução de equações do segundo grau com utilização do pacote <i>numpy</i> .			
i) Aplicar os comandos básicos da linguagem Python para automatizar soluções de funções do segundo grau e plotagem de gráficos com <i>Matplotlib</i> .	03 – Automatização de Plotagem de Gráficos de Função do 2º Grau	i) Funções quadráticas; ii) Obtenção de raízes e vértice da função; iii) Plotagem do gráfico da função;	i) Estrutura sequencial de comandos em Python; ii) Estrutura de decisão encadeada em Python (IF, ELIF, ELSE); iii) Automatização de cálculos para resolução de funções quadráticas com utilização do pacote <i>numpy</i> . iv) Automatização e plotagem do gráfico da função quadrática com utilização da biblioteca <i>Matplotlib</i> ;			

i) Calcular a equação reduzida da elipse; ii) Identificar os elementos principais da elipse.	04 – A Elipse e suas equações reduzidas	i) As Cônicas/Elipses; ii) Equações reduzidas da Elipse; iii) Cálculo das equações reduzidas da Elipse.	Não se aplica	Recursos de Apoio: multimídia, tutoriais, scripts.	Tarefas Investigativas	Avaliação02 referente aos RTIs 04, 05 e 06.
i) Calcular os elementos principais da Elipse a partir de sua equação reduzida; ii) Aplicar os comandos básicos da linguagem Python para automatizar soluções de equações reduzidas da Elipse para obtenção de seus elementos principais.	05 – Automatização e Obtenção dos elementos da Elipse	i) Cálculo para obtenção dos elementos da Elipse.	i) Automatização de cálculos para obtenção de elementos da Elipse, utilização de comandos básicos em python e pacote <i>numpy</i> .			
i) Aplicar os comandos básicos da linguagem Python para automatizar soluções de plotagem de Elipses e marcações de seus pontos;	06 – Plotagem do gráfico da Elipse	i) Esboçar o gráfico da Elipse; ii) Elementos da Elipse no gráfico.	i) Utilização de recursos da biblioteca <i>Matplotlib</i> e comandos da linguagem python para plotagem de gráficos e marcações de elementos da Elipse.			

Fonte: Elaboração própria

### 3.3.1.6 Perspectiva de ensino e aprendizagem

As atividades de ensino foram planejadas também na perspectiva de incentivar a investigação. Durante todo o processo de ensino, os estudantes são questionados sobre como e o porquê de tomar algumas decisões para a resolução das atividades propostas, são levados a buscar padrões por meio da exploração, a investigar e a comparar exemplos de códigos. Acreditamos que essa estratégia conduza o estudante a uma postura investigativa, fazendo com que os mesmos se familiarizem com esse tipo de atividade e que, futuramente, os mesmos possam ser levados espontaneamente a indagações que provoquem a investigação.

Quanto ao papel do professor, são realizadas intervenções para esclarecimentos, dialogando com os estudantes de forma investigativa, pensando “em voz alta”, auxiliando os estudantes na construção do conhecimento, estando, também, à disposição, durante todo o tempo de aula. Além da interação com o professor, os estudantes terão acesso a um conjunto de materiais, pensados e selecionados com o objetivo de propiciar suporte aos estudantes para resolução das atividades propostas, conforme elucidados na seção seguinte.

### 3.3.1.7 Recursos de Apoio

Com o intuito de auxiliar os estudantes quanto à aprendizagem dos conteúdos de Matemática e de programação, mencionados nas Seções 2.3.1.3 e 2.3.1.4, foi planejado um conjunto de recursos de apoio, tais como 16 scripts, 5 tutoriais e disponibilizados 25 vídeoaulas, sendo duas de própria autoria e as demais pertencentes a canais do YouTube. Os recursos de apoio foram distribuídos em 6 RTIs, sendo que, para cada um dos roteiros, o estudante tinha a sua disposição, vídeos e/ou tutoriais e/ou scripts que auxiliavam na resolução das tarefas investigativas abordadas no roteiro. Na seção seguinte, apresentamos aspectos relacionados ao estudo autônomo.

### 3.3.1.8 Apoio ao estudo autônomo

Para viabilizar o estudo autônomo, foram planejados horários de atendimento ao estudante, com o intuito de dirimir dúvidas individuais ou coletivas, exercitar os conteúdos trabalhados em sala de aula ou discutir qualquer aspecto relacionado à proposta de ensino-aprendizagem. A periodicidade dos horários de atendimento era semanal ou sob demanda, previstos sempre no contra turno dos estudantes.

Também foram disponibilizados todos os *softwares*, *off-line*, utilizados em PBI, bem como nos colocamos à disposição para auxiliar os estudantes na instalação destes *softwares* em

seus computadores pessoais. Além das atividades de ensino e aprendizagem, cabe destacar sobre a avaliação, conforme descrito na próxima seção.

### 3.3.1.9 Avaliação

A avaliação é entendida como um elemento norteador das ações que visam favorecer a aprendizagem, além de possibilitar um *feedback* sobre o desempenho dos estudantes. Para auxiliar o professor na correção dos instrumentos avaliativos (RTIs e Avaliações), adotamos *rubricas*. Segundo Brookhart (2013), uma rubrica é um conjunto coerente de critérios sobre o trabalho a ser realizado pelos estudantes que inclui descrições de níveis de desempenho (*performance*). Segundo Biggs e Tang (2011), o uso de rubricas ajuda a manter a confiança e a estabilidade na correção e julgamento, assim como é um recurso para prover *feedback* aos estudantes.

Em nosso trabalho, elaboramos uma rubrica analítica para avaliação do desempenho dos estudantes. Uma rubrica analítica é aquela que possui vários critérios, sendo cada um deles avaliado separadamente, fornecendo *feedback* mais específico para os estudantes (BENDER, 2014).

No Quadro 10, apresentamos a rubrica planejada e, como pode ser observado, nossa rubrica foi construída considerando 4 critérios (Raciocínio e cálculos, Construção e identificação dos gráficos, Soluções automatizadas para álgebra, Soluções automatizadas para gráficos) e quatro níveis de desempenho (Iniciante, Próximo de Proficiente, Proficiente e Avançado). Para a combinação critério-nível, há uma descrição da qualidade do desempenho.

Quadro 10. Rubrica avaliativa planejada

<b>Conceitos:</b>	Avançado	Proficiente	Próximo de Proficiente	Iniciante
<b>Habilidades</b>				
Raciocínio e cálculos (aritmética)	<p>Demonstra base de raciocínio matemático.</p> <p>Demonstra entendimento dos procedimentos para os cálculos, apresentando solução correta.</p>	<p>Demonstra base matemática e raciocínio correto para desenvolver uma resposta para a questão.</p> <p>Demonstra poucos erros quantos aos procedimentos para realização dos cálculos algébricos.</p>	<p>Demonstra alguma base matemática.</p> <p>Demonstra algum raciocínio correto, mas não suficiente para desenvolver a resposta da questão.</p> <p>Demonstra alguma habilidade com a aritmética, porém não consegue construir de forma correta uma solução algébrica, mesmo à mão.</p>	<p>Não apresenta base matemática.</p> <p>Não demonstra raciocínio correto ou justificável para elaborar uma solução.</p> <p>Não demonstra habilidade com a aritmética e não consegue construir uma solução algébrica à mão.</p>
Construção e identificação dos gráficos	<p>Demonstra habilidade para esboçar o gráfico que representa a solução do problema.</p> <p>Demonstra entendimento na marcação dos elementos do gráfico.</p>	<p>Demonstra habilidade para esboçar o gráfico que representa a solução do problema.</p> <p>Demonstra entendimento na marcação dos elementos do gráfico, com poucos erros.</p>	<p>Demonstra habilidade para esboçar o gráfico que representa a solução do problema proposto.</p> <p>Demonstra muitos erros na marcação dos elementos do gráfico.</p>	<p>Não demonstra habilidade para esboçar o gráfico do problema.</p>
Soluções automatizadas para a álgebra	<p>Demonstra entendimento dos procedimentos para a construção de programas com independência dos</p>	<p>Demonstra entendimento dos procedimentos para a construção de programas, com alguma independência</p>	<p>Demonstra entendimento dos procedimentos básicos para a construção de um programa simples e apenas</p>	<p>Não apresenta entendimento sobre os procedimentos básicos para a construção de um programa, mesmo que</p>

	<p>exemplos, para automatizar soluções algébricas para problemas simples e complexos.</p> <p>Demonstra entendimento de procedimentos para variação do programa, a fim de atender a resolução de outros tipos de problemas.</p>	<p>dos exemplos para automatizar soluções de equações algébricas para problemas simples e mais complexos.</p> <p>Demonstra poucos erros quando tenta alterar os exemplos para atender a resolução de outros tipos de problemas.</p>	<p>se baseando em exemplo, para automatizar soluções de equações algébricas para problemas simples.</p> <p>Demonstra muitos erros quando tenta alterar os exemplos para atender a resolução de outros tipos de problemas.</p>	<p>simples e baseado em exemplos, para automatizar uma solução algébrica.</p>
<p>Soluções automatizadas para os gráficos</p>	<p>Demonstra entendimento dos procedimentos para construção de programas com independência dos exemplos, para automatizar soluções de plotagem de gráficos.</p> <p>Demonstra entendimento dos procedimentos para customização de gráfico, com independência dos exemplos.</p> <p>Demonstra entendimento de procedimentos para variação do programa, a fim de atender a resolução de outros tipos de problemas.</p>	<p>Demonstra entendimento dos procedimentos para a construção de programas, com alguma independência dos exemplos para automatizar soluções de plotagem de gráficos para problemas simples e mais complexos.</p> <p>Demonstra entendimento de procedimentos para customização de gráfico, com alguma independência dos exemplos.</p> <p>Demonstra poucos erros quando tenta alterar os exemplos para atender a resolução de outros tipos de problemas.</p>	<p>Demonstra entendimento dos procedimentos básicos para a construção de um programa simples e apenas se baseando em exemplo, para automatizar soluções de plotagem de gráficos.</p> <p>Demonstra entendimento de procedimentos básicos para customização do gráfico, somente se baseando em exemplos.</p> <p>Demonstra muitos erros quando tenta alterar os exemplos para atender a resolução de outros tipos de problemas.</p>	<p>Não apresenta entendimento sobre os procedimentos básicos para construção de um programa, mesmo que simples e baseado em exemplos, para automatizar uma solução para plotagem de gráfico.</p> <p>Não apresenta entendimento sobre os procedimentos básicos para construção de um programa, mesmo que simples e baseado em exemplos, para customização de gráfico.</p>

Fonte: Elaboração Própria

Para avaliação do desempenho dos estudantes, adotamos dois tipos de instrumentos – os roteiros de tarefas investigativas e os roteiros avaliativos. Foram planejados seis RTIs, respondidos individualmente, mas os alunos eram livres para trocar ideias com os colegas, com o pesquisador, consultar material fornecido, outros materiais etc.

Quanto aos roteiros avaliativos<sup>41</sup>, foram aplicados dois: o primeiro (vide Apêndice D), ao final da realização dos três primeiros RTIs. Esse roteiro avaliativo foi constituído de assuntos relacionados às equações de segundo grau e funções quadráticas, automatização e plotagem de gráficos. O segundo roteiro avaliativo foi aplicado ao final do sexto RTI e compreendeu assuntos relacionados à geometria, automatização e plotagem do gráfico da elipse. Os dois roteiros avaliativos foram respondidos individualmente pelos alunos, não sendo permitida a interação com outros colegas ou a consulta a outras fontes de dados, além das fornecidas no próprio roteiro (fragmentos de códigos). Os dois roteiros avaliativos tinham como foco avaliar o desempenho dos estudantes quanto a automatização de resolução de equações e plotagem de gráficos.

Além da avaliação do desempenho dos estudantes, planejamos a realização de uma avaliação do processo de ensino. Nesse caso, a avaliação foi realizada com base no *feedback* dos estudantes, anotações no diário de campo, uma roda de conversa prevista para o final do curso e um questionário de avaliação final.

### 3.3.2 Implementação das Ações

A implementação de nossa proposta de ensino-aprendizagem, denominada PBI (Programação Baseada em Investigação), ocorreu no período de 24 de março a 29 de junho de 2017, no turno noturno, com 18 alunos do IFAC, matriculados na disciplina de Tecnologias da Informação e Comunicação Aplicadas ao Ensino do curso Superior de Licenciatura em Matemática.

A dinâmica da implementação está sumarizada no Quadro 11 e, como pode ser observado, a implementação teve como base a execução de cada um dos roteiros de tarefas investigativas, que, nesse contexto, foram aplicados em mais de um encontro<sup>42</sup>. Cabe ressaltar que os dias de aulas não foram distribuídos semanalmente, em virtude de professores do campus Cruzeiro do Sul estarem cursando pós-graduação. Dessa forma, foi necessário ajustar o

---

<sup>41</sup> Disponibilizamos no Apêndice D apenas o Roteiro Avaliativo 01 planejado, o outro roteiro poderá ser acessado por meio do endereço: <http://bit.do/esxqZ>.

<sup>42</sup> No contexto da pesquisa, encontro deve ser entendido como um dia de aula, que por sua vez, era composto por uma ou mais aulas de 50 minutos.

planejamento interno de execução de algumas disciplinas, permitindo a liberação de professores. A narrativa da aplicação dos roteiros foi sistematizada em horas.

Quadro 11. Dinâmica utilizada na aplicação dos Roteiros

Atividades/Roteiros	Dinâmica Utilizada
Apresentação do plano de ensino, diagnóstico de conhecimento prévio, questionário socioacadêmico e termo de consentimento livre e esclarecido.	Estudantes responderam os questionários e as tarefas individualmente; para o diagnóstico, tiveram à disposição fragmentos de código como recurso de apoio. Execução: dia 24/03, quatro aulas de 50 minutos.
Roteiros Tarefas Investigativas 1, 2, 3	Estudantes responderam as tarefas em grupos, podendo consultar uns aos outros e ao pesquisador; tiveram à disposição recursos de apoio (exemplos de códigos, vídeoaulas, tutoriais), podendo consultar outros. A interação total entre os estudantes era permitida, mas a resposta entregue individualmente. Execução: de 31/03 a 01/06, vinte e quatro aulas de 50 minutos.
Roteiro Avaliativo 01	Estudantes responderam as tarefas individualmente; tiveram à disposição somente fragmentos de códigos fornecidos pelo pesquisador. Execução: 02/06, três aulas de 50 minutos.
Roteiros Tarefas Investigativas 4 <sup>43</sup> , 5, 6	Estudantes responderam as tarefas em grupos, podendo consultar uns aos outros e ao pesquisador; tiveram à disposição recursos de apoio (exemplos de códigos, vídeoaulas, tutoriais), podendo consultar outros. A interação total entre os estudantes era permitida, mas a resposta entregue individualmente. Execução: de 03/06 a 22/06, dezesseis aulas de 50 minutos.
Roteiro Avaliativo 02	Estudantes responderam as tarefas individualmente; tiveram à disposição fragmentos de códigos fornecidos pelo pesquisador. Execução: dia 29/06, três horas aulas de 50 minutos.
Avaliação da Experiência de ensino	Realizada uma roda de conversa e aplicação do questionário. Execução: dia 29/06, uma hora aula de 50 minutos.

Fonte: Elaborado própria

No primeiro encontro (dia de aula), antes de aplicarmos a avaliação diagnóstica, socializamos a proposta de ensino-aprendizagem e destacamos a importância da aprendizagem de programação pelos futuros professores, mostrando aos estudantes o movimento mundial de incentivo ao aprendizado de programação. Ainda nesse encontro, os estudantes leram e assinaram o termo de livre consentimento, bem como preencheram o questionário socioacadêmico e, somente depois, responderam a avaliação diagnóstica. Ainda, discutimos e acordamos os melhores dias da semana e horários para o atendimento ao estudante e criamos um grupo de *WhatsApp* e lista de *e-mail* para facilitar a comunicação. Toda essa dinâmica ocupou 3,33 horas (4 aulas de 50 minutos cada).

<sup>43</sup> O RTI04 foi realizado como atividade extraclasse, tendo sua carga horária não contabilizada como horas em sala de aula.

O turno noturno, segundo o regulamento da instituição de ensino, tem seu início marcado para as 19 horas, porém, nesse dia, menos de 50% dos estudantes estavam presentes antes das 19h30min. Por se tratar do primeiro encontro, achamos prudente aguardar a chegada de um grupo maior de estudantes. O esperado era que conseguíssemos iniciar com no máximo 15 minutos de atraso, contudo o início só foi possível às 19h30min. Relatos dos estudantes atrelaram os atrasos à dependência de carona e a aspectos relacionados ao exercício de suas profissões, que muitas vezes tem horário de saída que conflitam com o início das aulas. Acordamos para tentar evitar atrasos longos nos próximos encontros, mas os estudantes, em muitas vezes, não conseguiam, já que se tratava de fatores impeditivos alheios as suas vontades. Essa foi uma problemática enfrentada durante toda a aplicação e que também contribuiu ajustes da carga horária para aplicação das atividades.

Exceto pelo primeiro encontro, todos os demais foram executados no laboratório de informática, pois o ambiente proporcionou condições plenas para utilização, inclusive do quadro branco. A partir de agora, vamos narrar a aplicação separadas por roteiros e por horários de atendimento ao estudante realizados entre a aplicação de um e outro roteiro.

A aplicação do RTI01<sup>44</sup> iniciou com 30 minutos de atraso e apenas um estudante não compareceu. Fizemos contato com esse estudante pelo grupo do *WhatsApp*, nos colocando à disposição para auxiliá-lo nesse início, além de disponibilizar horas de atendimento, ao qual, posteriormente, o estudante compareceu. Como previsto, iniciamos a aula apresentando o objetivo do roteiro e a dinâmica esperada para respondê-lo. Ainda, reforçamos o pedido para que os estudantes trouxessem fone de ouvido para todas as aulas, necessário para melhor aproveitamento dos recursos de apoio, que incluíam vídeoaulas. Socializamos com os estudantes conceitos iniciais de elementos e nomenclaturas utilizadas em programação de computadores, bem como demonstramos e exercitamos procedimentos que os estudantes iriam utilizar em todos os roteiros, como ter acesso aos materiais de apoio, salvar seus códigos e utilizar o editor de Python WingIde 101.

Conforme o planejado, executamos todo o RTI01 em conjunto com os estudantes, demonstrando uma postura investigativa, lendo, questionando, pensando em voz alta, criando hipóteses, testando-as, provocando nos estudantes questionamentos que os conduzissem a investigar, explorar os códigos e a se perguntarem o porquê dos procedimentos, considerando os problemas do roteiro e simulando outros problemas. Nesse momento, participantes da pesquisa perceberam e comentaram sobre o aprendizado de matemática e de programação em

---

<sup>44</sup> Está disponível no Apêndice B o RTI03 executado, sendo que todos os demais podem ser acessados pelo endereço: <http://bit.ly/2uLFJob>

conjunto. O estudante Est10 disse que “estamos aprendendo matemática e programação simultaneamente”.

Quanto ao tempo de aplicação do roteiro, planejamos em 3,33 horas (4horas aula), que não foram suficientes, pois, ao verificarmos a resolução dos estudantes, identificamos que 30% deles não haviam concluído as tarefas propostas. Por meio de diálogo com os estudantes e tendo como base nossas observações, percebemos uma falta de familiaridade com a leitura e com a abordagem proposta, que, associada aos atrasos para o início do roteiro, contribuíram para exceder a carga horária prevista. Primando pelo engajamento dos estudantes na proposta, optamos em retomar o RTI01 antes de prosseguir com o planejamento. Os estudantes que haviam concluído foram convidados a participarem da aula em forma de monitor, auxiliando os outros. Assim, o RTI foi concluído com um total de dois encontros, contabilizando aproximadamente 6 aulas. Para o RTI 01, em todos os horários de atendimento, nenhum aluno compareceu.

Antes de iniciar o RTI02, conforme o planejado, revisamos a última tarefa do roteiro anterior, socializando e discutindo com os estudantes as suas respostas à tarefa, bem como verificando a existência de outras possibilidades de resolução. Na sequência, na mesma data, iniciamos o segundo roteiro. A dinâmica planejada para a execução desse roteiro, primava por uma condução mais autônoma por parte dos estudantes, porém, a pedido de 100% dos participantes da pesquisa, iniciamos o roteiro em conjunto, mantendo, na primeira tarefa, a mesma estratégia utilizada no RTI01. A solicitação dos estudantes foi pautada na falta de vivência com a dinâmica dos roteiros, e, dessa forma, atendemos ao pedido e buscamos demonstrar possibilidades de condução e organização do raciocínio para responder as tarefas investigativas.

Quanto aos aspectos relacionados à carga horária, foram necessárias 5 horas (6 aulas) para a aplicação, aproximadamente 1h20min a mais do que o planejado, sendo necessário um segundo encontro para aplicação. O segundo encontro coincidiu com as vésperas de um feriado, sendo necessária uma sensibilização dos estudantes para comparecerem, e o resultado é que tivemos a ausência de 3 deles.

Durante a aplicação do RTI02, foi possível identificar que a maioria dos estudantes estava compreendendo a lógica de programação e, nesse contexto, sua associação com a Matemática. Antes de finalizar o encontro, foi possível realizar a retomada da última tarefa do roteiro, discutindo com os participantes da pesquisa suas soluções propostas, bem como demonstrando alternativas para a resolução da tarefa. Os estudantes que demonstraram maiores dificuldades com a programação de computadores, foram convidados a comparecer no horário

de atendimento ao estudante, juntamente com os estudantes ausentes e demais que quisessem participar.

No atendimento ao estudante, tivemos a presença de dois estudantes, e, ao dialogarmos sobre as dúvidas, as dificuldades se mostravam associadas à lógica de programação. Com o intuito de dirimir as dúvidas, revisamos os conceitos e comandos básicos utilizados na linguagem de programação *Python* e desenvolvemos uma série de *scripts* com comandos e funções utilizadas no roteiro, como entrada e saída de dados, manipulação de variáveis e estruturas de seleção (*if/else*).

A partir das dificuldades demonstradas pelos estudantes no atendimento e percebendo em seus relatos que eram comuns aos demais, preparamos um roteiro extraclasse, denominado AT HOME, com problemas que deveriam ser resolvidos por meio da programação com *Python*. Nesse roteiro, disponibilizamos recursos de apoio, como vídeos e tutoriais, além de indicar cursos *on-line*, gratuitos e que têm por objetivo desenvolver a lógica de programação e favorecer a utilização da linguagem *Python*. Cabe ressaltar que nesse momento da pesquisa, devido a feriados e cumprimento de créditos do mestrado, tivemos um intervalo de duas semanas sem aplicação, tempo dado aos estudantes para a entrega do roteiro. As dúvidas e interação com o professor poderia ser feita por e-mail ou por *WhatsApp*. Embora tenham sido disponibilizados estes canais, os estudantes não enviaram dúvidas.

Com o intuito de disponibilizar, no campus, recursos tecnológicos para o estudo autônomo, considerando que 38,89% dos estudantes não possuíam computadores em casa, instalamos os *softwares* necessários nos computadores da biblioteca e negociamos com a gestão do campus a possibilidade de acesso ao laboratório de informática aos estudantes em horários alternativos. Dois dias antes do prazo para a entrega das atividades do roteiro AT HOME, os estudantes solicitaram, via *WhatsApp*, prorrogação do prazo para entrega, demonstrando, assim, que os mesmos não fizeram a atividade com antecedência.

Como havíamos agendado horários de atendimento antes do início do RTI03, doze estudantes compareceram e pediram para utilizar o momento para resolver as atividades do roteiro AT HOME. Apesar de não ter sido o planejado e o almejado, possibilitamos o horário para a realização das atividades, porém, mesmo sem ser obrigatório, nenhum dos estudantes iniciou qualquer um dos cursos *on-line* indicados no roteiro.

Precedendo o início do RTI03, discutimos e corrigimos as tarefas propostas no roteiro AT HOME, cujo objetivo era exercitar a lógica de programação e alguns comandos do *Python*. Durante a discussão, cinco estudantes apresentaram soluções diferentes para os problemas propostos, como na organização da estrutura de seleção (*if/else*). Assim, socializamos a lógica

utilizada por cada um deles, corrigindo os erros e destacando que, em programação, é possível percorrer caminhos diferentes para chegar ao mesmo resultado.

Com o intuito de instigar a investigação e motivar o raciocínio dos participantes da pesquisa, o que facilita a compreensão e resolução dos problemas dos roteiros, modificamos alguns itens das tarefas do RTI03. As alterações tiveram como foco provocar questionamentos sobre outras possibilidades para a resolução da tarefa e de cálculos matemáticos, utilizando a programação de computadores. No Quadro 12, ilustramos as principais alterações nas tarefas do roteiro.

Quadro 12. Comparativo entre o planejado e o executado no RTI03.

Tarefa Planejada	Tarefa Executada
<p>1.3 Após a análise do script, responda as seguintes perguntas:</p> <p>a) O script consegue calcular as raízes da função? E as coordenadas do vértice? Por quê?</p> <p>b) Insira os códigos no script, para que o mesmo:</p> <p>i) calcule as coordenadas do vértice e as imprima, considerando duas casas decimais;</p> <p>ii) Como você resolveu o item i) desse problema?</p> <p>c) Salve seu script em sua pasta de respostas, atribua um nome significativo a ele e que identifique o item repondido.</p>	<p>1.4 Após a análise do script, responda as seguintes perguntas:</p> <p>d) No script, existe outra forma de calcular o valor de delta. Responda:</p> <p>i) Qual seria?</p> <p>ii) Faça a alteração no script.</p> <p>iii) As alterações que você fez funcionaram corretamente? Por quê?</p> <p>e) No script, existe outra forma de extrair a raiz de delta. Responda:</p> <p>i) Qual seria?</p> <p>ii) Faça a alteração no script.</p> <p>iii) As alterações que você fez funcionaram corretamente? Por quê?</p> <p>f) O script consegue calcular as raízes da função? E as coordenadas do vértice? Por quê?</p> <p>g) Insira os códigos no script, para que o mesmo:</p> <p>i) Calcule as raízes da função;</p> <p>ii) Calcule as coordenadas do vértice e as imprima, considerando duas casas decimais;</p> <p>iii) Como você resolveu o item i) desse problema?</p> <p>iv) Como você resolveu o item ii) desse problema?</p> <p>v) Você verificou se funciona? Quais</p>

	<p>resultados foram impressos? Informe-os.</p> <p>h) Salve seu script em sua pasta de respostas, atribua um nome significativo a ele e que identifique o item repondido.</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Fonte: Elaboração própria

Ainda, em outras tarefas deste roteiro, incluímos questionamentos, como o porquê de determinada resposta ou impressão em tela gerada pelos *scripts*. É importante ressaltar que as alterações apresentadas no Quadro 12, deram-se a partir do *feedback* dos participantes da pesquisa na execução do RTI02, e das dificuldades percebidas no roteiro AT HOME.

No que diz respeito à carga horária destinada à aplicação do RTI03, projetamos inicialmente 6,67 horas (8 aulas), porém foram necessárias 10 horas (12 aulas) para o cumprimento das tarefas propostas no roteiro. O horário de início desse roteiro se manteve com os mesmos atrasos dos anteriores, ou seja, em média 30 minutos. Até esse momento, foram destinadas noites completas para a aplicação do roteiro (4 aulas/3,33 horas), porém, por questões institucionais, tivemos um encontro com apenas duas aulas (1,67 horas). Essa experiência demonstrou que, para a dinâmica proposta, duas aulas não são produtivas, pois os estudantes, em sua totalidade, afirmaram que o tempo é insuficiente para a execução, de maneira que, quando começam a entender a tarefa e a construir seus programas, já é tempo de finalizar o encontro.

A interação e o envolvimento dos estudantes nas tarefas desse roteiro se intensificaram, sendo que, constantemente, identificávamos questionamentos para si e para os outros de como e o porquê de determinada resposta ou execução de *scripts* fornecidos como exemplos. Observando o trabalho dos estudantes, deparamo-nos com situações claras de exploração e testes, como a alteração de cores, tipo de traçado e espessura para a plotagem da linha que define uma parábola do segundo grau.

Quando estávamos na metade da aplicação desse roteiro, 27,7% dos estudantes solicitaram a possibilidade de um atendimento extra, em horário alternativo ao acordado anteriormente. Ao discutirmos a proposta da turma, em que muitos alegavam impossibilidade de comparecimento no horário de atendimento extraclasse por incompatibilidade com o turno de trabalho, agendamos esse atendimento para um sábado à tarde. Planejamos o atendimento das 15h às 18h, porém iniciamos às 15h20min e finalizamos às 19 horas, sendo que apenas 38,8% dos participantes da pesquisa compareceram.

A retomada das atividades desse roteiro ocorreu em um encontro específico para esse fim, no qual socializamos os caminhos que os estudantes percorreram em busca de soluções para as tarefas. Este foi o momento que antecedeu a nossa primeira avaliação, no qual aproveitamos para fazer uma revisão dos conteúdos abordados e dirimir as dúvidas dos estudantes. Ao discutir aspectos gerais dos roteiros, os participantes da pesquisa relataram que o RTI03 era muito longo, o que tornava sua execução cansativa.

O Roteiro Avaliativo<sup>45</sup> 01 foi realizado em um encontro posterior. Conforme planejado, este roteiro avaliativo tinha como objetivo verificar a aprendizagem individual quanto aos assuntos tratados nos roteiros anteriores. A avaliação foi realizada em 2,50 horas (3 aulas), superando o planejado anteriormente (1,67h ou 2 aulas) dado que alguns estudantes chegaram atrasados e solicitaram mais tempo.

No que diz respeito aos conteúdos de matemática e programação, previstos nesse roteiro avaliativo, os mesmos não foram alterados, porém, aportados por nossas vivências com as aplicações dos roteiros até o momento, fizemos alterações significativas na estrutura da avaliação e na organização das tarefas, que podem ser verificadas em detalhes, no Apêndice E. De maneira sucinta, a avaliação passou a possuir quatro partes, conectadas entre si:

- i) Resolução de equação à mão;
- ii) Automatização de resolução de equações, considerando a resolução de um problema matemático;
- iii) Plotagem do gráfico, considerando o problema proposto na parte ii);
- iv) Análise e reflexão, onde solicitamos que os estudantes comunicassem, por exemplo, aspectos relacionados ao raciocínio empregado na resolução da tarefa;

O RTI04 possuía somente assuntos matemáticos relacionados às elipses, tendo como base o estudo de suas equações reduzidas. Inicialmente, esse roteiro foi planejado para ser executado em 3,33 horas (4 aulas), em sala de aula, seguindo a mesma dinâmica dos roteiros anteriores. Porém, devido a questões institucionais que impactaram nos horários de aplicação da pesquisa, associada à iminência do término do semestre letivo, em comum acordo com os estudantes, passamos esse roteiro para ser executado extraclasse. Para proporcionar maior conforto aos participantes da pesquisa, enviamos todo o material via e-mail, bem como disponibilizamos, na coordenação do curso, cópias impressas do roteiro e folha de respostas.

---

<sup>45</sup> Disponibilizamos no Apêndice E o roteiro avaliativo 01 (executado), sendo que o roteiro avaliativo pode ser acessado pelo endereço: <http://bit.do/esxD2>.

Para dirimir as dúvidas, mantivemos contato via e-mail e grupo de *WhatsApp*, dando todo o suporte necessário para a resolução das tarefas.

Quanto ao RTI05, mantendo a dinâmica proposta até o momento, antes de iniciá-lo, retomamos as tarefas propostas na avaliação, discutindo e retirando dúvidas. Na sequência, iniciamos o recolhimento das repostas dos estudantes ao RTI04, momento em que percebemos que alguns estudantes estavam utilizando o tempo em sala para finalizar as tarefas propostas. Logo em seguida, iniciamos uma discussão sobre as dificuldades na resolução das tarefas do roteiro, momento em que os estudantes relataram que estavam vendo este conteúdo (elipses) pela primeira vez. Ao serem questionados, pois alguns responderam anteriormente (questionário socioacadêmico) terem estudado em Geometria Analítica, relataram que no momento que assinalaram as alternativas do questionário, não tinham certeza do que se tratava. Aproveitamos para dirimir dúvidas matemáticas relacionadas ao estudo de elipses.

No que diz respeito ao planejamento do RTI05, baseados nos relatos dos estudantes apontando para roteiros menores, optamos em reduzir alguns conteúdos relacionados a elipses e à programação. Nesse contexto, era prevista a automatização de resolução de equações das elipses, considerando seu estudo com o eixo maior ora vertical e ora horizontal, tendo o centro na origem (coordenadas 0 para x e 0 para y) e fora da origem (coordenadas -2 para x e 1 para y, por exemplo). Após a alteração, focamos a programação para automatizar a resolução de equações algébricas, no contexto das elipses, considerando o eixo maior na vertical e na horizontal, mas limitamos as coordenadas do centro na origem. Mesmo após o replanejamento, durante a aplicação do roteiro, demonstramos aos estudantes as alterações necessárias para atender a automatização de resolução de equações, considerando quaisquer coordenadas do centro.

Quanto à carga horária, o planejamento previa 5 horas para a aplicação desse roteiro, porém, foram necessárias 6,67 horas (8 aulas) para conclusão das atividades. No decorrer da aplicação, identificamos que as dificuldades dos estudantes, antes concentrada na programação de computadores, passou a ter como foco os conteúdos matemáticos estudados no roteiro. Assim, sem perder a postura investigativa, foram necessárias um número maior de intervenções, comparadas às realizadas nos roteiros anteriores, sendo necessário, constantemente, retomar os conceitos e procedimentos matemáticos relacionados a elipses.

Seguindo a mesma dinâmica dos roteiros anteriores, utilizamos os minutos que antecederam o início do RTI06 para retirar dúvidas a respeito das tarefas do roteiro anterior. Considerando que as discussões das tarefas do RTI05 foram mais intensas, mitigando as

dificuldades dos estudantes com os assuntos das elipses, fizemos uma breve revisão da álgebra envolvida no conteúdo.

Em se tratando dos conteúdos do RTI06, replanejamos segundo as alterações realizadas no RTI05, ou seja, focamos na plotagem de gráficos de elipses, considerando as coordenadas do seu centro posicionadas na origem (coordenada x igual a 0 e coordenada y igual a zero). Essa alteração contribuiu para que a carga horária inicialmente planejada fosse suficiente, mantendo as 6,67 horas (8 aulas).

Durante a aplicação desse roteiro, da mesma forma que no RTI05, fizemos intervenções constantes para a retomada dos conceitos Matemáticos envolvendo as elipses. Uma das dificuldades comumente identificadas está relacionada a confusões no momento da representação gráfica baseada em uma equação. Em diálogo com os estudantes, nesse momento da aplicação, todos afirmaram estar estudando elipses pela primeira vez, justificando suas dificuldades e limitações.

Ao finalizar o RTI06, foi possível, no mesmo encontro, retomar e discutir a última tarefa com os participantes da pesquisa, bem como dirimir dúvidas gerais, recapitulando os assuntos tratados até o momento. Considerando que nosso próximo encontro em sala foi destinado para a aplicação do Roteiro Avaliativo 02, propusemos um momento para atendimento ao estudante, agendado para o dia que antecedeu a avaliação, porém nenhum estudante compareceu.

Para a aplicação do Roteiro Avaliativo 02, baseados nas experiências adquiridas na aplicação do Roteiro Avaliativo 01, destinamos um tempo de 2,50 horas (3 aulas). A mesma estratégia foi aplicada na organização do roteiro e dos conteúdos, mantendo a estrutura utilizada na primeira avaliação, conforme pode ser observado no Apêndice F. Na aplicação dessa avaliação, 27,8% dos estudantes chegaram com mais de 30 minutos de atraso, o que fez com que esse grupo de estudantes tivesse que dar celeridade na resolução das tarefas.

Após os estudantes finalizarem a resolução das tarefas propostas no Roteiro Avaliativo 02, ainda no mesmo encontro, nos reunimos para uma roda de conversa e aplicação do questionário final da disciplina. Pelo adiantar da hora e por ser o último encontro, priorizamos a roda de conversa, momento em que os estudantes apresentaram os aspectos positivos, negativos e suas impressões, dada a vivência com PBI, tendo esta dinâmica 50 minutos de duração (1 aula). No que diz respeito ao questionário, explicamos o seu propósito e, após dialogar com os estudantes, disponibilizamos o mesmo em uma versão impressa, já que alguns estudantes relataram dificuldades com acesso à internet e em outra versão *on-line*, por meio da ferramenta *Google Forms*. Os estudantes que optaram pela versão impressa, responderam as

questões e entregaram na mesma noite; os demais responderam no dia seguinte, utilizando a ferramenta *on-line*.

Ainda na fase da implementação, ajustamos aspectos relacionados à rubrica avaliativa, tendo como foco as características que deveriam ser identificadas nas respostas dos estudantes para que os mesmos fossem classificados em um dos níveis de aprendizagem. Essas mudanças, partiram do desencadeamento de processos reflexivos sobre a nossa prática, aportados pelo *feedback* dos estudantes durante toda a aplicação dos roteiros. Na seção seguinte, discutiremos aspectos da avaliação, bem como apresentaremos a rubrica utilizada para correção das tarefas dos roteiros.

### 3.3.3 Avaliação das Ações

Como uma das etapas previstas no ciclo da pesquisa-ação, a avaliação teve a função de nortear as ações executadas para a melhoria contínua da prática.

Em nossa pesquisa, utilizamos alguns recursos para subsidiar a avaliação das ações, tais como: rubrica avaliativa; *feedback* dos estudantes durante a aplicação dos roteiros; observações anotadas no diário de campo; roda de conversa no final do curso e um questionário de avaliação final.

A rubrica, por exemplo, apresentada no Quadro 13, foi utilizada para avaliar a aprendizagem dos estudantes. Embora o foco do trabalho esteja na automatização de resolução algébrica e plotagem de gráficos, os estudantes, ao desenvolverem as tarefas investigativas, realizaram atividades que envolviam outras habilidades, como o raciocínio e cálculos, e construção e identificação dos gráficos. Tais habilidades, juntamente com a automatização de resolução algébrica e plotagem de gráficos, foram previstas na rubrica avaliativa que deu suporte a avaliação do aprendizado dos estudantes.

Quadro 13. Rubrica avaliativa aplicada

Conceitos:	Avançado	Proficiente	Próximo de Proficiente	Iniciante	Categoria de resposta
<b>Habilidades</b>					
Raciocínio e cálculos (aritmética)	Demonstra entendimento do que era solicitado na tarefa, identificação dos conceitos e fórmulas necessários a sua resolução e emprega cálculos totalmente corretos.	Demonstra entendimento do que era solicitado na tarefa, identificação da maioria dos conceitos e fórmulas necessárias a sua solução, apresenta pequenos erros no cálculo (jogo de sinal, resultado de uma operação aritmética).	Demonstra entendimento parcial do que era solicitado na tarefa, identifica apenas alguns conceitos e fórmulas necessárias a sua solução e apresenta muitos erros de cálculo.	Não demonstra entendimento do que era solicitado na tarefa, há tantos erros nos procedimentos matemáticos que a tarefa não pode ser resolvida.	
Construção e identificação dos gráficos	Demonstra habilidade para esboçar o gráfico que representa a solução do problema.  Demonstra entendimento na marcação dos elementos do gráfico.	Demonstra habilidade para esboçar o gráfico que representa a solução do problema.  Demonstra entendimento na marcação dos elementos do gráfico, com poucos erros.	Demonstra habilidade para esboçar o gráfico que representa a solução do problema proposto.  Demonstra muitos erros na marcação dos elementos do gráfico.	Não demonstra habilidade para esboçar o gráfico do problema.	
Soluções Automatizadas para a álgebra	Constrói programas para automatizar soluções algébricas para problemas simples <sup>46</sup> e complexos <sup>47</sup> . Os	Constrói programas para automatizar soluções algébricas para problemas simples e complexos.	Constrói programas para automatizar soluções algébricas para problemas simples e complexos. Os	Não consegue construir programas para automatizar soluções algébricas, mesmo que para problemas simples.	

<sup>46</sup> Problema simples: consideramos como problemas simples aqueles cuja solução envolve menos restrições e, por conseguinte, emprego de estruturas mais simples de programação, tal como, estrutura sequencial e de seleção simples. Por exemplo, uma equação do primeiro grau foi considerada neste trabalho com um problema simples, pois envolvia apenas como restrição verificar se o coeficiente de  $x$  era diferente de zero e a solução em termos de programação envolvia uma estrutura de seleção simples.

<sup>47</sup> Problemas complexos: consideramos como problemas complexos aqueles cuja solução envolve maior número de restrições e, por conseguinte, emprego de estruturas mais complexas de programação, tal como, estrutura sequencial aninhada. Por exemplo, uma equação do segundo grau foi considerada neste trabalho com um problema complexo, pois envolvia verificar além do coeficiente de  $x^2$ , as condições para cálculo do delta, o que consequentemente, impactava no emprego de estrutura de seleção aninhadas para a construção do programa.

	<p>programas construídos são diferentes dos exemplos trabalhados em sala de aula, demonstrando uma construção mais autônoma. Além disso, os programas são construídos sem erros de estrutura (estrutura repetição, seleção etc.), indentação, formatação (atende a formatação de entrada e saída), e sem erro de lógica /compreensão do problema (a solução atende ao domínio da tarefa e apresenta o que foi solicitado).</p>	<p>Contudo, os programas construídos seguem os exemplos trabalhos em sala de aula, não demonstrando construção autônoma do aluno. Mesmo baseando-se nos exemplos, os programas podiam apresentar mínimos erros (por exemplo, formatação de entrada e saída). Estes erros, por sua vez, não comprometiam a lógica e o raciocínio empregado, atendendo integralmente ao que era solicitado na tarefa.</p>	<p>programas construídos seguem apenas os exemplos trabalhos em sala de aula, não demonstrando construção autônoma do aluno. Mesmo baseando-se nos exemplos, os programas apresentavam alguns erros que comprometiam o atendimento integral do que era solicitado na tarefa.</p>	<p>Nem mesmo consegue usar os exemplos trabalhos em sala para responder ao que é solicitado.</p>	
<p>Soluções automatizadas para os gráficos</p>	<p>Constrói programas para automatizar a plotagem de gráficos, sendo que os programas construídos são diferentes dos exemplos trabalhados em sala de aula, demonstrando uma construção mais autônoma. Plota corretamente todos os elementos do gráfico, bem como legendas e identificações dos eixos do plano cartesiano. Demonstra entendimento para customizar (alterar cores, traçado das linhas,</p>	<p>Constrói programas para automatizar plotagem de gráficos. Contudo, os programas construídos seguem os exemplos trabalhos em sala de aula, não demonstrando construção autônoma do aluno. Mesmo baseando-se nos exemplos, os programas podiam apresentar mínimos erros (por exemplo, marcação no gráfico dos elementos solicitados na tarefa). Estes erros, por sua vez, não comprometiam a</p>	<p>Constrói programas para automatizar plotagem de gráficos. Contudo, os programas construídos seguem os exemplos trabalhos em sala de aula, não demonstrando construção autônoma do aluno. Mesmo baseando-se nos exemplos, os programas apresentavam alguns erros que comprometiam o atendimento integral do que era solicitado na tarefa.</p>	<p>Não consegue construir programas para automatizar a plotagem de gráficos, mesmo que para problemas simples. Nem mesmo consegue usar os exemplos trabalhos em sala para responder ao que é solicitado.</p>	

	espessura das linhas, bordas etc.) elementos do gráfico.	lógica e o raciocínio empregado, atendendo integralmente ao que era solicitado na tarefa.			
--	----------------------------------------------------------	-------------------------------------------------------------------------------------------	--	--	--

Fonte: Elaboração Própria

A avaliação da proposta de ensino-aprendizagem foi feita durante todo o processo, por meio de observações do próprio pesquisador e também do *feedback* e desenvolvimento dos alunos. Além disso, no final, realizamos uma roda de conversa e a aplicação de um questionário, disponibilizado impresso e por meio do *Google Forms*. Na roda de conversa, convidamos os estudantes a falar sobre a vivência nessa proposta de ensino-aprendizagem, apontando suas principais dificuldades, benefícios, bem como limitações da aplicação do curso. Tomados em conjunto (a aplicação da rubrica aos roteiros desenvolvidos pelos estudantes, as observações do pesquisador, o feedback dos estudantes durante todo o processo, a roda de conversa, a resposta ao questionário de avaliação) estes instrumentos subsidiaram nossa resposta às questões de pesquisa, conforme será apresentado no Capítulo 3

### 3.4 Técnicas e Instrumentos de coleta de dados

Para a coleta dos dados em nossa pesquisa, fizemos uso de técnicas e instrumentos sumarizados no Quadro 14.

Quadro 14. Técnicas e instrumentos de coletada de dados na pesquisa ação

FASE	TÉCNICA DE COLETA DE DADOS	INSTRUMENTO DE COLETA DE DADOS
PLANEJAMENTO	Vivência do pesquisador na Licenciatura Questionário socioacadêmico Avaliação diagnóstica	Currículo da Licenciatura e ementa da disciplina de Tecnologias da Informação e Comunicação Aplicadas ao Ensino.
IMPLEMENTAÇÃO	Observação	Diário de Campo
	Vivência dos estudantes durante os roteiros	Respostas e <i>feedback</i> dos estudantes às tarefas investigativas
AVALIAÇÃO	Roda de Conversa	Diário de Campo
	Aplicação de Questionários	Questionários
	Avaliação Desempenho dos estudantes	Aplicação da Rubrica aos Roteiros

Fonte: Elaboração própria

Para coleta de dados, utilizamos instrumentos já mencionados neste trabalho, como, por exemplo, questionário socioacadêmico, avaliação diagnóstica, avaliação dos roteiros. Ainda, durante a execução da pesquisa, todas as percepções do pesquisador diante do processo de ensino e dos estudantes, foram devidamente registradas. Esses instrumentos de coleta de dados forneceram, ao longo de todo processo, subsídios para a investigação e ação. Além disso, permitiram responder as questões de pesquisa e consolidar o produto educacional derivado desta pesquisa.

A roda de conversa, segundo Melo e Cruz (2014), apresenta-se como um rico instrumento metodológico de aproximação entre o pesquisador e os participantes da pesquisa, possibilitando uma comunicação dinâmica e produtiva entre os mesmos. Em nossa pesquisa, fizemos uso da roda de conversa na avaliação final do curso, momento em que os participantes da pesquisa puderam dialogar sobre aspectos relacionados à trajetória da pesquisa.

Para registro de nossas percepções, bem como os fatos ocorridos durante a aplicação do curso e as posturas dos participantes da pesquisa, fizemos uso do diário de campo, conhecido também como notas de campo (GERHARDT; SILVEIRA, 2009). O instrumento possibilitou o registro de observações e reflexões importantes durante a aplicação do curso, favorecendo as mudanças necessárias durante o curso, com o intuito de proporcionar uma melhor experiência de aprendizagem, além de contribuir com a compreensão do processo como um todo.

Como dito anteriormente, para a avaliação da aprendizagem dos estudantes, utilizamos as respostas encontradas nos RTIs, classificando em níveis de aprendizagem baseada em rubrica. Todos os resultados das análises são discutidos e apresentados no Capítulo 3.

### 3.5 Perspectiva da Análise

Em posse de todos os dados coletados no decorrer da pesquisa, organizamos os mesmos para análise. Na obra de Bauer e Gaskell (2000), os autores abordam inúmeros métodos para a análise de dados qualitativos, tais como: a análise de conteúdo classista; análise argumentativa; análise do discurso; análise da conversação e da fala; análise retórica. Passeggi (2008), além dos anteriores, cita métodos como a análise da narrativa, análise proposicional, análise (textual) de linguística e análise de enunciação. Cada um dos métodos de análise citados possui características e especificidades que os definem.

Em nosso trabalho, optamos por não utilizar um método específico para a análise dos dados. Pautados nas orientações de Bogdan e Biklen (1994), organizamos os dados de acordo com os instrumentos e contextos que os mesmos foram produzidos. Na sequência, a análise foi feita separadamente, por instrumento de coleta, buscando em todos eles, elementos relevantes para a nossa discussão e que pudessem subsidiar as respostas as nossas questões de pesquisa. No próximo capítulo, discutimos os resultados relacionados a análise de dados.

## 4 ANÁLISE DOS RESULTADOS

Neste capítulo, discutiremos os resultados alcançados por meio da aplicação de nossa proposta de ensino-aprendizagem, com o objetivo de responder nossas questões de pesquisa:

- Questão de Pesquisa 01: Em que aspectos PBI melhora as habilidades dos alunos para automatização de resolução de equações algébricas?
- Questão de Pesquisa 02: Em que aspectos PBI melhora as habilidades dos alunos para automatização da plotagem de gráficos?
- Questão de Pesquisa 03: Quais as limitações desta aplicação de PBI?

No que diz respeito à categorização dos estudantes, classificamos os mesmos seguindo os critérios definidos na rubrica, como avançado, proficiente, próximo de proficiente e iniciante. Os estudantes que faltaram e/ou que não entregaram (não devolveram para o pesquisador) qualquer um dos roteiros, foram classificados na categoria não entregaram.

Inicialmente, faremos uma síntese dos resultados obtidos para as nossas questões de pesquisa a fim de fornecer ao leitor uma visão geral dos ganhos e limitação de nossa proposta de ensino-aprendizagem. Na sequência, apresentamos uma análise detalhada para cada questão de pesquisa, a qual justifica a síntese dos resultados aqui apresentada.

### 4.1 Síntese dos Resultados

Iniciaremos com a primeira questão de pesquisa, que aborda: Em que aspectos a PBI melhora as habilidades dos estudantes para automatização de resolução de equações algébricas?

A análise dos dados forneceu evidências de que PBI melhorou as habilidades dos estudantes para programar soluções, com vistas à automatização de resolução de equações algébricas referentes aos conteúdos envolvendo equações do segundo grau e das elipses.

A aprendizagem de programação dos estudantes se dá de forma progressiva. De modo geral, quando avaliamos esses estudantes individualmente (nos roteiros avaliativos), verificamos que o desempenho deles está mais concentrado no nível proficiente (vide rubrica apresentada no Quadro 13), isto é, os alunos são capazes de construir programas, contudo os programas construídos seguem os exemplos trabalhados em sala de aula e, em alguns casos, apresentavam alguns poucos erros (por exemplo formatação de entrada e saída) que não comprometiam o atendimento do que era solicitado. No roteiro avaliativo 2, por exemplo, 67% dos estudantes alcançaram o nível proficiente. Isto demonstra que os alunos foram capazes de reutilizar código, uma prática comum e incentivada por desenvolvedores de *softwares*, pois o objetivo é reaproveitar códigos existentes e que muitas vezes já foram testados e aperfeiçoados, otimizando assim tempo e esforço no desenvolvimento (VEREJÃO, 2004). Quando um aluno

é capaz de reutilizar código, ele demonstra que compreende a sua lógica e, quando necessário, faz adequações a fim de resolver um problema

Para o alcance do nível avançado, os alunos precisariam dar um “salto” em termos de aprendizagem, que era o de criar programas de forma mais autônoma, diferenciando-se dos exemplos fornecidos em sala de aula. Embora o nível avançado não tenha sido alcançado de forma mais frequente pelos estudantes, consideramos que os resultados obtidos revelam melhoria da aprendizagem, principalmente se considerarmos que os alunos não tinham habilidades de programação, sendo esta a primeira experiência no currículo da Licenciatura. Na avaliação diagnóstica, por exemplo, 100% dos alunos demonstraram nível de desempenho considerado iniciante, isto é, não conseguiam construir qualquer programa simples, mesmo quando lhes eram apresentados exemplos.

Quanto à segunda questão de pesquisa, que trata: Em que aspectos PBI melhora as habilidades dos estudantes para automatização da plotagem de gráficos?

De modo similar, a análise dos dados forneceu evidências de que PBI melhorou as habilidades dos estudantes para programar soluções para plotagem de gráficos, considerando os conteúdos de funções do segundo grau e elipses.

Também para esta habilidade, ao considerarmos as avaliações individuais (nos roteiros avaliativos), verificamos que o desempenho dos alunos está mais concentrado no nível proficiente, alcançado no roteiro avaliativo 2, por 61,1% dos alunos.

No caso dos programas para plotagem de gráficos, esse resultado é bastante promissor, principalmente, porque os programas tinham mais linhas de códigos e manipulavam funções mais complexas. Os alunos terem conseguido compreender a estrutura dos códigos fornecidos e responder as tarefas propostas, baseando-se neles, demonstram um resultado de aprendizagem significativo.

Também para essa habilidade, na avaliação diagnóstica 100% dos alunos demonstraram nível de desempenho considerado iniciante, isto é, não conseguiam construir qualquer programa simples, mesmo quando lhes eram apresentados exemplos.

A terceira questão de pesquisa aborda: Quais as limitações desta aplicação de PBI?

No decorrer da aplicação da proposta de ensino e aprendizagem, identificamos limitações, sendo que algumas delas foram corrigidas no andamento da pesquisa e outras foram propostas no produto educacional. Dentre elas, destacamos a necessidade de iniciar o aprendizado de programação de uma forma lúdica, tornando as primeiras experiências dos estudantes mais brandas. Também, vimos a necessidade de ajustar a carga horária total da proposta de ensino e aprendizagem, bem como a carga horária destinadas aos RTIs.

Aspectos, como a falta de cultura para o estudo autônomo, foram observados nos estudantes, no decorrer da aplicação, bem como a não utilização de ferramentas de comunicação instantânea, como o *whatsapp* e de *e-mail* para interação com o pesquisador e retiradas de dúvidas, o que poderia tornar o aprendizado mais dinâmico.

Com base nessas limitações, a proposta de ensino de ensino e aprendizagem foi ajustada para consolidar o produto educacional desta dissertação, que é um Guia Didático para Professores, onde apresentamos a Programação Baseada em Investigação (PBI). Nesse guia didático, demonstramos aos professores de informática que atuam na Licenciatura em Matemática, uma possibilidade de estudantes de licenciatura aprenderem programação de computadores de forma progressiva, baseada em exemplos a partir de atividades de investigação. Na seção seguinte, apresentamos e discutimos em detalhes as três questões de pesquisa.

#### 4.2 Detalhamento dos Resultados da Primeira Questão de Pesquisa

Conforme descrito anteriormente, a primeira questão de pesquisa é: Em que aspectos PBI melhora as habilidades de automatização de resolução de equações algébricas?

Para respondermos a esta questão de pesquisa, vamos nos valer dos dados obtidos dos estudantes em três momentos: avaliação diagnóstica, realização dos roteiros (os três primeiros sobre funções e os três últimos sobre elipses) e os roteiros avaliativos 1 e 2 (intermediária e final).

Iniciamos pela avaliação diagnóstica, que teve por objetivos identificar os conhecimentos prévios dos estudantes e fornecer subsídios para um possível replanejamento do primeiro roteiro de tarefas investigativas.

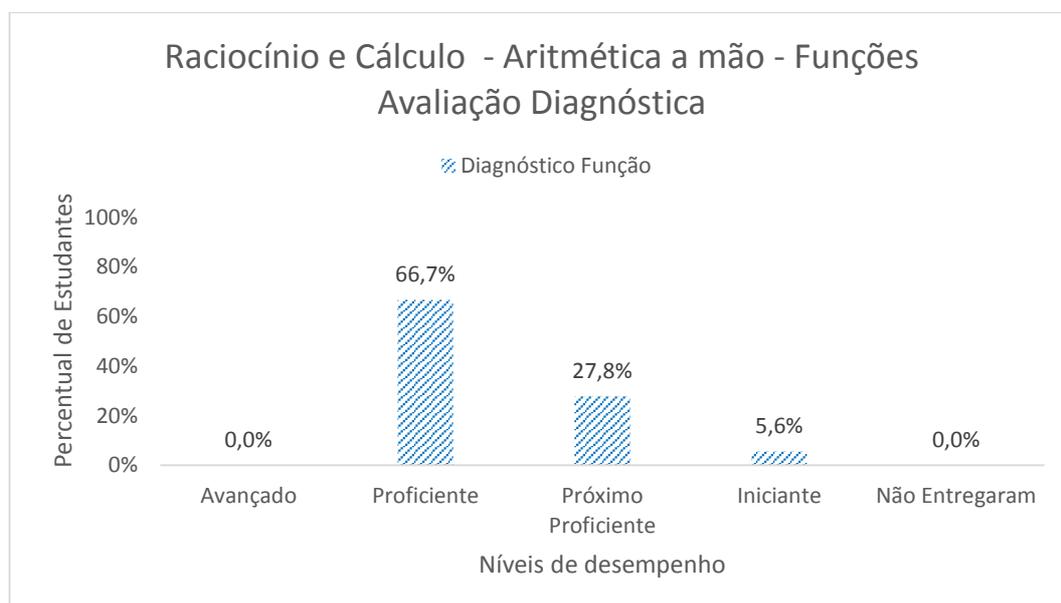
A avaliação diagnóstica foi composta por tarefas envolvendo conteúdos de funções do segundo grau e elipses. Para a resolução das tarefas, os estudantes eram inqueridos a solucionar questões à mão e também com o uso do computador, a fim de construir programas que automatizassem equações algébricas quanto à plotagem de gráficos.

No que diz respeito às tarefas que requeriam construção de programas para automatização de resolução de equações tanto para funções quanto para elipses, 100% dos alunos obtiveram desempenho iniciante (vide Quadro 12), demonstrando não conseguir construir qualquer programa, mesmo com o fornecimento de trechos de códigos. Os estudantes deixaram em branco as questões e, nem mesmo os dois estudantes que haviam dito ter alguma experiência prévia com programação, tentaram responder as tarefas. Esse era um aspecto de

certo modo previsto pelo pesquisador, haja vista que o curso de licenciatura em questão não prevê atividades com programação de computadores no currículo.

Mesmo se tratando da resolução de tarefas que envolviam funções do segundo grau com resolução dos cálculos feitos à mão, nenhum dos alunos teve desempenho categorizado como avançado (vide Figura 6), isto é, responder a tarefa com precisão nos cálculos, atendendo na totalidade o que era pedido na tarefa. A maioria dos estudantes (66,7%) teve seu desempenho categorizado como proficiente, isto é, suas soluções apresentavam pequenos erros de cálculo, sendo mais comum erros de “jogo de sinal” e de cálculos aritméticos envolvendo as operações básicas, principalmente envolvendo multiplicação de números inteiros. Apesar do assunto pertencer ao currículo da educação básica, pouco mais de um quarto dos estudantes teve o nível de desempenho classificado como próximo de proficiente, pois apresentaram muitos erros e/ou deixaram de responder parte das tarefas que foram solicitadas. A porcentagem de estudantes categorizada como iniciante refere-se ao caso daqueles estudantes que entregaram em branco as questões, porém havia vestígios de que a resposta havia sido apagada.

Figura 6. Diagnóstico da habilidade de raciocínio e cálculos realizados à mão no contexto de funções do segundo grau.



Fonte: Elaboração própria

Nas tarefas que envolviam conteúdos relacionados às elipses, para tarefas solicitando cálculos à mão, 100% dos estudantes tiveram desempenho categorizado como iniciante. Nesse caso, apenas um estudante tentou resolver de fato a tarefa, tendo esboçado a resposta e apagado posteriormente. Os demais deixaram a tarefa em branco, transparecendo ausência de tentativa em resolvê-la. Cabe ressaltar que a categoria não entregou refere-se aos estudantes que

deixaram de devolver os roteiros para correção, impossibilitando a correção de quaisquer atividades neles contidas.

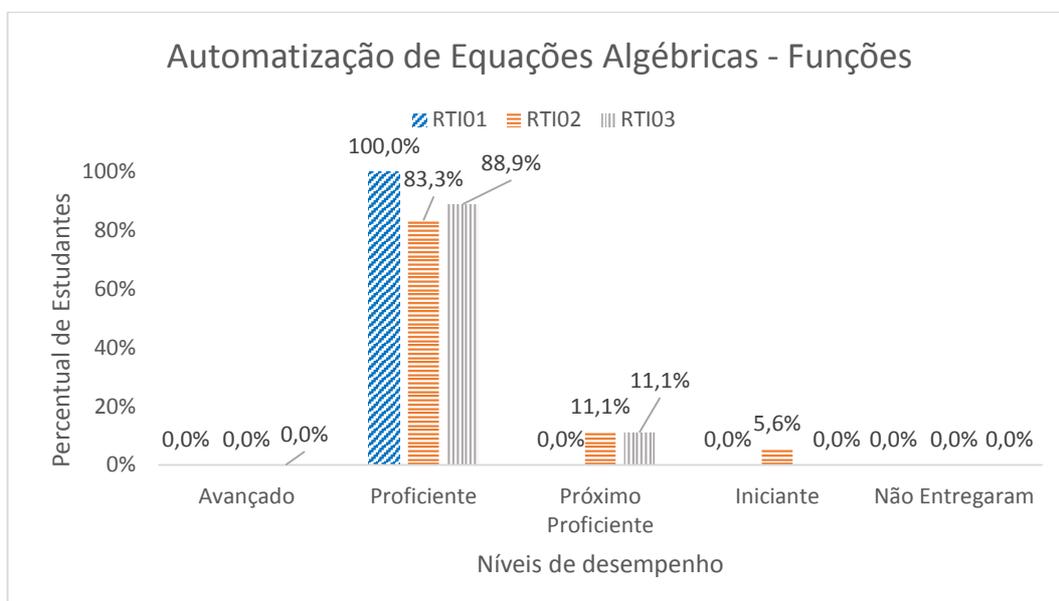
O desempenho dos estudantes no conteúdo de elipse para resoluções feitas à mão pode ser justificado pelo fato de que a Geometria é apontada como uma das áreas da Matemática com maior dificuldade de aprendizado por estudantes da Educação Básica, configurando-se em objeto de estudo de diversos pesquisadores que têm se debruçado na busca pela compreensão dessas dificuldades, bem como em estratégias que proporcionem melhorias no ensino e na aprendizagem de assuntos relacionados a ela (RESENDE; MESQUITA, 2013; CORTEZ, 2014; DE VILLIERS, 2010). Ainda, o ensino de conteúdos relacionados às cônicas, pertencentes à Geometria Analítica, comumente estudados no terceiro ano do ensino médio, em muitos casos, não acontece para a maioria dos estudantes e, quando abordados, são vistos de forma rápida e superficial (BONFIM, 2015).

Com o intuito de compreender melhor as razões para os resultados obtidos na avaliação diagnóstica, principalmente os relacionados à resolução de tarefas envolvendo elipses, conversamos com os estudantes e estes relataram não terem tido contato com este conteúdo no ensino médio e nem na graduação, apesar do conteúdo pertencer à área de Geometria Analítica e estar formalmente presente tanto no currículo da Educação Básica (comumente no 3º ano do ensino médio), quanto da Licenciatura em Matemática (BRASIL, 2017; BRASIL 2002a). Especificamente, no caso da instituição onde este estudo foi realizado, a matriz curricular possui a disciplina de Geometria Analítica no sexto período da licenciatura em Matemática e, ao observar a ementa, verificamos a presença do estudo das cônicas, apesar de todos os estudantes afirmarem que o conteúdo não foi abordado.

No que diz respeito à resolução dos RTIs, ao observamos o desempenho dos estudantes na automatização de resolução de equações algébricas, agora na perspectiva do desenvolvimento dos roteiros de tarefas investigativas, verificamos que para os três primeiros roteiros que tratavam sobre o estudo de funções, mais de 80% dos estudantes (vide Figura 7) tiveram seu desempenho categorizado como proficiente, isto é, eles foram capazes de reutilizar os exemplos de código trabalhados em sala de aula ou fornecidos como recursos de apoio na implementação de novos programas para resolver as tarefas propostas. Contudo, nesses roteiros nenhum aluno demonstrou em seus programas o emprego de lógica para elaboração da resposta ou estrutura de código diferente daqueles empregados em sala de aula ou nos recursos de apoio, justificando o fato de não haver estudantes categorizados com o nível avançado, conforme a rubrica utilizada na avaliação.

Cabe ressaltar que, em programação, reuso de código é uma prática comum e incentivada por desenvolvedores de *softwares*, pois o objetivo é reaproveitar códigos existentes e que muitas vezes já foram testados e aperfeiçoados, otimizando, assim, tempo e esforço no desenvolvimento (VEREJÃO, 2004). O desenvolvedor que é capaz de reutilizar o código compreende a sua lógica e, quando necessário, faz adequações a fim de resolver um problema. Em nossa proposta, o estudante que foi capaz de reutilizar códigos e trechos de códigos para resolução das tarefas investigativas foi classificado como proficiente. Dentro das expectativas do pesquisador, era esperado que a dinâmica de ensino pudesse prover a habilidade dos estudantes de reutilizarem os códigos para comporem suas soluções, e, nesse sentido, os resultados demonstram-se promissores.

Figura 7. Resultado dos Roteiros referentes à automatização de resolução de equações algébricas – Funções do segundo grau



Fonte: Elaboração própria

Ao observarmos a Figura 7, é possível notar que, no primeiro roteiro, 100% dos estudantes alcançaram o nível de desempenho proficiente e isto justifica-se pelo fato de que esse roteiro foi resolvido em conjunto com o pesquisador, estratégia adotada com o intuito de familiarizar os estudantes com a dinâmica de resolução dos roteiros que exigia dos estudantes uma abordagem de investigação, a fim de entender e explorar o problema, organizar os dados, formular conjecturas, implementar o programa e testá-lo para verificar se o mesmo resolvia adequadamente o problema e ainda comunicar a ideia empregada na solução.

Uma prerrogativa para a implementação de uma abordagem de investigação diz respeito à autonomia do estudante para explorar o problema e as possíveis soluções. No

desenvolvimento desses roteiros, observamos que os estudantes apresentavam resistências ao estudo autônomo. Por exemplo, para resolver as tarefas solicitadas, os alunos deveriam buscar por si e com base nos recursos de apoio os subsídios para responder as tarefas solicitadas. Contudo, muitas vezes, os estudantes tentavam pular etapas, tentando obter a resposta diretamente do pesquisador ou de outros colegas que estavam em fases mais avançadas do roteiro de tarefas investigativas, resistindo à exploração do material indicado e ao estudo autônomo. Esta abordagem de ensino que adotamos na pesquisa, muitas vezes, não é uma prática vivenciada pelo estudante no decorrer de sua formação e consideramos aceitável esse tipo de dificuldade, em virtude da não vivência.

Tanto os PCNs quanto a Base Nacional Comum Curricular enfatizam a importância do desenvolvimento e construção da autonomia, bem como do senso crítico pelos estudantes durante toda a Educação Básica, que deve oferecer condições para o desenvolvimento de seus estudantes (BRASIL, 2002; BRASIL, 2018). Da mesma forma, Ponte et.al (2016) destacam que, em uma abordagem investigativa, cabe ao professor desenvolver tarefas que estimulem o raciocínio e contribua com o desenvolvimento da autonomia e da cooperação entre os estudantes. Dessa forma, no decorrer da aplicação da pesquisa, envolvemos os estudantes em tarefas dentro e fora de sala de aula, que poderiam ser desenvolvidas em grupo. As tarefas possuíam recursos de apoio, como vídeos, tutorias e exemplos de código que poderiam ser utilizados pelos estudantes como suporte ao estudo autônomo. Em alguns roteiros, inserimos dicas de acesso a cursos *on-line* direcionados ao contexto da pesquisa e que auxiliariam os estudantes do início ao final da resolução dos roteiros.

Observamos também que era uma dificuldade comum entre os estudantes compreender o problema, o que era solicitado na tarefa e, conseqüentemente, organizar os dados para a iniciar o estudo de uma solução. Uma das fases para a resolução de problemas é a compreensão do mesmo, onde o estudante deve ser capaz de identificar suas partes principais e incógnitas, por exemplo (POLYA, 1995).

Para mitigar esta dificuldade dos estudantes, em sala de aula e nos horários de atendimento individual, utilizamos a dinâmica de ler em voz alta o enunciado do problema em conjunto com o aluno e também “pensamos em voz alta” em possibilidades de estruturar o problema, organizar os dados e encaminhar uma solução. Ao pensar em voz alta, o estudante tinha a possibilidade de ver, na prática, um exemplo de como ele próprio poderia proceder, além de ajudar o estudante a refletir sobre o problema. Tal estratégia foi utilizada sempre que necessária.

O professor, ao pensar em voz alta (em inglês *Think aloud*) auxilia o estudante a compreender os processos mentais utilizados na resolução do problema. Ao utilizar esta estratégia, o professor se questiona, pensando em voz alta, e inicia o processo de transferência da responsabilidade pelo uso da estratégia para o estudante (SILBEY, 2018). Também utilizada na investigação matemática, pensar em voz alta auxilia o estudante a elaborar e testar suas conjecturas na busca pela solução de uma tarefa (PONTE et al, 2016).

Nos roteiros, havia tarefas que requeriam dos alunos a comunicação por escrito da estratégia de pensamento empregada para resolução dos problemas. Expressar argumentações sobre o raciocínio empregado em dada resolução é uma habilidade requerida pelos estudantes, quando se trata da disciplina de Matemática, explícita no PCN+, quando estabelece da importância da comunicação na competência denominada comunicar e representar (BRASIL 2002). A comunicação é uma importante habilidade que deve ser motivada adequadamente. Para tanto, é necessário que o professor crie atividades nas quais, para respondê-las, os estudantes devem se expressar de forma oral e/ou escrita (DANIELSON; MARQUEZ, 2016). Com o desenvolvimento dos roteiros, a habilidade de comunicação era prevista e notamos que os estudantes apresentaram dificuldades em expressar sistematicamente o que ele, de fato, utilizou para resolver a tarefa proposta. A Figura 8 exemplifica as respostas de um estudante frente aos questionamentos feitos pelo pesquisador.

Figura 8. Exemplo de resposta às tarefas investigativas

Fragmento de uma tarefa investigativa referente ao RTI02
<p>1.4) Abra e execute o script3R02 , disponíveis na pasta de scripts. Insira os valores das variáveis, conforme disposto na equação que você montou no item 1.0 (<math>a=1</math>, <math>b=2</math>, <math>c=-15</math>) e responda:</p> <p>a) Qual a diferença entre os script3R02 e script2R02? Detalhe sua resposta, referenciando as linhas e o que cada um dos comandos executam.</p>
Resposta do Estudante Est10
<p>1.4)</p> <p>a) Em um script aparece valor de delta e no outro não, no script 3R02 acrescenta-se novos comandos</p> <p>b) no script entre a linha 11 e 15.</p>

Figura 9. Script2R02 fornecido para os estudantes

```

script2R02.py
1  '''Roteiro 02 Item 1.3 '''
2
3  import numpy # Importando Módulos e Bibliotecas
4  import math
5
6  a=float(input("Insira o valor de a:")) # Interação com o usuário, insere
5os valores e converte em float, necessário para trabalhar com números reais.
7  b=float(input("Insira O valor de b:"))
8  c=float(input("Insira o valor de c:"))
9
10 d= math.pow(b,2)-(4*a*c) # aplicação da fórmula para obtenção de delta, com
5auxílio da função math.pow na obtenção do quadrado da variável, armazena o
5valor de delta na variável d
11
12 r=math.sqrt(d) # Utilizando a Função math.sqrt, calcula a raiz de Delta e
5armazena na variável r
13 x1=(-b+r)/(2*a) # calcula o primeiro valor de x e armazena na variável x1
14 x2=(-b-r)/(2*a) # calcula o segundo valor de x e armazena na variável x2
15 print ("Raiz X1= %.2f" % x1) # imprime a raiz x1, considerando 2 casas
5decimais
16 print ("Raiz X2= %.2f" % x2) # imprime a raiz x2, considerando 2 casas
5decimais

```

Fonte: Elaboração própria com WingIde

Figura 10. Script3R02 fornecido para os estudantes

```

script3R02.py
1  ''' Roteiro 02 Item 1.4 '''
2
3  import numpy # Importando Módulos e Bibliotecas
4  import math
5
6  a=float(input('Insira o valor de a:')) # Interação com o usuário, insere
5os valores e converte em float
7  b=float(input('Insira O valor de b:'))
8  c=float(input('Insira o valor de c:'))
9
10 d= math.pow(b,2)-(4*a*c) # aplicação da fórmula para obtenção de delta, com
5auxílio da função math.pow na obtenção do quadrado da variável, armazena o
5valor de delta na variável d
11 if d<0 : # verifica se delta é positivo, caso seja negativo
5imprime a mensagem e o valor de delta, considerando duas casas decimais %.2f
12     print ('Delta negativo= %.2f' % d +', impossível extrair raizes no
5conjunto dos números reais')
13
14 else :
15     print ('Delta: %.2f.' % d) # Se o Delta for positivo, imprime o
5valor de Delta
16     r=math.sqrt(d) # Utilizando a Função math.sqrt, calcula a raiz de
5Delta e armazena na variável r
17     x1=(-b+r)/(2*a) # calcula o primeiro valor de x e armazena na
5variável x1
18     x2=(-b-r)/(2*a) # calcula o segundo valor de x e armazena na
5variável x2
19     print ('Raiz X1= %.2f' % x1) # imprime a raiz x1, considerando 2
5casas decimais %.2f
20     print ('Raiz X2= %.2f' % x2) # imprime a raiz x2, considerando 2
5casas decimais %.2f

```

Fonte: Elaboração própria com WingIde

No enunciado ilustrado na Figura 8, a tarefa consistia em fazer com que o estudante explorasse os *scripts* (Figuras 9 e 10), descobrindo a função dos comandos e sua importância no código. Vale ressaltar que todos os *scripts* possuíam comentários<sup>48</sup> após os comandos nas linhas de código e caberia ao estudante analisar os scripts, comparar os códigos e comentários

<sup>48</sup> Comentário, em programação, são textos inseridos nos *scripts* com o intuito de explicar o que é feito em determinada linha ou blocos de linhas do *script*/programa, facilitando o entendimento dos comandos utilizados na construção do programa. Em Python, comentários podem ser inseridos após o símbolo # (cerquilha).

e em seguida responder o que foi solicitado na tarefa. Uma explicação completa e correta para a tarefa é apresentada no Quadro 14.

Quadro 15. Exemplo de uma resposta correta para a atividade da Figura 7

1.4) a) No Script2R02, percebemos que não há comandos para verificar o valor de delta, sendo assim, se o delta for positivo o programa continua sendo executado, mas se for negativo será impresso em tela um erro, ficando a cargo do usuário identificá-lo.  
 Já no Script3R02, foram inseridos novos comandos, iniciando na linha 11 (logo após o cálculo do valor de delta) com o *if*, que verifica o valor atribuído a delta e se esse valor for negativo, o comando na linha 12 informa ao usuário: “Delta Negativo, impossível extrair raízes dentro do conjunto dos números reais”. A linha 14 possui o comando *else*, onde, se o valor de delta for positivo, há um encaminhamento no programa que conduz, a impressão em tela do valor de delta e na sequência o cálculo das raízes.

Fonte: Elaboração própria

Conforme pode ser observado no Quadro 14, a comparação deveria levar em consideração a inserção da estrutura condicional *if* e *else*, onde o *if* verifica se uma condição é verdadeira, ou seja, se o valor de delta é negativo, conduzindo o programa para executar as linhas seguintes e o *else* é responsável por informar ao programa que, caso a condição verificada no *if* seja falsa, as linhas seguintes serão executadas.

Na resposta do estudante (Figura 8), há indicações de que o mesmo não se atentou a estrutura condicional (*if/else*) que diferencia os dois programas e, que no caso do script3R02, esta estrutura é responsável por estabelecer restrições para o cálculo das raízes reais a partir do valor de delta. Quando feita à mão, essa operação pode ser feita implicitamente pelos estudantes, mas quando se constrói um código, elas devem ser tornadas explícitas para que o programa execute corretamente. Algumas percepções do pesquisador anotadas no diário de bordo, no que se refere ao estudante Est10, apontam que frequentemente o estudante deixou de assistir aos vídeos de apoio e de ler os tutoriais de suporte que tinham como objetivo auxiliá-lo, postura que pode justificar o fato de não atender corretamente o que foi solicitado no problema.

Embora o estudante tenha esboçado uma resposta (Figura 8), ele não conseguiu comunicar de forma completa e correta o impacto das mudanças nos códigos. Essa dificuldade foi verificada em 60% dos estudantes, ao responderem à mesma tarefa. Em nossa pesquisa, isto traz indicações de que é importante trabalhar a capacidade de expressão e comunicação escrita dos estudantes, porque eles como futuros professores precisam ter esta habilidade para auxiliar seus alunos. Do ponto de vista da pesquisa, estas respostas, quando dadas de forma tão incompleta, sinalizavam também possíveis dificuldades de entendimento e da necessidade de reforçar a aprendizagem.

À medida em que os estudantes desenvolviam as tarefas propostas nos três primeiros roteiros, quanto à automatização de resolução de equações algébricas, os mesmos demonstraram ser capazes de prover soluções utilizando a programação de computadores para a resolução dos problemas. Os programas desenvolvidos pelos estudantes possuíam complexidades diferentes e o nível de dificuldade era incrementado em cada roteiro, incidindo nas linhas de código e comandos necessários para a construção da solução automatizada, conforme exemplo representado nas Figuras 11 e 12.

Figura 11. Scripts construídos pelo Est03 nos RTIs 01 e 02.

(a)

```

1 import numpy
2 import math
3
4 #equação: 2*x + 43 = 0  a*x + b = c
5
6 a=2
7 b=43
8 c=0
9 x=(c-b)/a
10 print(x)

```

(b)

```

1 import numpy
2 import math
3
4 a=float(input('informe, por favor o valor de a:'))
5 b=float(input('informe, por favor o valor de b:'))
6 c=float(input('informe, por favor o valor de c:'))
7
8 d= math.pow(b,2)-(4*a*c)
9 r=math.sqrt(d)
10 x1=(-b+r)/(2*a)
11 x2=(-b-r)/(2*a)
12 print ('Raiz X1= %.2f' % x1)
13 print ('Raiz X2= %.2f' % x2)

```

Fonte: Respostas dos estudantes às tarefas

Figura 12. Scripts construídos pelo Est03 nos RTIs 02 e 03

(a)

```

1 import numpy
2 import math
3
4 a=float(input ('Escolha um valor para a:'))
5 b=float(input ('Escolha um valor para b:'))
6 c=float(input ('Escolha um valor para c:'))
7
8 d= math.pow(b,2)-(4*a*c)
9 if d<0 :
10     print ('Delta negativo= %.3f' % d +', impossível
11     extrair raízes no conjunto dos números reais')
12 else :
13     print ('Delta: %.3f.' % d)
14     r=math.sqrt(d)
15     x1=(-b+r)/(2*a)
16     x2=(-b-r)/(2*a)
17     print ('Raiz X1= %.3f' % x1)
18     print ('Raiz X2= %.3f' % x2)

```

(b)

```

1 from pylab import *
2 import matplotlib.pyplot as plt
3 import numpy
4 import math
5
6 a=float(input ('Dê um valor para a:'))
7 b=float(input ('Dê um valor para a:'))
8 c=float(input ('Dê um valor para a:'))
9
10 d= (b*b)-(4*a*c)
11 if d<0 :
12     print ("Delta negativo= %.2f" % d +', impossível
13     extrair raízes em Reais")
14 else:
15     print ("Delta: %s." % d)
16     r= d**(1/2)
17     x1=(-b+r)/(2*a)
18     x2=(-b-r)/(2*a)
19     print ("Raiz X1= %.2f" % x1)
20     print ("Raiz X2= %.2f" % x2)
21
22 # calculando vértice
23 xv= -b/(2*a)
24 yv= -d/(4*a)
25 print ('Vertice xv= %.2f' % xv)
26 print ('Vertice yv= %.2f' % yv)

```

Fonte: Respostas dos estudantes às tarefas

Na figura 11(a), apresentamos o código do estudante Est03 referente à automatização da solução de uma equação do primeiro grau, na qual o programa calcula o valor de  $x$ . Nesse *script*, há a versão mais simples do código, seguindo apenas uma estrutura de controle. Portanto, não há restrições sobre valores, isto é, o estudante não utiliza estrutura de seleção (if/else) para

verificar se o coeficiente de  $x$  é diferente de zero. Ainda, nesse *script*, não há leitura de dados via teclado, ou seja, os valores são atribuídos às variáveis diretamente no *script*.

O programa desenvolvido pelo estudante, representado na Figura 11(b), apresenta uma solução para automatização de uma equação do segundo grau, na qual o código é mais complexo, há leitura de dados via teclado e inclusão de estruturas para a obtenção do valor de delta e sua raiz utilizando recurso da biblioteca *math*, dispostas nas linhas 8 e 9 do programa. Ainda nesse programa, o estudante foi capaz de formatar adequadamente, com duas casas decimais, a impressão em tela dos valores possíveis para  $x$ , dispostos nas linhas 12 e 13. Ao se deparar com uma equação do segundo grau, que por si só possui um nível de complexidade maior do que uma equação do primeiro grau, o estudante pode até organizá-la e resolvê-la mentalmente, mas, para automatizar a solução utilizando a programação, é necessário que o raciocínio empregado seja transcrito, de forma lógica, em linguagem de programação.

O programa representado pela Figura 12(a) apresenta um aprimoramento do programa 11(b). Nele, o estudante propôs uma solução para uma equação do segundo grau, onde o valor de delta é criticado. Com um grau de complexidade maior, o estudante implementou uma estrutura de decisão, composta pelos comandos *if* e *else*, responsáveis, nesse *script*, por verificar se o valor de delta é negativo e encaminhar o programa para a execução das próximas linhas, de acordo com o resultado dessa verificação (delta positivo ou negativo). Estruturas de decisão envolvem um ou mais testes lógicos, que serão verdadeiros ou falsos, sendo seguidos de comandos que realizarão determinadas ações de acordo com o resultado. Para que o estudante empregue corretamente os comandos de estrutura de decisão, como *if* e *else* (linhas 9 e 12 respectivamente do *script*), é necessária uma organização mental prévia sobre a estrutura necessária na solução e posterior habilidade para a escrita na linguagem de programação. Na formatação de impressão, linhas 10, 13, 17 e 18, o estudante foi capaz alterar o número de casas decimais para três, o que pode demonstrar entendimento em algumas variações na sintaxe da função *print*.

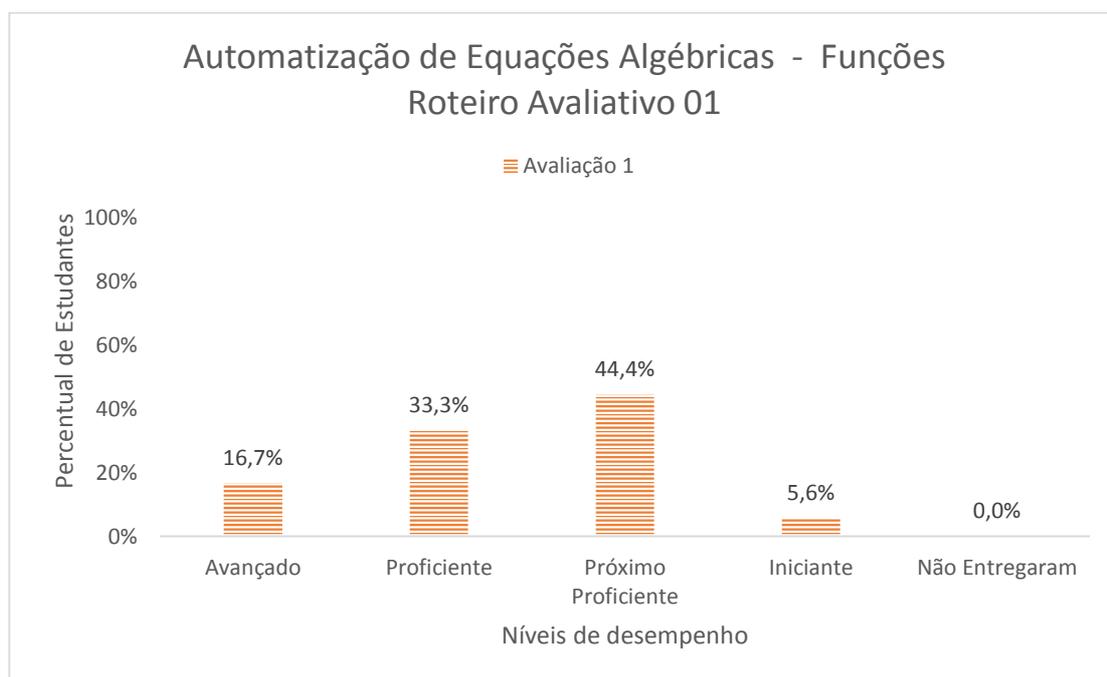
A Figura 12(b) apresenta uma solução para uma tarefa, pertencente ao terceiro roteiro de tarefas investigavas, na qual foi solicitado ao estudante automatizar o cálculo para obter os valores de delta, das raízes da função e coordenadas do vértice de uma parábola pertencente à uma função do segundo grau. Nesse programa, podemos observar que o estudante utilizou maneiras diferentes, das utilizadas nos artefatos anteriores, para calcular o valor de delta e extrair sua raiz quadrada (linhas 10 e 16 respectivamente).

As respostas representadas pelas Figuras 11 e 12 demonstram um crescimento gradativo, no qual os estudantes iniciam com um código de menor complexidade e no decorrer

das tarefas, vão resolvendo problemas que exigem códigos mais complexos. Isso fica demonstrado, por exemplo, pela utilização da biblioteca *math* nas linhas 8 e 14 da Figura 12(a), que calcula a potenciação e a raiz quadrada de determinado número e pela variação de comandos de programação que o estudante utilizou para realizar cálculos semelhantes (potenciação e raiz quadrada) nas linhas 10 e 16 da Figura 12(b).

Após o término dos três primeiros roteiros, aplicamos o roteiro avaliativo 01, com o intuito de verificar se os estudantes eram capazes de resolver as tarefas propostas, utilizando programação de computadores para automatizar a resolução de cálculos pertencentes às funções do segundo grau, como a obtenção dos zeros da função e coordenadas dos vértices. O nível de dificuldade das questões da avaliação foi compatível com o que fora empregado no desenvolvimento dos roteiros, porém, durante a avaliação, os estudantes ficaram limitados à utilização dos recursos de apoio que a eles disponibilizamos, como pequenos trechos de código, sem possibilidade de interação com outros colegas. O desempenho dos estudantes no roteiro avaliativo 01 está representado na Figura 13.

Figura 13. Desempenho no Roteiro Avaliativo 01 - Funções



Fonte: Elaboração própria

Conforme pode ser observado na Figura 13, 16,7% dos estudantes foram classificados no nível de aprendizagem avançado, pois apresentaram soluções estruturadas de forma correta e diferentes dos exemplos abordados nos roteiros, demonstrando ser capazes de construir programas de maneira mais autônoma. Já 33,3% dos estudantes construíram programas que

respondiam adequadamente às tarefas solicitadas na avaliação, porém, em seus *scripts*, nos deparamos com os códigos organizados de forma semelhante aos desenvolvidos durante os roteiros. As soluções dos estudantes classificados com desempenho próximo de proficiente, 44,4%, continham erros em comum. Os maiores equívocos estavam na organização da crítica ao valor de  $a$  e ou ao valor de  $\Delta$ , tendo os testes lógicos e estruturas condicionais (*if/else*) implementadas de maneira que comprometiam a execução correta do programa.

A tarefa proposta na avaliação, referente à automatização de resolução de equações algébricas no contexto de funções, solicitava que o estudante desenvolvesse um código para automatizar a resolução de cálculos envolvendo funções do segundo grau e que atendesse aos seguintes requisitos:

- i) Solicitar do usuário os valores das variáveis;
- ii) Verificar o valor da variável  $a$  e caso seja positivo, imprimir seu valor e informar ao usuário que o programa só resolve equações cujo valor de  $a$  seja menor do que zero;
- iii) Calcular, verificar e imprimir o valor de  $\Delta$ , se negativo ou positivo, com duas casas decimais. Não esquecendo de que o valor de  $\Delta$  também deverá ser verificado;
- iv) Calcular e imprimir as raízes da função e coordenadas do vértice. Com duas casas decimais;

Para demonstrarmos o atendimento dos itens i ao iv, apresentamos o programa construído pelo estudante Est16, classificado como avançado, ilustrado por meio da Figura 14.

Figura 14. Resposta do estudante Est16 a tarefa da avaliação

```

1  from pylab import *
2  import matplotlib.pyplot as plt
3  import numpy # Importando Módulos e Bibliotecas
4  import math
5
6  a=float(input("Diga-me o valor de a: "))
7  b=float(input("Diga-me o valor de b: "))
8  c=float(input("Diga-me o valor de c: "))
9
10 if a>0:
11     print('Variável a é maior que zero, não será executado
12     pelo programa. Ele vale: %.2f' % a)
13
14 else:
15     delta= (b*b)-(4*a*c)
16
17     if delta<0 :
18         print ('Impossível extrair raízes em Reais,
19         delta negativo: %.2f' % delta)
20
21     else:
22         print('Delta é igual a: %.2f ' % delta)
23         r= (delta**(1/2))
24         x1= (-b+r)/(2*a)
25         x2= (-b-r)/(2*a)
26         print('X1 é igual a: %.2f' %x1)
27         print('X2 é igual a: %.2f' %x2)
28
29         # calculando e imprimindo vértice
30         xv= (-b)/(2*a)
31         yv= (-delta)/(4*a)
32         print('Coordenada x do vértice: %.2f' % xv)
33         print('Coordenada y do vértice: %.2f' % yv)

```

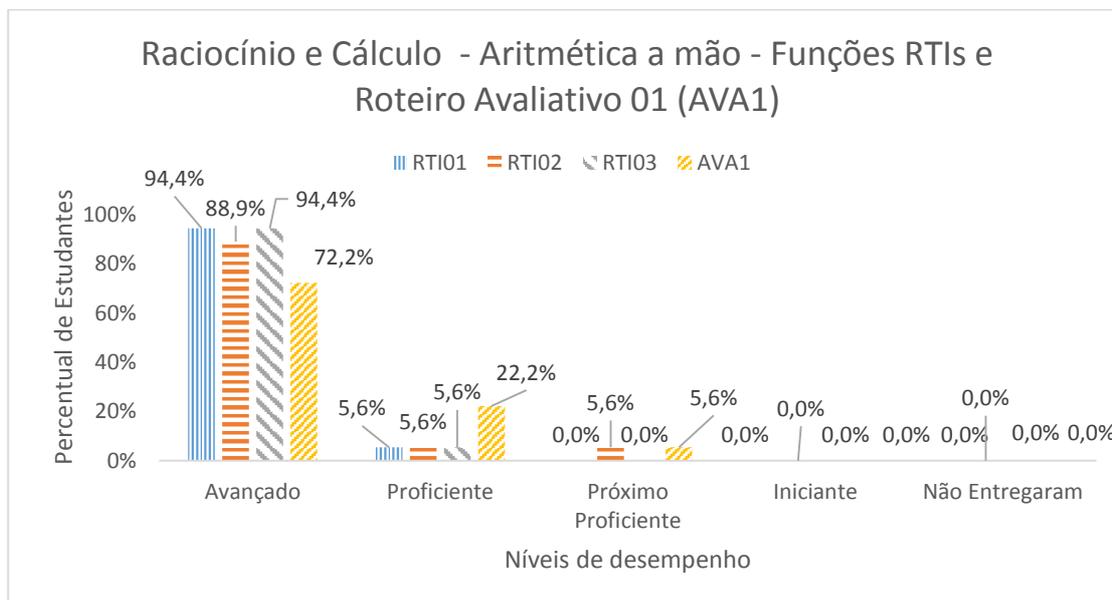
Fonte: Resposta do estudante.

O item i) foi implementado pelo estudante conforme descrito entre as linhas 6 e 8 da Figura 14. Na linha 10, o estudante iniciou um teste lógico utilizando o *if* para verificar o valor da variável *a*, atendendo ao item ii) da avaliação. Dependendo do valor da variável *a*, o programa conduz a execução para a linha 13, que verifica o valor de delta e, se for positivo, o programa encaminha a execução dos comandos inseridos dentro no *else*, localizado na linha 19 da Figura 14. A implementação efetuada pelo estudante entre as linhas 13 e 25 atendem o solicitado no item iii) da tarefa. Ao calcular e imprimir as coordenadas do vértice, comandos inseridos entre as linhas 28 e 31 da Figura 14, o estudante implementa a solução do item iv) e encerra o programa resolvendo corretamente tudo que foi solicitado na tarefa. Logo no início do script, linhas 6,7 e 8, a estudante cria seu próprio texto para solicitar a entrada de dados via teclado. Da mesma forma, a organização da estrutura condicional, *if/else*, também é criação do estudante, que não se prendeu aos exemplos utilizados nos roteiros e nas formas apresentadas para solução de problemas similares, demonstrando autonomia, o que contribuiu com a classificação do estudante no nível de desempenho avançado.

Até este momento, discutimos aspectos relacionados ao desempenho dos estudantes na construção de programas, para automatizar resoluções de cálculos no contexto de funções do segundo grau. Quando comparamos os resultados do roteiro avaliativo 1 (Figura 13) com os resultados da avaliação diagnóstica, em que 100% dos estudantes tiveram seu desempenho categorizado como iniciante, observamos uma evolução dos estudantes na aprendizagem, concentrando-se nos níveis proficiente e próximo de proficiente.

Os estudantes também melhoraram seu desempenho, quando se trata de calcular a mão. Como demonstra a Figura 15, na avaliação 1, mais de 70% foram classificados no nível de aprendizagem avançado, apresentando um crescimento quando comparada à avaliação diagnóstica em que nenhum estudante teve seu desempenho classificado como avançado e 67% estavam como proficientes. Suspeitamos que a melhora no desempenho possa ser efeito dos roteiros de tarefas investigativas que solicitavam resolução à mão, também a própria atividade de programar exige que os estudantes compreendam como se faz à mão, para posteriormente converterem em linguagem de programação.

Figura 15. Resultado dos RTIs e Roteiro Avaliativo 01 referentes ao raciocínio e cálculo à mão – Funções do segundo grau



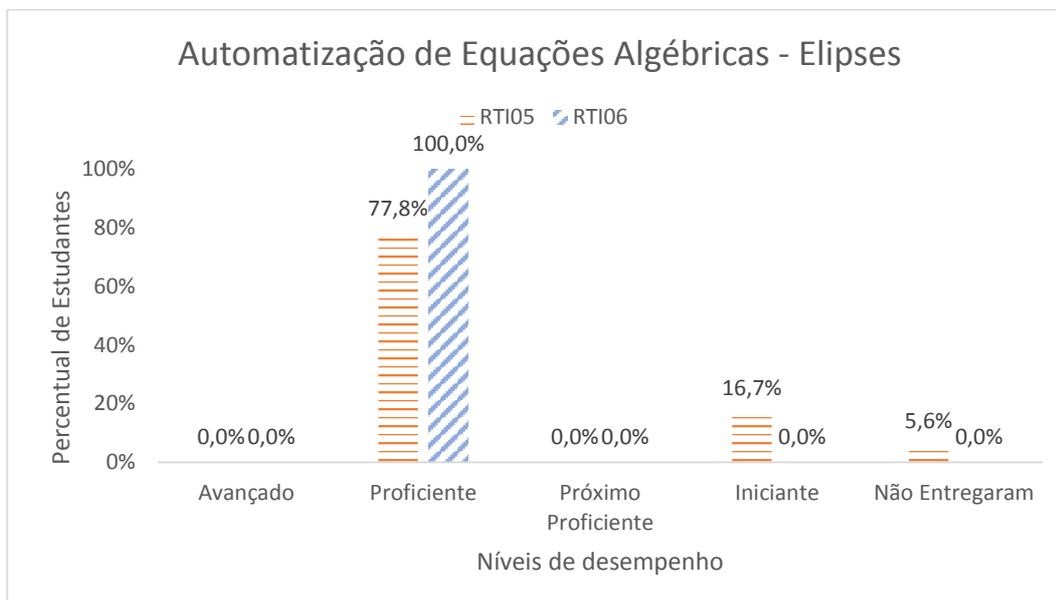
Fonte: Elaboração própria

Os estudantes que tiveram seu nível de desempenho classificado como proficiente são aqueles que apresentaram alguns erros na álgebra, necessária para a resolução do que foi solicitado na tarefa.

Quanto aos conteúdos de elipses, estes foram abordados nos roteiros 4, 5 e 6, sendo avaliados por meio do roteiro avaliativo 2. Considerando que todos os estudantes afirmaram não ter estudado elipses anteriormente, o roteiro 4 foi preparado com atividades estritamente matemáticas, com o objetivo de direcionar os estudos para os conteúdos de elipses, onde as atividades foram realizadas à mão, sem perder de vista a postura investigativa. Durante todo o percurso da aplicação, tanto do roteiro 4 quanto dos roteiros seguintes, demos ênfase às explicações matemáticas pertencentes às elipses, de acordo com as demandas dos estudantes, utilizando o quadro branco e material de apoio, como tutoriais e vídeos. Cabe destacar que, nessa fase da aplicação, relacionada à conteúdos de elipses, as dúvidas dos estudantes estavam mais concentradas na parte matemática, do que na programação necessária para implementação da solução.

A automatização de resolução de equações algébricas referente à elipse esteve presente nos roteiros 5, 6 e no roteiro avaliativo 02, tendo os resultados sumarizados por meio do gráfico exposto pela Figura 16.

Figura 16. Resultado dos RTIs referentes à automatização de resolução de equações algébricas – Elipses

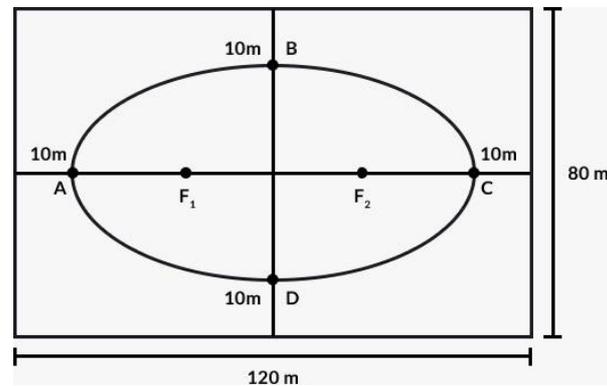


Fonte: Elaboração própria

Quanto à automatização da resolução de cálculos relacionados às elipses, presente nos roteiros 4 e 5, os dados revelados pelo gráfico na Figura 16 mostram que, no roteiro de tarefas investigativas 5, mais de 75% dos estudantes foram classificados no nível de aprendizagem proficiente, demonstrando habilidades para reutilizar os códigos fornecidos como material de apoio e adaptá-los na resolução das tarefas solicitadas no roteiro. Os estudantes que não lograram êxito na construção de seus códigos foram classificados como iniciante, representando 16,7%. Os 5,6% representam um estudante que faltou no dia da aplicação do roteiro e não entregou a atividade em oportunidade posterior.

Na aplicação desses roteiros, utilizamos a mesma dinâmica proposta na pesquisa, adequando nossa intervenção às necessidades dos estudantes. Uma típica tarefa pertencente aos roteiros que envolviam assuntos de elipses era fazer com que o estudante automatizasse o cálculo para a obtenção de elementos de elipses, tais como dimensão dos eixos e semieixos, coordenadas dos focos, distância focal e excentricidade. Uma das tarefas propostas no roteiro de tarefas investigativas 5 apresentava uma praça em um formato retangular, medindo 80m x 120 m, onde, em seu centro, deveria ser construído um jardim em formato de uma elipse, e os pontos marcados por F1 e F2 representam os locais onde seriam inseridos os postes de iluminação da praça, conforme ilustra a Figura 17.

Figura 17. Praça



Fonte: Roteiro de Tarefas Investigativas 5

Dadas as informações representadas por meio da Figura 17, os estudantes deveriam construir um programa com as seguintes características:

- i) Solicitar do usuário os valores das variáveis, de acordo com a equação reduzida da elipse em questão;
- ii) Calcular e imprimir a distância entre os dois postes;
- iii) Calcular e imprimir a excentricidade da praça.

Para responder aos itens i), ii) e iii), apresentamos o programa construído e apresentado como resposta à tarefa pelo estudante Est07, classificado no nível de desempenho proficiente, representado por meio da Figura 18.

Figura 18. Programa do estudante Est07 para responder aos itens i), ii) e iii).

```

1  '''Atividade 1.5'''
2  import numpy # Importando Módulos e Bibliotecas
3  import math
4
5  a=float(input('Digite o valor de a: '))
6  b= float(input('Digite o valor de b: '))
7  x0=float(input('Digite o valor de x0: '))
8  y0=float(input('Digite o valor de y0: '))
9
10 c= (a**2 - b**2)**0.5
11 foco1= (a**2 - b**2)**0.5
12 foco2= -(a**2 - b**2)** 0.5
13 dfocal= 2*c
14 x= foco1/a
15
16 print ("A excentricidade é: %.2f" % x)
17 print ('A distancia entre os dois postes é de: %.2f' % dfocal)

```

Fonte: Programa do estudante Est07

Com base nos exemplos trabalhados durante a aplicação da pesquisa, o estudante implementou por meio de programação de computadores a solução da tarefa solicitada no roteiro. Ao observarmos a Figura 18, notamos que o item i) foi respondido a partir da construção

das linhas de códigos inseridos pelo estudante nas linhas 5 a 8. Entre as linhas 10 e 14, o estudante implementou os cálculos solicitados nos itens ii) e iii), de forma sequencial. As impressões dos valores, devidamente formatadas com duas casas decimais, foram implementadas nas linhas 16 e 17, atendendo integralmente ao que foi solicitado na tarefa.

Nesses roteiros, os estudantes conseguiam empregar conhecimentos adquiridos nos roteiros anteriores, tendo menos dificuldades em estruturar o código, que, nesse contexto, apresentavam um grau de complexidade menor quanto a programação. Todo o arcabouço da lógica de programação empregada, assim como os operadores aritméticos, foi importado da experiência com os primeiros roteiros desenvolvidos pelos estudantes. É importante destacar que algumas observações anotadas em nosso diário de bordo mostram que, nesse momento da aplicação, roteiros 5 e 6, alguns estudantes eram capazes de implementar corretamente soluções como as demonstradas na Figura 18, sem a necessidade de recorrer aos exemplos de códigos, como é o caso do estudante Est07.

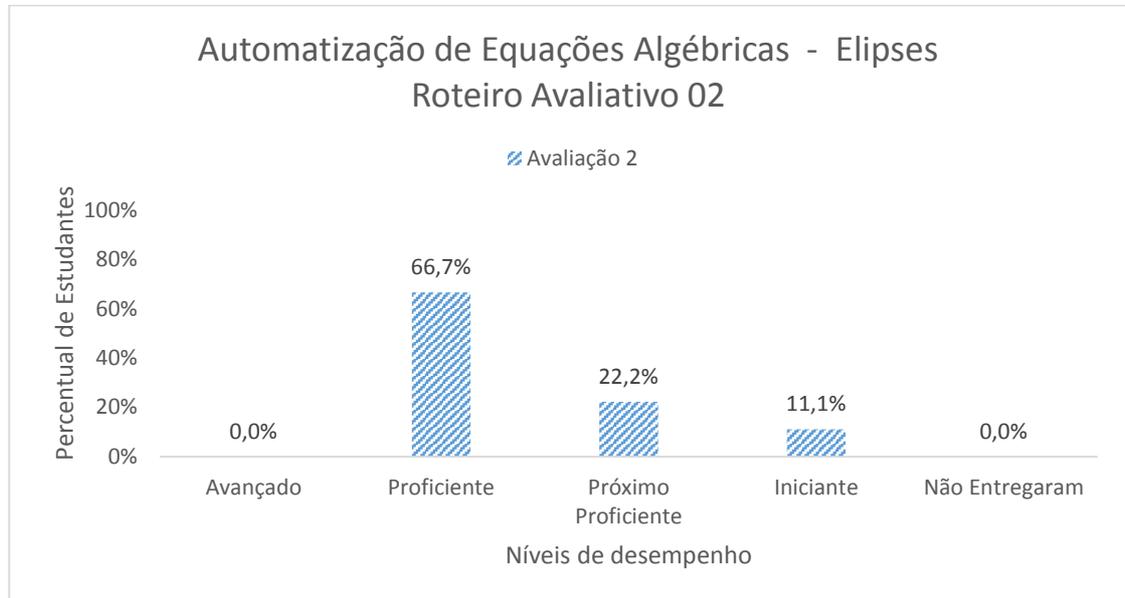
Os estudantes cujo desempenho nos roteiros é iniciante são aqueles que normalmente chegavam atrasados tanto nas aulas quanto nas avaliações. Cabe ressaltar que nenhum deles dispunha de computador em casa e só acessavam a internet pelo celular, conforme dados revelados pelo questionário socioacadêmico. Ainda, nenhum deles conseguiu participar do atendimento aos estudantes (extraclasse), alegando incompatibilidade de horários, e afirmaram não terem estudado fora de sala. Na tentativa de amenizar o problema e atender a este e a outros casos semelhantes, agendamos quatro horas de atendimento em um sábado, no período vespertino, porém, mesmo assim, nenhum deles compareceu.

Respeitando a mesma dinâmica dos roteiros anteriores, os estudantes foram provocados a comunicar por escrito a estratégia utilizada na resolução das tarefas investigativas, explicando o como e o porquê de suas respostas. Corroborando a análise feita anteriormente, os estudantes foram capazes de construir os artefatos para automatizar suas soluções, mas, no momento de explicarem o seu raciocínio, as respostas são abreviadas, incompletas e/ou ausentes, como deixar de responder ao questionamento de por que presente nas tarefas.

Ao final dos últimos de três roteiros de tarefas investigativas, roteiros 4, 5 e 6, aplicamos o roteiro avaliativo 02 que tinha como um dos objetivos verificar se os estudantes eram capazes de prover soluções para automatizar a resolução de cálculos, necessários para resolver problemas matemáticos no contexto de elipses. Na elaboração da avaliação, as tarefas foram planejadas obedecendo ao mesmo grau de dificuldade das resolvidas pelos estudantes durante a execução dos roteiros. A dinâmica da aplicação da segunda avaliação foi semelhante à utilizada na primeira, ou seja, os estudantes tinham ao seu dispor pequenos trechos de código

que poderiam nortear a implementação de suas soluções. Os resultados da segunda avaliação são apresentados na Figura 19.

Figura 19. Resultado do Roteiro Avaliativo 02 referente à automatização de resolução de equações algébricas - Elipses



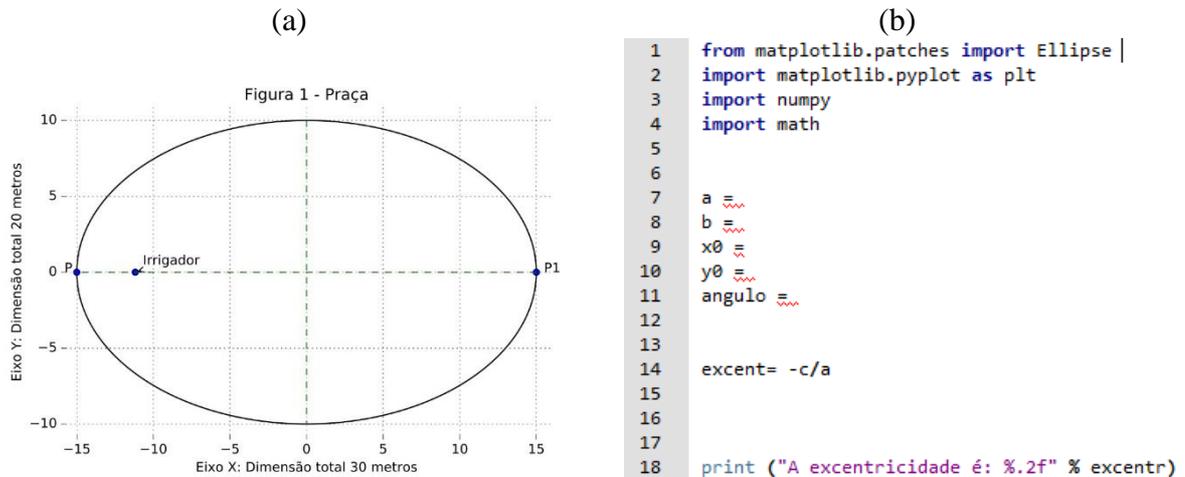
Fonte: Elaboração própria

Podemos observar, na Figura 19, que mais de 65% dos estudantes implementaram soluções que respondiam adequadamente, conforme critérios estabelecidos na rubrica, ao que fora solicitado na tarefa, sendo classificados como proficientes, isto é, foram capazes de prover soluções baseadas nos exemplos de códigos e tarefas dos roteiros desenvolvidos. A tarefa proposta na avaliação solicitava aos estudantes que desenvolvessem um programa para resolver um problema no contexto de elipse, que atendesse aos seguintes requisitos:

- i) Solicitar ao usuário o valor das variáveis;
- ii) Calcular e imprimir a distância entre o irrigador e os pontos P e P1 na praça;
- iii) Calcular e imprimir a excentricidade da praça.

Para auxiliar os estudantes, fornecemos o enunciado completo e um gráfico representando a praça e seus elementos, conforme pode ser observado na Figura 20(a). Como recurso de apoio, fornecemos um trecho de código (incompleto), representado pela Figura 20(b).

Figura 20. Representação da praça (a) e script de apoio (b)



Fonte: Roteiro avaliativo 2

Como resposta aos itens i), ii) e iii), o estudante Est07, classificado no nível de aprendizagem proficiente, apresentou o programa conforme ilustrado na Figura 21.

Figura 21. Representação de solução implementada pelo estudante Est07 em resposta à tarefa do roteiro avaliativo 2

```

1  from matplotlib.patches import Ellipse # Importando módulos e bibliotecas
2  import matplotlib.pyplot as plt
3  import numpy
4  import math
5
6
7  a=float(input('Digite o valor de a: '))
8  b=float(input('Digite o valor de b: '))
9  x0=float(input('Digite o valor de x0: '))
10 y0=float(input('Digite o valor de y0: '))
11 angulo = 360
12
13  c=((a**2-b**2)**0.5)
14  excent= c/a
15  foco= ((a**2-b**2)**0.5)
16  p=(a - foco)
17  p1=(foco + a)
18  print ("A excentricidade é: %.2f" % excent)
19  print ("A distancia do Irrigador ate o ponto P é de: %.2f" % p, "metros.")
20  print ("A distancia do Irrigador ate o ponto P1 é de: %.2f" % p1, "metros.")

```

Fonte: Resposta do estudante Est07

O item (i) da tarefa, foi implementado entre as linhas 7 e 10, conforme podemos observar na Figura 20. Para responder ao item (ii), o estudante implementou os comandos dispostos nas linhas 13, 15, 16, 17, 19 e 20. Já o item (iii), que solicita o cálculo e a impressão da excentricidade, foi implementado nas 14 e 18.

Ao compararmos o programa desenvolvido pelo estudante (Figura 21) com o *script* de apoio fornecido (Figura 20 (b)), podemos notar que a solução implementada pelo estudante foi realizada quase que de forma autônoma, demonstrando seu entendimento quanto aos

procedimentos necessários para resolver a tarefa por meio da programação de computadores. Acreditamos que tal resultado justifica-se pelo fato de os estudantes já terem passado por todo o processo de ensino e aprendizagem, tendo a oportunidade de importarem conhecimentos anteriores, como entrada de dados, formatações e operadores lógicos. Ao compararmos o desempenho dos estudantes na avaliação diagnóstica com o roteiro avaliativo 02, percebemos uma significativa melhora, considerando que a maioria dos estudantes deixou o nível de aprendizagem iniciante e passou para proficiente.

Se compararmos os dados do roteiro avaliativo 01, para automatização de resolução de equações para funções e elipses, verificamos que os estudantes progrediram na aprendizagem, tendo seu desempenho mais efetivo concentrado no nível proficiente. Cabe ressaltar que 50% dos estudantes classificados como próximo de proficientes e 100% dos classificados como iniciante pertencem ao grupo de estudantes que não entregavam os roteiros, que chegavam atrasados, que alegavam não terem tempo e estrutura (computador e internet) para estudos extraclasse e que muitas vezes não concluíam as atividades propostas, mesmo em sala de aula. Situações como as destacadas, podem ter contribuído com o número de estudantes classificados como próximo de proficientes no roteiro avaliativo 02, pois ao deixar de realizar atividades em sala e extraclasse, considerando a dinâmica adotada na aplicação dos roteiros, os estudantes comprometiam o avanço no seu aprendizado.

#### 4.3 Detalhamento dos Resultados da Segunda Questão de Pesquisa

Conforme mencionado anteriormente, a segunda questão de pesquisa é: Em que aspectos PBI melhora as habilidades de plotagem de gráficos?

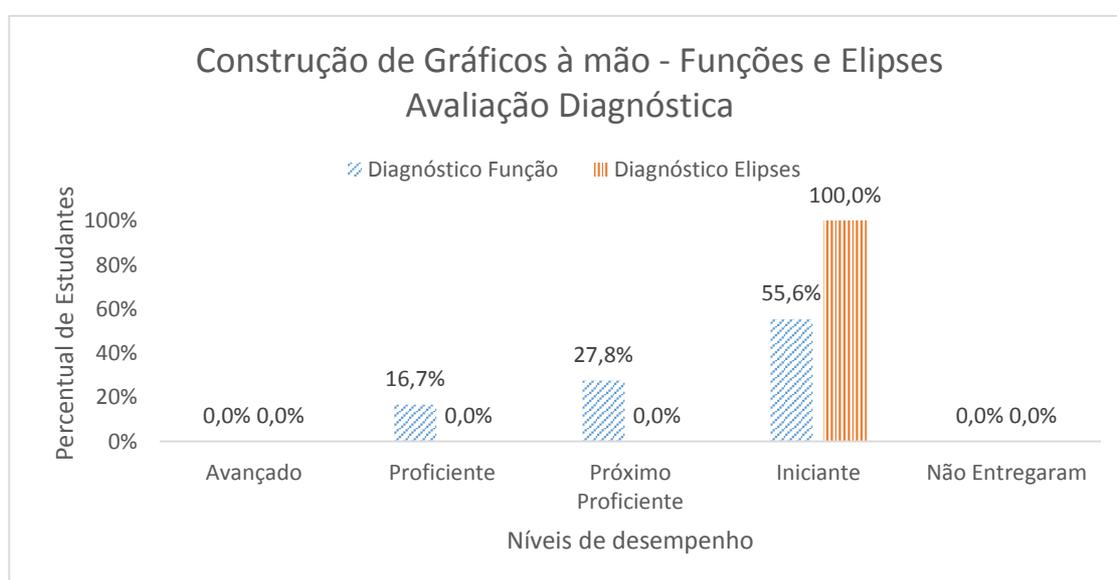
Obedecendo a mesma sistemática utilizada para responder a primeira questão de pesquisa, responderemos a esta, pautados em dados obtidos a partir das respostas dos estudantes na avaliação diagnóstica, nos roteiros de tarefas investigativas (roteiro 3 sobre funções e 6 sobre elipses) e nos roteiros avaliativos 1 e 2.

Conforme mencionado na Seção 3.3.1.3, os conteúdos relacionados a gráficos, abordados em nossa proposta de ensino, fazem parte de assuntos comumente estudados na Educação Básica e, no nosso caso, os gráficos representavam as funções do segundo grau e as equações das elipses presentes nos roteiros de tarefas investigativas. No que diz respeito à automatização de plotagem de gráficos, na avaliação diagnóstica, 100% dos alunos tiveram seu desempenho classificado como iniciante, tanto no contexto de funções do segundo grau, quanto no de elipses. Da mesma forma com o ocorrido na automatização de resolução de equações

algébrica, esse resultado era o esperado, já que a programação de computadores não compõe o currículo da licenciatura em questão.

Mesmo quando as tarefas solicitavam a construção do gráfico à mão, nenhuma das respostas dos estudantes foi classificada no nível de desempenho avançado (vide Figura 22), tanto em funções quanto em elipses, uma vez que, para este nível, os elementos solicitados (para o caso de função, pontos da função, coordenadas do vértice, eixos do plano cartesiano e para o caso de elipses, coordenadas dos focos, do centro e identificação dos eixos) deveriam estar corretamente apresentados.

Figura 22. Diagnóstico da construção e identificação de gráficos à mão no contexto de funções do segundo grau e elipses.



Fonte: Elaboração própria

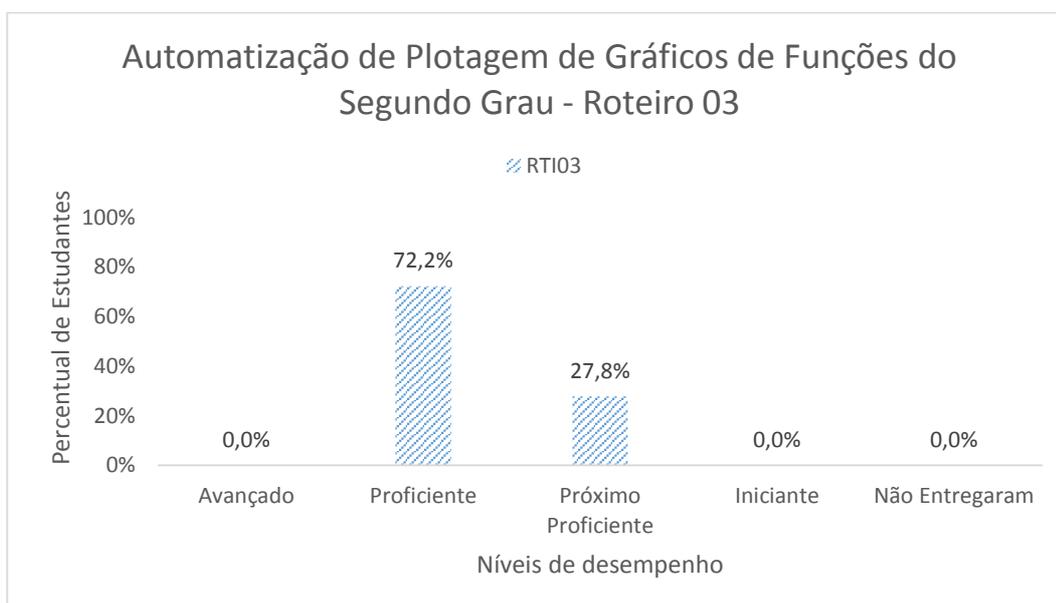
Quanto ao gráfico das funções, 16,7% dos estudantes tiveram seu desempenho classificado como proficiente. Nesse caso, embora esses estudantes tracem corretamente o gráfico, eles deixaram de identificar elementos solicitados na tarefa, como os zeros da função e coordenadas do vértice. Os 27,8% dos alunos classificados no nível de aprendizagem próximo de proficientes são aqueles que cometeram erros ao traçar o gráfico, como a inversão de coordenadas do vértice da parábola, associados a equívocos na identificação dos elementos no gráfico. Apesar do estudo de gráficos de funções do segundo grau pertencer ao ensino médio, a maioria dos estudantes (55,6%) tiveram seu nível de desempenho classificado como iniciante, pois deixaram as tarefas em branco, com vestígios de que iniciaram a construção do gráfico e ou traçaram o plano cartesiano e não avançaram, por exemplo. O estudante deve ser capaz de utilizar diferentes formas de comunicação e representação de suas ideias, e, no contexto desta

pesquisa, destacamos a representação gráfica, presente na competência de representação e comunicação dos PCN+ e também demandadas como competência e habilidade na Matemática, conforme a BNCC, que também destacam a importância do estudo das funções e conseqüentemente a sua representação gráfica (BRASIL, 2002; BRASIL, 2018). O ensino e aprendizagem de funções do segundo grau tem sido objeto de pesquisa em diversos trabalhos, onde os autores corroboram a dificuldade existente na compreensão da representação algébrica e gráfica, tanto na Educação Básica quanto na Superior (MARGARINUS, 2013; MPAKA, 2010; OLIVEIRA, 2006).

Na atividade que solicitava o esboço do gráfico da elipse, os desempenhos dos estudantes foram classificados como iniciantes, corroborando o desempenho obtido na resolução algébrica à mão da equação da elipse, discutida na seção 4.2. O desempenho dos estudantes nos gráficos relacionados a elipses pode ser compreendido pelo fato de relatarem de não terem tido contato com o assunto, tanto no ensino médio quanto na graduação, conforme já abordado anteriormente.

No que diz respeito aos RTIs, os mesmos foram pensados e desenvolvidos de forma com que os conteúdos propostos fossem incrementais e com graus de dificuldade crescente. Dessa forma, a plotagem de gráfico de uma função do segundo grau pertencia às atividades que envolviam a programação de computadores no terceiro roteiro de tarefas investigativas, no qual os estudantes viram funções do segundo grau e a construção da parábola que a representa. Ao observarmos os dados revelados na Figura 23, um pouco mais de 70% dos estudantes foram classificados como proficientes, atendendo ao que era solicitado nas atividades do roteiro, mas não foram além, ou seja, suas soluções foram construídas utilizando os exemplos trabalhados nos roteiros e materiais de apoio. Cabe ressaltar que, de acordo com a nossa rubrica, pequenos equívocos, como uma formatação na plotagem de um marcador no gráfico, que não comprometiam a execução do código, foram tolerados, mantendo o estudante no nível de aprendizagem proficiente. Essa tolerância ocorreu porque percebíamos que o estudante compreendia os elementos do gráfico, embora tivesse pequenas dificuldades específicas com a programação.

Figura 23. Resultado do roteiro 03 quanto à automatização de plotagem de gráficos no contexto de funções do segundo grau.



Fonte: Elaboração própria

Quanto aos estudantes classificados no nível de aprendizado próximo de proficiente, 80% deles deixaram de entregar algumas atividades pertencentes ao roteiro e ou entregaram programas incompletos. Apenas 20% deles entregaram todas as atividades, porém seus programas possuíam erros no código que impediam a plotagem correta do gráfico solicitado, como a obtenção dos pontos para plotagem da parábola.

Para a plotagem de gráficos, os estudantes puderam fazer uso das habilidades já adquiridas no desenvolvimento de programas para automatizar a resolução de equações algébricas, acrescentando ainda recursos da linguagem de programação Python, necessários para a plotagem dos gráficos da função. A dinâmica utilizada na resolução das tarefas foi a mesma empregada nos demais roteiros, baseada em exemplos, tendo os estudantes à disposição um conjunto de recursos de apoio, como vídeos, tutoriais e a intervenção do pesquisador.

A programação necessária para a plotagem correta de um gráfico de uma parábola, incluindo todos os elementos pertencentes ao gráfico, como eixos das ordenadas e abscissas, linhas e grades e ambiente de plotagem, é mais complexa que as estruturas de códigos vistas anteriormente, além de ser necessária uma quantidade de linhas de código superior às utilizadas para automatização de resolução de equações algébricas. Dessa forma, nos exemplos de códigos fornecidos, foram inseridos mais comentários, a fim de facilitar a compreensão dos estudantes. Dada a complexidade dos códigos, esses cuidados, associados a intervenções que buscavam elucidar conteúdos de programação utilizados na plotagem de gráfico, foram tomados para facilitar a aprendizagem dos estudantes. Para discutirmos um pouco mais sobre aspectos

relacionados à programação, apresentamos um trecho de código do estudante Est15, representado na Figura 24, desenvolvido como resposta à determinada tarefa do roteiro, classificado no nível de aprendizagem proficiente.

Figura 24. Trecho de código utilizado como resposta à tarefa do roteiro pelo estudante Est15.

```

30     #Obtendo pontos para plotagem do gráfico
31     x = np.arange(x1 -1, x2 +1, 0.001)
32     y = (k*(x**2)) + (w*x) + (h)
33     fig = plt.figure()
34     ax = fig.add_subplot(111, aspect='equal')
35     ax.plot(x, y ,linewidth =1.2, color='b') # Esta linha plota o gráfico.
36
37     #Configurando o ambiente de plotagem
38     ax.set_aspect('equal')
39     ax.grid(True, which='both')
40     ax.set_xlim(x1- 1, x2+ 1) #Limites da Área do gráfico eixos x e y
41     ax.set_ylim(yv -1, 1)
42
43     # Reirando linhas a direita
44     ax.yaxis.tick_left()
45     # Retirando linhas no topo
46     ax.spines['top'].set_color('none') # Bordas do limite do plano cartesiano
47     ax.spines['left'].set_color('none') # Bordas do limite do plano cartesiano
48     ax.spines['right'].set_color('none') # Bordas do limite do plano cartesiano
49     ax.spines['bottom'].set_color('none') # Bordas do limite do plano cartesiano
50     ax.xaxis.tick_bottom()
51
52     # Plotando linhas dos Eixos x e y
53     plt.plot((x1, x2), (0,0), ls= '--', color='black')
54     x = [xv,xv]
55     y = [0,yv]
56     plt.plot(x,y, ls= '--', color='black')
57
58     #Plotando pontos importantes
59     plt.plot(xv,yv, marker='o' , color='g', ls='') # marca as coordenadas do vértice no gráfico
60     plt.plot(x1,0, marker='o' , color='g', ls='') # marca o x1 no gráfico
61     plt.plot(x2,0, marker='o' , color='g', ls='')
62
63     #Anotando pontos importantes
64     ax.annotate('X1', xy=(x1,0), xytext= (x1 -0.5, -0.5),arrowprops=dict(arrowstyle="->",facecolor='
65     'black')) #anota (escreve) o x1 no gráfico
66     ax.annotate('X2', xy=(x2,0), xytext= (x2 -0.5, -0.5),arrowprops=dict(arrowstyle="->",facecolor='
67     'black'))
68     ax.annotate('V', xy=(xv,yv), xytext= (xv -0.5, yv -0.5),arrowprops=dict(arrowstyle="->",
69     facecolor='black')) #anota (escreve) o Vértice no gráfico
70
71     # Label - Inse as legendas no gráfico
72     ax.set_title('Gráfico Item 1.6')
73     ax.set_xlabel('Eixo X')
74     ax.set_ylabel('Eixo Y')
75
76     # Exibe Gráfico
77     plt.show()

```

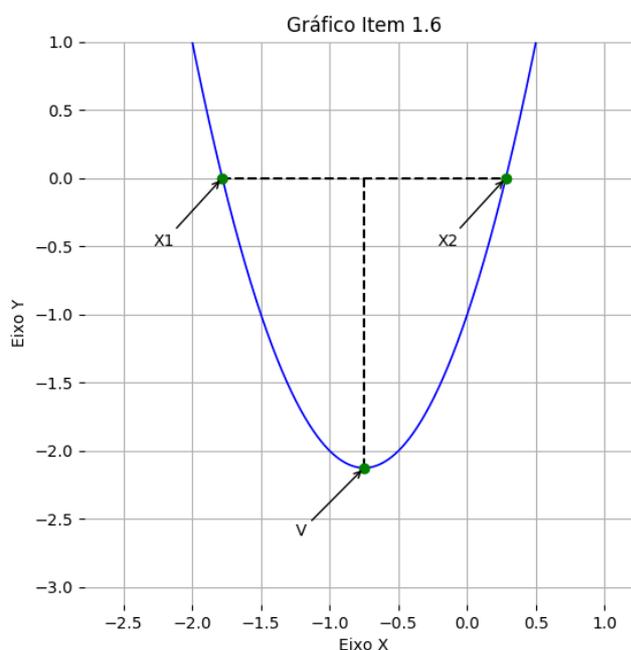
Fonte: Resposta do Estudante Est15

Na Figura 24, fizemos um recorte do programa construído pelo estudante, deixando apenas os códigos responsáveis pela plotagem do gráfico (linha de código 30 a 74), as linhas de código antecedentes diziam respeito à programação necessária para automatizar a resolução da função e determinação dos elementos necessários para a plotagem do gráfico, como as coordenadas do vértice e zeros da função. O trecho da tarefa referente à plotagem de gráfico solicitava a automatização do gráfico de uma função do segundo grau, com as seguintes características:

- i) Marcações e identificações das raízes da função;
- ii) Marcação e identificação do vértice;
- iii) Identificação dos eixos x e y;
- iv) Legenda do gráfico como: Gráfico Item 1.6.

A resposta ao item i) está implementada no programa por meio das linhas de código 65, 66, 69 e 70. O item ii) foi atendido pelas linhas de código 64 e 71; o item iii) foi respondido nas linhas 75 e 76 e o item iv) na linha 74. As demais linhas de código existentes no programa são responsáveis pela obtenção dos pontos para plotagem da parábola (linhas 31 e 32), configuração do ambiente de plotagem (linhas 38 a 41) e customização do gráfico, como ajustes de bordas, linhas e cores. Apesar de não ter sido solicitado na tarefa, as linhas de códigos referentes à obtenção dos pontos para plotagem da parábola e configuração do ambiente de plotagem, por exemplo, estavam implícitas na tarefa e o estudante já tinha conhecimento da necessidade de utilizá-las corretamente para responder integralmente aos itens solicitados na tarefa. Para uma melhor visualização, apresentamos na Figura 25 o resultado da plotagem do gráfico de uma função do tipo  $f(x) = 2x^2 + 3x - 1$ .

Figura 25. Resultado da execução do programa descrito na Figura 23, plotagem do gráfico de  $f(x) = 2x^2 + 3x - 1$ .



Fonte: Resultado da execução do programa do estudante Est15

Para que o estudante conseguisse chegar ao resultado exemplificado na Figura 25, foi necessária a compreensão de alguns recursos de Python, como a função *arange*, *plot*, *ax.annotate* e de inúmeros comandos responsáveis por customizar o gráfico, como *facecolor* e

*arrowprops*. Dessa forma, ao compreender a função de cada um dos comandos e funções listados, o estudante é capaz de ajustar as linhas de código para atender a diferentes demandas de plotagem de gráfico, como quando o valor da variável  $a$  de uma função do segundo grau é negativo e inclusive personalizá-lo, trocando cores das linhas do gráfico, ponta de setas, linhas e grades do ambiente de plotagem e outros elementos.

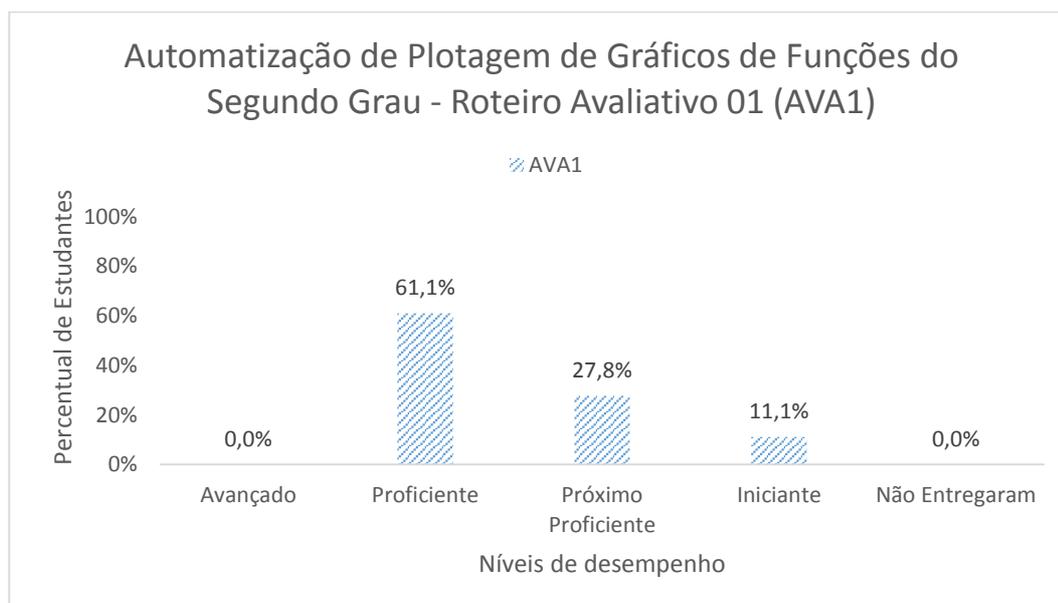
Em nossa aplicação, a codificação necessária para plotagem de gráficos, considerando o valor da variável  $a$  da função do segundo grau negativo, ou seja, concavidade da parábola para baixo, necessitava de ajustes diferentes de quando a concavidade é para cima, considerando, neste caso, o valor da variável  $a$  positivo. A última tarefa do roteiro três solicitava que os estudantes automatizassem a plotagem de gráficos para a variável  $a$  positiva e negativa. Para construir suas soluções, os estudantes tiveram que criar, baseadas em exemplos, estruturas condicionais que criticassem o valor da variável  $a$  de forma com que o programa desse o encaminhamento correto para a plotagem do gráfico. Apesar de alguns códigos necessários para essa solução se repetirem, o raciocínio e a lógica requeridos dos estudantes foram superiores às atividades desenvolvidas anteriormente e, ao concluí-la, os estudantes classificados como proficientes apresentaram um programa de aproximadamente 120 linhas de código pronto para plotar gráficos de funções independentemente do sinal da variável  $a$ .

A dinâmica de aplicação das atividades pertencentes à plotagem de gráfico no contexto de funções do segundo grau e presentes nesse roteiro seguiu a mesma estratégia do roteiro utilizado para automatização de resolução de equações algébricas. Assim, pautadas em princípios da investigação matemática, as tarefas do roteiro conduziam os estudantes à exploração, formulação de conjecturas, implementação de suas soluções e testes para verificar se programas atendiam corretamente o solicitado na tarefa, comunicando o raciocínio empregado em sua solução. Em suas respostas, encontramos características similares às discutidas na Seção 3.2, na resolução dos roteiros, já que, em muitos casos, as tarefas que abordavam automatização de plotagem de gráficos eram sequenciais às que possuíam automatização de resolução de equações algébricas.

Quanto aos resultados do roteiro avaliativo 1, cabe ressaltar que plotagens de gráficos no contexto de funções do segundo grau pertenciam às atividades do terceiro roteiro de tarefas investigativas, juntamente com atividades de automatização, discutidas na Seção 4.2. Dessa forma, a mesma avaliação 1 possuía tarefas que abordavam automatização de resolução de equações e de plotagem de gráficos. A tarefa da avaliação possuía o mesmo nível de complexidade das utilizadas no roteiro e a dinâmica de aplicação foi a mesma da automatização de resolução de equações algébricas, ou seja, os estudantes tinham a sua disposição trechos de

código que poderiam ser utilizados na construção de suas repostas à tarefa do roteiro. Pelo fato da programação necessária para a plotagem de gráficos ser mais extensa e complexa, o trecho de código fornecido aos estudantes como material de apoio, possuía mais números de linhas de código do que o fornecido na automatização de resolução de equações algébricas. Os dados demonstrados na Figura 26 nos auxiliará na discussão dos resultados da avaliação.

Figura 26. Resultados da avaliação 1 referente à plotagem de gráficos no contexto de funções do segundo grau



Fonte: Elaboração Própria

Um pouco mais de 60% dos estudantes tiveram seu nível de aprendizagem classificados como proficiente, ou seja, demonstraram habilidades para automatizar a plotagem de gráficos, conforme metodologia empregada, atendendo às tarefas do roteiro. Em seus programas utilizados como repostas, nos deparamos com linhas de códigos semelhantes às utilizadas nos roteiros e no *script* fornecido como apoio, mantendo a mesma estrutura e lógica por nós fornecida. Em virtude disso, não houve estudante classificado no nível de aprendizagem avançado. Conforme também podemos observar na Figura 26, 27,8% dos estudantes foram classificados no nível de aprendizagem próximo de proficiente, pois suas soluções para plotagem de gráficos apresentavam problemas que comprometiam o atendimento do solicitado na tarefa. Os problemas comumente encontrados estão associados à ausência ou equívocos nas linhas de códigos responsáveis pela plotagem dos elementos da parábola, como na sintaxe das funções *Python plt.plot* e *ax.annotate*, responsáveis pela plotagem e marcação das coordenadas do vértice e zeros da função.

Dentre os estudantes classificados no nível de aprendizagem próximo de proficiente, 80% deles pertencem ao grupo de estudantes que alegaram indisponibilidade de tempo para

estudos extraclasse, bem como não conseguiam participar dos momentos de atendimento individual e ou coletivo com o professor. Considerando que a codificação para plotagem de gráficos é mais complexa do que os códigos trabalhados anteriormente, acreditamos que a falta de dedicação dos estudantes para sanar suas próprias dúvidas pode ter contribuído para os resultados.

A atividade proposta no roteiro avaliativo 01 solicitava que o estudante provesse uma solução para plotar uma parábola, tendo como base determinada função do segundo grau. O programa desenvolvido pelo estudante deveria manter as linhas e grades no ambiente de plotagem e atender aos seguintes requisitos:

- i) Marcações e identificações das raízes da função;
- ii) Marcação e identificação do vértice;
- iii) Legendas com nomes significativos;

Para discutirmos o atendimento aos itens i) ii) e iii), utilizaremos o programa apresentado pelo estudante Est12, classificado como proficiente. Considerando que a solução do estudante possui mais de 70 linhas de código, recortamos somente os blocos de comandos responsáveis pelo atendimento dos itens supracitados, conforme demonstrado na Figura 27.

Figura 27. Trecho de código utilizado como resposta à tarefa do roteiro pelo estudante Est12.

```

61      plt.plot(xv,yv, marker='o' , color='g', ls='') # marca as coordenadas do vértice no gráfico
62      plt.plot(x1,0, marker='o' , color='g', ls='') # marca as raízes da função
63      plt.plot(x2,0, marker='o' , color='g', ls='') # marca as raízes da função
64
65      #Anotando pontos importantes x1, x2 e v
66      ax.annotate('X1', xy=(x1,0), xytext= (x1 -0.5, -0.5),arrowprops=dict(arrowstyle="->",
67      facecolor='black'))
68      ax.annotate('X2', xy=(x2,0), xytext= (x2 -0.5, -0.5),arrowprops=dict(arrowstyle="->",
69      facecolor='black'))
70      ax.annotate('v', xy=(xv,yv), xytext= (xv -0.5,yv -0.5),arrowprops=dict(arrowstyle="->",
71      facecolor='black'))
72
73      # Label - Insere as legendas no gráfico
74      ax.set_title("grafico 3")
75      ax.set_xlabel("eixo x")
76      ax.set_ylabel("eixo y")

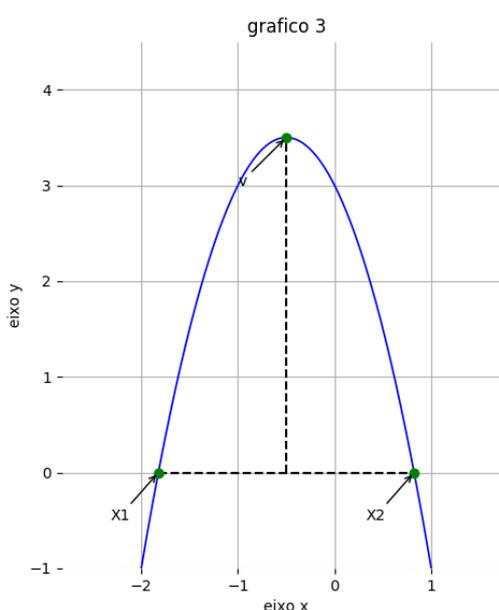
```

Fonte: Resposta do Estudante Est12

Ao observarmos a Figura 27, podemos identificar o atendimento ao item i) nas linhas de código 62, 63, 66 e 67. Já o item ii) foi respondido pelas linhas de código 61 e 68. As legendas solicitadas no item iii) são encontradas nas linhas de código 71 a 73. Para que o estudante pudesse responder adequadamente este tipo de tarefa, além do entendimento da sintaxe das funções de programação, é necessária a compreensão do plano cartesiano, bem

como a movimentação por coordenadas. Para exemplificar, observe a seguinte linha de código `plt.plot(x1,0, marker='o', color='g', ls='')`. A função `plt.plot` é utilizada para marcar no gráfico o  $x_1$  da função, para tanto, é necessário que o estudante aponte as coordenadas  $x$  e  $y$  no plano cartesiano, que nessa função estão dispostas no trecho de código (linhas 62 e 63) `plt.plot(x1,0`. O  $x_1,0$  da função indicam as coordenadas  $x$  e  $y$  respectivamente, ou seja, o marcador será plotado na interseção do valor calculado de  $x_1$  no eixo  $x$  e 0 (zero) em  $y$ . Para melhor compreensão, observem a parábola plotada como resultado do código do estudante, representada na Figura 28.

Figura 28. Resultado da execução do programa descrito na Figura 27, Est12.



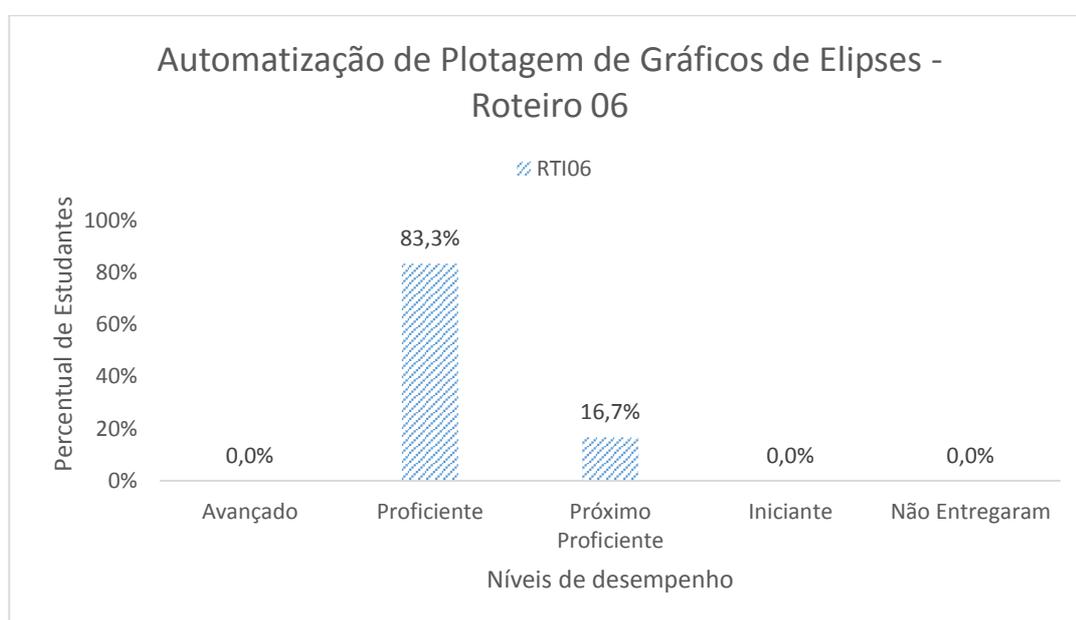
Fonte: Resultado da execução do programa do estudante Est12

Dentre os estudantes classificados no nível de aprendizagem próximo de proficiente, 80% deles pertencem ao grupo de estudantes que alegaram indisponibilidade de tempo para estudos extraclasse, bem como não conseguiam participar dos momentos de atendimento individual e ou coletivo com o professor. Considerando que a codificação para plotagem de gráficos é mais complexa do que os códigos trabalhados anteriormente, acreditamos que a falta de dedicação dos estudantes para sanar suas próprias dúvidas pode ter contribuído para os resultados.

A automatização de plotagem de gráficos no contexto das elipses esteve presente no roteiro de tarefas investigativas 6, sendo os estudantes avaliados por meio do roteiro avaliativo 02. A dinâmica de aplicação do roteiro 6 segue a mesma utilizada nos demais roteiros, sendo que, nessa fase, ainda eram necessárias intervenções para sanar dúvidas a respeito de conteúdos

matemáticos relacionados às elipses. Como podemos observar na Figura 29, 83,3% dos estudantes tiveram seu nível de desempenho classificado como proficiente, ou seja, suas soluções para automatização de gráficos atendiam o que era solicitado nas tarefas, mas seus códigos eram baseados nos exemplos trabalhados em sala de aula e nos recursos de apoio fornecidos. As respostas dos estudantes que possuíam erros no código que comprometiam o atendimento do que era solicitado na tarefa, como plotagem e marcações dos focos no gráfico, identificação dos eixos do plano cartesiano e/ou semieixos das elipses plotadas, representam apenas 16,7 % dos estudantes.

Figura 29. Resultado da automatização de plotagem de gráficos no contexto de elipses roteiro 6



Fonte: Elaborado pelos autores

De maneira gradual, os estudantes foram explorando os códigos fornecidos como recurso de apoio com o objetivo de compreender as funções e comandos de programação de computadores necessários para plotar uma elipse, já que, nesse caso, parte deles se diferenciava dos utilizados para plotagem do gráfico de uma função do segundo grau. O primeiro *script* explorado plotava uma elipse, porém sem marcação alguma de seus elementos e, para que os estudantes pudessem perceber o potencial da programação de computadores em outros contextos, solicitamos, em determinada tarefa do roteiro, que os mesmos plotassem uma circunferência, fazendo alterações neste mesmo *script*. Para melhor discussão, apresentamos na Figura 30 um recorte do bloco de programação fornecido como exemplo e as modificações feitas pelos estudantes, nesse caso, Est02, classificado como proficiente nesse roteiro, tendo sua resposta demonstrada na Figura 31.

Figura 30. Trecho do Script1R06, dado com exemplo, que plota uma elipse

```

10 a = 8 # dimensão do semieixo maior da Elipse
11 b = 6 # dimensão do semieixo menor da Elipse
12 x0 = 0 # coordenada do centro da Elipse, x
13 y0 = 0 # coordeanda do centro da Elipse, y
14 angulo = 360 # ângulo da Elipse
15
16 ells = Ellipse(xy=[ x0, y0], width=2*a, height=2*b, angle=angulo, color= 'black', linewidth =1.2) ↵
    ↵#ells recebe as coordenadas do centro [x0,y0], em seguida determinamos a largura do eixo " ↵
    ↵width=2*a, seguida pela altura do eixo " height=2*b". Determinamos o angulo da Elipse por meio de ↵
    ↵angle=angulo e a espessura da linha de contorno da Elipse " linewidth =1.2"
17 fig = plt.figure(0)
18 ax = fig.add_subplot(111, aspect='equal') # equalização do gráfico

```

Fonte: Roteiro de tarefa investigativas 6

Figura 31. Resposta do estudante Est02

```

10 a = 5 # dimensão do semieixo maior da Elipse
11 b = 5 # dimensão do semieixo menor da Elipse
12 x0 = 0 # coordenada do centro da Elipse, x
13 y0 = 0 # coordeanda do centro da Elipse, y
14 angulo = 360 # ângulo da Elipse
15
16 ells = Ellipse(xy=[ x0, y0], width=2*a, height=2*b, angle=angulo, color= 'black', linewidth =1.2)
17 fig = plt.figure(0)
18 ax = fig.add_subplot(111, aspect='equal') # equalização do gráfico

```

Fonte: Resposta do Estudante Est02 a tarefa do roteiro 6

O código representado pela Figura 30, ao ser executado, plotava uma elipse. A tarefa do roteiro solicitava aos estudantes que fizessem as devidas alterações para que o código passasse a plotar uma circunferência de raio 5 cm. Ao compararmos os códigos representados nas Figuras 30 e 31, podemos perceber que as variáveis  $a$  e  $b$ , dispostas nas linhas de código 10 e 11, respectivamente, recebem o valor do comprimento dos semieixos da elipse, com a variável  $a$  determinando o semieixo horizontal e  $b$ , o vertical. Por meio de exploração ao código, podemos observar na Figura 31 que o estudante alterou os valores dessas variáveis, atribuindo a elas o valor solicitado na tarefa, ou seja, 5. Ao fazer essa alteração, o estudante pode demonstrar entendimento na sintaxe da função *ells*, linha de código 16, responsável por receber os valores das variáveis e dar os encaminhamentos iniciais no programa para a plotagem da elipse.

Com um nível de complexidade um pouco maior, determinada tarefa do roteiro fornecia uma equação da elipse e solicitava que o estudante plotasse seu gráfico com as seguintes características:

- i) Plotar e marcar os focos;
- ii) Plotar e marcar o Centro;

iii) Plotar e marcar os limites dos eixos x e y (A, A1, B, B1).

Para resolver aos itens i) ii) e iii), o estudante Est12, classificado no nível de aprendizagem proficiente, construiu um programa, que, por possuir mais cerca de 90 linhas de código, apresentaremos somente os blocos responsáveis pelo atendimento aos itens, representados na Figura 32.

Figura 32. Resposta do estudante Est12

```

63 # plotando pontos importantes da elipse
64 foco1= [foco1, y0]
65 foco2= [foco2, y0]
66 focos=[foco1,foco2]
67 plt.plot(*zip(*focos), marker='o', color='b', ls='')
68 centro=[x0, y0]
69 plt.plot((*centro), marker='o', color='g', ls='')
70
71 # Marcando os pontos A,A1, B, B1
72 plt.plot( -a + x0, y0, marker='o' , color='black', ls='')
73 plt.plot( a + x0, y0, marker='o' , color='black', ls='')
74 plt.plot( x0, -b + y0, marker='o' , color='black', ls='')
75 plt.plot( x0, b + y0, marker= 'o' , color='black', ls='')
76
77 # Plotando Eixos
78 plt.plot((a+ x0, -a +x0), (y0,y0), ls= '--', color='g')
79 plt.plot((x0,x0),(b +y0, -b +y0), ls= '--', color='g')
80
81 # Plotando anotações
82 ax.annotate('Foco1', xy=foco1, xytext=[foco1[0]+0.5, foco1[1]+0.5],arrowprops=dict(arrowstyle="->",
83 facecolor='black'))
83 ax.annotate('Foco2', xy=foco2, xytext=[foco2[0]+0.5, foco2[1]+0.5],arrowprops=dict(arrowstyle="->",
84 facecolor='black'))
84 ax.annotate('Centro', xy=centro, xytext=[centro[0]+0.5, centro[1]+0.5], arrowprops=dict(arrowstyle="
85 &"->",facecolor='black'))
85 ax.annotate('A', xy=(a+ x0, y0), xytext= (a+ x0 +0.5, y0))
86 ax.annotate('A1', xy=( a+ x0,y0), xytext= (-a+ x0 -0.8, y0))
87 ax.annotate('B', xy=(x0,b+ y0), xytext= (x0, b+ y0 +0.5))
88 ax.annotate('B1', xy=(x0,-b +y0), xytext= (x0, -b+ y0 -0.8))

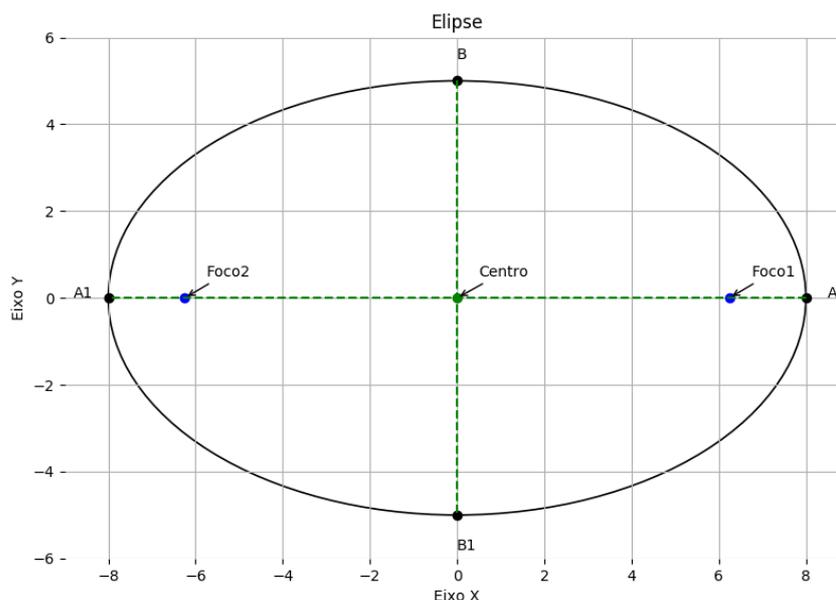
```

Fonte: Resposta do Estudante Est12 a tarefa do roteiro 6

Ao observarmos a Figura 32, podemos perceber que as linhas de código estão separadas em blocos de linhas de código, sendo que logo no início (linhas de código 64 a 66) temos a organização dos focos em listas<sup>49</sup>, conceito novo de programação estudado e implementado por meio de exemplos e que, nesse contexto, auxiliou na marcação dos focos e do centro da elipse. A marcação dos limites dos eixos x e y foi implementada pelo estudante nas linhas de código 72 a 75, sendo que, para tanto, as noções de coordenadas no plano cartesiano eram desejáveis na implementação. As anotações, (plotar a letra A, a palavra Centro, por exemplo) foram implementadas nas linhas de códigos 82 a 88, sendo que o estudante pode novamente se deparou com coordenadas do plano cartesiano para posicionar corretamente no gráfico, os elementos plotados. Para melhor clareza dos resultados produzidos pelo programa do estudante, apresentamos a elipse plotada e representada na Figura 33.

<sup>49</sup> Lista é uma estrutura de dados, organizadas por itens de forma linear que podem ser acessados por meio de índices, iniciando do zero, que representam cada elemento e a sua posição.

Figura 33. Elipse plotada utilizando o programa do estudante Est12

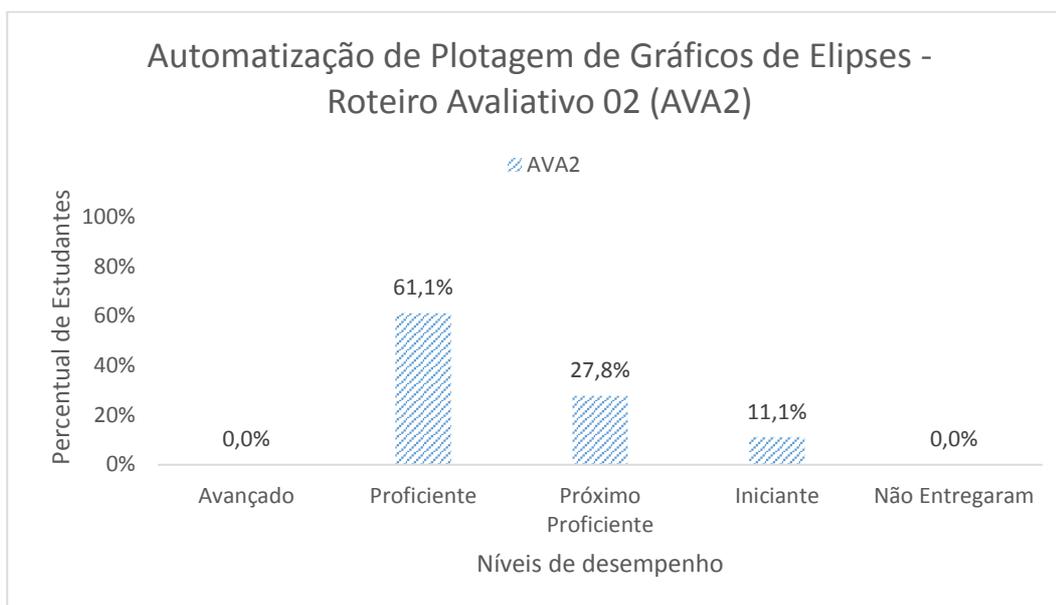


Fonte: Resposta do Estudante Est12 a tarefa do roteiro 6

As tarefas ainda solicitavam que os estudantes explorassem elementos do programa responsáveis pela customização do gráfico. Dessa forma, eles puderam experimentar alterações em cores, espessura e tipo de traçado utilizado nas linhas de todo o gráfico, por exemplo. Ao observar a Figura 33, gostaríamos de ressaltar que aspectos relacionados à estética foram pontuados e observados pelo estudante, já que cuidados relacionados às distâncias entre as letras B e B1, do ponto marcado no traço da elipse, deveriam ser inseridas no código para produzir a plotagem desses elementos de forma simétrica.

Ao final do RTI06, aplicamos o roteiro avaliativo 2, que, dentre outras tarefas, possuía uma específica para a automatização de plotagem de gráficos no contexto das elipses. A dinâmica utilizada nesse roteiro avaliativo seguiu a mesma proposta do roteiro avaliativo 01, diferenciando-se apenas pelos assuntos estudados. Na elaboração do roteiro de avaliativo 02, as tarefas foram planejadas com grau de dificuldade similar ao enfrentado pelos estudantes no desenvolvimento dos roteiros, tendo os recursos de apoio limitados a trechos de códigos que poderiam nortear a resposta do estudante, e, ainda, a interação dos estudantes era permitida somente com pesquisador/professor. Quanto ao desempenho dos estudantes, apresentamos os dados sumarizados no gráfico exposto na Figura 34.

Figura 34. Resultado da Roteiro Avaliativo 2



Fonte: Elaboração própria

Conforme podemos observar nos dados representados na Figura 34, mais de 60% dos estudantes tiveram seu desempenho classificado como proficiente, ou seja, apresentaram suas soluções, baseadas nos scripts e exemplos de código trabalhados nos roteiros, que plotassem corretamente o gráfico da elipse e os elementos solicitados na tarefa. Do total de alunos, 27,8% tiveram seu nível de aprendizagem classificado como próximo de proficiente, pois seus programas apresentados como resposta as tarefas, possuíam equívocos ou ausência de códigos que comprometiam o atendimento do solicitado na tarefa. Típicos erros encontrados nos scripts dos estudantes, classificados como próximo de proficiente, estão relacionados aos códigos responsáveis pela plotagem e marcação dos elementos solicitados na tarefa, como a representação dos focos e as extremidades do eixo horizontal. A tarefa da avaliação era semelhante à discutida na Figura 32 e seu resultado era semelhante ao representado na Figura 33, com algumas especificidades, como, ao invés de plotagem do foco, o estudante deveria plotar o irrigador.

Ainda sobre os estudantes classificados como próximo de proficientes, 75% deles tiveram este mesmo desempenho no roteiro avaliativo 01, possuindo as mesmas características discutidas anteriormente, ou seja, pertencem ao grupo de estudantes que afirmavam não dispor de tempo para estudos extraclasse e/ou não dispunha de recursos, como computador para estudarem fora de sala de aula e frequentemente chegavam atrasados nas aulas e avaliações. O mesmo pode ser dito dos estudantes cujo desempenho estão classificados como iniciantes, sendo que, nesse caso, observamos que 50% dos estudantes conseguiram desenvolver a tarefa

à mão. Portanto as dificuldades estavam relacionadas à programação. Cabe ressaltar que a dinâmica utilizada em sala de aula co-responsabiliza o estudante para o estudo. O não acompanhamento das atividades, por quaisquer motivos, e o não estudo fora de sala de aula comprometem o desempenho, porque toda a construção do conhecimento se dá progressivamente no decorrer das atividades.

Ao compararmos o desempenho dos estudantes na avaliação diagnóstica, referente à automatização de plotagem de gráficos no contexto de elipses, ao desempenho no roteiro avaliativo 02, podemos perceber que mais de 60% dos estudantes deixaram o nível de aprendizagem iniciante e passaram para proficiente, demonstrando em seus programas, habilidade para automatizar plotagem de gráficos, baseada em exemplos de código.

#### 4.4 Detalhamento dos Resultados da Terceira Questão de Pesquisa

Nossa terceira questão de pesquisa é: Quais as limitações desta aplicação de PBI?

Para responder essa questão de pesquisa, valemo-nos dos *feedbacks* dos estudantes, coletados durante a aplicação da proposta de ensino-aprendizagem, anotações no diário de campo, roda de conversa e questionário final (vide Apêndice H).

No que diz respeito às limitações de nossa proposta de ensino-aprendizagem, percebemos a necessidade de introduzir programação de forma lúdica, utilizando, por exemplo, *games* disponibilizados por plataformas como a Programaê<sup>50</sup>, que iniciam os estudantes com programação em blocos. O objetivo é inicializar a lógica de programação de uma maneira mais branda, o que pode facilitar a aprendizagem dos conteúdos dos roteiros. Apesar de não ter sido possível implementar em nossa pesquisa, inserimos essa dinâmica em nosso produto educacional, sugerindo que seja aplicada antes do início dos roteiros.

A carga horária inicialmente planejada sofreu alterações, o que possibilitou a aplicação da pesquisa, conforme necessidades dos participantes. Contudo, baseados em relatos coletados na roda de conversa e na resposta à pergunta do questionário de avaliação final, detectamos a necessidade de aumentar a carga horária final e redistribuir a carga horária destinada ao estudo autônomo, para favorecer os roteiros presenciais, proporcionando um maior conforto para o professor e para os estudantes na aplicação. A proposta de redimensionamento da carga horária foi inserida no produto educacional, onde a mesma passou a ser de 80 horas.

Quanto aos roteiros, nossas percepções, associadas aos relatos dos estudantes, apontaram dificuldades no cumprimento das tarefas propostas, dentro do tempo inicialmente

---

<sup>50</sup> <https://studio.code.org/s/frozen/stage/1/puzzle/1>

planejado para cada roteiro. Como a carga horária para a aplicação da pesquisa era limitada, ajustamos alguns roteiros no decorrer da aplicação e os demais roteiros foram ajustados para o produto. Com o intuito de corrigir problemas como esses, dividimos roteiros em duas partes e ajustamos a carga horária prevista para aplicação de cada um dos roteiros, como no RTI03, que foi dividido, separando a plotagem de gráficos da função do segundo grau com a concavidade da parábola para baixo. Esta alteração também foi inserida no produto educacional.

Uma especificidade de nossa aplicação é que os encontros não eram regulares, com horários e dias fixos do início ao fim da aplicação. Consideramos que a falta de regularidade nos encontros pode não favorecer a aprendizagem, em virtude de intervalos longos entre um encontro e outro. No produto, chamamos a atenção para a desejável regularidade nos encontros, bem como sugerimos que aconteçam no mínimo uma vez por semana, se possível duas.

Tratando-se dos recursos de apoio, percebemos nos estudantes a necessidade de buscar outros materiais para auxiliar na aprendizagem, principalmente quanto aos conteúdos de elipses, álgebra necessária na redução de suas equações e também na obtenção de seus elementos e plotagem de seus gráficos. Dessa forma, sugerimos no produto educacional que o professor, ao identificar a falta de familiaridade dos estudantes com determinados conteúdos, selecione e/ou indique materiais complementares, além dos propostos nos roteiros.

Outro aspecto observado e identificado é a falta de cultura dos estudantes quanto ao estudo autônomo, habilidade necessária para a abordagem utilizada em nossa pesquisa, bem como a falta de hábito para participarem das horas de atendimento ao estudante. Também observamos que ferramentas como o *WhatsApp* e *e-mails*, disponibilizadas para troca de informações e retirada de dúvidas, são subutilizadas pelos estudantes. Acreditamos que sejam necessárias ações que promovam mudanças desses comportamentos nos estudantes.

Identificamos, na roda de conversa, relatos sobre a ausência de tempo para estudos extraclasse. Com o intuito de orientá-los, propomos no produto educacional que o professor aborde aspectos relacionados a estratégias para gerenciamento do tempo, indicando caminhos que podem conduzir o estudante a administrar melhor as horas de seus dias, favorecendo os momentos para o estudo. Quanto ao exercício da autonomia, vimos como um caminho, alterações na prática dos professores que atuam nas licenciaturas, de maneira que os mesmos possam desenvolver estratégias que motivem o desenvolvimento da autonomia em seus estudantes, na sala de aula e fora dela.

## CONSIDERAÇÕES FINAIS

Nosso trabalho diz respeito a uma proposta de ensino-aprendizagem, denominada Programação Baseada em Investigação – PBI, no qual estudantes de licenciatura aprendem programação de computadores de maneira progressiva, baseada em exemplos e em atividades investigativas. Em nossa abordagem, PBI tem como finalidade desenvolver habilidades de programação para automatização de resolução de equações algébricas e plotagem de gráficos, no contexto de funções do segundo grau e das elipses. Embora este estudo tenha sido estanciado para esses conteúdos, a aprendizagem adquirida pode ser importada para outros conteúdos da Matemática.

Esta proposta de ensino foi avaliada por meio de sua aplicação com estudantes matriculados na disciplina de Tecnologias da Informação e Comunicação Aplicadas ao Ensino do Curso Superior de Licenciatura em Matemática do Instituto Federal do Acre – Campus Cruzeiro do Sul, nos meses de março a junho do ano de 2017. A análise dos dados revelou que PBI proporcionou melhorias na habilidade de programação de computadores nos estudantes, tanto para automatização de resolução de equações algébricas quanto à plotagem de gráficos. Os estudantes demonstraram desempenho de proficiência, ou seja, baseados em exemplos, foram capazes de desenvolver artefatos utilizando a linguagem de programação *Python* e pacotes, para resolverem as tarefas investigativas.

Este trabalho de dissertação foi consolidado em um produto educacional que se refere a um guia didático intitulado “Programação Baseada em Investigação – PBI Programação de Computadores aplicada à resolução de equações algébricas e plotagem de gráficos”. O guia é composto por orientação didática, ao professor e ao estudante, além de congrega recursos necessários para a implementação por outros professores de informática que atuam em cursos de Licenciatura em Matemática.

Dentre as contribuições deste trabalho, destacamos: (i) a formação dos licenciandos em Matemática para adoção de programação de computadores como recurso para facilitar a própria aprendizagem e, futuramente, o ensino; (ii) auxílio aos professores de Informática que atuam nas Licenciaturas e que podem encontrar em PBI formas de facilitar a aprendizagem e contextualização da programação na Licenciatura em Matemática; (iii) incentivo aos licenciandos, que passaram pela aprendizagem com PBI, para utilizarem programação de computadores como parte de sua própria atuação docente como professor de Matemática.

Em termos de produção científica, tivemos um trabalho publicado nos anais do XXIII Workshop de Informática na Escola, premiado como melhor artigo na trilha<sup>51</sup> em que foi submetido e um capítulo publicado no livro intitulado Formação de Professores e Estratégias de Ensino: Perspectivas Teórico-Práticas<sup>52</sup>.

Como trabalhos futuros, destacamos a importância de aplicar novamente a proposta na disciplina de Tecnologias de Informação e Comunicação Aplicadas ao Ensino, bem como promover a capacitação de professores de Informática que atuam em cursos de Licenciatura em Matemática para a aplicação de PBI. No que diz respeito aos desdobramentos da pesquisa, seria interessante identificar a atuação docente dos estudantes que passaram por PBI para verificar se suas atividades docentes incluíram programação de computadores.

---

<sup>51</sup> Formação De Recursos Humanos Para O Uso Das Tecnologias Digitais De Informação E Comunicação (TDIC) Na Educação" do XXIII WIE (Workshop de Informática na Escola), evento promovido pelo VI CBIE (Congresso Brasileiro de Informática na Educação), realizado pela Sociedade Brasileira de Computação (SBC), sediado em Recife-PE, no período de 30 de Outubro a 02 de Novembro de 2017. Disponível em <https://bit.ly/2LF0SGZ>.

<sup>52</sup> SOUZA. A. C. R et al.[Org.] **Formação de professores e estratégias de ensino**: perspectivas teórico-prática. Curitiba: Appris, 2018.

## REFERÊNCIAS

- AFINI, D. C.; JÚNIOR, S. C. J; CARDOSO, A. Explorando o SuperLogo na formação inicial de professores de matemática por meio da construção de pavimentações do plano: aprendendo com a sua criatividade. In: XIX Workshop de Informática na Escola. 2013, Limeira. **Anais... do II CBIE**, 2013.
- ANDALOSSI, K. E. **Pesquisas-ações: ciências, desenvolvimento, democracia**. São Carlos: EdUFSCar, 2004.
- BARCELOS, T. S.; SILVEIRA, I. F. Pensamento Computacional e Educação Matemática: Relações para o Ensino de Computação na Educação Básica. In: XX Workshop sobre Educação em Computação. 2012, Curitiba. **Anais... do XXXII CSBC**, 2012.
- BAUER, M. W.; GASKELL, G. **Pesquisa qualitativa com texto, imagem e som: um manual prático**. Petrópolis: Vozes, 2002.
- BENDER, W. N. **Aprendizagem baseada em projetos: educação diferenciada para o século XXI**. Porto Alegre: Penso, 2014.
- BENITTI, F. B. V. et al. Experimentação com Robótica Educativa no Ensino Médio: ambiente, atividades e resultados. In: XVº Workshop de Informática na Escola. 2009, Bento Gonçalves. **Anais... do XXIX CSBC**, 2009.
- BERS, M. U. **Young programmers: Think playgrounds, not playpens**. Jackson, 2014. Disponível em < <http://www.tedxjackson.com/talks/young-programmers-think-playgrounds-not-playpens/>>. Acesso em: 27 ago. 2017.
- BIGGS, J.; TANG, C. **Teaching for Quality Learning at University**. 4. ed. Berkshire, England: Society for Research into Higher Education & Open University Press, 2011.
- BLIKSTEIN, P. **O pensamento computacional e a reinvenção do computador na educação**. 2008. Disponível em <[http://www.blikstein.com/paulo/documents/online/ol\\_pensamento\\_computacional.html](http://www.blikstein.com/paulo/documents/online/ol_pensamento_computacional.html)>. Acesso em: 02 out. 2017.
- BOGDAN, R.; BIKLEN, S. **Investigação qualitativa em educação: uma introdução à teoria e aos métodos**. Porto: Porto Editora, 1994.
- BONFIM, R. S. **Cônicas: Situações Didáticas para o Ensino Médio**. Dissertação (Mestrado em Matemática). São José do Rio Preto, 2015. Instituto de Biociências, Letras e Ciências Exatas, UNESP.
- BRASIL. Ministério da Educação, Secretaria da Educação Média e Tecnológica. **Parâmetros Curriculares Nacionais: ensino médio**. Brasília: Ministério da Educação, 1999.
- BRASIL. Ministério da Educação. **Base Nacional Comum Curricular: Educação é a Base, Ensino Médio**. Brasília: Ministério da Educação, 2018.
- BRASIL. Ministério de Educação. Secretaria de Ensino Fundamental. **Parâmetros Curriculares Nacionais: Matemática, Ensino de Quinta a Oitava séries**. Brasília/DF, 1998.
- BRASIL. Parecer CNE/CES 1.302/2001. Diretrizes curriculares nacionais para os cursos de matemática, bacharelado e licenciatura. **Diário Oficial da União**, Brasília, 05 mar. 2002b, Seção 1, p. 15. Disponível em: <<http://bit.ly/2uR3JpN>>. Acesso em: 13 maio 2017.

BRASIL. Resolução CNE/CP 1/2002. **Institui diretrizes curriculares nacionais para a formação de professores da educação básica, em nível superior, curso de licenciatura, de graduação plena.** Brasília/DF, 2002a.

BRASIL. Resolução CNE/CP 2/2015. Define as Diretrizes Curriculares Nacionais para a formação inicial em nível superior (cursos de licenciatura, cursos de formação pedagógica para graduados e cursos de segunda licenciatura) e para a formação continuada. **Diário Oficial da União**, Brasília, 2 de julho de 2015 – Seção 1 – pp. 8-12. Brasília/DF, 2015. Disponível em: <<https://goo.gl/FwKhT2>> Acesso em: 05 set. 2017.

BRASIL. Secretaria de Educação Média e Tecnológica. **PCN+ Ensino Médio: orientações educacionais complementares aos Parâmetros Curriculares Nacionais. Ciências da Natureza, Matemática e suas Tecnologias.** Brasília: Ministério da Educação, 2002.

BRAUMANN, C. A. Divagações sobre investigação matemática e seu papel na aprendizagem matemática. In: **XI Encontro de Investigação em Educação Matemática.** Coimbra: Secção de Educação e Matemática da Sociedade Portuguesa de Ciências e Educação, 2002. p. 1-19..

BROOKHART, S. M. **How to create and use rubrics for formative assessment and grading.** Alexandria, VA: ASCD, 2013.

CAMARGO, I.; BOULOS, P. **Geometria Analítica: um tratamento vetorial.** 3. ed. São Paulo: Prentice Hall, 2005.

CORTEZ, A. B. **As dificuldades na visualização espacial de alunos do 4º ano do 1º Ciclo do Ensino Básico.** Dissertação (Mestrado em Ensino). Beja, 2014. Instituto Politécnico de Beja, Escola Superior de Educação.

CURCI, A. P. F. **O software de programação Scratch na formação inicial de do professor de Matemática por meio da criação de objetos de aprendizagem.** Dissertação (Mestrado Profissional em Ensino de Matemática). Londrina, 2017. Programa de Pós-Graduação em Ensino de Matemática, Universidade Tecnológica Federal do Paraná.

DANIELSON, C.; MARQUEZ, E. **Performance Tasks and Rubrics for High School Mathematics: Meeting Rigorous Standards and Assessments.** 2. ed. New York : Routledge, 2016.

DANTE, L.R. **Matemática contexto e aplicações, volume único.** 3. ed. São Paulo: Ática, 2009.

DE VILLIERS, M. Algumas reflexões sobre a Teoria de Van Hiele. **Educação Matemática Pesquisa**, São Paulo, v. 12, n. 3, p. 400-431, 2010.

DURÃES, M. N. Educação Técnica e Educação Tecnológica Múltiplos Significados no Contexto da Educação Profissional. **Educação & Realidade, Rio Grande do Sul**, v. 34, n. 3, p. 159-175, 2009.

ELOY, A. A.S.; LOPES, R. D.; ANGELO, I. M. Uso do Scratch no Brasil com objetivos educacionais: uma revisão sistemática. **Renote.** Rio Grande do Sul, v. 15, n.1, 2017.

ESTEBAN, M. P. S. **Pesquisa qualitativa em educação: fundamentos e tradições.** Porto Alegre: AMGH, 2010.

FARAGO, J. L. et al. **Ensino Médio: 3ª série.** Curitiba: Positivo, 2012. 3 v. v. 1.

FERNANDES, Daniela. **Brasil avança em conhecimento básico de matemática, mas continua atrás em ranking.** PARÍS, 2016. Disponível em <[http://www.bbc.com/portuguese/noticias/2016/02/160209\\_ocde\\_alunos\\_baixa\\_performance\\_pai\\_df](http://www.bbc.com/portuguese/noticias/2016/02/160209_ocde_alunos_baixa_performance_pai_df)>. Acesso em: 01 set. 2016.

- GATTI, B. A.; NUNES, M. N. R. Formação de professores para o ensino fundamental: estudo de currículos das licenciaturas em pedagogia, língua portuguesa, matemática e ciências biológicas. **Textos FCC**, v. 29, p. 155, 2009.
- GERHARDT, T. E.; SILVEIRA, D. T. [org.] **Métodos de Pesquisa**. 1. ed. Porto Alegre: Editora da UFRGS, 2009.
- IDRIS, Ivan. **NumPy Cookbook**. 2. ed. United Kingdom: Packt Publishing Ltd, 2015.
- IEZZI, Gelson et al. **Matemática ciência e aplicações: ensino médio**. 7. ed. São Paulo: Saraiva, 2013. 3 v. v. 3.
- LEITHOLD, Louis. **O Cálculo com Geometria Analítica**. 3. ed. São Paulo: Harbra, 1994.
- LOTERO, L. A. A. Teoría de la carga cognitiva, diseño multimedia y aprendizaje: un estado del arte. **Magis. Revista Internacional de Investigación en Educación**, v. 5, n. 10, 2012.
- LUTZ, M. **Learning python**. O'Reilly Media, Inc., 2013.
- MAGARINUS, R. **Uma proposta para o ensino de funções através da utilização de objetos de aprendizagem**. Santa Maria, 2013. Dissertação (Mestrado Profissional em Matemática). Centro de Ciências Naturais e Exatas, Universidade Federal de Santa Maria.
- MELO, M. C. H. de; CRUZ, G. de C. Roda de conversa: uma proposta metodológica para a construção de um espaço de diálogo no ensino médio. **Revista Imagens da Educação**, v. 4, n.2, p. 31-39, 2014.
- MENDONÇA, A. P. **Alinhamento Construtivo: Fundamentos e Aplicações**. In: Gonzaga, Amarildo M. (Organizador). Formação de Professores no Ensino Tecnológico: Fundamentos e Desafios. 1a. ed. ISBN 978-85-444-0369-3. Curitiba, PR: CRV, 2015. p. 109 – 130.
- MENEZES, N. N. C. **Introdução à Programação com Python**. São Paulo: Novatec, 2010.
- MPAKA, M. **O ensino e aprendizagem do gráfico da função quadrática com e sem auxílio do Software Winplot**. Lisboa, 2010. Dissertação (Mestrado em Ciências da Educação). Instituto de Ciências da Educação, Universidade Lusófona de Humanidades e Tecnologias.
- NETO, A. J. B. **A construção de Instrumentos Matemáticos didáticos com Tecnologia Digital: uma proposta de empoderamento para licenciandos em Matemática**. Tese (Doutorado em Educação Matemática). São Paulo, 2015. Pontifícia Universidade Católica de São Paulo PUC-SP.
- NUNES, D. J. **Ciência da Computação na Educação Básica**. Porto Alegre, 2011. Disponível em <<http://www.adufrgs.org.br/artigos/ciencia-da-computacao-na-educacao-basica/>>. Acesso em: 26 maio 2016.
- OLIVEIRA, F. C. **Dificuldades na construção de gráficos de funções**. Natal, 2006. Dissertação (Mestrado em Ensino de Ciências Naturais e Matemática). Centro de Ciências Exatas e da Terra, Universidade Federal do Rio Grande do Norte.
- PAPERT, S. **A máquina das crianças: repensando a escola na era da informática**. Porto Alegre: Ed. Artmed, 2008.
- PAPERT, S. **Mindstorms: Children, computers, and powerful ideas**. Basic Books, Inc., 1980.
- PARENTE, C. M. D.; DO VALLE, L. E. L. R.; DE MATTOS, M. J. V. M. [org.] **A formação de professores e seus desafios frente às mudanças sociais, políticas e tecnológicas**. Porto Alegre: Penso Editora, 2015.

- PASSEGGI, M. C. Metodologias para a análise e interpretação de fontes autobiográficas: Uma análise semântica. In: PASSEGGI, Maria da Conceição, BARBOSA, Tatyana Mabel Nobre (Org.). **Memórias, memoriais: pesquisa e formação docente**. Natal: RN: EDUFRN; São Paulo: Paulus, 2008, p. 60-91.
- PINTO, A. S. **Scratch na aprendizagem da matemática no 1º ciclo do ensino básico: estudo de caso na resolução de problemas**. Dissertação (Mestrado em Estudo da Criança). Braga, 2010. Instituto de Educação, Universidade do Minho.
- POLYA, G. **A arte de resolver problemas: um novo aspecto do método matemático**. Rio de Janeiro: Interciência, 1995.
- PONTE, J. P. et al. **Investigações matemáticas na sala de aula**. 3. ed. Belo Horizonte: Autêntica Editora, 2016.
- PONTE, J. P. et al. O trabalho do professor numa aula de investigação matemática. **Quadrante**, p. 41-70, 1998a.
- PONTE, J. P. et al. **Histórias de investigações matemáticas**. 1998. Disponível em: <[https://www.researchgate.net/publication/261178171\\_Historias\\_de\\_investigacoes\\_matematicas](https://www.researchgate.net/publication/261178171_Historias_de_investigacoes_matematicas)>. Acesso: 10 out. 2016.
- PONTE, J. P. Investigação sobre investigações matemáticas em Portugal. **Investigar em Educação**, p. 93-169, 2003.
- POZO, J. I. et al. **A solução de problemas: aprender a resolver, resolver para aprender**. Porto Alegre: Artmed, v. 3, 1998.
- QEDU. **Aprendizado dos alunos: Brasil**. Disponível em: <<http://qedu.org.br/brasil/aprendizado>>. Acesso: 23 mar. 2017.
- QEDU. **Aprendizado dos alunos: Brasil**. Disponível em: <<http://www.qedu.org.br/brasil/aprendizado>>. Acesso: 01 set. 2016.
- REIS, G. L.; SILVA, V. V. **Geometria Analítica**. 2. ed. Rio de Janeiro: LTC, 1996.
- RESENDE, G.; MESQUITA, M. G. B. F. Principais dificuldades percebidas no processo ensino-aprendizagem de matemática em escolas do município de Divinópolis, MG. **Educação Matemática Pesquisa**, São Paulo, v. 15, n. 1, p. 199-222, 2013.
- RESNICK, M. Learn to code, code to learn. **EdSurge**, 2013.
- SAHA, A. **Doing Math With Python**. San Francisco: No Starch Press, Inc., 2015.
- SANTOS, D. T. **O uso de Algoritmos e Programação no Ensino de Matemática**. Campinas, 2015. Dissertação (Mestrado em Matemática). Instituto de Matemática, Estatística e Computação Científica, Universidade Estadual de Campinas.
- SERRAZINA, L. et al. **O papel das investigações matemáticas e profissionais na formação inicial de professores**. 2002. Disponível em: <[http://spiem.pt/DOCS/ATAS\\_ENCONTROS/2002/2002\\_04\\_LSerrazina.pdf](http://spiem.pt/DOCS/ATAS_ENCONTROS/2002/2002_04_LSerrazina.pdf)>. Acesso: 10 jan. 2018
- SILBEY, R. **Math Think-Alouds**. Disponível em <<https://www.scholastic.com/teachers/articles/teaching-content/math-think-alouds>>. Acesso em: 20 abr. 2018.
- SMOLE, K. C. S.; DINIZ, M. I. S. **Matemática: ensino médio**. 7. ed. São Paulo: Saraiva, 2010. 3 v. v. 2.

SMOLE, K. C. S.; DINIZ, M. I. S. **Matemática: ensino médio**. 7. ed. São Paulo: Saraiva, 2010a. 3 v. v. 3.

SOCIEDADE Brasileira de Computação. **SBC realiza reunião para discutir a nova versão da Base Nacional Comum Curricular**. Disponível em

<<http://www.sbc.org.br/noticias/1693-sbc-realiza-reuniao-para-discutir-a-nova-versao-da-base-nacional-comum-curricular>>. Acesso: 26 maio 2016.

SOUZA, E. C; YONEZAWA, W. M. Princípios de Programação no Ensino de Matemática utilizando Processing 2. In: XII Encontro Nacional de Educação Matemática. 2016, São Paulo. **Anais...** do XII Encontro Nacional de Educação Matemática, 2016.

STEVENS, D.D; LEVI, A. **Introduction to rubrics: an assessment tool to save grading time, convey effective feedback, and promote student learning**. Sterling, Virginia, 2005.

THIOLLENT, M. **Metodologia da pesquisa-ação**. 18. ed. São Paulo: Cortez, 2011.

TRIPP, D. Pesquisa-ação: uma introdução metodológica. **Revista Educação e Pesquisa**. São Paulo, v. 31, n. 3, p. 443-466, set/dez. 2005.

VAREJÃO, F. M. **Linguagem de programação: conceitos e técnicas**. Rio de Janeiro: Elsevier, 2004

WISEU, F.; OLIVEIRA, I. B. Open-ended tasks in the promotion of classroom communication in mathematics. **International Electronic Journal of Elementary Education**, v. 4, n. 2, p. 287, 2012.

WING, J. M. Computational thinking. **Communications of the ACM**, v. 49, n. 3, p. 33-35, 2006.

WINTERLE, P. **Vetores e geometria analítica**. 2. ed. São Paulo: Pearson Education do Brasil, 2014.

WOLFRAM, C. **Computer Based Math**. 2016. Disponível em <[http://computerbasedmath.org/images/resources/CBM\\_brochure.pdf](http://computerbasedmath.org/images/resources/CBM_brochure.pdf)>. Acesso em: 08 set. 2017.

## APÊNDICE A – Roteiros de Tarefas Investigativas planejados

---

### AUTOMATIZAÇÃO E PLOTAGEM DE GRÁFICOS DE FUNÇÃO DO 2º GRAU

RTI 03 - Roteiro de Tarefas Investigativas 03

Carga Horária prevista: 08 horas aula (50 Minutos cada)

#### DESCRIÇÃO E INFORMAÇÕES GERAIS

Sejam bem-vindos ao nosso terceiro Roteiro de Tarefas Investigativas (RTI). O Roteiro foi construído para facilitar sua aprendizagem. Para tanto, é necessário que você leia e o siga sistematicamente, sem pular nenhuma das etapas.

#### RESULTADOS PRETENDIDOS DA APRENDIZAGEM

- *Calcular* elementos da função do segundo grau;
- *Identificar* os comandos básicos da linguagem Python para automatizar soluções de funções do segundo grau;
- *Relacionar* os procedimentos dos cálculos matemáticos para solução de funções do segundo grau feitos em papel, com os procedimentos empregados em programação;
- *Aplicar* os comandos básicos da linguagem Python para automatizar soluções de funções do segundo grau e plotagem de gráficos com *Matplotlib*.

#### CONTEÚDOS ABORDADOS

Neste roteiro, vamos tratar de conteúdos relacionados à Matemática e à Programação, conforme descritos a seguir:

- Matemática:
  - Funções quadráticas;
  - Obtenção de raízes e vértice da função;
  - Plotagem do gráfico da função;
- Programação:
  - Estrutura sequencial de comandos em Python;

- Estrutura de decisão encadeada em Python (IF, ELIF, ELSE);
- Automatização de cálculos para resolução de funções quadráticas com utilização do pacote numpy.
- Automatização e plotagem do gráfico da função quadrática com utilização da biblioteca Matplotlib;

## RECURSOS DE APOIO

### Recurso Didático

Nº	Conteúdos	Tipo de Mídia
01	Funções do segundo grau	<a href="#">Tutorial</a>
02	Numpy Arange	<a href="#">Tutorial</a>
03	Estruturas de decisões IF, ELIF, ELSE	<a href="#">Vídeo</a>
04	Funções do segundo grau	<a href="#">Vídeo</a>

**Observação:** Conteúdos disponíveis na pasta Material de suporte

### Códigos utilizados no Roteiro

Nomes	Descrição
<a href="#">script1R03</a>	Raízes da função e coordenadas do vértice
<a href="#">script2R03</a>	Plotando gráfico da função variável $a > 0$
<a href="#">script3R03</a>	Plotando gráfico da função variável $a < 0$

**Observação:** Códigos disponíveis na pasta Material de suporte

## TAREFAS INVESTIGATIVAS

### Abrindo a trilha!

Vamos explorar um pouco a função do segundo grau, também conhecida por funções quadráticas. Em nosso estudo, vamos considerar apenas o conjunto de números Reais na resolução da função e plotagem do seu gráfico, sendo o valor de *delta*, pertencente ao conjunto dos números reais. Sabemos que no estudo de funções quadráticas ou do segundo grau, normalmente necessitamos extrair as raízes ou chamadas zeros da função. Por uma questão de padronização em nosso estudo, adotaremos as nomenclaturas  $x_1$  e  $x_2$  para as raízes ou zeros da função, ressaltando que quando tivermos uma raiz negativa, a chamaremos de  $x_1$ .

Para construção do gráfico, faremos uso das raízes da função e coordenadas do vértice.

2.0) Vamos começar com um problema simples.

Considere um número, cujo seu quadrado aumentado de seu seu triplo é igual a 4. Qual é esse número?

- a) Analisando o problema proposto, extraia do mesmo a função que o representa, descrevendo-a em papel.
- b) Resolva a equação que você montou e informe:
  - i) Os passos que você utilizou para a resolução
  - ii) As raízes da função;
  - iii) As coordenadas do vértice;
- c) Como você resolveu o problema? Utilizou algum recurso didático disponibilizado no roteiro?

Caso tenha alguma dúvida quanto aos passos para resolução de funções do 2º grau e ou construção de seu gráfico, leia o **tutorial01** e/ou assista ao **vídeo 04**.

Agora, utilizando o Wing IDE, abra e execute o script1R03. Provavelmente o script retornou um erro, não se preocupe, você irá corrigi-lo. Leia todas as linhas de código e seus respectivos comentários, isto o ajudará no entendimento do funcionamento do script.

1.5 Após a análise do script, responda as seguintes perguntas:

- i) O script consegue calcular as raízes da função? E as coordenadas do vértice? Por quê?
- j) Insira os códigos no script, para que o mesmo:
  - i) calcule as coordenadas do vértice e as imprima, considerando duas casas decimais;
  - ii) Como você resolveu o item *i*) desse problema?

- k) Salve seu script em sua pasta de respostas, atribua um nome significativo a ele e que identifique o item repondido.

1.6 No papel, esboce o gráfico da função extraída do problema item 1.0). Marque no gráfico os seguintes elementos:

- a) Os zeros da função;
- b) Coordenadas do vértice;

### **Combustível!**

Agora, abra e execute o script2R03. O script deve ter plotado um gráfico. Feche-o, note que o script imprimiu algumas informações. Observe a linha 33 do script, estamos utilizando a *função arange* na obtenção dos valores que compõe o intervalo entre  $x_1$  e  $x_2$  (raízes da função). Feche o script2R03. Para entender melhor a *função arange*, leia o **tutorial02**, em seguida, abra um novo script e faça os exemplos utilizados no tutorial, cada exemplo em um script. Não esqueça de importar as bibliotecas! Se preferir, altere os valores dos exemplos e verifique os resultados. Salve os scripts que você criou em sua pasta de respostas, utilize o a palavra combustivel na composição do nome do arquivo (script).

1.7 Execute o script2R03 e responda:

- a) As coordenadas do vértice e raízes da função foram impressas pelo script?
- b) Acrescente as linhas de código necessárias para para impressão das raízes e coordenadas do vértice da função, em seguida responda:
  - i) Como você respondeu a letra *b* do item 1.3)? Você utilizou algum outro script para respondê-la? Se positivo, informe-o.
- c) Você identifica a *função arange* no script? Em qual linha de código?
- d) Observe a linha de código seguinte a da *função arange*.
  - i) O que está sendo calculado nela? Qual a ligação com a linha anterior?
- e) Observe atentamente o gráfico plotado na execução do script2R03 e responda:

- i) O gráfico plotado é semelhante ao que você esboçou no papel?
  - ii) Quais elementos não foram plotados?
- f) Volte a analisar o script e em seguida responda:
- i) Quais linhas de código são responsáveis pela plotagem das marcações das raízes da função e coordenadas do vértice?
  - ii) Quais linhas de código são responsáveis pela plotagem da identificação das raízes da função ( $x_1$ ) e coordenadas do vértice ( $V$ )?
- g) Faça as alterações necessárias no código para que sejam plotadas as marcações e identificações das raízes da função e responda:
- i) Como você fez as alterações no código? Você efetuou testes para verificar se as alterações funcionaram corretamente?
- h) Analise o gráfico novamente e observe as legendas do eixo. As raízes da função e coordenadas do vértice estão plotadas no gráfico?
- i) As legendas estão corretas?
  - ii) Caso a resposta ao item acima seja negativa, quais as alterações necessárias no código para corrigi-las? Como você as fez? Informe o número das linhas de código alteradas.
- i) Salve seu script em sua pasta de respostas, atribua a ele um nome significativo e que identifique a questão (1.3).

### **Pegue seu Kit de Sobrevivência!**

- 1.8 Construa um programa que automatize a resolução e plotagem do gráfico da função do segundo grau, com as seguintes características:
- a) O usuário informe os valores das variáveis;
  - b) Calcule o valor de delta e verifique se o mesmo é negativo e imprima seu valor;

- c) Calcule e imprima os zeros da função;
- d) Calcule e imprima as coordenadas do vértice;
- e) Plote o gráfico da função com os seguintes características:
  - v) marcações e identificações das raízes da função;
  - vi) marcação e identificação do vértice;
  - vii) Identificação dos eixos  $x$  e  $y$ ;
  - viii) Nome do gráfico como: Gráfico Item 1.4.
- f) Salve seu script em sua pasta de resposta. Não esqueça de atribuir um nome significativo a ele.
- g) Agora responda:
  - i) Como você construiu seu script?
  - ii) Você utilizou outro script para construir o seu? Caso positivo, qual?
  - iii) Você verificou se seu script funciona? Como?
  - iv) Com quais valores você testou? Informe-os. Funcionou corretamente?

### **Explorando a Trilha**

Sabemos que nas funções do segundo grau, quando  $a$  é menos do que zero, a parábola tem sua concavidade voltada para baixo. Os scripts produzidos neste roteiro, que plotam o gráfico da função, precisam de alguns ajustes para plotar corretamente o gráfico, considerando funções onde o valor da variável  $a$  seja menor do que zero. Desta forma, vamos lembrar que a *função* `np.arange` em nosso script, calcula todos os valores que compõe o intervalo entre meus zeros da função ( $x_1$  e  $x_2$ ), necessários para a determinação dos meus pontos em  $y$ , utilizados na plotagem do gráfico.

- 1.9 Abra, observe atentamente as linhas de código e em seguida execute `script3R03`.
- a) O gráfico foi plotado?
  - b) Quais os valores que estão atribuídos as variáveis  $a$ ,  $b$  e  $c$ ?
  - c) No papel, faça um esboço da parábola que deveria ter sido plotada.

### **Pausa para o Lanche! Tutoria em um minuto!**

1.10 Agora, abra novamente o script3R03 e faça as alterações necessárias para que o mesmo possua as seguintes características:

- a) Usuário deve informar os valores das variáveis;
- b) Imprima dos valores de delta, zeros da função e coordenadas do vértice;
- c) Plote o gráfico da função, considerando a variável  $a < 0$ ;
  - i) Quais linhas você alterou? Por quê?
- d) Plote o gráfico da função com os seguintes características:
  - i) marcações e identificações das raízes da função;
  - ii) marcação e identificação do vértice;
  - iii) Identificação dos eixos x e y;
  - iv) Nome do gráfico como: Gráfico Item 1.6.
- e) Agora responda:
  - i) Você identificou algum padrão (techos de código que se repetem) nos scripts? Se sim, detalhe sua resposta, informando-os.
  - ii) Como você construiu seu script?
  - iii) Você utilizou outro script para construir o seu? Caso positivo, qual?
  - iv) Você verificou se seu script funciona? Como?
  - v) Com quais valores você testou? Informe-os. Funcionou corretamente?
- f) Salve o script em sua pasta de respostas. Não esqueça de atribuir um nome significativo a ele e que identifique sua resposta.

### **No final da trilha, um Oásis!**

Construímos até agora, de forma incremental, scripts que possibilitam a automatização de resolução e plotagem do gráfico de funções do segundo grau, no conjunto dos números reais. Foram necessários ajustes no script para que o mesmo seja capaz de plotar corretamente gráficos de funções, considerando a variável  $a > 0$  e situações

considerando a variável  $a < 0$ . Agora, para uma melhor automação, é necessário que façamos um script que verifique o valor da variável  $a$  e posteriormente execute a sequencia de códigos corretas, caso a variável  $a$  seja maior ou menor que zero. As estruturas de decisões, IF (se), ELIF (senão se) e Else(senão) podem ser utilizadas na construção de scripts como esses. Para compreender melhor, assista ao **vídeo 03**, disponível na pasta Material de suporte.

1.11 Considere o seguinte problema:

Para facilitar a correção de suas provas e resolução de exercícios, você irá criar um programa que automatize a resolução de funções do segundo grau e plotagem de seu gráfico, dentro do conjunto dos números reais. O programa deverá apresentar as seguintes características:

- a) Solicitar do usuário os valores das variáveis  $a$ ,  $b$  e  $c$ , da função quadrática;
- b) Calcular, verificar e imprimir o valor de delta, se negativo ou positivo. Com duas casas decimais;
- c) Verificar o valor da variável  $a$  e, se maior ou menor que zero, tomar a decisão correta;
- d) Calcular e imprimir as raízes da função e coordenadas do vértice. Com duas casas decimais;
- e) Plotar o gráfico da função com as seguintes características:
  - i) marcações e identificações das raízes da função;
  - ii) marcação e identificação do vértice;
  - iii) Identificação dos eixos  $x$  e  $y$ ;
  - iv) Nome do gráfico a seu critério.
- f) Salve o script em sua pasta de respostas. Não esqueça de atribuir um nome significativo a ele e que identifique sua resposta.
- g) Como você construiu o script? Você conseguiu desenvolver todo o programa sozinho? Caso sua resposta seja não, em que momento precisou

recorrer aos scripts já trabalhados? Qual o script você utilizou?

- h) O script que você construiu é diferente dos que já vimos até agora? Por quê?

Referências dos Recursos de Apoio disponibilizados no roteiro.

<b>Nº</b>	<b>Conteúdos</b>	<b>Acesso</b>
01	Funções do segundo grau	<a href="#">Link</a>
02	Numpy Arange	<a href="#">Link</a>
03	Estruturas de decisões IF, ELIF, ELSE	<a href="#">Link</a>
04	Funções do segundo grau	<a href="#">Link</a>

## APÊNDICE B – Roteiros de Tarefas Investigativas aplicados

### AUTOMATIZAÇÃO E PLOTAGEM DE GRÁFICOS DE FUNÇÃO DO 2º GRAU

RTI 03 - Roteiro de Tarefas Investigativas 03

Carga Horária prevista: 08 horas aula (50 Minutos cada)

**Nome do Estudante:** \_\_\_\_\_

**Curso:** \_\_\_\_\_ **Turma:** \_\_\_\_\_ **Data:** \_\_\_\_\_

#### DESCRIÇÃO E INFORMAÇÕES GERAIS

Sejam bem-vindos ao nosso terceiro Roteiro de Tarefas Investigativas (RTI). O Roteiro foi construído para facilitar sua aprendizagem. Para tanto, é necessário que você leia e o siga sistematicamente, sem pular nenhuma das etapas.

#### RESULTADOS PRETENDIDOS DA APRENDIZAGEM

- *Calcular* elementos da função do segundo grau;
- *Identificar* os comandos básicos da linguagem Python para automatizar soluções de funções do segundo grau;
- *Relacionar* os procedimentos dos cálculos matemáticos para solução de funções do segundo grau feitos em papel, com os procedimentos empregados em programação;
- *Aplicar* os comandos básicos da linguagem Python para automatizar soluções de funções do segundo grau e plotagem de gráficos com *Matplotlib*.

#### CONTEÚDOS ABORDADOS

Neste roteiro, vamos tratar de conteúdos relacionados à Matemática e à Programação, conforme descritos a seguir:

- Matemática:
  - Funções quadráticas;
  - Obtenção de raízes e vértice da função;
  - Plotagem do gráfico da função;
- Programação:
  - Estrutura sequencial de comandos em Python;

- Estrutura de decisão encadeada em Python (IF, ELIF, ELSE);
- Automatização de cálculos para resolução de funções quadráticas com utilização do pacote numpy.
- Automatização e plotagem do gráfico da função quadrática com utilização da biblioteca Matplotlib;

## RECURSOS DE APOIO

### Recurso Didático

Nº	Conteúdos	Tipo de Mídia
01	Funções do segundo grau	<a href="#">Tutorial</a>
02	Numpy Arange	<a href="#">Tutorial</a>
03	Estruturas de decisões IF, ELIF, ELSE	<a href="#">Vídeo</a>
04	Funções do segundo grau	<a href="#">Vídeo</a>
<b>Observação:</b> Conteúdos disponíveis na pasta Material de suporte		

### Códigos utilizados no Roteiro

Nomes	Descrição
<a href="#">script1R03</a>	Raízes da função e coordenadas do vértice
<a href="#">script2R03</a>	Plotando gráfico da função variável $a > 0$
<a href="#">script3R03</a>	Plotando gráfico da função variável $a < 0$
<b>Observação:</b> Códigos disponíveis na pasta Material de suporte	

## TAREFAS INVESTIGATIVAS

### Abrindo a trilha!

Vamos explorar um pouco a função do segundo grau, também conhecida por funções quadráticas. Em nosso estudo, vamos considerar apenas o conjunto de números Reais na resolução da função e plotagem do seu gráfico, sendo o valor de *delta*, pertencente ao conjunto dos números reais. Sabemos que no estudo de funções quadráticas ou do segundo grau, normalmente necessitamos extrair as raízes ou chamadas zeros da função. Por uma questão de padronização em nosso estudo, adotaremos as nomenclaturas  $x_1$  e  $x_2$  para as raízes ou zeros da função, ressaltando que quando tivermos uma raiz negativa, a chamaremos de  $x_1$ .

Para construção do gráfico, faremos uso das raízes da função e coordenadas do vértice.

3.0) Vamos começar com o seguinte problema.

Considere um número, cujo seu quadrado aumentado de seu seu triplo é igual a 4. Qual é esse número?

- a) Analisando o problema proposto, extraia do mesmo a função que o representa, descrevendo-a em papel.
- b) Resolva a equação que você montou e informe:
  - iv) Os passos que você utilizou para a resolução;
  - v) As raízes da função;
  - vi) As coordenadas do vértice;
- c) Como você resolveu o problema? Utilizou algum recurso didático disponibilizado no roteiro?

Caso tenha alguma dúvida quanto aos passos para resolução de funções do 2º grau e ou construção de seu gráfico, leia o **tutorial01** e/ou assista ao **vídeo 04**.

Agora, utilizando o Wing IDE, abra e execute o script1R03. Provavelmente o script retornou um erro, não se preocupe, você irá corrigi-lo. Leia todas as linhas de código e seus respectivos comentários, isto o ajudará no entendimento do funcionamento do script.

1.3 Após a análise do script, responda as seguintes perguntas:

- e) No script, existe outra forma de calcular o valor de delta. Responda:
  - iii) Qual seria?
  - iv) Faça a alteração no script.
  - v) As alterações que você fez funcionaram corretamente? Por quê?
- f) No script, existe outra forma de extrair a raiz de delta. Responda:
  - i) Qual seria?

- ii) Faça a alteração no script.
- iii) As alterações que você fez funcionaram corretamente? Por quê?
- g) O script consegue calcular as raízes da função? E as coordenadas do vértice? Por quê?
- h) Insira os códigos no script, para que o mesmo:
  - iii) Calcule as raízes da função;
  - iv) Calcule as coordenadas do vértice e as imprima, considerando duas casas decimais;
  - v) Como você resolveu o item *i*) desse problema?
  - vi) Como você resolveu o item *ii*) desse problema?
  - vii) Você verificou se funciona? Quais resultados foram impressos? Informe-os.
- i) Salve seu script em sua pasta de respostas, atribua um nome significativo a ele e que identifique o item repondido.

1.4 No papel, esboce o gráfico da função extraída do problema item 1.0). Marque no gráfico os seguintes elementos:

- c) Os zeros da função;
- d) Coordenadas do vértice;

### **Combustível!**

Agora, abra e execute o script2R03. O script deve ter plotado um gráfico. Feche-o, note que o script imprimiu algumas informações. Observe a linha 33 do script, estamos utilizando a *função arange* na obtenção dos valores que compõe o intervalo entre  $x_1$  e  $x_2$  (raízes da função). Feche o script2R03. Para entender melhor a *função arange*, leia o **tutorial02**, em seguida, abra um novo script e faça os exemplos utilizados no tutorial, cada exemplo em um script. Não esqueça de importar as bibliotecas! Altere os valores dos exemplos e verifique os resultados. Salve os scripts que você criou em sua pasta de respostas, utilize o a palavra *combustivel* na composição do nome do arquivo (script).

1.5 Execute o script2R03 e responda:

- j) As coordenadas do vértice e raízes da função foram impressas pelo script? Por quê?
- k) Acrescente as linhas de código necessárias para para impressão das raízes e coordenadas do vértice da função, em seguida responda:
  - ii) Como você respondeu a letra *b* do item 1.5)? Você utilizou algum outro script para respondê-la? Se positivo, informe-o.
- l) Você identifica a *função arange* no script? Em qual linha de código?
- m) Observe a linha de código seguinte a da *função arange*.
  - ii) O que está sendo calculado nela? Qual a sua relação com a linha anterior?
- n) Observe atentamente o gráfico plotado na execução do script2R03 e responda:
  - iii) O gráfico plotado é semelhante ao que você esboçou no papel?
  - iv) Sua concavidade é para cima ou para baixo? Por quê?
  - v) Quais elementos não foram plotados? Por quê?
- o) Volte a analisar o script e em seguida responda:
  - iii) Qual comando responsável pela plotagem do gráfico neste script?
  - iv) Quais linhas de código são responsáveis pela plotagem das marcações das raízes da função e coordenadas do vértice?
  - v) Quais linhas de código são responsáveis pela plotagem da identificação das raízes da função ( $x_1$ ) e coordenadas do vértice ( $V$ )?
- p) Faça as alterações necessárias no código para que sejam plotadas as marcações e identificações das raízes da função e responda:
  - ii) Como você fez as alterações no código?
  - iii) Você efetuou testes para verificar se as alterações funcionaram corretamente?

- q) Analise o gráfico novamente e observe as legendas do eixo.
  - iii) As legendas estão corretas? Por quê?
  - iv) Caso a resposta ao item acima seja negativa, quais as alterações necessárias no código para corrigi-las? Como você as fez? Informe o número das linhas de código alteradas.
- r) Salve seu script em sua pasta de respostas, atribua a ele um nome significativo e que identifique a questão.

### **Pegue seu Kit de Sobrevivência!**

- 1.6 Construa um programa que automatize a resolução e plotagem do gráfico da função do segundo grau, com as seguintes características:
- h) O usuário informe os valores das variáveis;
  - i) Calcule o valor de delta e verifique se o mesmo é negativo e imprima seu valor;
  - j) Calcule e imprima os zeros da função;
  - k) Calcule e imprima as coordenadas do vértice;
  - l) Plote o gráfico da função com os seguintes características:
    - ix) marcações e identificações das raízes da função;
    - x) marcação e identificação do vértice;
    - xi) Identificação dos eixos x e y;
    - xii) Nome do gráfico como: Gráfico Item 1.6.
  - m) Salve seu script em sua pasta de resposta. Não esqueça de atribuir um nome significativo a ele.
  - n) Agora responda:
    - v) Como você construiu seu script?
    - vi) Você utilizou outro script para construir o seu? Caso positivo, qual?
    - vii) Você verificou se seu script funciona? Como?

- viii) Com quais valores você testou? Informe-os.
- ix) Funcionou corretamente?

### Explorando a Trilha

Sabemos que nas funções do segundo grau, quando  $a$  é menor do que zero, a parábola tem sua concavidade voltada para baixo. Os scripts produzidos neste roteiro, que plotam o gráfico da função, precisam de alguns ajustes para plotar corretamente o gráfico, considerando funções onde o valor da variável  $a$  seja menor do que zero. Desta forma, vamos lembrar que a função *np.arange* em nosso script, calcula todos os valores que compõe o intervalo entre meus zeros da função ( $x_1$  e  $x_2$ ), necessários para a determinação dos meus pontos em  $y$ , utilizados na plotagem do gráfico.

- 1.7 Abra, observe atentamente as linhas de código e em seguida execute script3R03.
- d) O gráfico foi plotado? Por quê?
  - e) Quais os valores que estão atribuídos as variáveis  $a$ ,  $b$  e  $c$ ?
  - f) No papel, faça um esboço da parábola que deveria ter sido plotada.

### Pausa para o Lanche! Tutoria em um minuto!

- 1.8 Agora, abra novamente o script3R03 e faça as alterações necessárias para que o mesmo possua as seguintes características:
- g) Usuário deve informar os valores das variáveis;
  - h) Imprima dos valores de delta, zeros da função e coordenadas do vértice;
  - i) Plote o gráfico da função, considerando a variável  $a < 0$ ;
    - ii) Quais linhas você alterou? Por quê?
  - j) Plote o gráfico da função com os seguintes características:
    - v) marcações e identificações das raízes da função;
    - vi) marcação e identificação do vértice;
    - vii) Identificação dos eixos  $x$  e  $y$ ;
    - viii) Nome do gráfico como: Gráfico Item 1.8.

k) Agora responda:

vi) Você identificou algum padrão (techos de código que se repetem) nos scripts? Se sim, detalhe sua resposta, informando-os.

vii) Como você construiu seu script?

viii) Você utilizou outro script para construir o seu? Caso positivo, qual?

ix) Você verificou se seu script funciona? Como?

x) Com quais valores você testou? Informe-os.

xi) Funcionou corretamente?

l) Salve o script em sua pasta de respostas. Não esqueça de atribuir um nome significativo a ele e que identifique sua resposta.

### **No final da trilha, um Oásis!**

Construímos até agora, de forma incremental, scripts que possibilitam a automatização de resolução e plotagem do gráfico de funções do segundo grau, no conjunto dos números reais. Foram necessários ajustes no script para que o mesmo seja capaz de plotar corretamente gráficos de funções, considerando a variável  $a > 0$  e situações considerando a variável  $a < 0$ . Agora, para uma melhor automação, é necessário que façamos um script que verifique o valor da variável  $a$  e posteriormente execute a sequência de códigos corretas, caso a variável  $a$  seja maior ou menor que zero. As estruturas de decisões, IF (se), ELIF (senão se) e Else (senão) podem ser utilizadas na construção de scripts como esses. Para compreender melhor, assista ao **vídeo 03**, disponível na pasta Material de suporte.

1.9 Considere o seguinte problema:

Para facilitar a correção de suas provas e resolução de exercícios, você irá criar um programa que automatize a resolução de funções do segundo grau e plotagem de seu gráfico, dentro do conjunto dos números reais. O programa deverá apresentar as seguintes características:

i) Solicitar do usuário os valores das variáveis  $a$ ,  $b$  e  $c$ , da função quadrática;

- j) Calcular, verificar e imprimir o valor de delta, se negativo ou positivo. Com duas casas decimais;
- k) Verificar o valor da variável  $a$  e, se maior ou menor que zero, tomar a decisão correta;
- l) Calcular e imprimir as raízes da função e coordenadas do vértice. Com duas casas decimais;
- m) Plotar o gráfico da função com as seguintes características:
- v) marcações e identificações das raízes da função;
  - vi) marcação e identificação do vértice;
  - vii) Identificação dos eixos  $x$  e  $y$ ;
  - viii) Nome do gráfico a seu critério.
- n) Salve o script em sua pasta de respostas. Não esqueça de atribuir um nome significativo a ele e que identifique sua resposta.
- o) Como você construiu o script? Você conseguiu desenvolver todo o programa sozinho? Caso sua resposta seja não, em que momento precisou recorrer aos scripts já trabalhados? Qual o script você utilizou?
- p) O script que você construiu é diferente dos que já vimos até agora? Por quê?

Referências dos Recursos de Apoio disponibilizados no roteiro.

Nº	Conteúdos	Acesso
01	Funções do segundo grau	<a href="#">Link</a>
02	Numpy Arange	<a href="#">Link</a>
03	Estruturas de decisões IF, ELIF, ELSE	<a href="#">Link</a>
04	Funções do segundo grau	<a href="#">Link</a>

## APÊNDICE C – Avaliação Diagnóstica

### AUTOMATIZAÇÃO DE RESOLUÇÃO DE FUNÇÕES QUADRÁTICAS E PLOTAGEM DO GRÁFICO. AUTOMATIZAÇÃO DE OBTENÇÃO DE ELEMENTOS DA ELIPSE E PLOTAGEM DE SEU GRÁFICO

Diagnóstico

#### OBJETIVOS

- Verificar se o estudante é capaz de resolver equações algébricas e construir gráficos;
- Verificar se o estudante é capaz de automatizar a resolução de equações algébricas e plotagem de gráficos.

#### RECURSOS DE APOIO

Recurso Didático

Nomes	Descrição
scriptPreFun	Esboço de código funções
scriptPreElls	Esboço de código elipses
<b>Observação:</b> Códigos disponíveis na pasta Material de suporte	

#### Atividades:

1.1 Considere o seguinte:

Dada uma função do tipo  $f(x) = 2x^2 + 3x - 4$ , faça:

- q) Calcule e informe o valor de delta e dos zeros da função;
- r) Calcule e informe as coordenadas do vértice;
- s) Esboce o gráfico da função e marque no gráfico os zeros da função e as coordenadas do vértice;

1.2 Considere a seguinte situação:

Uma elipse com as coordenadas do centro na origem, sendo eixo maior vertical medindo 4 metros e eixo menor medindo 2 metros.

- a) Analise a situação acima e faça:
- i) Organize os dados do problema em forma de uma equação reduzida da Elipse;
  - ii) Calcule e apresente as coordenadas do centro;
  - iii) Calcule e apresente as coordenadas dos focos;
  - iv) Calcule e apresente a distância focal;
  - v) Esboce o gráfico da Elipse.

1.3 Você sabe programar?

- a) ( ) sim                      b) ( ) não

**Caso sua resposta seja Sim, passe para as próximas questões.**

Para resolução do item 1.4), sugerimos a utilização do **scriptPreFun**.

1.4 Considerando ainda a função do item 1.1), utilizando a programação, desenvolva um programa com as seguintes características:

- a) Que solicite os valores das variáveis ao usuário;
- b) Calcule, verifique e imprima o valor de delta. Caso seja negativo, informe ao usuário que no conjunto dos números reais, não é possível extrair a raiz quadrada de números negativos;
- c) Calcule e imprima as raízes da função e coordenadas do vértice.
- d) Plote o gráfico da função com as seguintes características:
  - ix) marcações e identificações das raízes da função;

- x) marcação e identificação do vértice;
- e) Salve o script em sua pasta de respostas.

Para resolução do item 1.5), sugerimos a utilização do **scriptPreElls**.

- 1.5 Utilizando a programação, construa um programa que automatize a obtenção de elementos da Elipse, considerando equações como a que você organizou no item 1.2) letra a) número i). O programa deverá possuir as seguintes características:
- a) Solicitar ao usuário os valores das variáveis;
  - b) Calcular e imprimir as coordenadas dos focos;
  - c) Calcular e imprimir a distância focal;
  - d) Calcular e imprimir a dimensão dos eixos;
  - e) Plotar o gráfico da Elipse, marcar e plotar os seguintes pontos:
    - i) Centro;
    - ii) Focos;
    - iii) Limites dos eixos x e y;
  - f) Salve seu script em sua pasta de respostas, não esqueça de atribuir um nome significativo a ele.

## APÊNDICE D – Avaliação Planejada Roteiros 1,2 e 3

### AUTOMATIZAÇÃO DE RESOLUÇÃO DE FUNÇÕES QUADRÁTICAS E PLOTAGEM DO GRÁFICO

Roteiros 1, 2 e 3

#### OBJETIVOS

- Verificar se o estudante é capaz de automatizar a obtenção dos elementos principais da Elipse utilizando a programação, considerando as equações reduzidas da Elipse;
- Verificar se o Estudante é capaz de automatizar a plotagem de gráficos e marcações dos elementos da Elipse no mesmo.

#### RECURSOS DE APOIO

Recurso Didático

Nomes	Descrição
scriptPosFun	Esboço de código
<b>Observação:</b> Códigos disponíveis na pasta Material de suporte	

#### Atividades:

- 1.1 Dada função  $fx = -2x^2 - 2 + 3$ , faça:
- Calcule manualmente e informe os valores de delta e dos zeros da função;
  - Calcule e informe as coordenadas do vértice;
  - Esboce o gráfico da função;

Para resolução do item 1.2), sugerimos a utilização do **scriptPosFun**.

- 1.2 Considere o seguinte problema:

Você deve desenvolver um programa que automatize a resolução de funções do segundo grau e plotagem de seu gráfico, dentro do conjunto dos números reais, que resolva e plote gráficos somente de funções cujo valores da

variável  $a$  seja menor que zero. O programa ainda deverá apresentar as seguintes características:

- a) Solicitar do usuário os valores das variáveis  $a$ ,  $b$  e  $c$ , da função quadrática;
- b) Verificar o valor da variável  $a$  e caso não atenda o solicitado no enunciado do problema, imprimir e informar ao usuário;
- c) Calcular, verificar e imprimir o valor de delta, se negativo ou positivo. Com uma casa decimal;
- d) Calcular e imprimir as raízes da função e coordenadas do vértice. Com duas casas decimais;
- e) Plotar o gráfico da função com as seguintes características:
  - iv) marcações e identificações das raízes da função;
  - v) marcação e identificação do vértice;
  - vi) Legendas com nomes significativos;
- f) Salve o script em sua pasta de respostas.

## APÊNDICE E – Avaliação Executada dos Roteiros 1, 2 e 3

### AUTOMATIZAÇÃO DE RESOLUÇÃO DE FUNÇÕES QUADRÁTICAS E PLOTAGEM DO GRÁFICO

Roteiros 1, 2 e 3

#### OBJETIVOS

- Verificar se o estudante é capaz de automatizar a resolução de funções do segundo grau;
- Verificar se o Estudante é capaz de automatizar a plotagem de gráficos e marcações dos elementos de uma função do segundo grau.

#### RECURSOS DE APOIO

Recurso Didático

Nomes	Descrição
script_ava	Esboço de código
<b>Observação:</b> Códigos disponíveis na pasta Material de suporte	

#### Tarefas:

##### PARTE 1 – Resolução de equação a mão:

1.0 Considerando uma função  $f(x) = -2x^2 - 2x + 3$ , responda:

- Qual a direção da concavidade da parábola? Por quê?
- Calcule manualmente e informe os valores de delta e dos zeros da função;
- Calcule e informe as coordenadas do vértice;
- Esboce o gráfico da função;

##### PARTE 2 – Automatização de equações

Para automatizar o programa pedido no item 1.1, você pode fazer uso do **script\_eq**, que possui parte de códigos, incompletos, que você pode utilizar e modificar de acordo com a lógica que irá implementar no programa.

Para resolução do item 1.1), utilize o **script\_ava**.

1.1 Dada a seguinte situação:

Considerando que um professor de Matemática passou um problema envolvendo funções do segundo grau e que você por estudar python, gostaria de implementar por meio de programação. Desenvolva então um programa que automatize a resolução de funções do segundo grau, dentro do conjunto dos números reais, cujo valores da variável  $a$  seja menor que zero. O programa ainda deverá apresentar as seguintes características:

- g) Solicitar do usuário os valores das variáveis da respectiva função do segundo grau;
- h) Verificar o valor da variável  $a$  e caso não atenda o solicitado no enunciado do problema, imprimir seu valor e informar ao usuário que o programa só resolve equações cujo valor de  $a$  seja menor do que zero;
- i) Calcular, verificar e imprimir o valor de delta, se negativo ou positivo. Com duas casas decimais. Não esqueça de que o valor de delta também deverá ser verificado;
- j) Calcular e imprimir as raízes da função e coordenadas do vértice. Com duas casas decimais;
- k) Teste o programa com entrada de valores diversos, inclusive considerando a função  $f(x) = -2x^2 - 2x + 3$
- l) Salve seu script como (salvar como), atribuindo um nome significativo (exemplo: Parte 2)

### PARTE 3 – Plotagem de gráfico

Para a plotagem de gráfico, você pode fazer uso dos trechos de código, incompletos, disponibilizados no arquivo em bloco de notas denominado **graf.txt** que podem lhe ajudar a pensar em como automatizar, complementando assim o programa que você desenvolveu na parte 2 do roteiro.

#### 1.2 Dada a seguinte situação:

Agora, considerando uma continuidade do problema exposto na parte 2 do roteiro, o professor de Matemática solicitou a construção do gráfico da função. Por estudar python, você irá implementar a solução do problema por meio de programação. Lembre-se de que estamos trabalhando somente com funções que possuem o valor da variável  $a$  menor do que zero, com as características expostas abaixo.  
Obs: (Mantenha linhas e grades no ambiente de plotagem)

- a) Plotar o gráfico da função com as seguintes características:
- vii) marcações e identificações das raízes da função;
  - viii) marcação e identificação do vértice;
  - ix) Legendas com nomes significativos;
- b) Salve o script como (salvar como )em sua pasta de respostas, atribuindo um nome significativo e que identifique a questão (Ex: script\_parte3).

#### **PARTE 4 – Análise e reflexão**

- 1.3 Considerando o roteiro e o que você desenvolveu em cada parte, responda o que se pede abaixo.
- i) Analise seu programa resposta da parte 2 do roteiro e o programa **script\_eq** (fornecido). Quais os comandos inseridos para que o mesmo funcionasse corretamente? Responda de forma sintética.
  - ii) Explique a lógica que você utilizou na estrutura do IF/ELSE (ELIF)?
  - iii) Os limites da área do gráfico, conforme fornecido no bloco de notas (**graf.txt**) estavam corretos para a resolução do problema proposto? Em caso negativo, quais as alterações que você fez? Por quê?
  - iv) O script que você criou na parte 3 do roteiro funciona corretamente? Justifique sua resposta. **Obs:** Caso sua resposta seja negativa, descreva as razões que impediram de fazê-lo funcionar.
  - v) Caso seu script produzido na parte 3 do roteiro tenha funcionado corretamente, quais foram os valores que você atribuiu às variáveis para testá-lo?

**APÊNDICE F – Questionário Socioacadêmico**

**QUESTIONÁRIO SOCIOACADÊMICO**

Caro estudante: todos os dados fornecidos neste questionário são confidenciais e serão utilizados somente para fins acadêmicos. Em nenhuma hipótese os respondentes serão identificados.

<b>Curso:</b>	<b>Turma:</b>	<b>Turno:</b>
<b>Nome:</b>		
<b>Número celular:</b>	<b>whatsapp? ( ) Sim ( ) Não</b>	
<b>Outro Número de telefone:</b>		
<b>E-mail:</b>		

**Leia e responda as questões abaixo:**

**1. Qual a sua idade?**

R: \_\_\_\_\_ anos

**2. Localização da sua residência:**

a) ( ) Zona Urbana                      b) ( ) Zona Rural

Município: \_\_\_\_\_ Estado: \_\_\_\_\_

**3. Qual o meio de transporte que você utiliza para ir à escola?**

a) ( ) Carro ou Moto                      c) ( ) A pé                      e) ( ) Carona

b) ( ) Ônibus ou Van                      d) ( ) Bicicleta

**4. Você trabalha?**

a) ( ) Sim                                      b) ( ) Não

**5. Caso afirmativo, qual sua jornada de trabalho diária?**

a) ( ) 4 horas                                  c) ( ) 8 horas

b) ( ) 6 horas

**6. Fora da escola, para estudar e fazer pesquisas, você dispõe de computador:**

- a) ( ) Na minha residência      b) ( ) No meu trabalho  
c) ( ) Lan house                      d) ( ) Residência de amigos ou parentes  
e) ( ) Biblioteca pública          f) ( ) Não tenho acesso a computador fora da escola

**7. Fora da escola, você utiliza conexão de internet:**

- a) ( ) Banda Larga (OI ou Rádio)                      b) ( ) 3/4 G  
c) ( ) Não utilizo internet fora da escola

**8. Você cursou o ensino médio:**

- a) ( ) Todo em escola pública;      c) ( ) Todo em escola particular;  
b) ( ) Parte em escola pública, parte em escola particular.

**9. Você estudou equações e funções do 2º grau:**

- a) ( ) Apenas no ensino fundamental e médio      b) ( ) Ensino médio e graduação  
c) ( ) Não estudei

**10. Você estudou Elipses em Geometria Analítica:**

- a) ( ) Apenas no ensino médio                      b) ( ) Ensino médio e graduação  
c) ( ) Não estudei

**11. Fora da sala de aula, qual a média de horas semanais que você estuda?**

- a) ( ) \_\_\_\_ horas                      b) ( ) não estudo fora de sala

**12. Já teve algum contato com programação de computadores?**

- a) ( ) Sim                      b) ( ) Não

**13. Caso afirmativo, o contato com programação foi por meio:**

- a) ( ) presencial: curso, disciplina, oficinas etc.      c) ( ) Aprendi a programar sozinho.  
b) ( ) Programação em plataforma on-line.

**14. Caso afirmativo, qual ou quais linguagens de programação já utilizou?**

- a) ( ) Python                      c) ( ) C                      e) ( ) Pascal  
b) ( ) Java                      d) ( ) Delphi      f) ( ) Outras: \_\_\_\_\_

**15. Você se identifica com o curso de licenciatura em Matemática?**

- a) ( ) Sim, já atuo ou pretendo atuar na área      b) ( ) Sim, mas não pretendo atuar  
c) ( ) Não me identifico com o curso.

**16. Deixe seu comentário sobre o que julgar necessário:**

Somos gratos pelas contribuições!

## APÊNDICE G – Termo de Consentimento Livre e Esclarecido

### Termo de Consentimento Livre e Esclarecido

**PESQUISA:** Aliando Programação e Atividades Investigativas para automatização de Equações Algébricas e Plotagem de Gráficos: Um Estudo na Licenciatura em Matemática.

**COORDENAÇÃO:** Professor Cristiano José Ferreira - IFAC

**ORIENTAÇÃO:** Professora Dra. Andréa Pereira Mendonça – IFAM

1. **NATUREZA DA PESQUISA:** Você está sendo convidado a participar desta pesquisa que, tem como finalidade, investigar em que aspectos o processo de ensino que integra programação de computadores melhorou a aprendizagem dos estudantes de Licenciatura em Matemática para automatizar a resolução de equações algébricas e plotar gráficos.

2. **APLICAÇÃO DA PESQUISA:** A presente pesquisa será realizada no Instituto Federal de Educação, Ciência e Tecnologia do Acre-Campus Cruzeiro do Sul. As atividades serão desenvolvidas como parte da disciplina de Tecnologia da Informação e Comunicação Aplicadas ao Ensino.

3. **PARTICIPANTES DA PESQUISA:** Participarão desta pesquisa estudantes de licenciatura em Matemática, do Instituto Federal de Educação, Ciência e Tecnologia do Acre – Campus Cruzeiro do Sul.

4. **ENVOLVIMENTO NA PESQUISA:** Ao participar deste estudo você será convidado a preencher um questionário sócio acadêmico, juntamente com outros estudantes que aceitaram participar da pesquisa. Ao longo da pesquisa, serão desenvolvidas atividades no contexto de ensino e aprendizagem, envolvendo resolução de problemas e construção de programas de computadores. Estas atividades ficarão de posse do pesquisador.

5. **SOBRE O QUESTIONÁRIO:** Serão solicitadas algumas informações básicas e perguntas de múltipla escolha ou escolha simples sobre suas condições sociais e acadêmicas. Em momento algum, os participantes desta pesquisa serão identificados ou terão os dados publicados nominalmente.

6. RISCOS E DESCONFORTO: A participação nesta pesquisa não traz complicações legais de nenhuma ordem. Nenhum dos procedimentos utilizados oferece riscos à sua dignidade ou a sua saúde.

7. CONFIDENCIALIDADE: Todas as informações coletadas nesta investigação são estritamente confidenciais. Acima de tudo interessam os dados coletivos e não aspectos particulares de cada estudante. Os dados serão analisados apenas para fins acadêmicos. O objetivo da pesquisa é melhorar os processos e recursos destinados ao ensino e aprendizagem dos estudantes de licenciatura. Sua identidade não será divulgada ou publicada, exceto se devidamente autorizado.

8. BENEFÍCIOS: Ao participar desta pesquisa, você terá oportunidade de desenvolver habilidades de programação para a resolução de equações algébricas e plotagem de gráficos, numa perspectiva de resolução de problemas pautados na exploração e investigação. O desenvolvimento dessas habilidades pode contribuir com a sua formação, no sentido em que ampliam as possibilidades para utilizar a tecnologia no processo de ensino e aprendizagem, corroborando com movimentos nacionais e internacionais que apoiam e incentivam a aprendizagem de programação. Posteriormente, acreditamos que os resultados produzidos nesta pesquisa poderão contribuir com a formação de outros estudantes.

9. PAGAMENTO: Você não terá nenhum tipo de despesa por participar deste estudo, bem como não receberá nenhum tipo de pagamento por sua participação.

Após estes esclarecimentos, solicitamos sua contribuição com a referida pesquisa. Para tanto, preencha os itens que se seguem: **CONSENTIMENTO LIVRE E ESCLARECIDO**

Tendo em vista os itens acima apresentados, eu, de forma livre e esclarecida, concordo com os termos supracitados e confirmo minha participação na pesquisa.

---

Nome do estudante (Por extenso)

Cruzeiro do Sul, \_\_\_\_\_, \_\_\_\_\_ de 2017.

---

Cristiano José Ferreira

Cruzeiro do Sul, , de 2017.

Agradecemos a sua colaboração e colocamo-nos à disposição para esclarecimentos adicionais. O pesquisador responsável por esta pesquisa é o Prof. Cristiano José Ferreira mestrando do MPET do Instituto Federal do Amazonas, sob orientação da Professora Dra. Andréa Pereira Mendonça, docente do Mestrado Profissional em Ensino Tecnológico do Instituto Federal do Amazonas IFAM.

### APENDICE H – Questionário de Experiências e Melhorais

Caro estudante: todos os dados fornecidos neste questionário são confidenciais e serão utilizados somente para fins acadêmicos. Em nenhuma hipótese, os respondentes serão identificados.

<b>Curso:</b>	<b>Turma:</b>	<b>Turno:</b>
<b>Nome:</b>		

**1. Fora da sala de aula, qual a média de horas semanais que você estudou para nosso curso?**

- a) ( ) \_\_\_\_ horas                      b) ( ) não estudo fora de sala

**2. No contexto da Matemática, você considera que seus conhecimentos de programação de computadores para resolução de equações algébricas são:**

- a) ( ) Avançados                      c) ( ) Intermediário  
b) ( ) Básicos                          d) ( ) Insuficientes

**3. No contexto da Matemática, você considera que seus conhecimentos de programação de computadores para plotagem de gráficos são:**

- a) ( ) Avançados                      c) ( ) Intermediário  
b) ( ) Básicos                          d) ( ) Insuficientes

Aspectos do Roteiro

**4. As instalações dos programas relacionados a Python nos computadores da biblioteca contribuíram ou facilitaram seus estudos durante o curso?**

- a) ( ) Sim, utilizei sempre que necessário.  
b) ( ) Não, não foi necessária a utilização.

**5. O roteiro foi organizado de forma sistemática/ lógica e favoreceu a aprendizagem?**

- a) ( ) Sim, foi suficiente para favorecer a aprendizagem  
b) ( ) Sim, mas são necessárias adequações.  
c) ( ) Sim, mas não favoreceu a aprendizagem  
d) ( ) Não, não favoreceu a aprendizagem

**6. Caso sua resposta a questão anterior tenha ficado entre as alternativas b, c e d, justifique sua resposta indicando melhorias.**

R:

**7. Os materiais de suporte (vídeos, tutorias, exemplos de código), foram suficientes?**

- a)  Sim, foram suficientes.
- b)  Sim, mas busquei outros materiais para complementar.
- c)  Não, tive que buscar outros materiais.
- d)  Não, mas não busquei outros materiais.

**8. Caso sua resposta ao item anterior tenha sido a letra c, informe os tipos (vídeos, tutorias, exemplos de código) e conteúdo (equação do segundo grau, elipse etc.) que pesquisou.**

R:

**9. A abordagem utilizada no roteiro, baseada em investigação matemática, favoreceu sua aprendizagem quanto à automatização de equações algébricas.**

- a)  Discordo totalmente
- b)  Discordo
- c)  De acordo
- d)  Totalmente de acordo

**10. A abordagem utilizada no roteiro, baseada em investigação matemática, favoreceu sua aprendizagem quanto à automatização de plotagem de gráficos.**

- a)  Discordo totalmente
- b)  Discordo
- c)  De acordo
- d)  Totalmente de acordo

**11. A organização das aulas em roteiros, como os utilizados no curso, considera o estudante como um sujeito ativo na construção do seu conhecimento.**

- a)  Discordo totalmente
- b)  Discordo
- c)  De acordo
- d)  Totalmente de acordo

**12. Quais dos seguintes elementos abordados durante o curso, você usaria durante sua prática docente, adaptando-os ao seu contexto? Marque mais de uma se for o caso.**

- a)  Roteiros
- b)  Programação com Python
- c)  Tarefas Investigativas
- d)  Programação com outra linguagem
- e)  Nenhum

**13. A tecnologia utilizada no curso, na perspectiva da construção do conhecimento, pode favorecer a aprendizagem e engajar os estudantes.**

- a)  Discordo totalmente
- b)  Discordo
- c)  De acordo
- d)  Totalmente de acordo

**14. Na associação de matemática e programação, especificamente no contexto de equações algébricas, funções do segundo grau e seus gráficos, você sentiu mais dificuldades com a:**

- a)  Matemática
- b)  Programação
- c)  Matemática e Programação
- d)  Não tive dificuldade.

**15. Na associação de matemática e programação, especificamente no contexto de elipses, você sentiu mais dificuldades com a:**

- a)  Matemática
- b)  Programação
- c)  Matemática e Programação
- d)  Não tive dificuldades.

**16. Você frequentou os horários de atendimento ao estudante?**

- a)  Sim
- b)  Não. Por quê? Justifique:

**17. Caso a resposta ao item anterior tenha sido sim, responda esta pergunta, caso contrário passe para a próxima. Os horários disponibilizados para atendimento ao estudante auxiliaram na sua aprendizagem durante o curso?**

- a)  Auxiliaram muito
- b)  Auxiliaram pouco
- c)  Não auxiliaram

**18. Você se identifica com o curso de licenciatura em Matemática?**

- a)  Sim, já atuo ou pretendo atuar na área
- b)  Sim, mas não pretendo atuar
- c)  Não me identifico com o curso.

**19. Deixe seus comentários e contribuições.**

Somos gratos pelas contribuições!