

O SCRATCH PODE MELHORAR A APRENDIZAGEM DA LÓGICA DE PROGRAMAÇÃO?

João Padilha Moreira¹

RESUMO

Este artigo é um recorte da dissertação de mestrado onde se buscou compreender como o pensamento computacional aliado à ferramenta *Scratch* pode auxiliar nos processos de ensino e de aprendizagem de conceitos envolvendo o estudo de algoritmos e programação de computadores, no contexto da aplicação de TDICs (Tecnologias Digitais da Informação e da Comunicação). A pesquisa se fundamentou na afirmação de que a ferramenta *Scratch* permitiria que os processos de ensino e de aprendizagem de algoritmos e programação seriam potencializados devido ao melhor desempenho por parte dos alunos, na construção dos algoritmos. quanto o pensamento computacional combinado com a ferramenta *Scratch* pode auxiliar para a superação de defasagem de conhecimentos apresentados pelos alunos.

Palavras-chave: Algoritmos e programação de computadores, pensamento computacional, desenvolvimento do raciocínio lógico-matemático, Tecnologias Digitais da Informação e da Comunicação, tecnologia computacional *Scratch*.



Alcides Maya
FACULDADE E ESCOLA TÉCNICA

1. INTRODUÇÃO

Ao se pensar em Educação e nos processos de ensino e de aprendizagem, percebe-se que os movimentos em busca de metodologias diferenciadas, que possam estimular esses processos, têm ocorrido em diferentes áreas do conhecimento e em diferentes níveis de ensino. Nesse cenário, observa-se que a área tecnológica evoluiu muito e apresenta diferentes recursos que podem contribuir para que os processos de ensino e de aprendizagem sejam mais efetivos.

Segundo Miquelin (2009), a TI (Tecnologia da Informação) faz parte do cotidiano da maioria das pessoas. A procura por novas soluções envolve o profissional que desenvolve sistemas, ou seja, o programador ou desenvolvedor de *software*, entre outros que atuam na área de TI. Esses profissionais, quando atuam na área Educacional, podem auxiliar no aprimoramento dos processos de ensino e de aprendizagem, dentro e fora do contexto educacional. A TI agiliza a comunicação, facilita a vida das pessoas, desde a comunicação, serviços oferecidos, como: aplicativos, sistemas *web*, sistemas para empresas, entre outros. Nesse contexto, considerando o estudo de algoritmos e programação de computadores, faz-se necessários profissionais da área de TI, que desenvolvam soluções para atender ao mercado e, também, a área Educacional, otimizando os recursos que utilizem Tecnologias Digitais da Informação e da Comunicação.

De acordo com De Souza (2018), os sinais de um mundo globalizado, e em constante crescimento, demonstram que as transformações proporcionadas pelas TDICs têm demandado respostas imediatas nos sistemas de ensino formal. Nesse sentido, ressalta-se que os cursos voltados para a formação na área das TDICs, devem estar orientados para a formação de profissionais competentes, que compreendam e empreguem recursos para melhorar as formas de programação de computadores, para que os mesmos sejam capazes de analisar e desenvolver sistemas com um padrão de qualidade que atenda às necessidades de cada organização em particular.

A partir desse contexto, bem como da inserção do autor desta Dissertação em uma Instituição de Ensino Superior, atuando como coordenador do Curso Superior

de ADS (Análise e Desenvolvimento de Sistemas), além de ser um observador nas aulas da disciplina de Algoritmos e Programação, do referido curso, chegou-se à definição do problema de pesquisa: “Como a ferramenta *Scratch* pode auxiliar nos processos de ensino e de aprendizagem de algoritmos e programação?”

Baseando-se na aplicação da ferramenta *Scratch*, no referido contexto, pode ser potencializadora dos processos de ensino e de aprendizagem de algoritmos e programação de computadores. Utilizou-se a metodologia de pesquisa qualitativa, na qual o significado e intenção são considerados inerentes aos atos, relações e estruturas sociais (BAPTISTA, 2017). Nesse contexto, aplicou-se um instrumento de pesquisa, junto aos alunos ingressantes do curso mencionado anteriormente, em dois momentos distintos: pré-teste e pós-teste. A análise dos dados coletados possibilitou a compreensão de lacunas apresentadas pelos acadêmicos, em relação a aspectos que envolvem dificuldades no aprendizado de conceitos matemáticos. Essa análise também permitiu comprovar que o uso de tecnologia computacional, como a ferramenta *Scratch*, pode auxiliar na superação das mesmas.

A metodologia qualitativa, aplicada nesta Dissertação, é uma pesquisa indutiva, na qual o pesquisador desenvolve conceitos, ideias e fatos a partir de padrões encontrados. Geralmente, os métodos qualitativos são menos estruturados e mais flexíveis ao andamento da pesquisa e às ações do pesquisador (BAUER; GASKELL, 2010). Partiu-se de pressupostos de estudos anteriores que utilizaram esse tipo de abordagem, o que em princípio se caracteriza como uma revisão bibliográfica, que tem como finalidade coletar informações prévias levantando dados.

A pesquisa envolveu um estudo de caso, opção que decorre do interesse pela compreensão daquilo que é único e particular do sujeito pesquisado. Lüdke e André (1986, p.17), afirmam que “quando queremos estudar algo singular, que tenha valor em si mesmo, devemos escolher o estudo de caso”. O interesse pelo assunto abordado nesta pesquisa se justifica pela inserção do pesquisador no Curso de ADS, como coordenador de Curso e, também, como observador na disciplina de algoritmos e programação, que é uma disciplina do primeiro semestre do curso de ADS, e tem uma carga horária de 160 horas, sendo trabalhada em dois dias da semana: nas segundas e quartas-feiras. Adotou-se a ferramenta *Scratch* para apoiar os processos de ensino e de aprendizagem nesta disciplina, como auxílio na

superação das dificuldades dos acadêmicos na compreensão de conceitos matemáticos e de abstração da construção dos pseudocódigos.

Identificou-se, por intermédio de observações, apontadas pelo professor titular da disciplina e pelo acompanhamento estatístico de alunos que se matriculam na mesma desde o ano de 2014 até o ano de 2017, que um grande número de alunos trancavam, cancelavam e reprovavam nessa matéria, gerando uma grande evasão no curso de ADS. Esses dados podem ser melhor evidenciados na tabela da Figura 1, considerando o período de 2014 a 2017, apresentando índices que oscilam entre 51% a 80%, de alunos que trancaram, cancelaram ou reprovaram na disciplina de algoritmos de programação no curso de ADS.

Valaski e Paraiso (2012), apresentam que:

os índices de repetência e desistência dos alunos são altos nos primeiros períodos nos cursos de graduação na área de informática. Um dos principais motivos é a dificuldade que os alunos têm em disciplinas relacionadas à programação. Muitos alunos têm dificuldade em abstrair os conceitos básicos de programação. Por este motivo diversos *softwares* educacionais têm sido desenvolvidos para auxiliar nesta aprendizagem [...] (VALASKI; PARAISO, 2012, p. 9).

Essas observações também são evidenciadas por Gomes (2010). Esse autor coloca que se percebe, nas disciplinas com foco em programação de computadores, um índice elevado de reprovação, uma vez que muitos alunos que ingressam nos cursos de graduação ligados às áreas de computação têm, nesses cursos, o contato inicial com a programação nas referidas matérias. A maioria dos alunos apresenta dificuldades na compreensão de conceitos matemáticos. Além disso, por apresentar uma linguagem com uma sintaxe bem específica, as dificuldades ocasionam um grau elevado de evasão na disciplina. Pode-se fazer uma conexão entre essas dificuldades matemáticas a partir de seus reflexos na resolução de problemas propostos pelos professores, visto que em algumas atividades os alunos não conseguiram resolvê-las, pois não abstraíram a informação matemática para transformá-la na evidência lógica para a solução dos mesmos.

Justo (2019) destaca que

as impressões e dificuldades que os alunos [...] tiveram durante a sua vida escolar em relação à matemática, essas experiências marcaram de tal forma a vida acadêmica de muitos deles que estes desenvolveram um sentimento de aversão, de temor à matemática. Poucos são aqueles que relatam ter uma relação amigável com esse componente curricular (JUSTO, 2019, p.13).

Ao identificar as fragilidades dos acadêmicos na referida disciplina, o NDE² do curso buscou estratégias para auxiliar na superação das lacunas apresentadas. Entre essas estratégias destacam-se: a oferta de oficinas de nivelamento de matemática e a oferta de monitoria, para as quais foram selecionados dois acadêmicos que haviam sido aprovados, na disciplina, pois

a matemática é componente importante na construção da cidadania, na medida em que a sociedade se utiliza, cada vez mais, de conhecimentos científicos e recursos tecnológicos, dos quais os cidadãos devem se apropriar [...], recursos didáticos como jogos, livros, vídeos, calculadoras, computadores e outros materiais têm um papel importante nos processos de ensino e aprendizagem. Contudo, eles precisam estar integrados a situações que levem ao exercício da análise e da reflexão, em última instância, a base da atividade matemática (PCN, 1997, p. 19).

A maioria dos alunos matriculados na disciplina mencionada, apresenta uma série de lacunas na compreensão de conceitos matemáticos, dificultando a compreensão de matérias da área da computação e, conseqüentemente, do curso de ADS. Isso pode ser observado por meio dos resultados da aplicação de instrumentos elaborados nessa pesquisa (aplicados em dois momentos, pré-teste e pós-teste), que apontaram indícios de aspectos a serem (re) trabalhados com os alunos em sala de aula.

Acredita-se, nesse contexto, que a utilização da ferramenta *Scratch* possa contribuir no auxílio para superação das dificuldades na compreensão de conceitos

2A Comissão Nacional de Avaliação da Educação Superior (CONAES), no Parecer CONAES N.º 04, de 17 de junho de 2010, por meio do Art. 1o. criou o Núcleo Docente Estruturante (NDE), onde um curso de graduação deve constituir um grupo de docentes, com atribuições acadêmicas de acompanhamento, atuante no processo de concepção, consolidação e contínua atualização do projeto pedagógico do curso. Onde o NDE deve ser constituído por membros do corpo docente do curso, que exerçam liderança acadêmica no âmbito do mesmo, percebida na produção de conhecimentos na área, no desenvolvimento do ensino, e em outras dimensões entendidas como importantes pela instituição, e que atuem sobre o desenvolvimento do curso (BRASIL, 2010).

matemáticos dos acadêmicos ingressantes do curso de ADS, conforme mencionado anteriormente. O estudo de caso foi desenvolvido no segundo semestre de 2018. Os alunos foram selecionados por adesão voluntária, foram convidados a participar do estudo de caso a população total da disciplina, 20 acadêmicos, dos quais 11 acadêmicos assinaram a autorização para aplicação da entrevista semiestruturada, apresentada no Apêndice A. Os referidos alunos estavam matriculados na disciplina de algoritmos e programação, desenvolvendo os algoritmos no caderno e, também, no computador, realizando o chamado teste de bancada³ e, posteriormente, usando o *software VisuAlg*⁴, para o desenvolvimento de algoritmos em pseudocódigo⁵.

Lima Junior, Vieira e Vieira (2015) afirmam que na disciplina de algoritmos e programação os acadêmicos têm o primeiro contato com a área de programação e a base para qualquer curso que envolva a área de programação. Com base na afirmação dos autores, é também a matéria em que a maioria dos acadêmicos sente uma barreira de difícil transposição, pois nela se estudam aspectos envolvendo resolução de problemas, interpretação de textos e atividades, que evidenciam as dificuldades na compreensão de conceitos matemáticos.

A pesquisa se caracterizou por uma abordagem qualitativa, envolvendo a análise de dados, apresentados na Figura 1, e informações coletadas durante a aplicação da pesquisa. As tecnologias que viabilizaram a execução dos procedimentos e dos objetivos almejados foram a análise de documentos e o depoimento dos participantes, sendo alunos que ingressaram no primeiro semestre

3“Leite (2006), apresenta os passos para a implementação de um teste de bancada: uma das ferramentas muito empregadas para verificar se um pseudocódigo (ou algoritmo de um modo geral) está sendo executado corretamente é o chamado Teste de Bancada. Esse teste retrata fielmente o que pseudocódigo está executado, passo-a-passo, mostrando ao programador as etapas previstas no algoritmo. O procedimento para executar um teste de bancada é bem simples: 1. Elaborar uma tabela onde cada coluna se refere a cada variável envolvida e o resultado de uma operação em particular (ou observação pertinente); 2. Executar os passos previstos no algoritmo; 3. Verificar se os resultados obtidos são coerentes com os previstos; 4. Encerrar o teste após um número razoável de resultados corretos obtidos” (LEITE, 2006, p. 60).

4“O Visual Algoritmo (VisuAlg) é um programa que edita, interpreta e executa algoritmos com uma linguagem próxima do português estruturado como um programa normal de computador. (PASCHOALINI, 2015). O ambiente VisuAlg foi construído para executar programas escritos em algoritmo ou português estruturado para fins didáticos. É um ambiente onde o código é interpretado comando a comando, provendo entradas e saídas, e informando os erros quando existirem” (PIVA JUNIOR; ENGELBRECHT; NAKAMITI, 2018 p.5).

5“Pseudocódigo é uma etapa intermediária entre a descrição na língua portuguesa dos passos de um procedimento e uma especificação deste procedimento usando uma linguagem de programação real” (ROSEN, 2010, p. 854)

do curso de ADS, regularmente matriculados na disciplina de algoritmos e programação, da Faculdade FAQI⁶, uma Instituição de Ensino Superior privada, no campus localizado no município de Gravataí - RS, cidade da região metropolitana de Porto Alegre – RS. Constatou-se, conforme se verificar na tabela da Figura 1, um índice entre 51% e 80% de evasão na referida disciplina. Os acadêmicos compreendidos nesse índice da pesquisa mostravam dificuldade em acompanhar a disciplina de algoritmos e programação do curso de ADS, demonstrando dificuldades na compreensão de conceitos matemáticos e, também, no desenvolvimento de atividades que lhes eram propostas e exigidas para o nível de programação em computação. Nesse contexto, desenvolveu-se a pesquisa, aqui, apresentada, que envolveu uma observação com intervenção pedagógica utilizando a ferramenta *Scratch*, nos processos de ensino e de aprendizagem envolvendo algoritmos e programação de computadores. Verificou-se, a partir das ponderações dos acadêmicos, por meio da análise das pesquisas aplicadas pela CPA⁷, a necessidade de ouvi-los e descobrir o motivo de suas dificuldades, atividade inerente ao coordenador de curso, pois os acadêmicos levavam à coordenação as dificuldades que enfrentavam no início de sua formação e, especificamente, na disciplina mencionada.

Silva (2016) considera que as atividades relacionadas ao estudo da lógica de programação, por meio da ferramenta *Scratch*, como forma de potencializar os processos de ensino e de aprendizagem de algoritmos e programação, permitem a resolução de problemas e auxiliam nas dificuldades dos acadêmicos na compreensão de conceitos matemáticos. Para Moran (2000), a utilização TDICs possibilita a evolução de metodologias alternativas à tradicional, voltada ao processo de ensino meramente expositivo, o que, por vezes, auxilia o aprimoramento dos processos de ensino e de aprendizagem. Nesse contexto, a função do professor deve ser a de mediador do processo e não apenas transmissor de informações. Assim o computador pode ser um auxiliar para o professor nos processos de ensino e de aprendizagem, possibilitando transformar o meio para a aprendizagem e questionar as formas tradicionais de ensinar e aprender.

⁶Faculdade e Escolas técnicas – FAQI – www.qi.edu.br

⁷Brasil (2018), a sigla CPA é a Comissão Própria de Avaliação, instituída pelo SINAES – Sistema Nacional de Avaliação da Educação Superior, e é responsável pela implantação e pelo desenvolvimento de processos de avaliação institucional.

1.1. Justificativa da pesquisa

A partir de constatações abordadas e discutidas com os 05 docentes integrantes do Núcleo Docente Estruturante – NDE do Curso de ADS da FAQI - Gravataí, sobre as dificuldades que os alunos enfrentavam, especificamente na disciplina de algoritmos e programação, foi apresentado o levantamento realizado no período de 2014 a 2017, o qual deu origem à tabela apresentada na Figura 1, que evidencia, os dados da tabela mostram a evasão e as desistências.

Os problemas que os alunos enfrentavam são as dificuldades na compreensão de conceitos matemáticos. Então, este pesquisador, enquanto membro do NDE e coordenador do curso de ADS, decidiu iniciar a aplicação de uma pesquisa na referida disciplina, na condição de observador, pois a mesma estava sendo ministrada por outro professor. A proposta foi a de acompanhar os processos de ensino e de aprendizagem desenvolvidos com o uso de teste de bancada e com a construção de algoritmos em português estruturado, usando o software *VisuAlg*. Propôs-se, ao docente responsável pela disciplina, a realização de uma intervenção pedagógica, por meio da aplicação da tecnologia computacional *Scratch*, buscando investigar “Como a tecnologia computacional *Scratch* poderia auxiliar nos processos de ensino e de aprendizagem de algoritmos e programação?”. No contexto da pesquisa, constitui-se como objetivo geral o de compreender como a ferramenta *Scratch* aliada ao pensamento computacional poderia contribuir na aplicação de conceitos abordados na disciplina de algoritmos e programação, no curso de ADS. A partir disso, estabeleceram-se os seguintes objetivos específicos:

- Compreender como os alunos resolviam os algoritmos propostos pelo professor, na disciplina de algoritmos e programação, integrante do currículo do referido curso;

- Implementar (planejar, desenvolver e avaliar) cada verbo é um objetivo específico diferente uma proposta de uso da tecnologia computacional *Scratch* na disciplina de algoritmos e programação.

Relacionou-se a utilização das TDICs, inseridas nos ambientes educacionais, para vencer obstáculos e dificuldades na compreensão de conceitos matemáticos

em tarefas que usavam algoritmos, com o intuito de proporcionar uma solução ao problema abordado na investigação. Nessa direção, estabeleceu-se dois caminhos: o primeiro envolvia dificuldades na compreensão de conceitos matemáticos, para executar tarefas que abordavam o estudo de algoritmos e programação. O segundo, referia-se à utilização da ferramenta *Scratch*, que serviu como apoio aos processos de ensino e de aprendizagem para o observador e para os alunos matriculados no curso de ADS, onde para Yin (1994)

De facto, é importante ao rever qualquer documento compreender que foi escrito para alguns propósitos específicos e algumas audiências específicas outras que não aquelas do estudo do caso a ser feito. Neste sentido, o investigador do estudo do caso é um observador indirecto, e as provas do documentário reflectem uma comunicação entre outras partes tentando alcançar alguns outros objectivos. Ao tentar constantemente identificar estas condições, você é menos passível de ser mal direccionado por provas documentárias e mais passível de ser correctamente crítico a interpretar os conteúdos de tais provas (YIN, 1994, p.102).

A evolução das TDICs ganhou força a partir da década de 90 e continua em uma crescente evolução, pois as mesmas auxiliam no desenvolvimento de habilidades. Nesta pesquisa, o foco do uso das TDICs está no auxílio à superação das dificuldades dos acadêmicos na compreensão de conceitos matemáticos para resolver problemas na forma de algoritmos. É importante salientar que, as dificuldades enfrentadas pelos alunos que ingressam em um curso superior, especificamente no curso de ADS, estão relacionadas a essas dificuldades, pois a Informática é uma Ciência Exata, baseada em conceitos matemáticos que, por sua vez, são a base para a resolução dos algoritmos.

Para aprimorar os processos de ensino e de aprendizagem de algoritmos e de programação, torna-se necessário fortalecer o pensamento computacional com mecanismos que possibilitem ao educando entender a importância do conhecimento para solucionar os desafios propostos. Copi (1978) considera que a lógica é um estudo “dos métodos e princípios usados para distinguir o raciocínio correto e incorreto”. Isso significa que o pensamento se torna mais crítico em termos de ideias, opiniões e argumentos. Para Baião (2016) as tecnologias que ambos os documentos tratam de forma importantíssima:

“a necessidade de mudança na inclusão tecnológica no ambiente escolar para assim poder aproximar o conteúdo, o máximo possível da realidade do aluno inclusive para poder desenvolver competências que são necessárias para o estudante no século XXI” (BAIÃO, 2016, p.94).

Para os alunos resolverem os problemas propostos na disciplina de algoritmos e programação, é fundamental que entendam as proposições feitas pelo professor, para dar sequência às soluções de vários problemas e, assim, assimilar os conteúdos, entendendo qual é a finalidade do conhecimento para determinada situação. Com o auxílio das TDICs, é fundamental que as dificuldades na compreensão de conceitos matemáticos sejam acompanhadas pelos professores, que devem propor atividades e aplicar diferentes ferramentas e recursos, como a utilização de jogos educacionais digitais, que contribuam para a aprendizagem e motivem o interesse pelo conhecimento.

2. METODOLOGIA DA PESQUISA

Entende-se que o percurso metodológico adotado contempla, de fato, aspectos mais flexíveis, tanto para o pesquisador quanto para os alunos envolvidos durante a realização da pesquisa, assim como em relação aos conteúdos que foram ofertados durante a realização da disciplina de algoritmos e programação, realizada no 2º semestre de 2018.

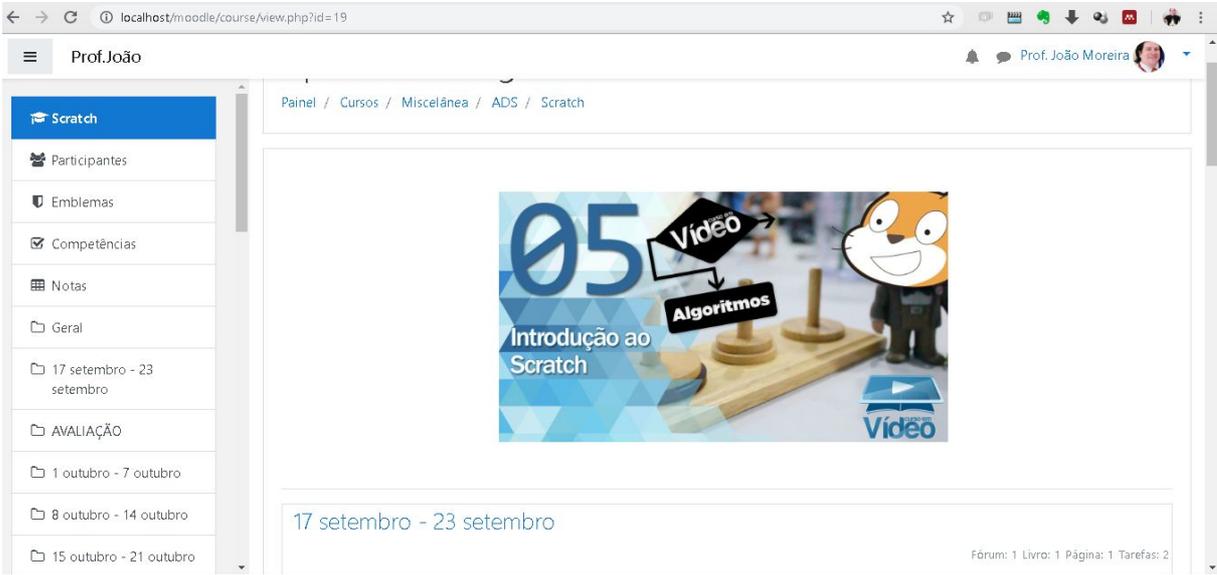
Segundo Lüdke e André (1986), quando se quer estudar algo singular, que tenha valor em si mesmo, deve-se escolher a metodologia do estudo de caso. Sendo assim, buscou-se compreender como a ferramenta *Scratch* poderia contribuir para a construção de conhecimentos abordados na disciplina de algoritmos e programação do Curso de ADS da FAQI caracterizando, assim, um estudo de caso, utilizando a técnica da abordagem de forma qualitativa e, respondendo ao problema da pesquisa, que é “Como a ferramenta *Scratch* pode auxiliar nos processos de ensino e de aprendizagem de algoritmos e programação? ”.

2.1. Local e participantes da pesquisa

A pesquisa foi aplicada nas Faculdades e Escolas Técnicas - FAQI, no Curso Superior de Tecnologia em ADS, no município de Gravataí, RS, em uma turma, regularmente matriculada na disciplina de Algoritmos e Programação. Da seguinte forma: aulas presenciais, utilizando-se a modalidade de EaD (Educação a Distância), utilizando-se para isso o AVA (Ambiente Virtual de Aprendizagem), a amostra compreende 11 alunos, com faixa etária entre 20 e 40 anos, sendo que os mesmos concordaram em participar para a utilização e divulgação dos dados (Apêndice A). A escolha, recaiu sobre o AVA MOODLE, para as atividades na modalidade de EaD (SILVA, 2016), visto como uma vantagem. O AVA MOODLE, é conhecido como um dos portais mais populares, pois possui módulos customizáveis interativos, permitindo aos usuários uma interação em que se atinge inúmeras possibilidades de atividades síncronas, ou seja, em tempo real, ou assíncronas, que podem ser feitas no tempo do usuário e depois sincronizadas.

Os recursos do AVA MOODLE, apresentam algumas ferramentas muito conhecidas como questionários, tarefas, chats, fóruns, entre outros módulos. Para Beluce (2012), a escolha das metodologias que são empregadas nos módulos interativos do ambiente virtual e que dinamizam nos processos de ensino e aprendizagem, definem os níveis de interatividade entre estudantes, tutores e professores. A partir dessa abordagem sobre a importância do AVA MOODLE, utilizou-se esse ambiente virtual, para aplicar as atividades não presenciais, bem como receber as atividades, executadas em casa.

Figura 2 - AVA MOODLE: Aprendendo Algoritmos com a ferramenta Scratch



The image shows a screenshot of a Moodle course page. The browser address bar indicates the URL is localhost/moodle/course/view.php?id=19. The page title is 'Prof. João'. The course navigation menu on the left includes 'Scratch', 'Participantes', 'Emblemas', 'Competências', 'Notas', 'Geral', and several date-based sections: '17 setembro - 23 setembro', '1 outubro - 7 outubro', '8 outubro - 14 outubro', and '15 outubro - 21 outubro'. The main content area displays a video player with a thumbnail for '05 Introdução ao Scratch'. The thumbnail features the Scratch character, a play button, and the text 'Algoritmos' and 'Vídeo'. Below the video player, there is a section for '17 setembro - 23 setembro' and a footer indicating 'Fórum: 1 Livro: 1 Página: 1 Tarefas: 2'.

Fonte: A pesquisa.

A Figura 3, apresenta as áreas de navegação do ambiente aprendendo algoritmos com a ferramenta *Scratch*, demonstrando como navegar, pelas áreas de navegação, de cabeçalho, de visualização do material, do sumário.

Figura 3 - Telas identificadas no AVA MOODLE.

The image shows a screenshot of a Moodle course page titled "Aprendendo Algoritmos com Scratch". The page is divided into several sections, with three specific areas highlighted by red boxes and labeled with arrows:

- Area de Cabeçalho:** Located at the top of the page, containing the course title "Aprendendo Algoritmos com Scratch" and a breadcrumb trail: "Página / Cursos / Miscelânea / ADS / Scratch / 17 setembro - 23 setembro / Introdução ao Scratch".
- Area de navegação:** Located on the left side of the page, containing a sidebar menu with options like "Scratch", "Participantes", "Emblemas", "Competências", "Notas", "Geral", and a calendar view for the current week (17 setembro - 23 setembro).
- Area de Visualização do material:** Located in the center of the page, containing the main content area titled "Introdução ao Scratch" with the sub-heading "1. Introdução" and a brief introductory text.
- Area do Sumário:** Located on the right side of the page, containing a "Sumário" section with a list of course items, each with a plus icon for expansion.

Fonte: A pesquisa.

A Figura 4, mostra a divisão do ambiente em semanas para acesso as atividades propostas, assim como local para entrega das atividades.

Figura 4 - Tela com divisão semanal.



Fonte: A pesquisa.

As Figuras 1,2, e 3 mostram aos alunos um ambiente de apoio para intervenção pedagógica no formato on-line, para resolução de atividades com o uso de algoritmos usando a ferramenta *Scratch*.

2.2. Etapas da pesquisa

Para dar conta do objetivo geral desta pesquisa, que é o de compreender como a ferramenta *Scratch* pode contribuir na aplicação de conceitos abordados na disciplina de algoritmos e programação, no curso de ADS, foi necessário verificar os conhecimentos dos participantes em relação ao entendimento e desenvolvimento de algoritmos. Dessa forma, apresentam-se as etapas de acordo com as fases.

2.2.1. Fase 1 – Pré-teste

- Apresentação da metodologia de trabalho;
- Preenchimento dos termos de consentimentos;
- Aplicação do pré-teste, utilizando caderno, lápis, borracha e editor de texto, conforme apresentado no Apêndice D.

2.2.2 Fase 2 – Atividades práticas com a ferramenta *Scratch*

- Desenvolvimento e acompanhamento das atividades;

- Análise da construção e do desenvolvimento de conceitos necessários para embasar os processos de ensino e de aprendizagem relacionados à disciplina de algoritmos e programação;
- Apresentação da metodologia de trabalho;
- Cadastro no *site* da ferramenta *Scratch* online⁸;
- Realização de uma aula expositivo-dialogada, sobre o funcionamento da ferramenta *Scratch*;
- Dinâmica interativa, com o a ferramenta *Scratch* para apropriar os recursos da mesma.

2.2.3 Fase 3 – Pós-teste

- Aplicação do pós-teste, após a utilização a ferramenta *Scratch*, como mostra o Apêndice D.

2.3. Instrumentos de coleta de dados

Os instrumentos de coleta de dados se apresentam a seguir:

- a) Ficha de acompanhamento para descrição das atividades e observações do pesquisador acerca dos alunos, apresentadas no Apêndice B;
- b) Impressões dos acadêmicos, coletadas ao longo do estudo, além de comentários espontâneos sobre os encontros, conforme mostra o Apêndice C;
- c) Pré-teste e pós-teste (Apêndice D); e
- d) Entrevista semiestruturada (Apêndice E).

3. REVISÃO DE TRABALHOS RELACIONADOS

Para dar conta desse estudo, buscou-se uma revisão dos trabalhos na área, iniciando a busca no Portal de Periódicos e no Banco Teses da CAPES⁹, no *Google Acadêmico*¹⁰ e na plataforma *Scopus*¹¹, utilizando as palavras-chaves: algoritmos e

⁸ Disponível em <https://scratch.mit.edu/projects/31876/>

⁹ <https://catalogodeteses.capes.gov.br/catalogo-teses/#/>

¹⁰ <https://scholar.google.com.br/scholar?q=encontros+bibli&hl=pt-BR&lr=>

¹¹ https://www.periodicos.capes.gov.br/?option=com_pcollection&mn=70&smn=79&cid=63

programação, raciocínio lógico-matemático, TDICs e ferramenta *Scratch*, no período compreendido entre os anos de 2013 a 2018. A busca apresentou 4090 resultados. Como critérios de seleção dos documentos, os mesmos foram separados em teses de doutorado e dissertações de mestrado, que abordassem a utilização a ferramenta *Scratch*, como auxiliar nos processos de ensino e de aprendizagem de programação, em diferentes níveis (Ensino Médio, Técnico e Ensino Superior).

Os critérios de inclusão e exclusão dos trabalhos para a realização da análise foram os seguintes:

1. Inclusão: apenas foram incluídos trabalhos que resultavam das pesquisas a partir das palavras-chave definidas previamente, e que correspondiam com a investigação e intervenção dentro do contexto educativo ligado ao tema estudado;

2. Exclusão: foram excluídos os trabalhos encontrados mais de uma vez, por exemplo, teses de doutorado e dissertações de mestrado, ligados à pesquisa e que foram referenciadas em outros documentos e trabalhos que tenham haviam constado em mais de uma pesquisa. Os trabalhos encontrados com a palavra-chave (ou com as expressões) “algoritmos e programação, raciocínio lógico-matemático, TDICs e ferramenta *Scratch*” foram referenciados apenas uma única vez. Essa análise teve, como objetivo principal, o de identificar a estratégia de aplicação da ferramenta *Scratch*, relacionando-a aos tipos de avaliação predominantes, os ciclos de ensino nos quais são realizadas essas investigações, intervenções e instrumentos de avaliação do rendimento acadêmico.

Os resultados para responder às questões de investigação formuladas foram analisados, documento a documento, sendo um total de 160 documentos como resultado das consultas realizadas aos repositórios científicos nacionais com base nas palavras-chave aplicadas.

Alunos que finalizam o ensino médio, e que ingressam no curso de ADS, apresentam dificuldades em disciplinas consideradas de conhecimento técnico, como lógica de programação, organização de computadores e sistemas operacionais.

Lummertz (2016) considera, em sua pesquisa, a abordagem qualitativa, buscando responder aos questionamentos sobre as dificuldades que os alunos

enfrentam no uso da ferramenta *Scratch*, na constituição de aspectos relacionados à Literacia Digital¹² e ao pensamento computacional, por meio da construção de jogos digitais com alunos ingressantes em um curso superior, especificamente no curso de ADS. Noronha (2016) aborda a aprendizagem, a formação dos alunos e as dificuldades que esses enfrentam no desenvolvimento do raciocínio matemático e no uso da ferramenta *Scratch*, propondo que se discuta sobre algoritmos, para a construção de conhecimentos. Sápiras (2017) usou, como proposta, uma oficina com jogos digitais com alunos de 7^a e 8^a série e fez experiências com uso da ferramenta *Scratch*, empregando métodos de análise fundamentados na experiência com os alunos, para medir os processos de ensino e de aprendizagem de alunos com diferentes níveis de conhecimentos e as dificuldades enfrentadas durante sua aplicação.

Para Rodrigues (2001), Gama (1987), Lévy (1993) e Leite (2010), a tecnologia transforma as organizações e a sociedades garantindo novas possibilidades de bem-estar e conforto, considerando que no passado as dificuldades eram maiores por desconhecimento ou por falta de uso de tecnologias, e que os avanços possibilitaram a descoberta e a utilização de algumas tecnologias que auxiliam a aprendizagem, a ampliação da capacidade de raciocinar e a superar as dificuldades encontradas a partir de sua capacidade de inventar. Os resultados obtidos mostram que o uso da ferramenta *Scratch*, apresenta resultados bastante favoráveis ao aprendizado dos alunos. Silva (2016) considera que o estudo da lógica de programação, com o apoio da ferramenta *Scratch*, é uma forma de potencializar o ensino de matemática para a solução de problemas. Para mediar o trabalho envolvendo a resolução de problemas ligados às dificuldades na compreensão de conceitos matemáticos, optou-se por utilizar uma ferramenta que possibilitasse o contato dos alunos com a programação de computadores, aproximando-os, também, da lógica inerente à programação que está relacionada à estruturação do pensamento por meio da criação de algoritmos.

De acordo com os estudos de Pinto (2010, p.31), “a programação pode ser vista como um meio ou como um fim em si mesma, considerando a programação como um meio, essa pode servir para resolver problemas significativos que se

12A Literacia Digital é considerada por Jenkins et al. (2009) como a capacidade de o indivíduo compreender e usar a informação contida em vários materiais digitais de modo a desenvolver seus próprios conhecimentos.

coloquem aos alunos”. Inácio (2016) trabalha o ensino, usando algoritmos e lógica de programação, mediados pela ferramenta *Scratch*, comparando a metodologia de “*ensinagem*” e aprendizagem, utilizada pelos professores, passando por todas as fases para desenvolver a cognição direcionada a verificar as dificuldades na compreensão de conceitos matemáticos. O autor considera importante a aplicação de uma nova metodologia de aprendizagem usando como orientação o uso da ferramenta *Scratch*.

Baião (2016) trabalha com a proposta de incorporar as TDICs na sala de aula, usando como objeto de estudo e análise o ensino de Física no Ensino Médio. Ressalta o uso da ferramenta *Scratch* combinada com a plataforma Arduino, como método experimental em sala de aula, baseando-se nas orientações do método construcionista, destacando a importância que essa ferramenta tem no aspecto lúdico da aprendizagem na escola. Um outro aspecto que se deve destacar segundo Baião (2016):

a necessidade de mudança de método e de inclusão de tecnologia em sala de aula para assim poder aproximar o conteúdo o máximo possível da realidade do aluno inclusive para poder desenvolver competências que são necessárias para o aluno do século XXI” (BAIÃO, 2016, p. 94).

Silva (2016) considera o uso da lógica de programação para a educação matemática no Ensino Médio: experiências com a ferramenta *Scratch* e as atividades relacionadas ao estudo de lógica de programação, facilitada com o apoio da ferramenta *Scratch*, como forma de potencializar o ensino de matemática no que diz respeito à resolução de problemas. Um aspecto relevante do trabalho da autora se relaciona à construção de um diálogo entre o uso de tecnologias na educação. Para Zoppo (2017), as aplicações e utilizações das tecnologias em sala de aula estimulam a curiosidade dos alunos em problemas derivados da Matemática. A autora considera que “a escola vem sendo impulsionada a repensar sua prática pedagógica, bem como selecionar dentre os recursos disponíveis quais podem contribuir nesse processo” (ZOPPO, 2017, p.8). O embasamento para essa reflexão está relacionado ao uso das TDICs na sala de aula, como forma de facilitar os processos de ensino e de aprendizagem.

Para Ribeiro (2015), os assuntos que são pertinentes e motivam a uma reflexão acadêmica e profissional, argumentam sobre a problemática que os profissionais da área de informática enfrentam, diariamente, com o crescimento da capacidade humana, que resulta da utilização generalizada de sistemas de computação. O autor também ressalta que, nos processos de ensino e de aprendizagem, ao valorizar o conhecimento de programação de computadores, observando-se, em vários países, iniciativas para sua introdução desde o ensino formal fundamental até ao Ensino Superior. O autor se refere ao ensino superior de ciências exatas, pois apresenta uma alta procura por pessoas graduadas que dominem programação. O estudo também apresenta evidências que comprovam que um grande número de acadêmicos abandona as disciplinas de programação. Ribeiro (2015) propõe a utilização do AVA Moodle, na formação dos alunos em computação e compara o nível de aprendizagem dos alunos em situações que envolvam a linguagem de programação Java, relacionando o grau de interação do aluno. Esta dissertação se fundamenta no trabalho de diversos autores, como: Delors (2006), Rodrigues (2001), Gama (1987), Lévy (1993) e Leite (2010), os quais evidenciam o uso das TDICs, nos processos de ensino e de aprendizagem.

Delors (2006) leva em conta que:

a evolução é o “único paradigma da mudança”, essas mudanças estão presentes no cotidiano social, no estilo, na forma e se inter-relaciona em um formato direto e indireto na vida educacional e profissional na capacidade de aprender e raciocinar usando o slogan “aprender a aprender”, muito debatido pelas instituições e difundido aos jovens na atualidade, e que faz parte do perfil dos alunos que ingressam nas faculdades (DELORS, 2006, p. 50).

Sabe-se que os estudos apresentados não se esgotaram, são reflexões importantes e que contribuem para o desenvolvimento científico e para futuros estudos. Para França et al. (2014) o século XXI tem o cunho da evolução do pensamento humano, além das fronteiras do cérebro físico. Ele se expandiu para o que se convencionou denominar pensamento computacional, ou seja, aquele em que uma máquina executa funções com maior rapidez e eficiência que o cérebro humano mediano. A presença dessa forma de expressão de raciocínio nas escolas permitiria a crianças e jovens alcançarem infinitas possibilidades criativas, lógicas e

lúdicas capazes de proporcionarem uma sociedade com níveis de inteligência indescritíveis.

A partir dessa concepção do autor, pode-se dizer que:

“A computação constitui uma área de conhecimento que permeia todas as atividades humanas, de forma que não se pode imaginar uma sociedade sem computadores e suas tecnologias. Não se pode imaginar o cidadão ignorante em Computação, enquanto ciência, já que, em qualquer atividade profissional, haverá pelo menos o uso de tecnologias da informação atrelada a um raciocínio computacional (algorítmico). Ainda, existem inúmeros problemas das áreas das Ciências Exatas, Humanas, Artes, e da realidade cotidiana que poderiam ser resolvidos com o seu auxílio” (FRANÇA *et. al*, 2014, p. 1505).

Por estar presente no cotidiano do cidadão comum, quanto daqueles que trabalham com pesquisas nas mais diversas áreas de conhecimento, a computação permeia as atividades humanas na atualidade. É impossível conceber um avanço tecnológico no qual a computação não se faça presente, pois foi a partir dela que se concretizou a maioria das descobertas que tornaram a vida das pessoas muito melhor, além dos notáveis progressos na Medicina, Astrofísica e na própria Tecnologia da Informação. Por esses motivos o pensamento computacional deve estar presente desde a educação básica até o ensino superior, pois por meio dele o conhecimento atingirá proporções maiores e melhores das vistas atualmente. Assim sendo,

“Assim, futuros sociólogos, economistas, músicos, educadores deverão interagir com profissionais da computação através de um pensamento interdisciplinar, fazendo-se necessário, naturalmente, o seu ensino e aprendizagem na educação básica, uma realidade nos países desenvolvidos. O ensino de conceitos básicos de computação – modelos de computação, algoritmos, complexidade computacional, autômatos, linguagens e arquitetura de computadores, entre outros – fornece o conhecimento necessário à formação do cidadão. (FRANÇA *et. al*, 2014, p.1505).

A amplitude do pensamento computacional é tão vasta, que pode interagir com as mais diversas áreas do conhecimento, sem perder sua finalidade e sem retirar a essência de outra área com a qual esteja interagindo. A partir de seus conceitos relacionados ao pensamento computacional, os processos de ensino e de aprendizagem adquirem dimensões melhores, pois a interdisciplinaridade na educação é uma realidade que a Computação proporciona. Isso é também um meio de melhorar as condições de vida em sociedade. Ao se observar a sociedade

moderna, percebe-se a necessidade de se desenvolver habilidades como aprender a aprender, aprender de forma colaborativa e atribuir lugar às tecnologias desenvolvidas advindas da Ciência da Computação, nesses processos, como destaca a Lei de Diretrizes e Bases (LDB), em seu Artigo 32, a qual afirma que, além desses conhecimentos, é objetivo da formação básica do cidadão “a compreensão do ambiente natural e social, do sistema político, da tecnologia, das artes e dos valores em que se fundamenta a sociedade” (SOUZA e SILVA, 1997, p. 53).

As questões tecnológicas são tão importantes para o processo de formação e educação de crianças e jovens, que os legisladores, cientes de sua importância para o desenvolvimento na maneira de ensinar e aprender a compreensão do pensamento computacional, incluíram essa dimensão na Lei de Diretrizes e Bases da Educação Nacional, projeto de lei art. 27-A à lei nº 9.394, de 20 de dezembro de 1996. A partir desse projeto as escolas, além de outros espaços educacionais, passaram a ter também o que se conhece como Laboratório de Informática, que possibilita a crianças e jovens estarem inseridos no universo da Computação. Esses espaços oferecem a oportunidade a professores e alunos estabelecerem uma relação de trocas em que irá se

revelando gradativamente a importância do pensamento computacional e sua aplicabilidade nas diversas áreas. Nesse contexto, atividades lúdicas apresentam-se como alternativa, podendo motivar a participação dos alunos e engajá-los em tarefas de aprendizagem. Para um melhor aproveitamento de materiais dispostos na literatura, como as atividades de Computação Desplugada, ficou evidente para os envolvidos a necessidade de adaptá-los para aplicação no cenário educacional brasileiro. Além disso, tendo em vista a formação em Computação, Pedagogia e Psicologia, os licenciandos em computação poderão produzir artefatos tangíveis e digitais que possibilitem o desenvolvimento do pensamento computacional nas escolas (FRANÇA *et. al*, 2014, p.1512).

Ao se unir os conhecimentos oriundos da Computação com os da Pedagogia é possível inovar a apresentação de materiais didático-pedagógicos e formas de desenvolver conteúdos que estimulem a aprendizagem e que atinjam um número maior de alunos, pois como é de senso comum cada aluno tem o seu jeito de aprender e o computador amplia essa possibilidade.

Para Blikstein (2008) dessa maneira, pode-se definir como pensamento computacional “saber usar o computador como um instrumento de aumento do poder cognitivo e operacional humano”. Para Wing (2016), o pensamento

computacional está baseado nos processos computacionais executados por máquinas ou por seres humanos. Os modelos computacionais dão condições para se desenvolver sistemas que não se conseguiria desenvolver, sem eles.

E assim se propõe uma pergunta? O que os seres humanos fazem melhor que as máquinas, e o que os computadores fazem melhor que as pessoas, todavia a pergunta principal fica, o que realmente é função dos computadores, pois, atualmente, tem-se apenas parte dessa resposta. Nesse sentido, Wing (2016) indaga que ao:

resolver um problema eficientemente, podemos questionar se uma solução aproximada é boa o suficiente e se falsos positivos ou falsos negativos são permitidos. O pensamento computacional é reformular um problema aparentemente difícil em um problema que sabemos como resolver, talvez por redução, incorporação, transformação ou simulação. Pensamento computacional é pensar recursivamente. É processamento paralelo. É interpretar código como dado e dado como código. É verificação de tipo como uma generalização da análise dimensional. É reconhecer ambas as virtudes e os perigos da utilização de aliás, ou dar a alguém ou alguma coisa mais de um nome. É reconhecer ambos o custo e o poder do endereçamento indireto e chamada de procedimento. É julgar um programa não apenas pela sua correte e eficiência, mas pela sua estética, e a interface de um sistema pela sua simplicidade e elegância. Pensamento computacional é usar abstração e decomposição ao atacar uma tarefa grande e complexa ou projetar um sistema complexo e grande. É a separação de interesses. É escolher uma representação apropriada para um problema ou modelagem dos aspectos relevantes de um problema para torná-lo tratável. É usar invariantes para descrever o comportamento de um sistema de forma sucinta e declarativa. É ter a confiança de que podemos usar, modificar e influenciar um sistema grande e complexo sem entender todos os seus detalhes com segurança, modularizar algo em antecipação de múltiplos usuários ou *prefetching* e *caching* em antecipação de um uso futuro (WING, 2016, p. 03).

Segundo Wing (2016), uma boa maneira de aproximar o que é o pensamento computacional é apresentando alguns exemplos do dia a dia, como quando seu filho vai para escola de manhã, ao fazer a mochila coloca todas as coisas que vai precisar para o dia; isso é *prefetching* e *caching*, seguindo a mesma analogia quando ele perde as luvas e se sugere que refaça o caminho de volta, passo a passo; a este processo, chama-se de *backtracking*. Outro exemplo, é quando se deve parar de alugar equipamentos e comprar os seus próprios equipamentos; isso são algoritmos online. Mais um exemplo, seria a escolha da fila do mercado; isso é modelagem de performance para sistemas com vários servidores. Outro bom exemplo, comparar a autonomia do telefone celular que continua funcionando

quando falta energia elétrica em casa, nesse caso, poder-se-ia compartilhar com o processo de independência de falha de redundância de projeto. Um processo no qual o sistema solicita “CAPTCHA”, para autenticar se é um *bot* ou um ser humano. O equivalente sem computação seria, exploração da dificuldade de resolução de problemas difíceis de inteligência artificial para enganar um sistema computacional.

Ainda fazendo alusão aos pensamentos de Wing (2016), o pensamento computacional é parte de um conjunto de habilidades que a comunidade acadêmica de uma forma geral, cientista e outras pessoas, vão precisar conhecer e utilizar, pois, a futurística computação ubíqua¹³ é realidade, assim como o pensamento computacional que amplia esse conceito será realidade amanhã. Nesse sentido, esta pesquisa busca no pensamento computacional e na experimentação de tecnologias computacionais como o *Scratch* para ofertar, além dos testes de bancadas, a utilização do *software Visualg* uma nova possibilidade de repetição e fixação dos conceitos por intermédio de uma tecnologia que busca a aplicação e a fixação do pensamento computacional em alunos da disciplina de Algoritmos de programação na Faculdade FAQI.

¹³Computação ubíqua ou computação pervasiva é um termo usado para descrever a onipresença da informática no cotidiano das pessoas.

4. REFLEXÕES SOBRE O RACIOCÍNIO LÓGICO MATEMÁTICO E O USO DA FERRAMENTA *SCRATCH*

Tendo Papert (1985), como precursor de técnicas de programação, utilizando a Linguagem LOGO, na educação no âmbito investigativo, as atividades foram realizadas, levando em conta não apenas o conteúdo que é apresentado, mas buscando a informação e se utilizando dela para a formação e construção de novos conceitos.

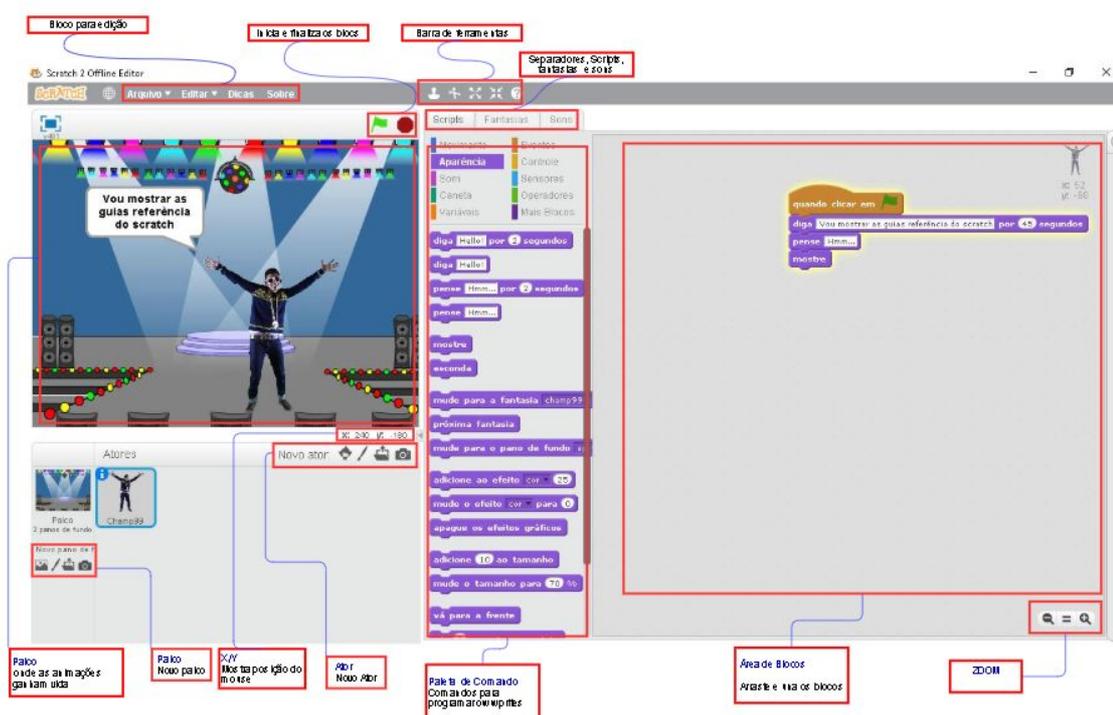
Laudares (1987) trabalha a educação matemática dentro da sala de aula, fazendo uma ligação direta com o objeto de estudo desta pesquisa, relacionando aspectos fundamentais para os alunos que ingressam em um curso superior e se defrontam com uma realidade distinta à enfrentada durante sua formação no ensino médio. Observa-se nesses alunos a falta de interesse e de participação, devido à deficiência em sua base de aprendizagem e que, ao se defrontarem com o conhecimento das TDICs na prática pedagógica, descobrem nelas um auxílio para reduzirem as dificuldades. Dessa forma, acabam se interessando pela solução de problemas apresentados em sala de aula. Nesse sentido, observa-se que o raciocínio lógico-matemático é uma forma de estruturar o pensamento de acordo com as normas estabelecidas pela lógica, permitindo chegar a uma determinada conclusão ou resolver um problema.

Resnick (2017) diz que a ferramenta *Scratch* foi idealizada e desenvolvida pelo grupo *Lifelung Kindergarten Group* do *Massachusetts Institute of Technology Media Lab* e, com essa tecnologia, pode-se organizar histórias, animações, até mesmo, jogos interativos e compartilhar um portfólio de atividades que são criadas, individualmente ou em comunidade, possibilitando que outras pessoas tenham acesso às criações. Essa ferramenta foi desenvolvida, originalmente, para que as crianças desenvolvessem uma linha de raciocínio criativa, além de provocar um pensamento diferenciado que permitisse que as mesmas trabalhassem de forma colaborativa. Ainda assim, a ferramenta *Scratch*, atualmente, é utilizada por pessoas de todas as idades. Para Resnick (2013), a possibilidade de escrever códigos para computadores é muito importante para a sociedade atual, pois ao se aprender a arte

da programação, especialmente, com a ferramenta *Scratch*, aprende-se importantes estratégias para a solução de problemas e apresentação de ideias.

Conforme observado na Figura 6, mostra-se todas as funcionalidades para usar a ferramenta *Scratch* por intermédio das guias de referências.

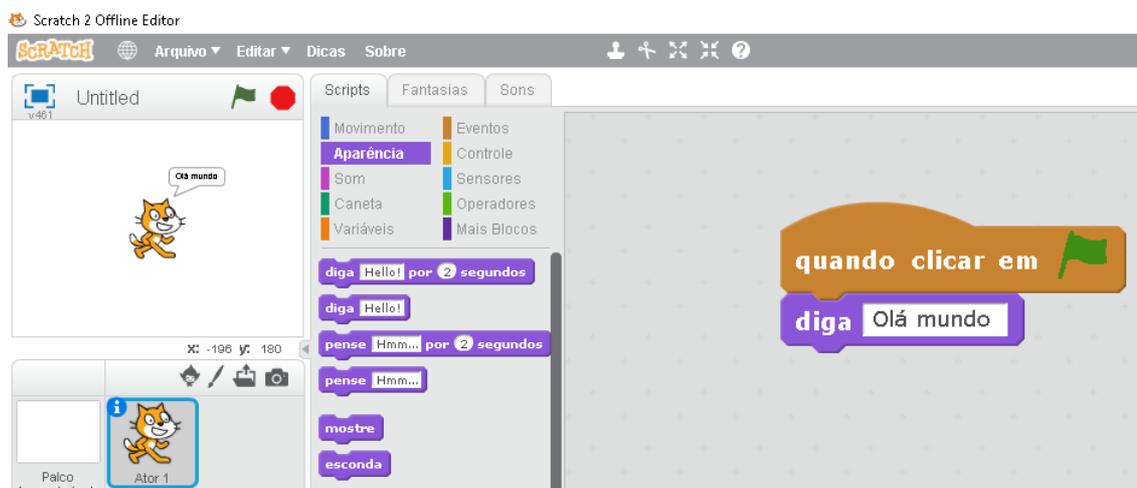
Figura 6 - Tela principal da ferramenta Scratch: Guia de Referência.



Fonte: A pesquisa.

A ferramenta *Scratch* para a construção de um algoritmo não necessita que se digite os pseudocódigos e, sim, que se arraste os blocos no palco ao lado, escrever na caixa de cor roxa, “olá mundo” e pressionar o ícone com uma bandeirinha verde para visualizar o resultado no palco ao lado. Conforme se apresenta na Figura 7.

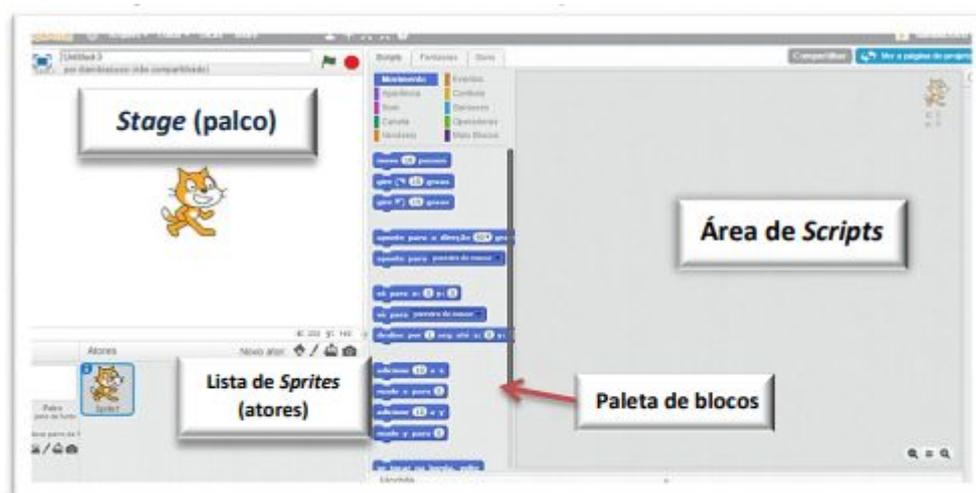
Figura 7 – Utilização básica da ferramenta *Scratch*.



Fonte: A pesquisa.

Para Biazussi (2016) proporcionar atividades interativas é ensinar por meio de tecnologias computacionais, que estejam disponíveis na rede de computadores e possam auxiliar o trabalho do professor. As tecnologias computacionais que permitem simulações na programação são recursos incríveis que permitem a evolução de habilidades na lógica, matemática e pensamento computacional, auxiliando nas resoluções dos problemas computacionais. Nesse sentido, o autor apresenta algumas figuras que demonstram a simplicidade na utilização da ferramenta *Scratch*, na figura 08, pode-se visualizar a divisão em quatro áreas de utilização da ferramenta *Scratch*, o *Stage* (palco), área de *scripts*, a lista de *sprites* (atores) e a paleta de blocos.

Figura 8 - As quatro áreas de trabalho da ferramenta *Scratch*.



Fonte: Biazussi, 2016.

A ferramenta *Scratch* pode ser usada em vários idiomas, se a versão que você está utilizando não é em português, basta clicar globo indicado na figura 09, e escolher a língua portuguesa. A figura 09, também demonstra a funcionalidade da barra de menu, abas e as ferramentas de cursor.

Figura 9 - Barras e ferramentas de cursor.



Fonte: Biazussi, 2016

Observa-se na figura 10, que na aba de movimento, arrastando as paletas de “para x”, combinando com: “diga olá” e mova 10 passos, consegue-se fazer o personagem interagir na tela da ferramenta *Scratch*.

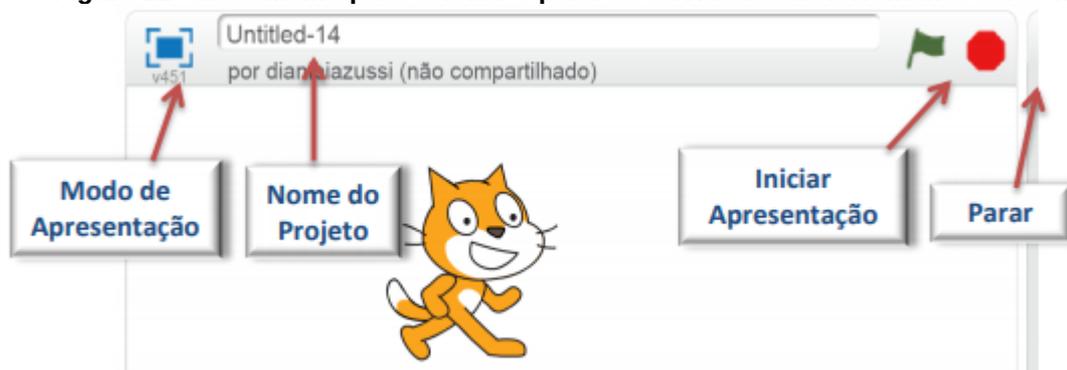
Figura 10 - Aba de movimentos.



Fonte: Biazussi, 2016.

Conforme apresentado na figura 10, pode-se construir uma interação rápida e envolvente com a ferramenta *Scratch*, todavia para fazer o *Sprite* se movimentar no palco é necessário, como apresentado na figura 11, clicar na figura verde para que a ação se inicie e na placa vermelha para a atividade finalize.

Figura 11 - Mecanismo para acionar e parar as atividades na ferramenta *Scratch*.



Fonte: Biazussi, 2016.

Com as informações apresentadas, neste capítulo, entende-se o que é a ferramenta *Scratch*, de onde veio e para que serve. Assim como os princípios básicos de funcionamento dessa tecnologia computacional, que será aplicada para investigar, como a ferramenta *Scratch* pode auxiliar na compreensão de conceitos abordados na disciplina de Algoritmos e Programação, bem como atender aos objetivos propostos nesta pesquisa.

4.1. Conhecimentos Matemáticos e Tecnologias Digitais da Informação e da Comunicação¹⁴

Entende-se que a Educação Matemática é produto de pesquisas que passam pela matemática pura, computacional e escolar, entre outras. Segundo Skovsmose (2018), a chamada matemática pura é uma [matemática](#) que não se preocupa com sua aplicação em uma área específica do conhecimento. A matemática

¹⁴Este capítulo foi publicado parcialmente em: MOREIRA, J.P.; GELLER, M. Uma experiência com o uso do SCRATCH na disciplina de Algoritmos e Programação no curso superior de ADS. In: 3º SEMINÁRIO DE GESTÃO E TECNOLOGIA: COMPETÊNCIAS PARA INOVAR. 2018. Disponível em: < <http://revista.faqi.edu.br/index.php/seminario/article/view/402> >. Acesso em 01 set. 2019.

computacional é a área da [matemática](#) que é aplicada especificamente para a resolução de problemas da [computação](#). A matemática trabalhada na escola tem como interesse e foco a aprendizagem dos alunos, e também se relacionada com áreas da psicologia e outras áreas que empregam a didática. Nesse contexto, estudou-se o trabalho de Machado (2005), que trabalha com aprendizagem, de forma a reforçar para os alunos a necessidade de compreender e raciocinar sobre o que é proposto em sala de aula, assim como organizar o pensamento e não somente memorizar e aplicar fórmulas de forma objetiva, mas como forma de explicar a função que representa. Ao se unir a teoria e a prática sobre a educação matemática.

4.2. Lógica de programação

É comum utilizar a lógica para entender que no cotidiano, pois ela está presente em na rotina das pessoas. Um exemplo: ao atravessar uma rua, a pessoa não se dá conta, mas precisa traçar um algoritmo em sua mente, como fará, por onde irá, deverá olhar atentamente para a esquerda, depois para a direita e, só então ir em frente, atravessando a rua. Assim como no momento de convidar os amigos para um churrasco, precisa-se criar um algoritmo para essa atividade, ou seja, ao dar início ao churrasco: quantos quilos de carne serão necessários, quantos espetos serão usados, usar ou não sal, o que fazer primeiro. Da mesma forma, para se trocar uma lâmpada, pegar a escada, trocar a lâmpada, descer da escada e ligar a chave elétrica. Esses exemplos mostram que a lógica é a forma de pensar corretamente ou de ordenar o pensamento.

Carvalho (2010) afirma que a lógica de programação é uma técnica usada para desenvolver atividades direcionadas à linguagem de programação, usando sequências lógicas. A definição dessas lógicas, quando alinhadas, permitem que os algoritmos tenham uma sequência, a qual, sendo seguida passo-a-passo, possibilita alcançar o objetivo pretendido, e que serão manipuladas de uma forma que se transformem em um algoritmo assim, dando origem a um programa de computador.

Alves (2018) apresenta que um computador que tenha um sistema operacional, que são produtos da lógica de programação, concebidos por profissionais que

conhecem e desenvolvem de forma sequencial programas que usam o algoritmo como linguagem computacional. Hunt e Homas (1999) dizem que a lógica de programação é uma técnica em que desenvolve uma sequência lógica em forma de algoritmo para alcançar objetivos predeterminados.

Alves (2018, p.3) questiona: “Mas afinal, que é um Algoritmo?”. O primeiro passo para se aprender programação não envolve computador, envolve educar a mente para que reconheça em detalhes os passos necessários para executar uma determinada tarefa”. Schank (2002) utiliza várias ferramentas que auxiliam na verificação da lógica da programação no computador, pois, podem ser utilizadas por crianças, uma vez que trabalham desde o nível de programação básica até a mais sofisticada. É importante ressaltar que, a programação não deve ser uma quantidade de código que pouquíssimas pessoas têm o entendimento do que representa, por ser difícil ou não conhecer como funciona. Conforme apresenta Carvalho (2010), a programação estabelece uma possibilidade, em que a forma de orientar atividades e possibilidades que instrui o computador para dar início à execução de uma tarefa, de forma autônoma.

5. ANÁLISE DOS RESULTADOS

Neste capítulo são apresentadas as análises do pré-teste e do pós-teste, além da implementação da proposta de uso da ferramenta *Scratch* na disciplina de Algoritmos e Programação. Destaca-se que o pesquisador ministrou as aulas sobre o uso da ferramenta *Scratch* com a autorização do professor da turma, com anuência dos alunos dos alunos.

Anterior a aplicação do pré-teste, o professor titular da turma pesquisada, ministrou para a turma os seguintes conceitos: apresentação da disciplina, Introdução a Lógica de Programação. Fluxograma e Pseudocódigo, Variáveis, Constantes e Memória. Comandos de entrada e saída, Operadores aritméticos e expressões aritméticas / Prioridade de operadores, teste de bancada. Introdução ao Português Estruturado / Pseudocódigo, *Visualg*, Exercícios em forma de pseudocódigo, Estruturas Condicionais, lista de exercícios sobre Estruturas Condicionais.

5.1. ANÁLISE DO PRÉ-TESTE

O pré-teste se consistiu na aplicação de dez questões (algoritmos), que envolviam os conteúdos trabalhados anteriormente pelo professor titular da disciplina (Apêndice D). Ao analisar os resultados do pré-teste, foi possível evidenciar 04 perfis distintos no grupo de acadêmicos participantes, como mostra a Figura 12. O critério adotado para dividir os perfis, a partir do pré-teste foi o número de acertos, resultando nos perfis apresentados a seguir:

Perfil 1: acadêmicos: F, H e K, que não acertaram nenhuma questão;

Perfil 2: acadêmicos: B, D e E, que acertaram 3 das 10 questões;

Perfil 3: acadêmicos: C, G, J, que acertaram 4 questões;

Perfil 4: acadêmicos: A e I, que acertaram 5 questões.

Figura 12 - Identificação dos perfis dos pesquisados no pré-teste.

Acadêmicos	1	2	3	4	5	6	7	8	9	10	acertos (*)
F	-	-	-	-	-	-	-	-	-	-	0
H	-	-	-	-	-	-	-	-	-	-	0
K	-	-	-	-	-	-	-	-	-	-	0
B	*	*	-	-	*	-	-	-	-	-	3
D	*	-	*	*	-	-	-	-	-	-	3
E	*	*	*	-	-	-	-	-	-	-	3
C	*	*	*	-	-	*	-	-	-	-	4
G	*	*	*	*	-	-	-	-	-	-	4
J	*	*	*	*	-	-	-	-	-	-	4
A	*	*	*	*	*	-	-	-	-	-	5
I	*	*	-	*	*	*	-	-	-	-	5

Fonte: A pesquisa.

5.1.1. Análise dos resultados da Questão 01

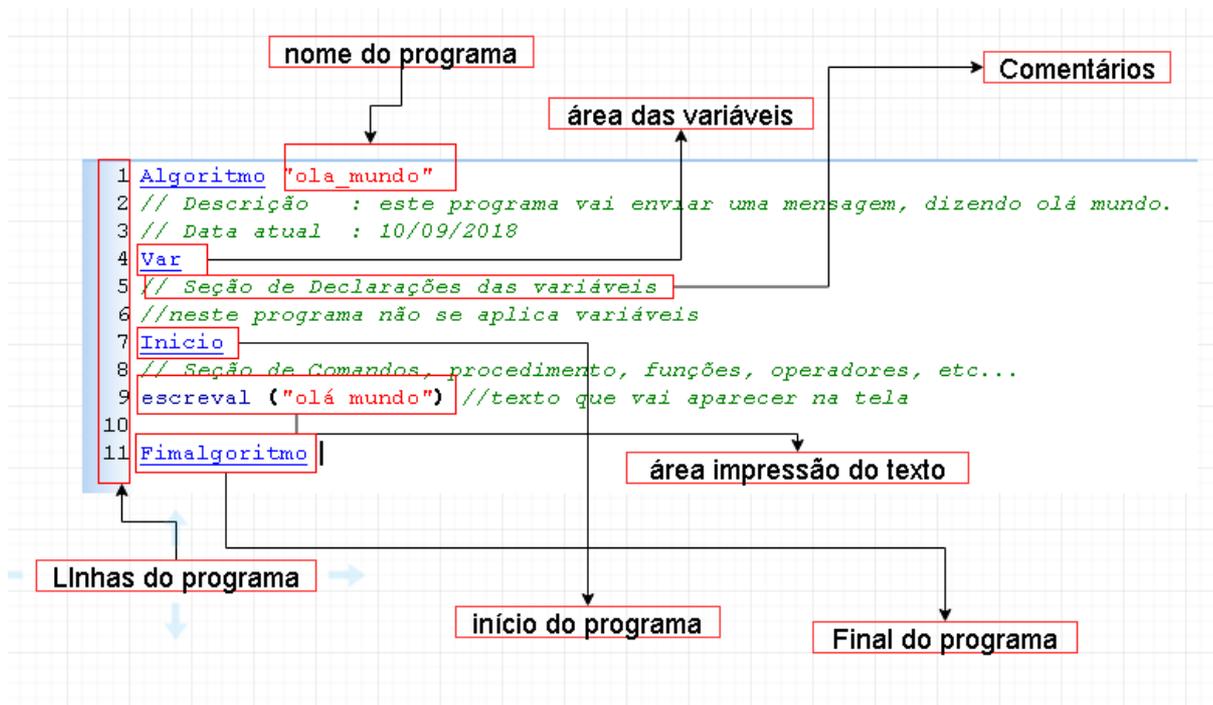
Nessa questão, solicitou-se que fosse desenvolvido um algoritmo, que necessitaria apresentar a expressão “*olá mundo*”, usando para isso o caderno, lápis, caneta ou editor de texto no computador, aplicando uma pseudolinguagem (português estruturado). Bilabila (2018) conceitua que o *portugol* é uma pseudolinguagem em que as instruções (ou comandos) dos algoritmos são escritas em Português. Assim, é possível visualizar a lógica da programação de uma maneira mais prática, retirando a dificuldade da aprendizagem ou da interpretação de outro idioma como, por exemplo, o inglês, que é predominantemente utilizado nas linguagens de programação. Assim mesmo, ainda se necessita que o educando tenha conhecimentos prévios. Esses conhecimentos facilitarão a manipulação da linguagem, assim como outros pré-requisitos, como o entendimento sobre variáveis, constantes, organização de uma linguagem de programação, entradas e saídas de dados, bem como o domínio de operadores lógicos são necessários para poder operacionalizar os algoritmos, permitindo que o programa seja executado de maneira adequada.

Na análise dessa questão, verificou-se que 3 acadêmicos não conseguiram desenvolver a atividade, foram os acadêmicos F, H e K, pois os mesmos, entregaram a atividade “incompleta”, para que os acadêmicos atingissem plenamente todas as etapas dessa questão, seria necessário que apresentassem os elementos destacados por Quierelli (2012): o algoritmo para realizar uma tarefa precisa ser alimentado com dados informados pelos usuários, na chamada entrada de dados. Sendo assim, um algoritmo é executado em três fases: entrada, processamento e saída, obedecendo aos seguintes passos: 1) as variáveis devem ser declaradas na seção *var*, pois, assim, elas podem ser usadas no algoritmo; 2) declarar a palavra reservada, *início*; 3) declarar a palavra reservada *escreva*, sinalizando um texto para a entrada de dados; 4) declarar a palavra reservada *leia* para armazenar o dado que será informado pelo usuário; 5), declarar uma *variável* para receber o cálculo matemático; 6) declarar a palavra reservada *escreva* e informar a *variável* que vai imprimir o resultado final do algoritmo e, finalmente, no passo 7) finalizar apresentar o algoritmo com a palavra reservada *finalgoritmo*. Caso esses passos não sejam todos cumpridos, o algoritmo seria considerado incorreto ou incompleto. Quierelli (2012) apresenta o conceito de variável, destacando que:

todo programa de computador por menor que seja deve fazer uso de variáveis para poder trabalhar de acordo com a proposta para a qual foi projetado, mas afinal, o que é uma variável? Imaginemos a memória RAM do computador que é a memória volátil, ou seja, que mantém os dados enquanto o computador estiver ligado e após o desligamento do mesmo esse valor se perdem, é nela que ocorre todo o processamento das informações do computador e é nela também que se guardam os valores digitados pelo usuário possam ser armazenadas em memória nós temos que reservar um pequeno espaço na mesma para que as informações sejam colocadas lá. Esse espaço na memória que o programador reserva no seu programa é chamado de variável. (QUIERELLI, 2012, p.18)

Na Figura 13, apresenta-se a resolução modelo para a questão 01, com todas as nuances que devem ser seguidas para que a questão referida seja considerada correta.

Figura 13 - Resolução pré-teste questão 01 – Atividade modelo.



Fonte: A pesquisa.

No início deste capítulo, evidenciaram-se 04 perfis para os acadêmicos pesquisados, em que o primeiro traz o recorte dos acadêmicos: F, H e K, que não acertaram nenhuma questão no pré-teste e, em virtude disso, destaca-se o caso do acadêmico F, que entregou a atividade incompleta, conforme se verifica na Figura 09.

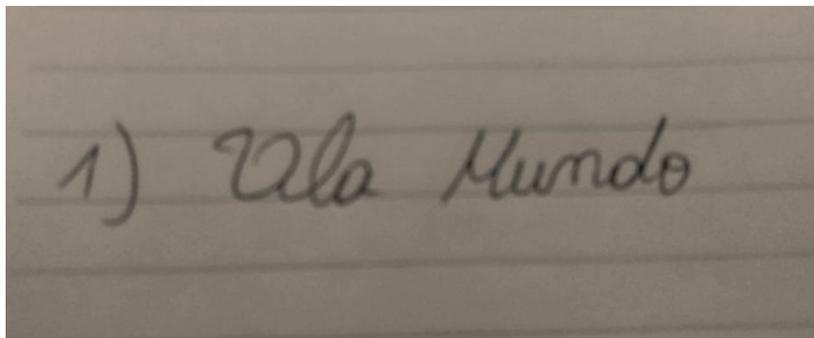
Petry (2015, p.1) afirma que:

A disciplina de algoritmos, que faz parte do currículo de alguns cursos de graduação da área tecnológica, é uma barreira para muitos alunos que se iniciam na área. Em um algoritmo, o aluno precisa desenvolver seu raciocínio lógico propondo soluções de problemas.

Com base nas afirmações de Petry (2015), verifica-se que a resolução apresentada pelo acadêmico F não consegue apresentar a solução esperada para o problema, pois o mesmo apresenta a expressão "olá mundo" de forma literal. Não utilizou nenhum recurso solicitado em programação, o que não constitui um algoritmo computacional. Dessa forma, o acadêmico não atingiu completamente as

etapas solicitadas, tornando sua atividade errada, conforme apresentado na Figura 14.

Figura 14 - Resolução pré-teste, questão 01 – Acadêmico.



Fonte: A pesquisa.

5.1.2. Análise dos resultados da Questão 02

Nessa questão, solicitou-se o desenvolvimento de um algoritmo que somasse dois valores e apresentasse o resultado ao final do algoritmo. Esse recorte apresenta a necessidade de que o acadêmico, enquanto sujeito ativo na construção do conhecimento, compreenda os esquemas de representação, que se equivalham e possam se aliar aos elementos matemáticos.

Para o entendimento dos números inteiros e suas operações, nesse sentido os PCN (BRASIL, 1998, p.67) evidenciam que:

o uso de símbolos e da linguagem Matemática para representar números pode ser estudado do ponto de vista histórico e também do ponto de vista prático. Neste ciclo os alunos têm boas condições para perceber que os números têm múltiplas representações e compreender melhor as relações entre representações fracionárias e decimais, frações equivalentes, escritas percentuais e até a notação científica.

Ao avaliar essa questão, verificam-se que 04 acadêmicos a erraram e 07 acertaram-na. Os 04 acadêmicos que erraram essa questão foram os acadêmicos D, F, H, e o acadêmico K, que entregou a atividade em branco. Para que os acadêmicos atingissem plenamente todas as etapas dessa questão seria necessário que apresentassem os elementos presentes na Figura 15.

Figura 15 - Resolução pré-teste questão 02 – Atividade Modelo.

```

1 Algoritmo "soma_dois_numeros"
2 // Descrição: somar dois numeros e apresentar o resultado.
3 // Data atual: 10/09/2018
4 Var
5 // Seção de Declarações das variáveis
6 numeroa, numerob : inteiro //aqui colocasse as variaveis
7 soma : real //aqui colocasse a variavel para soma
8 Inicio
9 // Seção de Comandos, procedimento, funções, operadores, etc...
10 escreval ("digite um valor numerico") //aqui pede ao usuario
11 //que digite o primeiro número
12 leia(numeroa) //aqui armazena o valor informado pelo
13 // usuario para numeroa
14 escreval ("digite outro valor numerico") //aqui pede ao usuario
15 //que digite o segundo número
16 leia(numerob) //aqui armazena o valor informado pelo
17 //usuario para numerob
18 soma <- numeroa + numerob //aqui é a equação do algoritmos
19
20 escreval("o valor da soma é : ",soma) //aqui se imprime
21 //o resultado final
22 Finalgoritmo

```

Fonte: A pesquisa.

Aqui, evidencia-se a resolução da atividade apresentada pelo acadêmico D, como mostra a Figura 11. Esse acadêmico demonstra compreensão dos preceitos da construção de algoritmos em pseudocódigos, pois declara as variáveis X e Y do tipo inteiro, inicia o algoritmo com a palavra reservada *início*, solicita a entrada de dados pelo usuário por meio da palavra reservada *escreva* ("informe X"), porém não aloca na memória essa informação, pois se esperava uma palavra reservada *leia* (X), a qual não foi evidenciada. Novamente, solicita o armazenamento da variável y, pela palavra reservada *escreva* ("informe Y"). Comete o mesmo lapso em não declarar o armazenamento da variável, por meio da palavra reservada *leia* (y), no que tange ao a lógica de programação. Ocorre novo erro, pois se esperava uma variável, (Z), para armazenar a soma das variáveis (X+Y), com o seguinte comando $Z \leftarrow (X+Y)$, o que não é evidenciado em seu algoritmo. Outro comando não é evidenciado é a palavra reservada *escreva* ("o valor da soma é: Z".) que apresenta o resultado para o usuário, mesmo assim, mantém a estrutura correta com o encerramento do algoritmo por meio da palavra reservada, *fim*, o qual encerra o

algoritmo. Pode-se inferir que o acadêmico D compreendeu o que foi solicitado no enunciado. Ainda assim, pelas dificuldades na compreensão de conceitos matemáticos, e pelas necessidades da estrutura da construção de pseudocódigos em português estruturado, o acadêmico se esqueceu de elencar princípios básicos para o funcionamento desse algoritmo. Sua avaliação final, portanto, é de que o algoritmo está incompleto.

Na disciplina de algoritmos e programação a relação da informática e da matemática são relações de interligação e dependência, verificáveis a partir de exigências definidas pelos PCN (BRASIL, 1998, p.64) que compreende que alguns dos objetivos da Matemática são:

resolver situações-problema envolvendo números naturais, inteiros racionais e a partir deles ampliar e construir novos significados da adição, subtração, multiplicação, divisão, potencialização e radiciação. - Identificar, interpretar e utilizar diferentes representações dos números naturais, racionais e inteiros, indicadas por diferentes notações, vinculando-as aos contextos matemáticos e não matemáticos [...].

Figura 16 - Resolução pré-teste, questão 2 – acadêmico D.

```

2 - VAR
    X, Y : INTEIRO
INICIAR
ESCREVER ("INFORME X")
ESCREVER ("INFORME Y")
    X + Y
FIM
  
```

Fonte: A pesquisa.

5.1.3. Análise dos resultados da Questão 03

Nessa questão, solicitou-se o desenvolvimento de um algoritmo que resolvesse o seguinte problema: calcular quantos anos uma pessoa está completando com base em sua data de nascimento e a data atual. Esse problema se caracteriza por

uma atividade que deve ser construída em pseudocódigo. O cerne da atividade é um problema matemático de subtração.

Nessa questão, 05 acadêmicos erraram e 06 acertaram, respectivamente, a questão 3. Ao se tabular os dados, verificou-se que os acadêmicos que erraram essa questão foram os denominados de B e F, pois os mesmos entregaram a atividade incompleta, e os acadêmicos I, H e K, que entregaram a atividade em branco. Para que os acadêmicos atingissem plenamente todas as etapas dessa questão, seria necessário que apresentassem os elementos presentes na Figura 17.

Figura 17 - Resolução pré-teste questão 03 – Atividade Modelo.

```

Área dos programas ( Edição do código fonte ) -> Nome do arquivo: [IDADE.ALG]
1 Algoritmo "IDADE" //no do algoritmo
2 // Disciplina : [Linguagem e Lógica de Programação]
3 Var
4 // Seção de Declarações das variáveis
5 ano_nasc, ano_atual: inteiro
6 idade: inteiro
7
8 Inicio
9 // Seção de Comandos, procedimento, funções, operadores, etc...
10 escreval ("digite o ano que estamos")
11 leia (ano_nasc)
12 escreval ("digite o ano que você nasceu")
13 leia (ano_atual)
14
15 idade <- ano_atual - ano_nasc //calculo
16
17 escreval ("A sua idade é : ", idade) //saida com o resultado
18
19 Fimalgoritmo
20

```

Fonte: A pesquisa.

Analisou-se, também, a resolução formulada pelo acadêmico F, a qual foi considerada como incorreta, pois o mesmo entregou a atividade incompleta, conforme se verifica na Figura 18.

Saliba (1992) apresenta que:

existem diversas formas de representação de algoritmos, mas não há um consenso com relação à qual seria a melhor delas. O critério usado para classificar hierarquicamente estas formas está diretamente ligado ao nível de detalhe ou, inversamente, ao grau de abstração oferecido. O processo de abstração é uma abordagem dada à solução do problema, onde se consideram apenas os aspectos que são importantes para sua solução. Algumas formas de representação de algoritmos tratam os problemas apenas em nível lógico, abstraindo-se de detalhes de implementação, muitas vezes relacionados com alguma linguagem de programação específica. Por outro lado, existem formas de representação de algoritmos que possuem uma maior riqueza de detalhes e, muitas vezes, acabam por obscurecer a ideia principal, o algoritmo, dificultando seu entendimento [...] (SALIBA, 1992, p.9).

A resolução apresentada pelo acadêmico F estava incorreta, pois o mesmo apresentou um cálculo matemático literal, não utilizando nenhum dos recursos solicitados. Dessa forma, não atingindo as etapas solicitadas anteriormente.

Para resolver um cálculo é necessário acionar diversos mecanismos cognitivos, e isso demanda em uma atividade complexa. Segundo Bastos (2006), a existência de processos verbais da informação, das percepções, dos reconhecimentos dos números, da chamada discriminação viso espacial, atende e auxilia a raciocinar em um formato sintático, nas memórias de longo e de curto prazo. Justo (2009) considera que, para resolver problemas, há a possibilidade de apresentar atividades que não dispensam o sentido para construir os conhecimentos, os quais fundamentam a construção desses conceitos de como se dá o processo de aprendizagem, e com a solução de problemas que os alunos colocam situações em prática, simulando a sua vida diária, adquirindo e resgatando conhecimentos previamente adquiridos e os adaptando às novas situações que serão apresentadas. Para Quaranta e Wolman (2006, p.19), “[...] a resolução de problemas é uma atividade indispensável para construir o sentido dos conhecimentos. Sendo assim, os problemas são um meio fundamental para o ensino de um conceito”. De acordo com o apresentado por Justo (2009), o direcionamento do problema necessita que o aluno abstraia a situação, por meio da semântica, que estabelece as relações apresentadas pelo problema para, somente depois verificar se está operacionalização matemática o ajudará a resolver e apresentar a solução.

Figura 18 - Resolução pré-teste questão 03 – acadêmico F.

Handwritten calculation on lined paper:

$$\begin{array}{r} 2018 \\ - 1995 \\ \hline 0023 \end{array}$$

23 anos

Fonte: A pesquisa.

5.1.4. Análise dos resultados da Questão 04

Nessa questão, solicitou-se o desenvolvimento de um algoritmo que resolvesse o seguinte problema: a conversão de reais em dólares, convencionando que U\$ \$1,00 (um dólar), equivaleria a R\$ 2,22 (dois reais e vinte e dois centavos). Sendo assim, se tenho R\$ 428 reais, quantos dólares terei? Esse problema, para sua resolução, necessita da construção de um algoritmo em português estruturado e sua resolução está centrada em esquemas matemáticos ligados à divisão, para a qual Silva (2014) diz:

pretendíamos mostrar aos alunos a possibilidade de construção de caminhos alternativos de uma mesma situação de divisão usando o método das subtrações sucessivas. [...]durante a situação de aprendizagem oferecida, a turma avançaria para além do algoritmo. Desejávamos que os alunos compreendessem o conceito de divisão envolvendo as ideias de “repartir em partes 27NCTM. Normas para o Currículo e a Avaliação em Matemática Escolar. Lisboa: APM, 2007 (Tradução portuguesa do documento americano). 102 iguais e de medida”, a fim de favorecer escolhas de estratégias mais elaboradas para resolver problemas. (SILVA, 2014, p.101).

Nessa questão, ocorreram 6 erros e 5 acertos. Ao se tabular os dados, verificou-se que 6 acadêmicos erraram esta questão. Os acadêmicos H e K, que entregaram a atividade incompleta e os acadêmicos B, C, E e F, que entregaram a atividade em branco. Para que os acadêmicos atingissem plenamente todas as etapas dessa questão seria necessário que apresentassem, os elementos presentes na Figura 19.

Figura 19 - Resolução questão 04 – questão modelo.

```

1
2
3 Algoritmo "conversao_dolar"
4 // Data atual :
5 Var
6 // Seção de Declarações das variáveis
7 //lembrando que a palavra real é um termo reservado para o sistema
8 //então deve-se aplicar reais
9 reais : inteiro //aqui colocasse as variaveis
10 conversao, dolares : real //aqui colocasse a variavel para soma
11 Inicio
12 // Seção de Comandos, procedimento, funções, operadores, etc...
13 escreval ("quantos reais você tem") //aqui pede ao usuario que digite
14 //o primeiro número
15 leia(reais)//aqui armazena o valor informado pelo usuario para numeroa
16
17 conversao <- reais / 2.22//aqui é a equação do algoritmos
18
19 escreval("Você tem : ",conversao:2:2," em dolares")//aqui se imprime o
20 //resultado final
21 //a função conversao:2:2, faz arredondamento para duas
22 //casas após a virgula
23 Fimalgoritmo

```

Fonte: A pesquisa.

Analisando-se a questão 04, ficou em evidência a atividade realizada pelo acadêmico H que entregou a atividade incompleta, conforme se verifica na Figura 20. Algumas das dificuldades que os acadêmicos apresentam, segundo Sánchez e Fernandez (2006, p.23), são:

o foco desejado para o processo de ensino e aprendizagem da matemática é a abordagem cognitiva, algo que trabalhe a compreensão dos alunos. O processo para iniciar o ensino da matemática, deve se partir da intuição e progressivamente aproximando da dedução". Desta forma ele visa construir o conhecimento matemático e não fica somente nas tentativas de modos mecânicos usados para resolução de problemas e algoritmo e por outro lado vincula a um planejamento de ensino e aprendizagem fundamentados no nível de raciocínio do aluno. Neste contexto, para os alunos que cursam o ensino fundamental, sua mente pode funcionar com o raciocínio verbal concreto, ou seja, por exemplificações, percepções de relação à aprendizagem escolar com que mais se tem próximo à realidade vivenciada em termos concretos.

A resolução apresentada pelo acadêmico H está incorreta, embora o mesmo demonstre compreensão dos preceitos da pseudolinguagem em português estruturado, pois denomina o algoritmo como conversão, declara a variável A do tipo inteiro, e declara as variáveis B e C do tipo real. Ainda inicia o algoritmo com o a palavra reservada, *início*, solicita a entrada de dados pelo usuário por meio da palavra reservada *escreva* ("digite o valor em reais"), armazena o valor na memória por meio da palavra reservada *leia* (a). Equivoca-se ao declarar $c \leftarrow (a*b)$, o que não

evidencia o cálculo matemático correto para proceder a conversão de reais para dólares. Outro lapso verificado é que não é descrito a palavra reservada *escreva* ("o valor da conversão é:", c.) que apresenta o resultado para o usuário. Ainda assim, mantém a estrutura da escrita com o encerramento do algoritmo por meio da palavra reservada *finalgoritmo*. Pode-se inferir que o acadêmico H compreendeu o que foi solicitado no enunciado do problema, **porém** não aplicou corretamente a conversão matemática e a estrutura da pseudolinguagem em português estruturado. A avaliação final é de que o algoritmo está incorreto.

Figura 20 - Resolução questão 04 – acadêmico H.

```
4) - algoritmo "conversao"  
  
var  
  a : inteiro  
  b, c: real  
  
Inicio  
  escreva ("digite valor em reais")  
  leia(a)  
  c <- a * b  
  
Final
```

Fonte: A pesquisa.

5.1.5. Análise dos resultados da Questão 05

Nessa questão, solicitou-se o desenvolvimento de um algoritmo que resolvesse o seguinte problema: a conversão entre graus *Celsius* e *Fahrenheit*, apresentando a seguinte fórmula para conversão: $C = F - 32 / 1.8$, que representa que os graus *Celsius* (C) equivalem à escala de graus *Fahrenheit* (F). Seus pontos de referência são o ponto de congelamento da água em 32°F, e o ponto de ebulição da água, em

212°F. Analisando-se os resultados dessa questão, verificou-se que 08 acadêmicos erraram e 03 acertaram. Os acadêmicos F e G que entregaram a atividade incompleta e os acadêmicos C, D, E, H, J e K, que entregaram as atividades em branco. Para que os acadêmicos atingissem plenamente todas as etapas desta questão seria necessário que apresentassem, os elementos presentes na Figura 21.

Figura 21 - Resolução questão 05 – questão modelo.

```

1 Algoritmo "conversao_graus" //Titulo do algoritmo
2 // Disciplina: Conversão de graus fahrenheit em celsius
3 // Data atual: 10/09/2018.
4 Var
5 // Seção de Declarações das variáveis
6 celsius, fahrenheit : inteiro //aqui colocasse as variaveis
7 conversao : real //aqui colocasse a variavel para soma
8 Inicio
9 // Seção de Comandos, procedimento, funções, operadores, etc...
10 escreval ("informe os graus fahrenheit") //aqui pede ao usuario que digite
11 // o primeiro valor
12 leia(fahrenheit)//aqui armazena o valor informado pelo usuario para graus
13 conversao <- (fahrenheit - 32) / 1.8 //aqui é a equação do algoritmos
14 escreval("Esta fazendo : ",conversao:2:2," graus Celsius")//aqui se imprime
15 // o resultado final
16 //a função conversao:2:2, faz arredondamento para
17 // duas casas após a virgula
18 Fimalgoritmo //final do algoritmo

```

Fonte: A pesquisa.

Algumas das dificuldades que os acadêmicos apresentam, encontram-se destacadas por Falckembach e Araujo (2006, p.909) que relatam que:

as dificuldades encontradas pelos alunos na Resolução de Problemas, via computador. A dificuldade está no reconhecimento dos procedimentos necessários para se chegar à solução do problema. Isso implica em ter que trabalhar de forma mais eficiente os processos cognitivos, em especial a abstração e a formalização, necessários à construção de um algoritmo, ou seja, à modelagem da solução do problema por meio da técnica de algoritmos pseudocódigos. Resolver problemas inclui uma predisposição para o questionamento de quais os processos cognitivos que devem ser utilizados, a fim de gerar o conhecimento necessário à resolução do problema. Envolve as operações de pensar, de análise, de síntese e de avaliação.

A resolução apresentada pelo acadêmico G foi parcial em relação ao que foi solicitado, pois se verificou que o mesmo demonstra compreensão dos preceitos da pseudolinguagem em português estruturado, pois denomina o algoritmo como graus, declara as variáveis *Celsius* e *Fahrenheit* do tipo inteiro e declara a variável conversão, com o tipo real. Seguindo a análise do algoritmo, é possível verificar que o acadêmico inicia o algoritmo com a palavra reservada, *inicio*, solicita a entrada de dados pelo usuário por meio da palavra reservada *escreval*, sendo que o "L", após o escreva, significa que deve deixar uma linha em branco entre a entrada e a saída de dados, ("informe, Fahrenheit"), porém esquece de armazenar o valor na memória,

não apresentando a palavra reservada *leia* (Fahrenheit). Apresenta o cálculo correto, aplicando a fórmula para converter *Fahrenheit* em *Celsius*, conversao <- (Fahrenheit – 32) /1.8. Não declara a palavra reservada, *escreva*, para imprimir na tela o resultado do cálculo, o que não evidencia o resultado matemático correto, para proceder à conversão. Mesmo assim, mantém a estrutura com o encerramento do algoritmo por meio da palavra reservada *finalgoritmo*. Pode-se inferir que o acadêmico H compreendeu o que foi solicitado no enunciado do problema, mas não aplicou corretamente a estrutura da pseudolinguagem em português estruturado. A avaliação final é de que o algoritmo está incorreto.

Figura 22 - Resolução questão 05 – acadêmico G.

```

algoritmo "graus"
var
celsius, fahrenheit: inteiro
conversao: real
Inicio
escreval("informe, fahrenheit")
conversao <- (fahrenheit - 32) / 1.8
fim

```

Fonte: A pesquisa.

5.1.6. Análise dos resultados da Questão 06

Nessa questão, solicitou-se que os acadêmicos desenvolvessem um algoritmo para calcular 60 por cento de imposto, sabendo que o valor total em dólares é \$4.280,00. A aplicabilidade da Matemática ao dia a dia dos alunos permite uma repercussão sobre a aprendizagem, estimulando e relacionando essas assimilações com a realidade nas quais estão inseridas.

Nesse contexto, Dias (2018) destaca que:

como o assunto de porcentagem está muito presente na comunidade escolar na qual desempenho o meu trabalho, aprofundi a minha pesquisa nessa área de estudo, na tentativa de contribuir para que meus alunos pudessem reconhecer que a Matemática da sala de aula pode ser aplicada no seu dia-a-dia, e que o domínio desse conhecimento os auxiliaria a torná-los participantes em sua sociedade. Observo que trabalhar com conceitos matemáticos, a partir de suas aplicações em situações do dia-a-dia do aluno pode levar a um aprendizado duradouro, agradável e criativa (DIAS, 2019, p.18).

Segundo Oliveira (2010), os cálculos, uma vez resolvidos por operações matemáticas, confundem-se com a própria história da Matemática, que apresenta que os homens, desde as épocas mais antigas e remotas, buscavam atividades e habilidades técnicas para aprimorar sua prática, ou seja, para operacionalizar as atividades e que, a essas, deram o nome de cálculos. O cálculo numérico se reporta aos algoritmos de adição, subtração, multiplicação e divisão, podendo ser considerados como técnica. O cálculo relacional reúne as operações de pensamento necessárias para trabalhar com relações envolvidas nas situações.

Nessa questão, verificou-se que 09 acadêmicos não acertaram, foram eles os acadêmicos F, G, H, J e K, que entregaram a atividade incompleta, além dos acadêmicos A, B, D e E, que entregaram as atividades em branco. Para que os acadêmicos atingissem plenamente todas as etapas dessa questão, seria necessário que apresentassem, os elementos presentes na Figura 23, que mostra o resultado da atividade realizada.

Figura 23 - Resolução questão 06 – questão modelo.

```

1 Algoritmo "imposto"
2 // Descrição: calcular o imposto.
3 // Data atual: 10/09/2018
4 Var
5 // Seção de Declarações das variáveis
6 valor_gasto : inteiro //aqui colocasse as variaveis
7 valor_pagar, imposto, percentual : real //aqui colocasse a variavel para soma
8 Inicio
9 // Seção de Comandos, procedimento, funções, operadores, etc...
10 escreval ("informe o valor gasto em produtos") //aqui pede ao usuario que
11 // digite o valor
12 leia(valor_gasto)//aqui armazena o valor informado pelo usuario
13 percentual <- 60/100
14 imposto <- (valor_gasto * percentual) //aqui é a equação do algoritmos
15 escreval("O valor do imposto a ser pago é de   : ",imposto)//aqui se imprime
16 // o resultado final
17 Fimalgoritmo

```

Fonte: A pesquisa.

Em seu artigo sobre as dificuldades que os acadêmicos apresentam no aprendizado de algoritmos, Falckembach e Araujo (2006) relatam que:

dificuldade na Resolução de Problemas considerando as dificuldades encontradas pelos alunos dos cursos da área de Ciências Exatas na Resolução de Problemas, via computador, e conseqüentemente no aprendizado de Algoritmos; considerando que esse conteúdo é de fundamental importância para os cursos da área de Informática; considerando também que cada aluno tem o seu ritmo de trabalho e que não é possível ao professor adequar-se às necessidades de cada aluno,

esses fatores justificaram a concepção e o desenvolvimento de um ambiente de aprendizagem para Algoritmos. O processo mental para a aprendizagem de Algoritmos é novo e isso representa uma dificuldade para o aluno que está iniciando na computação, gerando, muitas vezes, resistência ao aprendizado (FALCKEMBACH e ARAUJO, 2006, p.909).

A resolução apresentada pelo acadêmico G, foi parcial em relação ao que foi solicitado, pois o mesmo demonstra compreensão dos preceitos da pseudolinguagem em português estruturado. O mesmo denomina o algoritmo como imposto, declara a variável gasto com o tipo inteiro, mas não declara as variáveis pagar, imposto e percentual do tipo, real, inicia o algoritmo com a palavra reservada, *início*, solicita a entrada de dados pelo usuário por meio da palavra reservada *escreva* ("informe o valor"), armazena o valor na memória por meio da palavra reservada *leia* (gasto), atribui um percentual $\leftarrow (60 \cdot 100)$, $\text{Imposto} \leftarrow (\text{gasto} \cdot \text{percentual})$ o que evidencia o cálculo matemático, que necessita do conhecimento de juros simples e está correto para proceder à operação. Como não declarou as variáveis, que foram usadas para o cálculo no início do algoritmo, inviabiliza toda a operação e modifica o resultado final, embora declare corretamente a palavra reservada *escreva* ("o imposto é: ", imposto), e finaliza o algoritmo com a palavra reservada *fimalgoritmo*. Assim os resultados que serão processados e apresentados estarão totalmente distorcidos em função da organização escrita não ter sido apresentada de forma correta.

Figura 24 - Resolução questão 06 – acadêmico G.

```

algoritmo "imposto"
var
gasto : inteiro

Inicio
escrever("informe o valor")
ler(gasto)
percentual  $\leftarrow$  60 * 100
imposto  $\leftarrow$  (gasto * percentual)
escrever("o imposto é :", imposto)
Fim

```

Fonte: A pesquisa.

5.1.7. Análise dos resultados da Questão 07

Nessa questão, solicitou-se o desenvolvimento de um algoritmo que resolvesse o seguinte problema: calcular quanto alguém pagará, após 10 meses, tendo tomado emprestada a quantia de R\$ 5.000,00 reais, com taxas de 2% de juros ao mês. Para o desenvolvimento do algoritmo, foram elaborados problemas, que

demandavam trabalhar juros simples e compostos, visto que para Dias (2019, p.46), o conteúdo de juros composto se mostrou importante, pois:

para este grupo de alunos. Foram realizadas algumas atividades para serem trabalhadas na sala de aula. No decorrer da seção, destaco algumas delas. No primeiro momento, realizamos uma atividade para identificar e diferenciar juro simples de juro composto; e no segundo, trabalhamos o conceito de juro composto e problemas presentes no cotidiano dos alunos. No trabalho anterior, nos exemplos trazidos pelos alunos sobre juros, verifiquei a necessidade imediata de os alunos identificarem a diferença entre juros simples e juros compostos. Organizei uma atividade que consistia em responder, individualmente, qual a diferença entre juro simples e composto, que ajudaria a desencadear o processo de aprendizagem.

Ao se tabular os dados, verificou-se que os 11 acadêmicos erraram esta questão, sendo que os acadêmicos I, F e G, entregaram suas atividades incompletas e os demais em branco. Para que os acadêmicos atingissem plenamente todas as etapas **dessa** questão, seria necessário que apresentassem, os elementos presentes na Figura 25, que mostra o resultado da atividade realizada.

Figura 25 - Resolução questão 07 – questão modelo.

```

- Área dos programas ( Edição do código fonte ) -> Nome do arquivo: [questão 07 - imposto.alg] -
1 algoritmo "imposto"
2 // Disciplina : [Linguagem e Lógica de Programação]
3 var
4 //declarar as variáveis
5 t, capital, montante, taxa, juros : real
6
7 inicio
8 // Seção de Comandos
9 escreval ("valor que voce pegou de emprestimo")
10 leia(capital)
11 escreval ("qual porcentagem de juros")
12 leia(taxa)
13 escreval ("quantidades de parcelas")
14 leia(t)
15
16 taxa <- taxa /100
17 juros <- capital * taxa * t
18 montante <- capital + juros
19
20 escreval ("ao final de :",t)
21 escreval ("o valor a ser pago é de :", montante)
22
23 fimalgoritmo

```

Fonte: A pesquisa.

O Acadêmico G é destaque nessa atividade, uma vez que o mesmo entregou sua atividade com as solicitações realizadas pelo professor, referentes a resolução do algoritmo usando *pseudocódigo*. O algoritmo recebe o título de *imposto*, declarou a variável *gasto* com o tipo *inteiro*, declarou as variáveis *pagar*, *imposto* e *percentual*

com o tipo *real*, iniciou o algoritmo com a palavra reservada *início*, solicitou a entrada de dados pelo usuário por meio da palavra reservada *escreva* (“informe o valor”), armazena o valor na memória, apresentando a palavra reservada *leia* (*gasto*). Não apresentou o cálculo correto, demonstrando desconhecimento sobre juros, aplicando erroneamente a fórmula para o cálculo. A seguir, o acadêmico declara a palavra reservada para imprimir na tela, porém o resultado do cálculo, será impresso errado. Sendo assim, verifica-se que o acadêmico não construiu o resultado matemático correto para proceder ao cálculo do juro. Mesmo assim, mantém a estrutura da escrita com o encerramento do algoritmo por meio da palavra reservada *fim*, quando o correto seria a palavra reservada *finalgoritmo*. Pode-se inferir que o acadêmico G compreendeu o que foi solicitado no enunciado do problema, contudo não aplicou corretamente o cálculo matemático. A avaliação final é de que o algoritmo está incorreto, conforme se apresenta na Figura 26.

Figura 26 - Resolução questão 07 – acadêmico G.

```
algoritmo "juros"
var
  emprestimo, pagamento, juro, pagamentodez : real
Inicio
  escrever ("informe o valor")
  ler (emprestimo)
  juro <- 2/100
  pagamento <- emprestimo / juro
  pagamentodez <- pagamento * 10
  escreval ("O valor do imposto é :", pagamentodez)
Fim|
```

Fonte: A pesquisa.

5.1.8. Análise dos resultados da Questão 08

Nessa questão, solicitou-se o desenvolvimento de um algoritmo para informar a data de nascimento, a data atual e calcular a idade a partir desses dados de entrada. Posteriormente, solicitou-se que fosse feito um comparativo, se o usuário é maior ou menor de idade. Para se seguir uma linha de pensamento que embasasse a resolução dessas atividades, utilizou-se os PCNs+, que são os parâmetros referentes ao Ensino Médio (BRASIL, 2003), que apresentam uma visão sobre a Educação diferente da proposta de transmitir conhecimento de maneira estanque e de forma isolada, propondo que o foco sejam as competências, que possam ser articuladas, são evidenciados nos PCNs, que:

aprender matemática de uma forma contextualizada, integrada e relacionada a outros conhecimentos traz em si o desenvolvimento de competências e habilidades que são essencialmente formadoras, à medida que instrumentalizam e estruturam o pensamento do aluno, capacitando-o para compreender e interpretar situações, para se apropriar de linguagens específicas, argumentar, analisar e avaliar, tirar conclusões próprias[...] (BRASIL, 2003, p. 111).

Nessa questão, verificou-se que, dos 08 acadêmicos que não acertaram, 03 entregaram a atividade incompleta (sendo os acadêmicos, D, K e G). Os demais acadêmicos entregaram as atividades em branco. Ao se tabular os dados, verificou-se que para que os acadêmicos atingissem plenamente todas as etapas dessa questão, seria necessário que apresentassem, os elementos presentes na Figura 27, que mostra o resultado da atividade realizada.

Figura 27 - Resolução questão 08 – questão modelo.

```

4 // Descrição : Aqui você descreve o que o programa faz! (função)
5 // Autor(a) : Nome do(a) aluno(a)
6 // Data atual :
7 Var
8 // Seção de Declarações das variáveis
9 //lembrando que a palavra real é um termo reservado para o sistema
10 //então deve-se aplicar reais
11 data_nasc, data_atual : inteiro
12 idade : real //aqui colocasse a variavel para soma
13 Inicio
14 // Seção de Comandos, procedimento, funções, operadores, etc...
15 escreval ("informe o ano atual") //aqui pede ao usuario que digite o primeiro n
16 leia(data_atual)//aqui armazena o valor informado pelo usuario para numeroa
17 escreval ("informe o ano nascimento") //aqui pede ao usuario que digite o prime.
18 leia(data_nasc)//aqui armazena o valor informado pelo usuario para numeroa
19
20 idade <- data_atual - data_nasc
21
22 se ( idade >= 18) entao
23     escreval ("Você é de maior, e tem ", idade, " anos ")
24 senao
25     escreval ("Você é de menor, e tem ", idade, " anos ")
26 fimse
27
28 fimalgoritmo

```

Fonte: A pesquisa.

Evidencia-se, aqui, a resolução apresentada pelo acadêmico D, foram solicitados, os preceitos da representação na pseudolinguagem em português estruturado, no qual declaram as variáveis *datatual* e *datnasc*, do tipo inteiro, e inicia o algoritmo com a palavra reservada, *inicio*. Solicita a entrada de dados pelo usuário por meio da palavra reservada *escreva* (“informe data atual”), armazena o valor na memória por meio da palavra reservada *leia* (*datatual*), declara o cálculo de forma adequada para calcular a idade por meio da palavra reservada *idade <- datatual – datnasc*. Ainda assim, não evidencia conhecimentos sobre comandos condicionais, estudados com o professor titular da disciplina, por intermédio do teste de bancada e posteriormente com a utilização do *Visualg* os quais iriam conduzir o final do algoritmo, para destinar o resultado obtido para uma resposta se o usuário é maior

ou menor de idade. A avaliação final é de que o algoritmo está errado, conforme mostra a Figura 28.

Figura 28 - Resolução questão 08 – acadêmico D.

8 - VAR
 DATATUAL, DATNASC : INTEIRO
 INICIAR
 ESCREVER ("INFORME DATA ATUAL")
 LER (DATATUAL)
 ESCREVER ("INFORME DATA NASCIMENTO")
 IDADE ← DATATUAL - DATNASC
 FIM

Fonte: A pesquisa.

5.1.9. Análise dos resultados da Questão 09

Nessa questão, solicitou-se que os acadêmicos desenvolvessem um algoritmo que perguntasse um número, e que o resultado do algoritmo informaria se o mesmo é par ou ímpar. Nessa questão, evidencia-se que a maior dificuldade não está na lógica de programação, mas, sim, no que se encontra nos PCNs+ do Ensino Médio (BRASIL, 2003, p.112) destacando que:

A resolução de problemas é peça central para o ensino de Matemática, pois o pensar e o fazer se mobilizam e se desenvolvem quando o indivíduo está engajado ativamente no enfrentamento de desafios. Essa competência não se desenvolve quando propomos apenas exercícios de aplicação dos conceitos e técnicas matemáticos, pois, neste caso, o que está em ação é uma simples transposição analógica: o aluno busca na memória um exercício semelhante e desenvolve passos análogos aos daquela situação, o que não garante que seja capaz de utilizar seus conhecimentos em situações diferentes ou mais complexas.

Nessa questão, verificou-se que, nenhum acadêmico acertou a mesma. Ao se tabular os dados, verificou-se que para que os acadêmicos atingissem plenamente todas as etapas, seria necessário que apresentassem, os elementos presentes na Figura 29, que mostra o resultado da atividade realizada.

Figura 29 - Resolução questão 09 – questão modelo.

```

Área dos algoritmos ( Edição do código fonte ) -> Nome do arquivo: [par ou impar.ALG]
1 Algoritmo "par_ou_impar"
2 // Disciplina : [Linguagem e Lógica de Programação]
3 // Professor :
4 // Descrição : Aqui você descreve o que o programa faz! (função)
5 // Autor(a) : Nome do(a) aluno(a)
6 // Data atual :
7 Var
8 // Seção de Declarações das variáveis
9 //lembrando que a palavra real é um termo reservado para o sistema
10 //então deve-se aplicar reais
11 numero : inteiro
12 resultado, maior, menor : real //aqui colocasse a variavel para soma
13 Inicio
14 // Seção de Comandos, procedimento, funções, operadores, etc...
15 escreval ("informe o numero") //aqui pede ao usuario que digite o primeiro número
16 leia(numero) //aqui armazena o valor informado pelo usuario para numeroa
17
18 se (numero mod 2 = 0 ) entao
19
20 escreval ("Este número é par")
21 senao
22 escreval ("Este número é impar")
23 fimse
24
25 fimalgoritmo

```

Fonte: A pesquisa.

Considerando os resultados dessa questão, destaca-se que os acadêmicos G, F e K entregaram as atividades incompletas e, os demais acadêmicos, entregaram suas atividades em branco. Entre os resultados, salienta-se a resolução apresentada pelo acadêmico G. A análise da resposta do acadêmico permitiu verificar que o mesmo compreende os preceitos da construção de algoritmos em pseudocódigo, pois denomina o algoritmo como *par-ou-ímpar*. O referido acadêmico se equivoca ao não declarar as variáveis, o que inviabiliza o funcionamento do algoritmo, porém inicia o algoritmo com a palavra reservada *inicio*. Solicita a entrada de dados pelo usuário por meio da palavra reservada *escreva* (“*digite um valor numérico*”), armazena o valor na memória por meio da palavra reservada *leia* (*número*). Ainda assim, o acadêmico declara o cálculo de forma inadequada, visando identificar se o valor é par ou ímpar, por meio da instrução *se número = 2, escreva(“par”)*, *se número = 3 escreva(“ímpar”)*, o que não elucida o problema. Sendo assim, o acadêmico não evidencia os conhecimentos sobre comandos condicionais e, também, não demonstra conhecimentos sobre o operador resto, conceitos esses

que iriam conduzir ao final do algoritmo, e que foram ensinados nas aulas iniciais pelo professor titular da disciplina e para destinar o resultado obtido para uma resposta ao usuário se o número é par ou ímpar. A avaliação final é de que o algoritmo está incorreto, conforme mostra a Figura 30.

Figura 30 - Resolução questão 09 – acadêmico G.

```
"par-ou- impar"  
var  
numero: inteiro  
Iniciar  
escrever ("digite um valor numérico")  
ler(numero)  
se numero = 2  
escrever ("par")  
se numero = 3  
escrever ("ímpar")  
fim
```

Fonte: A pesquisa.

5.1.10. Análise dos resultados da Questão 10

Nessa questão, solicitou-se aos acadêmicos que elaborassem um algoritmo que calculasse o Índice de Massa Corpórea (IMC), que é um dos indicadores das condições físicas, utilizado mundialmente por profissionais da área de saúde por ser confiável, de simples determinação e de baixo custo (MOTA et al., 2011). Verificou-se que, nenhum acadêmico acertou a questão. Ao se tabular os dados, verificou-se que para os acadêmicos atingissem plenamente todas as etapas, seria necessário que apresentassem, os elementos presentes na Figura 31, que mostra o resultado da atividade realizada.

Figura 31 - Resolução da questão 10 – questão modelo.

```

Área dos algoritmos ( Edição do código fonte ) -> Nome do arquivo: [IMC.ALG]
1 Algoritmo "IMC"
2 Var
3   altura, peso, imc : real
4 Inicio
5   escreva ("Informe sua altura(em metros):")
6   leia (altura)
7   escreva ("Informe seu peso(em Kg):")
8   leia (peso)
9   imc <- peso/(altura * altura)
10  se (imc < 17) entao
11    escreva ("-MUITO ABAIXO DO PESO-")
12  senao
13    se (imc >=17 e (imc < 18.5)) entao
14      escreva ("-ABAIXO DO PESO-")
15    senao
16      se ((imc >= 18.5) e (imc <= 25)) entao
17        escreva ("-PESO IDEAL-")
18      senao
19        se ((imc > 25) e (imc <= 30)) entao
20          escreva ("-SOBREPESO-")
21        senao
22          se ((imc > 30) e (imc <= 35)) entao
23            escreva ("-OBESIDADE-")
24          senao
25            se ((imc > 35) e (imc <= 40)) entao
26              escreva ("-OBESIDADE SEVERA-")
27            senao
28              escreva ("-OBESIDADE MÓRBIDA-")
29          fimse
30        fimse
31      fimse
32    fimse
33  fimse
34 fimse
35 Fimalgoritmo
36

```

Fonte: A pesquisa.

Verificou-se que os 11 acadêmicos entregaram a atividades em branco. Para a resolução da atividade, os acadêmicos precisavam colocar em prática todos os conteúdos apresentados, até então, pelo professor titular da disciplina com relação à resolução de algoritmos, utilizando a conversão do pseudocódigo para português estruturado, inicialmente nominando o algoritmo, como *IMC*, depois na seção *Var*, declarar as variáveis: *altura*, *peso*, *imc*, com o tipo real, solicitar a entrada de dados com referente à altura do usuário, e armazenar essa variável, solicitar a informação do usuário com relação ao peso. Aplicar a fórmula do IMC:

Solicitou-se que os acadêmicos desenvolvessem um algoritmo que perguntasse qual a altura e o peso do usuário e que fosse aplicada a fórmula $IMC =$

massa / altura². Com base nesta fórmula, o algoritmo deveria escrever o resultado da indicação do nível de peso, de acordo com a tabela da Figura 32.

Figura 32 - Índices do IMC, para resolução da atividade 10.

Escala de peso	Legenda
Abaixo do 17	Muito abaixo do peso
Entre 17 e 18,5	Abaixo do peso
De 18,5 a 25	Peso ideal
De 25 a 30	Sobrepeso
De 30 a 35	Obesidade
De 35 a 40	Obesidade severa
40 ou mais	Obesidade mórbida

Fonte: A pesquisa.

5.2. ANÁLISE DO PÓS-TESTE

A análise do pós-teste consiste na aplicação de dez questões (algoritmos), que envolviam os conteúdos trabalhados anteriormente pelo professor titular da disciplina, utilizou-se o chamado teste de bancada, a utilização do *software Visualg*, aplicou-se um pré-teste o qual foi analisado anteriormente, e então se implementou a utilização da ferramenta *Scratch*. Ao analisar o resultado do pré-teste, foi possível evidenciar 05 perfis distintos que são apresentados na figura 33, em que os acadêmicos: H e K, não acertaram nenhuma questão; B, C e G, acertaram 05, questões; D acertou 07, questões; A, F e J, acertaram 09, questões e, por fim, E e I que acertaram 10, questões.

Figura 33 - Acertos por questão no pós-teste.

Acadêmicos	1	2	3	4	5	6	7	8	9	10	acertos (*)
H	-	-	-	-	-	-	-	-	-	-	0
K	-	-	-	-	-	-	-	-	-	-	0
B	*	*	*	-	*	-	-	-	*	-	5
C	*	*	*	-	-	*	-	-	*	-	5
G	*	*	*	*	*	-	-	-	-	-	5
D	*	*	*	*	-	-	-	*	*	*	7
A	*	*	*	*	*	*	*	*	*	*	9
F	*	*	*	*	*	*	*	*	*	*	9
J	*	*	*	*	*	*	*	*	*	*	9
E	*	*	*	*	*	*	*	*	*	*	10
I	*	*	*	*	*	*	*	*	*	*	10

Fonte: A pesquisa.

5.2.1. Análise dos resultados da Questão 01

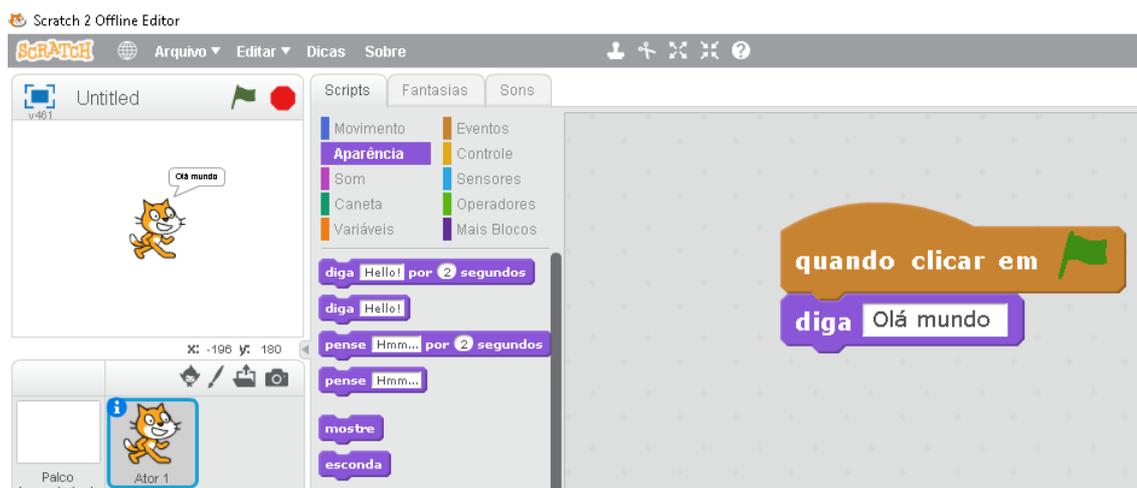
Nessa questão, solicitou-se que fosse desenvolvido um algoritmo, que necessitaria apresentar a expressão “olá mundo”, e para isso se empregou a tecnologia computacional *Scratch*. Analisando os resultados, verifica-se que houve a entrega de duas atividades em branco pelos acadêmicos H e K, e que os demais atingiram completamente as etapas solicitadas.

Silva (2016) corrobora ao trazer que:

[...] a lógica de programação para a resolução de problemas e as atividades em grupo na criação de algoritmos, [...] os alunos demonstraram-se satisfeitos em aprender Matemática via *Scratch*. Além disso, percebeu-se que essa abordagem pode contribuir para o desenvolvimento de competências e habilidades necessárias à formação do estudante desta etapa escolar, desde que sua consolidação se dê em etapas anteriores (SILVA, 2016, p.9).

Acredita-se, dessa forma, que utilizando a ferramenta *Scratch* para a construção do algoritmo não é necessário digitar os pseudocódigos e, sim, apenas arrastar os blocos. Conforme se evidencia na figura 34.

Figura 34 - Resolução da questão 01, utilizando a ferramenta Scratch – acadêmico I.



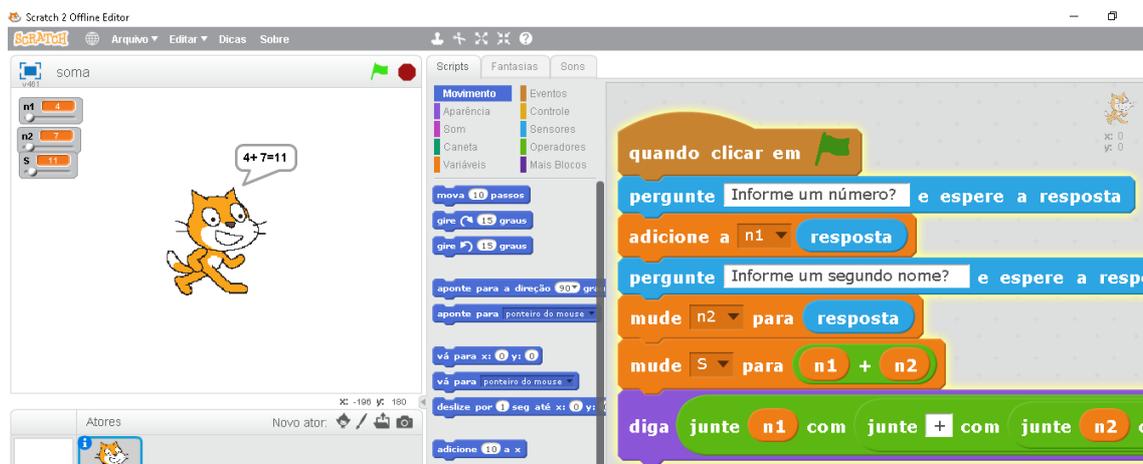
Fonte: A pesquisa.

5.2.2. Análise dos resultados da Questão 02

Na questão 02, solicitou-se o desenvolvimento de um algoritmo que somasse dois valores e apresentasse o resultado ao final do algoritmo. A análise demonstrou que 09 acadêmicos acertaram e 02 entregaram em branco a questão. Verificou-se que os acadêmicos que entregaram a questão em branco, foram H e K. Para que os acadêmicos atingissem plenamente todas as etapas, seria necessário que apresentassem os elementos presentes na Figura 36, na resolução da atividade realizada pelo acadêmico B, usando a ferramenta *Scratch*. Segundo Resnick (2013, p.24):

O programa *Scratch* propicia que seus usuários aprendam conceitos matemáticos e computacionais importantes, desenvolvam uma forma sistemática e criativa de pensar. Além disso, provoca o trabalho colaborativo, ou seja, potencializa as habilidades essenciais requeridas no século XXI.

Figura 35 - Resolução da questão 02, utilizando a ferramenta Scratch – acadêmico B.



Fonte: A pesquisa.

5.2.3. Análise dos resultados da Questão 03

Nessa questão, solicitou-se o desenvolvimento de um algoritmo que resolvesse o seguinte problema: calcular quantos anos uma pessoa está completando com base em sua data de nascimento e a data atual. O cerne da atividade é um problema matemático de subtração. Nessa questão, 02 acadêmicos erraram e 09 acertaram, respectivamente, a questão 3. Ao se tabular os dados, verificou-se que os acadêmicos que erraram esta questão foram os denominados de H e K, que entregaram a questão incompleta. Para que os acadêmicos atingissem plenamente todas as etapas, seria necessário que apresentassem no mínimo os elementos presentes na figura 37, usando o *Scratch*. A aprendizagem no *Scratch* pode ocorrer por meio do desenvolvimento da criatividade,

da sistematização do pensamento e do trabalho colaborativo por meio de uma linguagem de programação visual, com o agrupamento de blocos lógicos e manuseio de mídias de som e imagem, para a produção de histórias interativas, jogos e animações, permitindo o compartilhamento das criações de maneira online. O software possibilita trabalhar conceitos específicos de programação, como por exemplo, sincronia, interação, variáveis, execução paralela, lógica booleana, números randômicos etc. Além disso, auxilia na construção de conceitos matemáticos por meio da construção de figuras geométricas, manipulação de coordenadas cartesianas, realização de operações matemáticas, movimentação de objetos, utilização de operações lógicas por meio de condicionais e laços de repetição, entre outros (SILVA, 2016, p.21).

A partir da análise dos resultados da intervenção aplicada com alunos, segundo Pinto (2010) há alguma,

evidência de um maior empenho dos alunos quando resolvem problemas com o auxílio do Scratch. Afigura-se, deste modo, que o Scratch se constituiu como recurso adequado à resolução de problemas, uma vez que permitiu que os alunos tentassem procedimentos alternativos quando sentiam dificuldades. Este estudo evidencia as potencialidades do Scratch, defendendo que este tem algum potencial pedagógico e que poderá aumentar o interesse e a qualidade das aprendizagens efectuadas na área da Matemática (PINTO, 2010, p. iv).

Figura 36 - Resolução da questão 03, utilizando a ferramenta Scratch – acadêmico E.



Fonte: A pesquisa.

5.2.4. Análise dos resultados da Questão 04

Na questão 04, solicitou-se o desenvolvimento de um algoritmo que resolvesse o seguinte problema, a conversão de reais em dólares sabendo que U\$\$1,00 (um dólar), custaria R\$ 2,22, então se tenho R\$ 428 reais, quantos dólares terei?

Nessa questão, 04 acadêmicos erraram e 07 acertaram, verificou-se que 04 acadêmicos que erraram esta questão. Foram os acadêmicos B e C que a entregaram incompleta. Os acadêmicos H e K entregaram esta atividade em branco. Para que os acadêmicos atingissem plenamente todas as etapas, seria necessário que apresentassem os elementos presentes na figura 38, usando a ferramenta *Scratch*.

A utilização da ferramenta *Scratch* de forma livre desenvolve habilidades matemáticas, uma vez que:

[...] ainda trazem que todas as culturas utilizam habilidades para representar números e por si desenvolvem habilidades matemáticas, o que pode ser potencializado com a utilização de tecnologias. Porém, pensamos que o Scratch demonstra maiores possibilidades quando utilizado de forma livre pelos alunos, não sendo um trabalho pronto desenvolvido pelo professor [...] (SÁPIRAS, 2017, p.13).

Figura 37 - Resolução da questão 04, utilizando a ferramenta Scratch – acadêmico A.



Fonte: A pesquisa.

5.2.5. Análise dos resultados da Questão 05

Nessa questão, solicitou-se o desenvolvimento de um algoritmo que resolvesse o seguinte problema: a conversão entre graus *Celsius* e *Fahrenheit*, apresentando a seguinte fórmula para conversão: $C = F - 32 / 1.8$, que representa que os graus *Celsius* (C) equivalem a escala de graus *Fahrenheit* (F), seus pontos de referência são o ponto de congelamento da água em 32°F, e o ponto de ebulição da água, em 212°F.

Verificou-se que os 04 acadêmicos, que erraram essa questão, foram os acadêmicos H e K que entregaram a atividade incompleta e C e D que entregaram em branco. Para que os acadêmicos atingissem plenamente todas as etapas, seria necessário que apresentassem no mínimo os elementos presentes na figura 39, usando a ferramenta *Scratch*.

O relato de Sápiras, (2017), aponta que,

o Scratch é um sistema tecnológico que vai além das estimas de sua utilização, quando apresenta que: “Esse software possibilita ao usuário o desenvolvimento de programas e jogos eletrônicos. Entretanto, nosso olhar está nas potencialidades frente aos processos de ensino e aprendizagem da Matemática e sua relação com linguagens de programação”. (SÁPIRAS, 2017, p.14).

Figura 38 - Resolução da questão 05, utilizando a ferramenta Scratch – acadêmico F.



Fonte: A pesquisa.

5.2.6. Análise dos resultados da Questão 06

Nessa questão, solicitou-se que se desenvolvesse um algoritmo para calcular 60 por cento de imposto, sabendo que o total é \$4.280.00 dólares. Nessa questão, 05 acadêmicos erraram e 06 acertaram. Ao se tabular os dados, verificou-se que 05 acadêmicos erraram esta questão, foram o acadêmico B, D, G, H e K que entregaram a atividade incompleta. Para que os acadêmicos atingissem plenamente todas as etapas, seria necessário que apresentassem os elementos presentes na figura 40, usando a ferramenta *Scratch*. Durante a realização dessa pesquisa se evidencia uma evolução dos acadêmicos devido à utilização da ferramenta *Scratch*, pois nessa questão houve apenas dois acertos sem o uso da ferramenta *Scratch*, evoluindo para 06 acertos após sua implantação.

O que também se evidencia na fala de Sápiras (2017, p.44):

O Scratch se utiliza de uma interface gráfica que permite que programas sejam desenvolvidos como blocos de encaixar, lembrando o brinquedo Lego. Esse software não exige o conhecimento inicial referente à programação, por isso pode ser utilizado no aprendizado de conceitos matemáticos e computacionais. Cada bloco de encaixar contém um comando diferente que pode ser unido livremente, com o objetivo de trabalhar com diferentes mídias como som e imagens. Cabe destacarmos também que ele oferece a opção Português, facilitando a sua utilização por parte do aluno. Devido a todas essas características, o Scratch se constitui como uma linguagem de programação visual e permite ao usuário construir interativamente suas próprias histórias, animações, jogos, simuladores, ambientes visuais de aprendizagem, músicas e arte, em que toda ação de qualquer objeto deve ser programada e explicitada. Os blocos que são arrastados para uma área específica são conectados, formando a programação do ambiente.

Figura 39 - Resolução da questão 06, utilizando a ferramenta Scratch – Acadêmico C.



Fonte: A pesquisa.

5.2.7. Análise dos resultados da Questão 07

Nessa questão, solicitou-se o desenvolvimento de um algoritmo que resolvesse o seguinte problema: calcular quanto um cliente de um banco pagará, após 10 meses, tendo tomado emprestado a quantia de R\$ 5.000,00 reais, com taxas de 2% ao mês. Para o desenvolvimento do algoritmo foram elaboradas as atividades, que demandavam trabalhar juros. Houve 5 erros e 6 acertos, verificou-se que 6 acadêmicos erraram a questão, foram os acadêmicos B, C, D, G, H e K, que entregaram a atividade incompleta. Para que os acadêmicos atingissem plenamente

todas as etapas, seria necessário que apresentassem, os elementos, presentes na Figura 41, que mostra o resultado da atividade realizada.

Resnick (2013) diz que, o uso de tecnologia computacional como a ferramenta *Scratch*, além de poder contribuir para a construção de ideias computacionais, pode ser importante para a elaboração de estratégias de resolução de problemas, organização de projetos e comunicação de ideias.

Figura 40 - Resolução da questão 07, utilizando a ferramenta *Scratch* – acadêmico A.



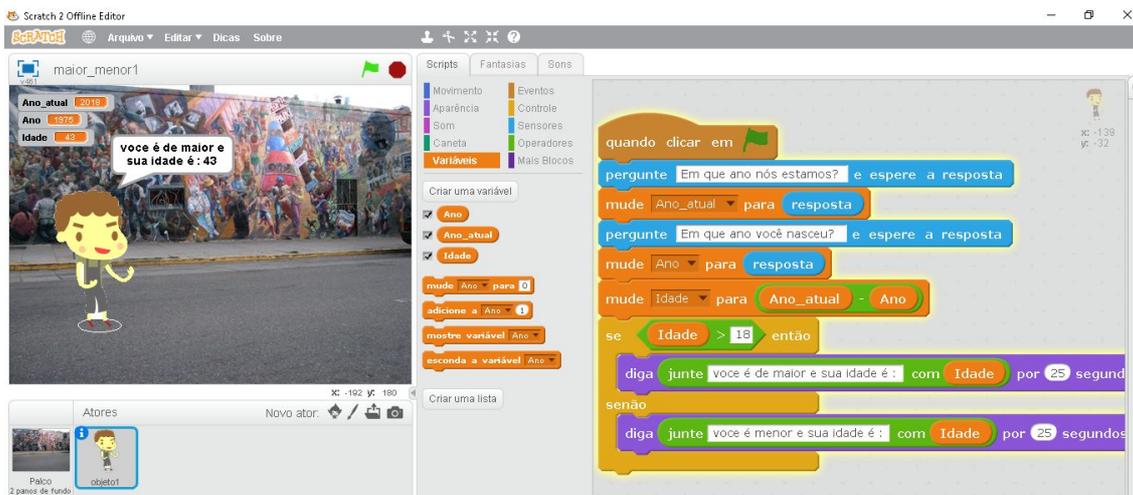
Fonte: A pesquisa.

5.2.8. Análise dos resultados da Questão 08

Nessa questão, solicitou-se o desenvolvimento de um algoritmo para informar a data de nascimento, a data atual, e calcular a idade e, depois, que fosse feito um comparativo, se o usuário é maior ou menor de idade.

Verificou-se que os 05 acadêmicos que erraram a questão e os 06 que acertaram, foram os acadêmicos B, C, G, H e K, que entregaram a atividade incompleta. Para que os acadêmicos atingissem plenamente todas as etapas desta questão, seria necessário que apresentassem, os elementos presentes na resolução da atividade, como mostra a Figura 42.

Figura 41 - Resolução da questão 08, utilizando a ferramenta Scratch – acadêmico D.



Fonte: A pesquisa.

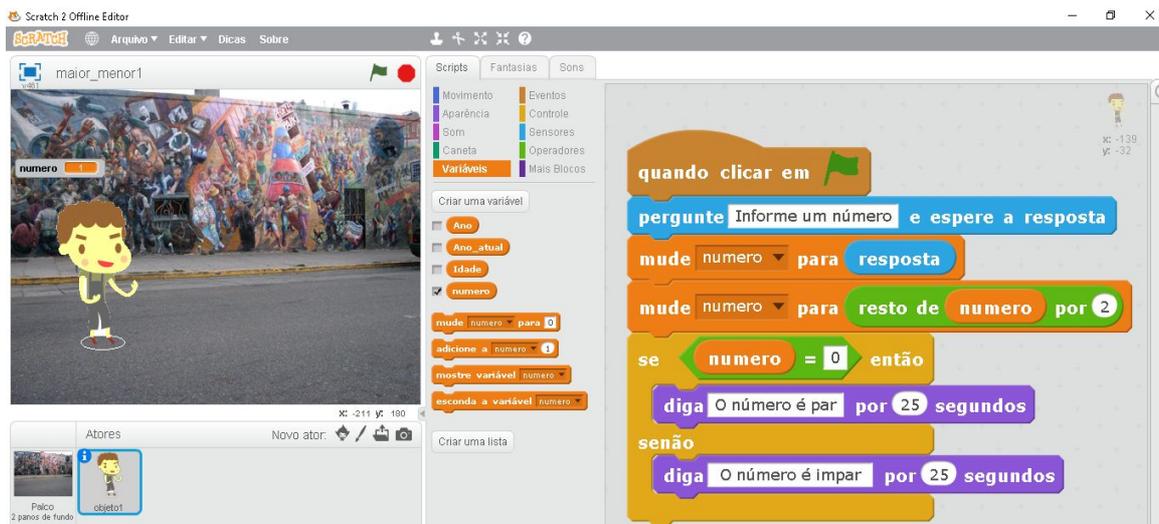
5.2.9. Análise dos resultados da Questão 09

Na questão 09, solicitou-se que os acadêmicos desenvolvessem um algoritmo que perguntasse um número, e respondesse se o mesmo é par ou ímpar. Verificou-se que os 03 acadêmicos que erraram que foram os acadêmicos G, H, e K entregaram a atividade incompleta, 08 acadêmicos acertaram. Para que os acadêmicos atingissem plenamente todas as etapas, seria necessário que apresentassem, os elementos presentes na Figura 43, que apresenta a solução da atividade realizada, usando a tecnologia computacional *Scratch*.

Nesse sentido, apresenta-se um dos motivos de se utilizar a ferramenta *Scratch*.

Esta tecnologia computacional possui uma sintaxe mais intuitiva que linguagem LOGO, e as outras linguagens de programação (como Pascal, Python, Fortran, Cobol, e C++), seus blocos de comandos são visíveis e possuem maior diversidade de comandos prontos, representados por blocos, os quais facilitam a produção de estórias multimídias interativas ou qualquer outro tipo de programação (VENTORINI, 2015, p.34).

Figura 42 - Resolução da questão 09, utilizando Scratch – acadêmico E.

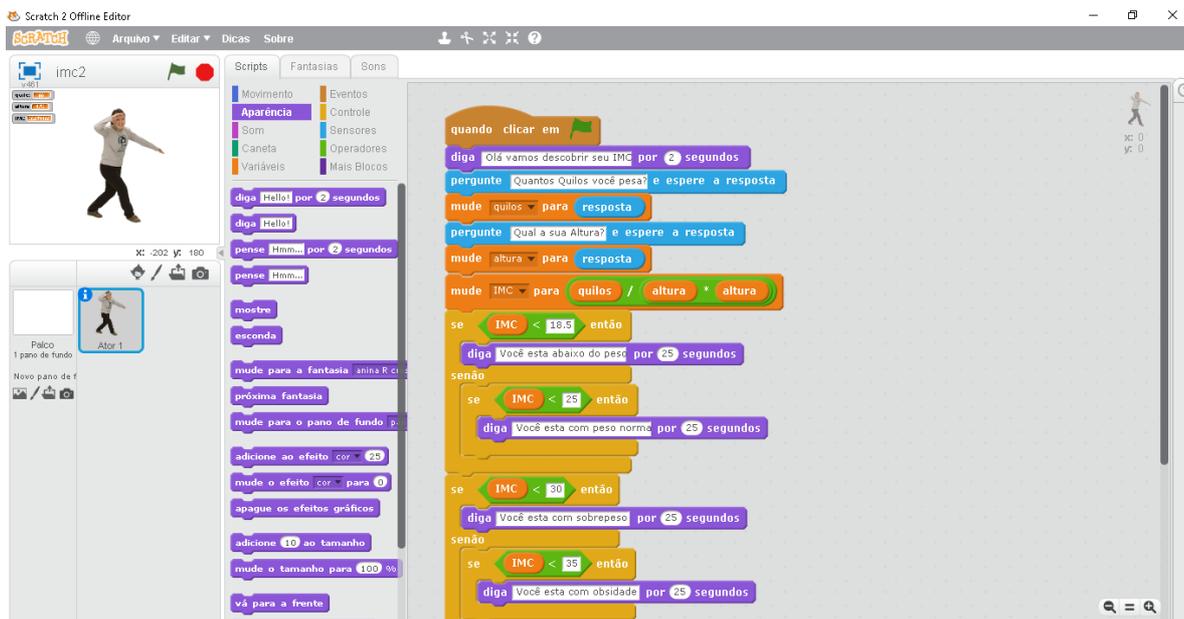


Fonte: A pesquisa.

5.2.10. Análise dos resultados da Questão 10

Nessa questão, solicitou-se que os acadêmicos desenvolvessem um algoritmo que perguntasse, qual sua altura e seu peso, seguindo a fórmula $IMC = \text{massa} / \text{altura}^2$. Para a resolução da atividade era preciso utilizar as informações presentes na figura 27. Verificou-se que 08 acadêmicos erraram e 03 acertaram. Ao se tabular os dados, verificou-se que os 08 acadêmicos que erraram a questão foram os acadêmicos A, B, C, F, G, H, J e K, que a entregaram de forma incompleta. Para que os acadêmicos atingissem plenamente todas as etapas, seria necessário que apresentassem, os elementos presentes na Figura 44, que apresenta o resultado da atividade.

Figura 43 - Resolução da questão 10, utilizando a ferramenta *Scratch* – acadêmico I.



Fonte: A pesquisa.

5.3. REFLEXÕES SOBRE A ENTREVISTA COM OS ALUNOS

No cotidiano acadêmico é bastante comum comentários como, “esse *aluno não tem os pré-requisitos ou ainda faltam conhecimentos básicos*”, quando um professor se defronta com isso, faz-se necessário investigar para conhecer. A busca pelo conhecimento envolve conhecer os alunos e encontrar uma proposta de intervenção que os conduza a superação. Após o término das atividades com a ferramenta *Scratch*, foi realizada uma entrevista semiestruturada com os alunos. Sobre a resolução de problemas que exigiam raciocínio lógico e o uso do algoritmo para resolver um problema de matemática de forma prática, destaca-se algumas das falas dos participantes da pesquisa:

Eu gosto bastante de resolver este tipo de questão, de achar a melhor solução para um problema, pois inclusive caso ele se repita eu já tenho a melhor solução. (Acadêmico A)

O meu raciocínio mudou parece que as coisas ficam mais claras, fica mais fácil. (Acadêmico D)

Sim, me ajuda bastante no raciocínio, em matemática. Eu gosto de resolver problemas da internet do estilo do sinótica no estilo, racha cuca, para mim é legal. Pois facilita bastante na parte do raciocínio, o exemplo maior é que consegue botar tudo na prática de modo mais fácil. Pois como eu te falei é

um, o maior benefício do Scratch é que quando mais a gente vai brincando com ele mais ele vai auxiliando no raciocínio. (Acadêmico E)

Sim bastante, é um negócio que quando tu terminas a sensação de dever cumprido é muito boa. Sim, no trabalho ali com algoritmos, consigo ver que melhorou bastante meu raciocínio, no meu trabalho eu já consigo aplicar isso, em coisas que eu teria muita dificuldade antes, um exemplo é na manipulação do excel, e em outras coisas. " (Acadêmico F)

Sim, bastante, não sei talvez, toda vida, pois começa ali no jogo da dama, mas nunca foi assim voltado no computador, pois eu sempre gostei de jogos que desafiassem como dama, xadrez, essas coisas assim é como eu visualizo os números e o raciocínio, ou seja eu estou gostando. (Acadêmico G)

Sim eu achei bem prático, na verdade eu tive que usar a lógica praticamente, pois tinha alguns problemas para resolver precisava usar a lógica, deixa eu tentar lembrar para dar um exemplo, a sim, acho que um bom exemplo é o exercício de conversão de dólar, onde a gente tinha que usar fórmulas matemáticas e a lógica e isso não é muito difícil, e assim ficava pronto para construir o algoritmo. Também foi muito tranquilo, pois os algoritmos vão aparecer para o usuário, ou seja, como eu já disse, prático. Gosto bastante, digamos por exemplo, antigamente eu não gostava, pois eu não entendia, depois eu comecei a gostar, um bom exemplo foi que comecei a usar o cubo mágico, e no começo eu usava somente como cola para resolver, mas depois eu comecei a raciocinar com funcionava aquilo e comecei a gostar de quebra cabeça e coisas que exigem mais raciocínio lógico e problemas difíceis de resolver, pois para mim, comecei a encarar com um jogo, pois no jogo tu tem dificuldades para passar, pois para mim a partir daí comecei a encarar as atividades de lógica com um jogo, pois em um jogo tem que chegar em um objetivo e tem dificuldades para passar, tem que pensar de certa forma, e para mim problemas lógicos são a mesma coisa". (Acadêmico H)

(...) matemática é o seguinte ela não é igual as outras áreas, biologia, história que nem, pois nestas áreas podem ter várias respostas para mesma pergunta e na matemática é só uma. Ou não tem né então por isso que gosto, pois aí é certo ou errado não tem enrolação. (...) é muito mais fácil do que fazer o cálculo inteiro usando as fórmulas, colocando a fórmula ele te dá o resultado inteiro ali né". (Acadêmico I)

Ajudou na verdade, ajudou bastante, pois é bem simples e foi bem simplificado ao invés de ficar pensando a maioria das vezes já estava lá as coisas mais simplificadas na tela para procurar no caso e aí com isso tu já te achava mais e aí já consegui resolver o problema. Eu gosto de resolver problemas que exigem raciocínio lógico, assim às vezes dá um pouco de dor de cabeça, mas é isso aí que se trata para mim e é exatamente por isso pois porque tu, tipo quanto pega o que é difícil e transporta e consegue resolver as coisas que são de um nível menor e assim fica mais fácil e prática é por isso os motivos que eu mais gosto. (Acadêmico J)

Depende, assim eu gosto de resolver problemas, mas se eu estiver muito cansado eu saio fora, mas se eu estou de boa assim eu curto bastante sim. Quando eu comecei a fazer o curso eu já tinha terminado o curso técnico e lá eu tive a cadeira de matemática aplicada a informática e isso me ajudou bastante, também já tinha feito uma cadeira de lógica e isso me deu um bom raciocínio lógico matemático, e então a gente já fica treinado com aquilo ali tem um raciocínio prático para matemática e isso faz toda a diferença, ou seja, ajuda bastante sim. (Acadêmico K).

Em relação a resolução de problemas, que demanda raciocínio lógico, e o uso do algoritmo, cabe destacar que

O computador oferece o algoritmo (a receita), não o pensamento. O aprendiz precisa pensar em como usar o algoritmo. Os alunos acham que o importante é chegar ao resultado certo, mas não é isso. (...) Hoje há problemas que requerem outras qualidades, como intuição. Se você entende os princípios, pode usar o computador para fazer o trabalho por você. Mas se não entende, não terá chance. (TALL, 2002, p.10)

Em relação a ferramenta *Scratch* como uma facilitadora do processo de aprendizagem e fonte de interesse pela programação, destaca-se os seguintes registros:

Por ele ser bem visual, eu acho que ele melhora e facilita o aprendizado né e até por ele ter temas imagens, animações, gera um interesse para quem tem uma certa dificuldade e ou temor por programação em um determinado momento. Eu acho que a parte mais fácil é realmente por ele ser visual, ou seja, ter uma lista de comandos que eu posso utilizar, mesmo que eu não saiba, alguns são intuitivos né, e poder ir simplesmente encaixando eles ali, facilita bastante e por isso eu não vi grande dificuldade, se existe alguma deve ser em seu eu fosse totalmente leigo né, talvez eu não entendesse alguns comandos, mas para quem já tem alguma noção de lógica é bem tranquilo. (Acadêmico A)

Eu gosto, gosto de resolver. Sim, ele é um uso mais fácil é mais objetivo né, pois o algoritmo que a gente faz ele tem que ser todo digitado. E no Scratch, ele é mais objetivo, pois basta pegar o comando pronto montar os blocos e ele já funciona né. (Acadêmico C)

Não, eu já conhecia antes, não utilizei muito, mas pelo que eu sabia era usado somente com crianças, mas achei interessante. Tudo que é novo, talvez deve ter uma complicação no início, mas ele é bem simples bem didático. Na sala de aula facilita bastante, tanto na sala de aula como fora, pois, auxilia inclusive na comunicação. (Acadêmico D)

Sim, porque é uma maneira mais didática que o que a gente tem trabalhado ultimamente, por ser mais visual mais interativo, isso facilita e chama mais a atenção". A facilidade foi encontrar os comandos muito mais fácil, pois eles já estão na tela. Dificuldade talvez a colocação dos blocos em ordem na tela, mas isso foi em função de não conhecer o programa ainda. (Acadêmico F)

Não entendi muito bem a pergunta, perai deixa eu pensar um pouco, com aquela aula eu achei interessante assim, o que eu entendi ali foi, gostei tá, porque eu não tinha visto ainda na prática com aquele cálculo como eu poderia utilizar ele, como o bonequinho se mexendo aquilo eu ainda não tinha conseguido visualizar que um cálculo, pois um cálculo gerar uma movimentação e gerar algum resultado e isso eu não tinha visto, neste caso sim. (...) Facilitou na forma de conseguir visualizar o resultado, mas assim eu não o achei muito fácil. Não é fácil, mas assim ele facilitaria de uma forma que não te deixa pensar muito, pois ao encaixar bloco, não me deixaria raciocinar muito, pois facilita, mas para mim eu preferiria da maneira tradicional. Eu não achei difícil, mas achei pouco tempo, mas não achei difícil. (Acadêmico G)

Eu nunca tinha usado o Scratch antes, foi a primeira vez que usei agora, já tinha ouvido falar que era educacional para trabalhar com um público bem

leigo, mas eu vi que ajuda bastante, embora eu já conheça bastante “JAVA” e essas coisas eu achei bem tranquilo, mas percebo que se eu não soubesse nada de programação, seria realmente uma boa alternativa para aprender. (Acadêmico H)

(...) ele facilitou muito, né ele é bem mais fácil né ele já tem os blocos prontos é só tu saber usar ele nas posições corretas né, e seguir. (Acadêmico I)

Facilitou e auxiliou bastante lá com os barulhinhos e toda interação que ele podia fazer tanto com as imagens com a programação em blocos. Ajudou bastante pois estimulou bastante pelas imagens pelos sons e por tudo que envolve o aplicativo, o aplicativo em si e te mostra os diversos tipos de interações que tu pode ter com a programação com o dia a dia com diversas situações ali que tu pode envolver tipo jogos para perguntas frequentes e diversas e que tu pode fazer perguntas e respostas automáticas como se fosse um NPC¹⁵ no caso, mas na verdade tu pode viver e é só uma simulação né onde o usuário pode usar o Scratch. (Acadêmico J)

Eu gostei muito, mas como eu já conheço bem programação eu achei um pouco cansativo, mas acho que para quem não conhece e está iniciando ele é bem intuitivo assim, pois tem vários aspectos que facilitam desenvolver a lógica assim. (Acadêmico K)

Ele é intuitivo, mas ainda tem que procurar conhecer certinho as coisas pois ele é intuitivo, mas até aprender, e logo que tu pega bem a manha dele aí vai tranquilo, para quem não conhece a lógica e facilita e fica muito fácil e sei lá achei ele bem prático. (Acadêmico K)

Sápiras (2017) indica que devido a suas características a ferramenta *Scratch* se constitui em uma tecnologia computacional que permite ao usuário construir, de forma interativa, suas histórias, jogos, simuladores, animações, sendo que toda a ação de qualquer tipo de objeto requer programação. Dessa forma, os blocos que são arrastados para uma determinada área são conectados, formando a programação do ambiente.

Sobre a utilização de tecnologias digitais em sala de aula, destacam-se as seguintes impressões:

Olha eu acho bem importante hoje em dia, pois em tudo que se faz no mundo trabalho envolve em algum momento tecnologias digitais, eu acho que a escola e o ensino devem aproveitar bastante isso pois fora da sala vai se usar bastante. Eu gostei bastante da pesquisa, eu acho que deu um outro momento e deu uma outra visão para a gente que estava com um conteúdo específico. (Acadêmico A)

Muito grande, pois tu tens todo o conhecimento do mundo nas palmas das tuas mãos ali. ”, perguntado sobre, qual é a opinião sobre esta pesquisa? É

15Um personagem não jogável (em inglês: *non-player character* ou NPC) é um [personagem](#) de qualquer [jogo eletrônico](#) que não pode ser controlado por um jogador, mas se envolve de alguma forma no enredo de um jogo.

bacana saber o andamento e talvez melhor em algum ponto é sempre bom ter um indicador. (Acadêmico F)

Na utilização de tecnologias digitais em sala de aula, começa pela parte financeira que tu tens um custo muito menor, pois pode usar tudo no computador e ou transferir para um pendrive, e é por isso que vejo benefícios nisso. (Acadêmico H)

É bom né tem que ter tecnologia digital para a gente ter mais experiência né, pois não adianta a gente fazer os algoritmos no papel, claro que é bom também, mas o teste de mesa no computador é outra coisa. (Acadêmico I)

Pois olha tecnologias digitais na sala de aula, facilita bastante pois eu acho que se todos os alunos lidarem com os problemas em comum usando programas que tu encontras quando tu tá aprendendo, né. Acho que é uma boa ideia para estimular o pessoal já que estão no ensino superior para eles voltarem a ver quão alegre e contente a gente pode ser programando e resolvendo os problemas e mostrar que a programação não é um bicho de 7 cabeças, pois para uns até pode ser às vezes, mas é só tu correr atrás que as coisas ficam simples. (Acadêmico J)

Tem os lados benéficos e se for mal-usada pode atrapalhar a aula, pois muito alunos não têm noção da dosagem. Mas bem utilizada tem uma grande importância". Perguntado sobre qual é a opinião sobre esta pesquisa? " Curti a ideia, achei bem da hora. (Acadêmico K)

Para Silva (2016), usando a ferramenta *Scratch*, os alunos reconheceram falta de dedicação em aprender da maneira tradicional, porém enquanto os alunos que declararam ter muito bom rendimento nas questões evidenciaram o sucesso, a autora apresenta que:

[...] um dos pontos analisados por esta investigação pretendia verificar se um ensino mediado pelo uso de tecnologias digitais proporcionaria aulas mais dinâmicas e motivadoras à aprendizagem, observou-se que os obstáculos encontrados nos processos de ensino-aprendizagem nesta era digital, parecem ter muito mais haver com o nível de interesse do aluno, do que com a capacidade dele em aprender o que está sendo proposto (SILVA, 2016, p. 101).

Sobre dificuldades e/ou facilidades durante o pré-teste e o pós-teste:

A gente saiu daquele conteúdo ainda trabalhando com algoritmos, mas a gente teve uma outra visão de outra forma de fazer a mesma coisa. (Acadêmico A)

Não tive dificuldade alguma em usar o Scratch, pois foi bem fácil e o que me facilitou mais é que ele é muito objetivo né, porque no caso né os comandos já vêm prontos, mas tem que saber usar eles no caso. (Acadêmico C)

A facilidade foi o computador, pois resolver os algoritmos no computador com o Scratch eu acho que é bem mais fácil assim. (Acadêmico G)

Gostei de usar o Scratch pois ele é uma ferramenta muito, autoexplicativa, muito prática para trabalhar com programação em bloco, ou

seja, uma ferramenta por si só fácil, tive uma dificuldade mínima para entender as escolhas dos blocos, mas quando entendi, tudo ficou certo. Vi que o Scratch é uma ferramenta muito boa, pois se tiver algum amigo que não conheça programação posso mostrar para que ele possa dar os primeiros passos na lógica. (Acadêmico H)

Usando o Scratch é mais fácil, pois aparentemente ele deixa as coisas mais fácil, do que tu ter que parar para pensar e ver como tu vai fazer né, pois ali no Scratch tem os blocos e é só seguir, e ele já coloca né. (Acadêmico I)

A maior dificuldade foi lembrar de algumas expressões lógicas que não estava lembrando, mas depois que lembrei de tudo ficou fácil novamente não teve assim uma complicação e o que facilitou bastante foi a formulação dos exercícios aplicados pois os primeiros estavam bem fáceis e iam refrescando e te envolvendo mais abrindo a memória para a realização dos algoritmos de programação no caso. (Acadêmico J)

Foi bem bacana, foi uma grande ideia assim. A dificuldade não achei muitas assim, mas geralmente assim é acho que quando dá um branco na hora de fazer a questão e a gente é porque a gente não interpretou bem direito, o que as questões estavam pedindo, mas as questões estavam bem equilibradas assim. (Acadêmico K)

Para Petry (2005), a dificuldade de entendimento na utilização da lógica de programação é um problema que se repete nas disciplinas de algoritmos nos cursos de graduação nas áreas de computação. Essas dificuldades registradas pelos acadêmicos são as maiores causas no processo de reprovação e do abandono dos cursos, em faculdades privadas ou públicas. Quando não identificadas e se não for proposta uma solução a tempo essas deficiências trazem imensos prejuízos aos alunos, professores das instituições e para o currículo do curso, atrasando a formação e, muitas vezes, tornando o nível de atividades práticas e intelectuais, incompatíveis com o que será proposto ao longo de sua formação acadêmica.

Sobre a participação na pesquisa:

Eu gostei bastante da pesquisa, eu acho que deu um outro momento e deu uma outra visão para a gente que estava com um conteúdo específico. (Acadêmico A)

Não sei pelo que a gente fez, foi uma coisa básica que a gente está aprendendo e exercitando nesta cadeira, achei meio difícil, embora fosse uma atividade básica para ver o que a gente está aprendendo sobre algoritmo. (Acadêmico C)

Como vou te explicar, se tenho dúvida na sala de aula posso usar as tecnologias aplicadas para fazer contato com o professor e tirar a dúvida, foi bem interessante, não tinham feito isso antes ainda e então foi bom. Eu acho que teve, teve sim, até porque eu não lembrava de algumas coisas, pois foi a mesma prova e ficou bem mais claro na segunda vez. (Acadêmico D)

Uma pesquisa válida, pois para quem está começando é essencial para quem não sabe nada, pois ele estimula, a gostar de programação ou lógica. Pois tive dificuldades na primeira prova e foi mais no entender, eu tive em entender o contexto, da palavra e o também eu estou com medo do tempo, e, pois, na segunda, foi bem mais tranquilo. (Acadêmico E)

Na primeira parte não foi muito tranquilo, mas na segunda, foi bem melhor, mas uma pena que não deu tempo de terminar tudo. (Acadêmico F)

Eu gostei, pois foi meio surpresa, eu achei um desafio ali, eu e a maioria não conseguimos fazer todos, mas pelo menos nos tentamos e foi bem desafiador para os primeiros dias de aula com base no que o professor já tinha trabalhado com a gente. (Acadêmico G)

Eu achei uma pesquisa muito interessante, pois como já disse eu nunca tinha usado o Scratch e agora eu vi como ele funciona e como eu disse foi difícil de entender no início, mas depois que eu entendi foi bem fácil. (Acadêmico H)

Pois é algo muito legal e é por isso que a programação se torna divertido e é aonde eu quero mostrar que a programação é algo para se gostar, pois com o Scratch me trouxe essa lembrança de que a programação pode ser algo muito legal. (Acadêmico J)

Um dado evidenciado ao final da pesquisa é que ao se verificar o novo mapa da evasão no ano de 2018, especificamente no período da aplicação desta pesquisa, houve uma variação no percentual de alunos evadidos, apresentando o menor índice do período entre 2014 a 2018. Conforme se verifica na figura 45, que apresenta uma oscilação entre 40% e 80% no índice. No período da aplicação da pesquisa que foi entre 13 de agosto de 2018 ao dia 21 de dezembro de 2018, registrou-se que entre os 20 alunos matriculados na disciplina, 02 trancaram a matrícula, 02 cancelaram e 04 reprovaram na disciplina, constituindo um índice de 40% do total dos inscritos, constituindo o menor índice no período evidenciado.

6. CONSIDERAÇÕES FINAIS

Para responder ao problema de pesquisa “Como a ferramenta *Scratch* pode auxiliar na compreensão de conceitos abordados na disciplina de algoritmos e Programação”, atendendo aos objetivos propostos na mesma, o estudo de caso contou com um cenário de estudos, definição da metodologia, aporte teórico, instrumentos, análise de pesquisa e apresentação dos resultados. A ferramenta *Scratch* também se mostrou uma tecnologia computacional que aliada ao pensamento computacional, serviu, para auxiliar a resolução dos algoritmos propostos, pois o objeto desta pesquisa foi a aplicação da ferramenta *Scratch* para a superação de dificuldades na compreensão de conceitos matemáticos, bem como na construção de algoritmos na disciplina de algoritmos de programação.

Além disso, o trabalho possibilitou uma reflexão sobre o uso de TDICs aliadas a um referencial pedagógico que considere a diversidade de pensamento e conhecimento trazidos pelos acadêmicos e que alinhado com o pensamento computacional promova um melhor desempenho por parte dos alunos na construção e desenvolvimento de algoritmos. Com o intuito de compreender melhor o uso da ferramenta *Scratch*. Buscou-se conhecer e assim mediar os diferentes níveis de conhecimento apresentado pelos alunos pesquisados. Destaca-se que durante o desenvolvimento deste estudo, houve a necessidade de entendimento de alguns pontos, como perceber que os alunos que ingressam no ensino superior em áreas mais técnicas têm um desempenho deficitário ao enfrentar disciplinas que demandam a utilização de cálculos matemáticos, dificuldade na compreensão de conceitos matemáticos, estudar algoritmos e, depois trabalhar esses mesmos conteúdos em programação. Entende-se que, cada etapa realizada durante a construção deste trabalho, permitiu outro olhar sobre o uso de uma tecnologia computacional diferente das usuais, além de contribuir para a construção de conceitos citados anteriormente. Outra dificuldade apresentada foi a compreensão da lógica de programação e da organização da escrita da linguagem de programação.

O aprendizado que resulta após o desenvolvimento deste estudo é de que as TDICs, além de representarem um diferencial, nos processos de ensino e aprendizagem representam um universo a ser explorado e expandido. É possível utilizar diferentes tecnologias e o pensamento computacional para auxiliar os alunos e os professores a se apropriarem, aprofundarem e ampliarem os conceitos de cada disciplina. O presente estudo não se esgota aqui, ao contrário, é uma porta para a efetivação da interdisciplinaridade construída a partir da junção de tecnologias computacionais e do pensamento computacional voltada para uma linha pedagógica que contemple as necessidades e dificuldades de um sujeito imerso em um universo tecnológico e que ainda assim precisa apresentar um melhor desempenho acadêmico em disciplinas essenciais que envolvem lógica e matemática. A partir dos resultados obtidos, vislumbrou-se a possibilidade de se utilizar o pensamento computacional e a ferramenta *Scratch* para auxiliarem alunos e professores de áreas técnicas em outro grande desafio educacional, que é a possibilidade de aplicar a ferramenta *Scratch* em disciplinas como, organização de computadores, em que poderia utilizar a ferramenta *Scratch* juntamente com Arduíno, com makey makey¹⁶. Essa ferramenta também poderia ser utilizada com outros periféricos físicos que poderiam auxiliar e facilitar o aprendizado em disciplinas como organização de computadores e sistemas operacionais, auxiliando a expressar em palavras, ideias e pensamentos de modo lógico, coeso e coerente. O universo do pensamento está em constante expansão e evolução, não é possível desenvolvê-lo apenas na área lógico-matemática, é preciso uni-la a outras tecnologias e ao pensamento computacional que estão disponíveis para o desenvolvimento de mais um passo significativo dentro do processo ensino-aprendizagem.

¹⁶Segundo Kenshima (2016) Makey Makey, é um dispositivo que você conecta na porta USB do seu computador, Raspberry Pi, televisor, etc. Através dele, você usa objetos comuns como dispositivo de entrada, sendo uma excelente opção de aprendizado de conceitos de eletrônica para adultos e crianças. O produto usa um tipo de comutação de alta frequência que detecta a conexão através de diversos materiais, e você pode utilizá-lo como um teclado ou mouse. Transformando praticamente tudo em um sensor touch.

REFERÊNCIAS

- ALVES, V. S. **A construção do conceito de número racional no sexto ano do ensino fundamental**. 2012. 185 f. Dissertação (Mestrado) - Curso de Mestrado em Ensino de Ciências e Matemática, Universidade Federal de Alagoas, Maceió, 2012. Disponível em: <<http://www.repositorio.ufal.br/bitstream/riufal/1217/1/A%20constru%C3%A7%C3%A3o%20do%20conceito%20de%20n%C3%BAmero%20racional%20no%20sexto%20ano%20do%20ensino%20fundamental.pdf>>. Acesso em: 25 fev. 2019.
- ALVES, G. F. O. **Lógica de programação para iniciantes**. 2018. Disponível em: <<https://mclp.dicasdeprogramacao.com.br/>>. Acesso em: 27 nov. 2018.
- BAIÃO, E. R. **Desenvolvimento de uma metodologia para o uso do Scratch for arduíno no ensino médio**. 2016. 101 f. Dissertação (Mestrado) - Curso de Programa de Pós-graduação em Educação, Universidade Estadual de Campinas, Campinas, 2016. Disponível em: <http://repositorio.unicamp.br/bitstream/REPOSIP/305320/1/Baiao_EmersonRodrigo_M.pdf>. Acesso em: 17 out. 2018.
- BAPTISTA, J. A. S. **Programação com scratch: desenvolvendo raciocínio algorítmico**. 2017. 70 f. Dissertação (Mestrado) - Curso de Mestrado Profissional em Matemática em Rede Nacional, PROFMAT, IMPA, Rio de Janeiro, 2017. Disponível em: <https://impa.br/wp-content/uploads/2018/02/TCC_2017_Jo%C3%A3o-Alvaro-Baptista.pdf>. Acesso em: 29 jul. 2018.
- BAUER, M.; GASKELL, G. **Pesquisa qualitativa com texto: imagem e som: um manual prático**. Petrópolis, RJ: Vozes, 2002.
- BASTOS, J. A. **O cérebro e a matemática**. São José do Rio Preto: Edição do Autor, 2006.
- BELUCE, A. C. **Estratégias de ensino e de aprendizagem e motivação em ambientes virtuais de aprendizagem**. 2012. 129 f. Dissertação (Mestrado) - Curso de Mestrado em Educação, Universidade Estadual de Londrina – Uel, Londrina, 2012. Disponível em: <http://repositorio.utfpr.edu.br/jspui/bitstream/1/1393/1/LD_PPGEN_M_Martiniano%2C%20Eziquiel_2015.pdf>. Acesso em: 18 dez. 2018.
- BIAZUSSI, D. C. G. **Construindo um jogo de desafios lógicos por meio do software scratch**. Palotina: Seed, 2016. 45 p. (ISBN 978-85-8015-094-0). Disponível em: <http://www.diaadiaeducacao.pr.gov.br/portals/cadernospde/pdebusca/producoes_pd>

e/2016/2016_pdp_mat_unioeste_dianicristinagoergen.pdf>. Acesso em: 29 set. 2019.

BILABILA, A. M. **CompAlg – Ferramenta de ensino e aprendizagem da lógica de programação**. 108 f. Dissertação (Mestrado) - Curso de Mestrado em Ciência de Computadores, Universidade do Porto, Porto, 2017. Disponível em: <<https://repositorio-aberto.up.pt/bitstream/10216/111643/2/261923.pdf>>. Acesso em: 17 dez. 2018.

BLIKSTEIN, P. O pensamento computacional e a reinvenção do computador na educação. 2008. Disponível em: <<http://bit.ly/1IXIbNn>>. Acesso em: 22 de set. 2019.

BRASIL. M. E. **O que é uma comissão permanente de avaliação (CPA)?** 2018. Disponível em: <<http://portal.mec.gov.br/par/127-perguntas-frequentes-911936531/educacao-superior-399764090/13122-o-que-e-uma-comissao-permanente-de-avaliacao-cpa>>. Acesso em: 26 set. 2019.

BRASIL. M. E. **Parâmetros curriculares nacionais: ensino médio**. Brasília, 1998.

BRASIL. M. E. **Parâmetros curriculares nacionais: ensino médio**. Brasília, 1999.

BRASIL. M. E. **PCNS+ ensino médio: orientações educacionais complementares aos Parâmetros Curriculares Nacionais**. Brasília, 2003. Disponível em: <<http://portal.mec.gov.br/seb/arquivos/pdf/CienciasNatureza.pdf>>. Acesso em: 04 fev. 2019.

CARVALHO, V. A. **Lógica de programação**. 2010. Disponível em: <http://proedu.rnp.br/bitstream/handle/123456789/290/Logica_Programacao_COR_capa_ficha_20110127.pdf?sequence=1&isAllowed=y>. Acesso em: 27 nov. 2018.

CERVO, A. L. BERVIAN, P. A. **Metodologia científica**. 5.ed. São Paulo: Prentice Hall, 2002.

COPI, I. M. **Introdução à lógica**. 2. ed. São Paulo: Mestre Jou, 1978.

DELORS, J. **Educação: um tesouro a descobrir**. Brasília: MEC - Unesco, 2006.

DIAS, R. V. **O uso de porcentagem no cotidiano dos alunos**. 122 f. Dissertação (Mestrado) - Curso de Programa de Pós-graduação em Educação em Ciências e Matemática, Pontifícia Universidade Católica no Rio Grande do Sul, Porto Alegre, 2019. Disponível em:

<<http://repositorio.pucrs.br/dspace/bitstream/10923/3070/1/000400101-Texto%2bCompleto-0.pdf>>. Acesso em: 03 mar. 2019.

FALCKEMBACH, G. A. M.; ARAUJO, F. V. de. **Aprendizagem de algoritmos: dificuldades na resolução de problemas**: ISSN 2359-2656. In: CONGRESSO SUL BRASILEIRO DE COMPUTAÇÃO, 2006, Florianópolis. Anais do congresso. Florianópolis: Sulcomp, 2006. v. 2, p. 909 - 916. Disponível em: <<http://periodicos.unesc.net/sulcomp/article/view/916>>. Acesso em: 27 fev. 2019.

FRANÇA, R. S. et al. **A disseminação do pensamento computacional na educação básica**: lições aprendidas com experiências de licenciandos em computação. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 34. 2014, Florianópolis. **XXXIV Congresso da Sociedade Brasileira de Computação**. Florianópolis: CSBC, 2014. p. 1506 - 1514. Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/wei/2014/0020.pdf>>. Acesso em: 21 set. 2019.

GAMA, R. **A tecnologia e o trabalho na história**. São Paulo: Nobel Eusp, 1987.

GIL, A. C. **Como elaborar projetos de pesquisa**. São Paulo: Atlas, 1991.

HUNT, A.; THOMAS, D. **The pragmatic programmer: from journeyman to master**. Pittsburgh: Addison-Wesley Professional, 1999.

INÁCIO, F. A. J. **Ensino de algoritmo e lógica de programação: modelo construtivista auxiliado pelo scratch**. 2016. 114 f. Dissertação (Mestrado) - Curso de Mestrado em Educação, IFTM - Instituto Federal de Educação, Ciência e Tecnologia do Triângulo Mineiro - Campus Avançado Uberaba Parque Tecnológico – Mg, Uberaba, 2016. Disponível em:

<http://www.iftm.edu.br/visao/loader_anexo_cursos.php?src=131117193326_20160502_ndash_flamarion_assis_jeronimo_inacio_ndash_ensino_de_algoritmo_e_logica_de_programacao__modelo_construtivista_auxiliado_pelo_Scratch.pdf>. Acesso em: 16 jan. 2019.

JENKINS, H. et al. **Confronting the challenges of participatory culture**: Media education for the 21st century. Mit Press, 2009.

JUSTO, J. C. R. **Resolução de problemas matemáticos aditivos: possibilidade da ação docente**. 2009. 197 f. Tese (Doutorado) - Curso de Doutorado em Educação, Ufrgs, Porto Alegre, 2009. Disponível em: <<https://www.lume.ufrgs.br/handle/10183/21445>>. Acesso em: 02 mar. 2019.

KENSHIMA, G. **Makey makey - tornando o mundo maker mais divertido**. 2016. Disponível em: <Makey Makey - Tornando o mundo maker mais divertido>. Acesso em: 01 out. 2019.

LAUDARES, J. B. **Educação matemática**. Belo Horizonte: Cefet/mg, 1987.

LEITE, L. S.; POCHO, C. L.; AGUIAR, M. M. (Org.). **Tecnologia educacional: descubra suas possibilidades na sala de aula**. 2. ed. Rio de Janeiro: Vozes, 2010.

LÉVY, P. **As tecnologias da inteligência: o futuro do pensamento na era da informática**. Rio de Janeiro: 34, 1993.

LUMMERTZ, R. S. **As potencialidades do uso do software scratch para a construção da literacia digital**. 2016. 133 f. Dissertação (Mestrado) - Curso de Ensino de Ciências e Matemática, Programa de Pós-Graduação em Ensino de Ciências e Matemática, Universidade Luterana do Brasil, Canoas, 2016. Disponível

em: <<http://ppgecim.ulbra.br/teses/index.php/ppgecim/article/view/263>>. Acesso em: 29 jul. 2018.

MACHADO, J. N. **Matemática e realidade: análise dos pressupostos filosóficos que fundamentam o ensino da matemática**. 6. ed. São Paulo: Cortez, 2005.

MORAN, J. M. **Novas tecnologias e mediação pedagógica**. 6. ed. Campinas: Papirus, 2000.

MORAN, J. M. **A educação que desejamos: novos desafios e como chegar lá**. Papirus, 2007. Disponível em: <<https://books.google.com.br/books?id=PiZe8ahPcD8C&printsec=frontcover&hl=pt-br#v=onepage&q&f=false>>. Acesso em: 23 jan. 2019.

MOTA, J. F.; RINALDI, A. Elisa M.; PEREIRA, A. F.; ORSATTI, F. L.; BURINI, R. C. **Indicadores antropométricos como marcadores de risco para anormalidades metabólicas**. *Ciência & Saúde Coletiva*, v. 16, p. 3901-3908, 2011.

NORONHA, F.T. **A construção do conhecimento de algoritmos no contexto do hibridismo tecnológico: análise da prática pedagógica aplicada no IFRS**. 2016. 119 f. Dissertação (Mestrado) - Curso de Educação, Unilasalle, Canoas, 2016. Disponível em: <<http://repositorio.unilasalle.edu.br/bitstream/11690/685/1/NORONHA.pdf>>. Acesso em: 29 jun. 2018.

PAPERT, S. **A máquina das crianças**. São Paulo: Penso, 1985.

PETRY, P. G. **Um sistema para o ensino e aprendizagem de algoritmos utilizando um companheiro de aprendizagem colaborativo**. 2015. 94 f. Dissertação (Mestrado) - Curso de Mestrado em Ciência da Computação, Universidade Federal de Santa Catarina, Florianópolis, 2005. Disponível em: <<https://repositorio.ufsc.br/bitstream/handle/123456789/102997/224490.pdf?sequence=1>>. Acesso em: 25 fev. 2019.

PIAGET, J. 1975. **Gênese das estruturas lógicas elementares**. Rio de Janeiro: Forense.

PINTO, A. S. **Scratch na aprendizagem da Matemática no 1.º Ciclo do Ensino Básico: estudo de caso na resolução de problemas**. 2010. 128 f. Dissertação (Mestrado) - Curso de Mestrado em Área de Especialização em Estudos da Criança Tecnologias de Informação e Comunicação, Universidade do Minho, Braga, 2010. Disponível em: <<https://repositorium.sdum.uminho.pt/bitstream/1822/14538/1/tese.pdf>>. Acesso em: 16 jan. 2019.

RESNICK, M. **Aprender a programar, programar para aprender**. 2013. Disponível em: <<http://eduteka.icesi.edu.co/articulos/codetolearn>>. Acesso em: 20 março. 2018.

RESNICK, M. **Sobre o Scratch**. 2017. Disponível em: <<https://Scratch.mit.edu/about/>>. Acesso em: 25 nov. 2018.

RIBEIRO, R. S. **Construção e uso de ambiente visual para o ensino de programação introdutória**. 2015. 59 f. Dissertação (Mestrado) - Curso de Mestrado em Ciência da Computação, Universidade de São Paulo, São Paulo, 2015. Disponível em: <<https://www.ime.usp.br/~cpg/teses/Dissertacao-RomenigdaSilvaRibeiro.pdf>>. Acesso em: 04 fev. 2019.

RODRIGUES, A. M. M. **Por uma filosofia da tecnologia**. In: GRINSPUN, M. P. S. Z. (Org). *Educação tecnológica: desafios e perspectivas*. São Paulo: Cortez, 2001.

SALIBA, W. L. C. **Técnicas de programação – uma abordagem estruturada**. São Paulo: Makron, McGraw-Hill, 1992.

SCHANK, Roger C. **Educational outrage: an occasional column**. 2002. Disponível em: < https://books.google.com.br/books?id=tUPfBQAAQBAJ&pg=PA260&lpg=PA260&dq=SCHANK,+Roger+C.+Educational+Outrage:+an+occasional+column&source=bl&ots=yoFK53CRIJ&sig=ACfU3U1y_foUWc4YxEQMYhMfv2w-m-g9_w&hl=pt-BR&sa=X&ved=2ahUKEwicuv0nOzkAhWKILkGHflyArMQ6AEwAAnoECAkQAQ#v=snipet&q=SCHANK&f=false >. Acesso em: 25 set. 2019.

SÁNCHEZ, H.; BRAVO, J. C. F., José A. O. **Ensino da matemática: fundamentos teóricos e bases psicopedagoga** – Porto Alegre: Artmed, 2006.

SÁPIRAS, F. S. **Relações entre a literacia digital e o ambiente scratch: um olhar por meio de perspectivas matemáticas com alunos do sétimo e oitavo anos do ensino fundamental**. 2017. 126 f. Dissertação (Mestrado) - Curso de Ensino de Ciências e Matemática, Programa de Pós-graduação em Ensino de Ciências e Matemática, Universidade Luterana do Brasil, Canoas, 2017. Disponível em: < <http://ppgecim.ulbra.br/teses/index.php/ppgecim/search/search> >. Acesso em: 29 jul. 2018.

SILVA, A. L. M. L. S. da. **A apropriação do conceito de divisão por alunos dos anos iniciais do ensino fundamental**. 2104. 175 f. Dissertação (Mestrado) - Curso de Programa de Pós-Graduação em Educação, Universidade Federal do Espírito Santo, Vitória, 2014. Disponível em:

<<http://repositorio.ufes.br/bitstream/10/1931/1/Apropria%C3%A7%C3%A3o%20do%20conceito%20de%20divis%C3%A3o%20por%20alunos%20dos%20anos%20iniciais%20do%20ensino%20fundamental.pdf>>. Acesso em: 04 mar. 2019.

SILVA, S. P. **O uso da lógica de programação para a educação matemática no ensino médio: experiências com o Scratch**. 2016. 135 f. Dissertação (Mestrado) - Curso de Programa de Pós-graduação em Ensino de Ciências e Matemática, Universidade Federal de Pelotas, Pelotas, 2016. Disponível em: <http://bdtd.ibict.br/vufind/Record/UFPL_5be2adebbb636984a4107e0d7b3e49a8>. Acesso em: 17 out. 2018.

TALL, D. A. **Resposta é o de menos. Entrevista**. Notícias no mundo da Educação, 2002. Disponível em:

<<http://novaescola.abril.com.br/noticias/fev0227/indexfev0227b.htm> >. Acesso em: 26 ago. 2018.

WING, J. **Pensamento computacional – Um conjunto de atitudes e habilidades que todos, não só cientistas da computação, ficaram ansiosos para aprender e usar**. Revista Brasileira Ensino de Ciência e Tecnologia. Ponta Grossa, v. 9, n. 2, p.

1-10, mai./ago. 2016. Disponível em:

< <https://periodicos.utfpr.edu.br/rbect/article/view/4711> >. Acesso em 22 setembro de 2019.

YIN, R. K. **Pesquisa estudo de caso: Desenho e Métodos**. 2. ed. Porto Alegre:

Bookmann, 1994. Disponível em:

<http://maratavarespsictics.pbworks.com/w/file/fetch/74440967/3-YIN-desenho%20e%20metodo_Pesquisa%20Estudo%20de%20Caso.pdf>. Acesso em: 29 set. 2019.

ZOPPO, B. M. **A contribuição do scratch como possibilidade de material didático digital de matemática no ensino fundamental i**. 2017. 137 f. Dissertação

(Mestrado) - Curso de Programa de Pós-graduação em Educação em Ciências e em Matemática, Universidade Federal do Paraná, Curitiba, 2017. Disponível em:

<<https://acervodigital.ufpr.br/bitstream/handle/1884/53394/R%20-%20D%20-%20BEATRIZ%20MARIA%20ZOPPO.pdf?sequence=1&isAllowed=y>>. Acesso em: 17 out. 2018.