INFORMÁTICA NA EDUCAÇÃO teoria & prática

Vol. 24 | N° 1 | 2021

ISSN digital ISSN impresso 1982–1654 1516–084X



Páginas 115-128

Elisângela Valevein Rodrigues

Instituto Federal do Paraná elisangela.rodrigues@ifpr.edu.br

Leandro Delgado de Souza

Instituto Federal do Paraná leandro.delgado@ifpr.edu.br





PORTO ALEGRE

RIO GRANDE DO SUL Brasil

Recebido em: março de 2021 Aprovado em: maio de 2021

Instituto de Hackers: o pensamento computacional aplicado ao Ensino Médio Integrado Profissionalizante

Hacker's Institute: the computational thinking on vocational learning

Resumo

O pensamento computacional desperta criatividade, espírito crítico, raciocínio lógico e a construção passo a passo de soluções do mundo real, por meio de fundamentos e técnicas computacionais. Nesta pesquisa, apresenta-se a promoção do ensino por meio do pensamento computacional no Ensino Médio Integrado, de uma instituição pública focada em educação profissional. Isso, mediante a realização de uma oficina de curta duração, utilizando linguagem de programação visual por blocos, com ferramentas como Code.Org e Scratch. A oficina teve sua construção e concepção baseadas na teoria de aprendizagem situada, de Jean Lave. Foi aplicada em dois momentos com turmas distintas, e ao final de sua aplicação foi avaliada pelos estudantes. Notou-se resultado positivo e motivador quanto à metodologia utilizada e à linguagem de blocos. Assim, considera-se que este estudo atendeu satisfatoriamente aos objetivos apresentados e que novas propostas de ensino, utilizando-se dessas premissas, possuem caráter válido e pertinente.

Palavras-chave: Educação Profissional. Pensamento Computacional. Aprendizagem Situada.

Abstract

The computational thinking awakes creativity, critical spirit, logical reasoning e pass-by-pass construction of real world solutions, using computational fundaments and techniques. This research seeks to relate the promotion of teaching through computational thinking in integrated high school, from a public institution focused on professional education. A short workshop was conducted using block visual programming language with tools such as Code.Org and Scratch. The workshop was built and designed based on Jean Lave's theory of situated learning. It was applied in two moments with different classes and, at the end of its application was evaluated by the students. Positive and motivating results were noted regarding the methodology used and the block language. Thus, it is considered that this study satisfactorily met the objectives presented and that new teaching proposals using these premises have a valid and pertinent character.

Keywords: Vocational Learning. Computational Thinking. Situated Learning.

1. Introdução

O termo pensamento computacional, como é conhecido hoje, surgiu a partir da publicação realizada por Wing, sendo que tanto humanos quanto máquinas possuem capacidade de solucionar problemas baseados no poder e nos limites da computação (WING, 2006). O pensamento computacional está baseado em quatro pilares: abstração, decomposição, reconhecimento de padrões е algoritmo (BRACKMANN et al., 2017). Esses pilares trabalham para que os estudantes adquiram capacidade de transformar problemas grandes em menores e, assim, fragmentem-nos, sendo esse o pilar da decomposição. isso, há a possibilidade de verificar minuciosamente cada um dos problemas, podendo identificar padrões ou dificuldades já solucionadas, reconhecendo padrões, dando atenção apenas ao que importa ou tem relevância para a solução do problema existente, pilar da abstração; e possibilitando, então, criar um método computacional, digital ou manual como solução do problema, pilar do algoritmo.

O pensamento computacional é uma habilidade que deve estar disponível para todas as pessoas, não sendo uma exclusividade daquelas ligadas à computação (WING, 2016). "É uma área recente e ainda tem muito para ser explorado, principalmente do ponto de vista educacional" (QUADROS; BICHO; ADAMATTI, 2020). Além disso, o pensamento computacional é comparado à alfabetização do século atual, em que vários postos de trabalho exigem das pessoas um conhecimento, mesmo que básico, de programação de computadores (MARIMON BOUCINHA et al., 2017). Essa nova metodologia pode ser adotada em sala de aula por educadores nas suas práticas, durante todas as etapas e em consonância com a alfabetização e o letramento digital (QUADROS; BICHO; ADAMATTI, 2020).

Trabalhar o pensamento computacional, com estudantes do Ensino Médio Técnico, contribui para que eles tenham uma formação voltada para a autonomia, levando à sua emancipação tecnológica, aumentando a criatividade, o raciocínio lógico e a capacidade de solucionar problemas. Isso tudo contribui para sua emancipação e atuação no mundo do trabalho. Nesse sentido, trabalhar o pensamento computacional com os estudantes, com base na teoria de aprendizagem situada de Jean Lave, pode ser um fator inovador, impactando diretamente no processo de ensino-aprendizagem deles.

A formação dos estudantes de forma autônoma, emancipadora e de outros atributos já mencionados, é parte de uma política educacional praticada nos Institutos Federais de Ciência e Tecnologia, também do Colégio Pedro II e dos CEFETs, que são, por sua lei de criação 11.892/2008, as instituições federais de referência em educação profissional. Saviani(2007) cita que nessas instituições "[...] no ensino médio a relação entre educação e trabalho, entre conhecimento e a atividade prática deverá ser tratada de maneira explícita e direta". O pensamento computacional visa fortalecer a relação educação-trabalho, ao propiciar a união de conhecimento e prática, que está muito presente nas atividades envolvendo pensamento computacional e programação de computadores, pois estes se referem a capacidades de modelagem, análise e solução de problemas(SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 2018).

Assim, o objetivo deste estudo foi estimular e avaliar o pensamento computacional para estudantes do Ensino Técnico Integrado ao Ensino Médio, por meio de uma proposta de ensino (oficina de curta duração), para cursos que não estejam diretamente relacionados com áreas pontualmente ligadas à computação.

2. Referencial Teórico-Metodológico

O referencial teórico vai trazer informações sobre a teoria e a metodologia utilizadas para a construção da referente pesquisa. Os conceitos abordados serão: Pensamento Computacional, Educação Profissional, Institutos Federais de Ciência e Tecnologia, Prática Pedagógica, Programação de Computadores, Ensino de Programação, Lógica de Blocos e a Prática da Aprendizagem.

2.1 Referencial Teórico

2.1.1 Pensamento Computacional

Com o avanço da tecnologia, a educação tem recebido ferramentas e métodos auxiliares no processo educativo, e as Tecnologias da Informação e Comunicação (TICs) estão muito presentes nas salas de aula contemporâneas, que têm modificado o paradigma da aprendizagem para algo mais profundo e contextual, possibilitando experimentações, resoluções de problemas, simulações, entre outros (RAMOS, 2015).

Grupos como o Pensamento Computacional¹ ou o Grupo Computação na Escola², além de outros sociedades grupos como civis, órgãos governamentais ou entidades representativas da área, são incentivadores da aplicação e do uso das TICs no processo de ensino-aprendizagem, reconhecendo o papel que a tecnologia vem adquirindo no método educativo (RAMOS, 2015).

A tecnologia e a educação modificaram e ainda modificarão o mundo do trabalho. Dessa forma, para a formação dos futuros profissionais, além do uso das TICs, práticas de computação são atividades essenciais para estar em consonância com o mundo de trabalho atual e futuro (VON WANGENHEIM; RODRIGUES NUNES; DOS SANTOS, 2014).

Para isso, o pensamento computacional aborda a solução ou resolução de problemas reais, por meio de métodos computacionais, pelo uso do computador ou algum outro dispositivo semelhante, para coletar e analisar dados, apresentando-os de várias maneiras diferentes, incluindo simulações, automatização das soluções; realizar processamento de dados a fim de se atingir maior eficiência e eficácia da solução ou procedimento, sistematização ou padronização da solução do problema (MARIMON BOUCINHA et al., 2017). Tudo isso também é possível utilizando os conceitos de computação de forma desplugada, ou seja, sem o uso do computador, utilizando apenas seus conceitos (VON WANGENHEIM; RODRIGUES NUNES; DOS SANTOS, 2014).

relevante elucidar que o pensamento computacional não significa que os seres humanos devam pensar como os computadores, mas sim que fundamentos utilizem técnicas, computacionais aliados à inteligência humana para a abordagem e solução de problemas, que vai muito além de programar uma solução computacional (RIBEIRO; FOSS, 2017).

Apesar de o conceito de Wing ainda ser o mais utilizado, ainda há divergências conceituais, porém, praticamente todos convergem para a intersecção entre computação, conhecimento disciplinar e algoritmo (ARMANDO, 2016).

Ribeiro e Foss (2017) definem que o pensamento computacional possui relação com seus pilares: abstração - necessária para dados e processos, e as técnicas de construção de soluções (algoritmos); análise - consiste em técnicas de análise de algoritmos quanto à sua correção e eficiência, sob diferentes aspectos; e automação - envolve a mecanização das soluções ou de suas partes, permitindo que máquinas nos ajudem a solucionar o problema.

Brackmann et al. (2017), com base em pesquisas da Code.Org³, Liukas⁴ e BBC Learning⁵ citam quatro pensamento computacional: decomposição, reconhecimento de padrões, abstração e algoritmo. A decomposição trata de os estudantes se depararem com um problema grande e transformá-lo em um menor e mais simples de ser resolvido, para que assim possam reconhecer padrões nos problemas, ou encontrar soluções já existentes para os pequenos problemas. Dessa forma, é possível focar apenas nos problemas específicos ou naqueles que não possuem e precisam de solução, essa seria a capacidade de abstração. Por fim, a pessoa é capaz de construir um algoritmo, que nada mais é que uma solução computacional, um processo passo a passo para a solução do problema, que pode ser feita computacionalmente ou não.

2.1.2 Educação Profissional

A promulgação da lei 11.892 de 29 de dezembro de 2008 criou e concretizou a implantação de algumas instituições de ensino profissional no Brasil, sendo abrangidos por essa lei os Institutos Federais de Ciência e Tecnologia, o Colégio Pedro II, os Centros Federais de Educação Tecnológica do Rio de Janeiro e de Minas Gerais, além da Universidade Tecnológica Federal do Paraná, que fazem parte da Rede Federal de Educação Profissional, Científica e Tecnológica. Essas instituições por finalidade desenvolvimento e a qualificação dos cidadãos, permitindo-lhes a devida formação para atuação em qualquer setor econômico, bem como atendimento às demandas regionais (BRASIL, 2008).

Os objetivos elencados na lei 11.892/2008 demonstram que não se trata apenas de ensinar, mas oferecer uma formação integral aos estudantes. Vejamos alguns deles.

¹ Disponível em: www.computacional.com.br.

² Disponível em: www.computacaonaescola.ufsc.br.

³CODE, ORG. Instructor Handbook—Code Studio Lesson Plans for Courses One, Two, and

⁴LIUKAS, L. *HelloRuby*: adventures in coding. Feiwel & Friends, 2015. ⁵BBC LEARNING, B. What is computational thinking? 2019. Disponível em: http://www.bbc.co.uk/education/guides/zp92mp3/revision. Acesso em: 25 abr. 2019.

- Ministrar educação profissional e técnica de nível médio [...];
- Realizar pesquisas aplicadas [...];
- Desenvolver atividades de extensão, [...], em articulação com o mundo do trabalho [...];
- Estimular e apoiar processos educativos que levem à geração de trabalho e renda e à emancipação do cidadão [...].

Nota-se que a formação dos estudantes dos Institutos Federais é voltada para capacitá-los de forma que possam integrar os conhecimentos adquiridos com cultura, pesquisa e tecnologia (BRASIL, 2010).

As instituições de ensino profissional possuem uma flexibilidade curricular garantida tanto pela Lei 11.892/2008 quanto pelo documento Concepções e Diretrizes, da SETEC/MEC, para que os currículos possam atender a demandas e arranjos produtivos locais. Além do atendimento ao contexto do qual cada campus está inserido. A estrutura curricular dessas instituições tem como propósito "uma formação profissional e tecnológica contextualizada, banhada de conhecimentos, princípios e valores que potencializam a ação humana na busca de caminhos mais dignos de vida." (BRASIL, 2010).

Estudantes oriundos de classes trabalhadoras ou detentoras de meio de produção, sem distinção, por meio dessa educação e da flexibilidade curricular, alcançam uma capacidade criativa e autônoma, política e intelectual (ARAUJO; FRIGOTTO, 2015). Com relação à formação profissional contextualizada, a integração dos conteúdos pela prática pedagógica contribui para que os estudantes consigam compreender as relações entre os conteúdos teóricos transmitidos em sala de aula e as atividades e os desafios práticos reais do mundo do trabalho (ARAUJO; FRIGOTTO, 2015), tendo dimensões como trabalho, ciência e cultura integrados por esse processo educativo (CIAVATTA; RAMOS, 2011).

Importante ressaltar que a educação profissional não trata do ensino de uma profissão por si só, ou prepara o cidadão para a execução de tarefas rotineiras, mas transmite ao acadêmico condições para que ele compreenda o mundo do trabalho em todas suas vertentes, seu dinamismo, as condições socioprodutivas, além do entendimento integral de suas atividades provenientes de sua profissão, de forma crítica, criativa e autônoma (BRASIL, 2010).

Uma educação pública, gratuita, laica e universal, pode proporcionar aos estudantes o conhecimento necessário para que eles compreendam a relação educação-trabalho, o que é diferente da vida profissional, mas se trata de um momento vivido com base na experimentação do tentar e de conhecer algo (FRIGOTTO, 2007).

Dessa forma, a educação profissional e tecnológica contemporânea, em sua concepção, objetiva uma formação dos estudantes como sujeitos atuantes, críticos e participativos. Nessa perspectiva, o desenvolvimento do pensamento computacional pode proporcionar os conhecimentos suficientes para tal concepção, por meio de estratégias que estimulem o desenvolvimento de seu conhecimento e seu aprendizado e, assim, não sejam meros reprodutores de conteúdo.

2.1.3 Ensino de Programação

A inserção da computação no cotidiano das escolas brasileiras ainda precisa de investimento e de novos métodos pedagógicas. Porém, pode-se utilizar a tecnologia de uma forma mais abrangente no ensino de lógica de programação, possibilitando desenvolver o raciocínio e a criatividade, auxiliando na tomada de decisões (BATISTA; HORTA; FONSECA, 2020).

O ensino de programação trata-se, em muitos casos, de passar uma linguagem de programação aos estudantes, mostrando técnicas, sintaxe, lista de comandos e afins, o que torna a aprendizagem um processo custoso e nada convidativo e chamativo para eles (FONTES; SILVA, 2008). Essa forma de ensinar proporciona elevados índices de evasão e reprovação em disciplinas relacionadas à programação de computadores, pois não é uma tarefa trivial. Requer conhecimentos matemáticos, raciocínio lógico, abstração, e uma capacidade de solução de problemas (JUNQUEIRA, 2016).

Um programa de computador ou algoritmo é uma sequência de passos e instruções visando atingir ou solucionar um problema, um mecanismo de atividades e ações detalhadas e ordenadas para realizar alguma tarefa (FORBELLONE; EBERSPACHER, 2005). Isso mostra que realmente um programa de computador é algo bastante complexo, não trivial, em que há dados de entrada e/ou saída, que podem ser textos, números reais, números inteiros, conjunto de números, planilhas, arquivos formatados, entre outros.

Para que esses dados de entrada sejam processados e convertam-se em dados de saída, existem inúmeras técnicas de programação que

podem ser utilizadas (FORBELLONE; EBERSPACHER, 2005). Tudo isso contribui para que o processo de ensino-aprendizagem de programação torne-se complexo (DE JESUS; BRITO, 2009).

Pensando na aplicação de disciplinas programação em cursos que não são da área de computação, o processo de ensino-aprendizagem pode se tornar ainda mais complexo e difícil.

Para contrapor a essa metodologia tradicional, projetos como a Hora do Código⁶, a Escola de Hackers (Projeto da Prefeitura de Passo Fundo) e o Projeto Jovens Hackers (Instituto Federal de São Paulo), contribuem no processo de aprendizagem de técnicas de programação de computadores, em praticamente todos os níveis de ensino, utilizando softwares com linguagem de programação visual por blocos como estratégia de Sociedade própria Brasileira Computação (SBC) incentiva que em séries iniciais, na matriz curricular, já sejam contemplados componentes que envolvam lógica de programação, com metodologia muito distinta da praticada em nível de graduação ou além.

O termo Hacker, utilizado neste e nos estudos supracitados, possui relação com o surgimento de grupos de estudantes norte-americanos, que em laboratórios, garagens, porões passaram a utilizar da tecnologia e da programação para resolver problemas de projetos propostos (PRETTO, 2010). Ou seja, esse termo tem relação com a criação de soluções tecnológicas para problemas reais, e essa prática apresenta conexão com a aprendizagem situada proposta por Jean Lave, que defende o vínculo dos socioculturais aspectos vivenciados adquiridos anteriormente conhecimentos pelo indivíduo e a solução de problemas.

2.2 Referencial Metodológico

2.2.1 Linguagem de Programação por Bloc

A programação por meio de linguagem de blocos não é tão recente, contudo, essa técnica emergiu com a popularização e crescente utilização de softwares e ferramentas como Scratch, SNAP!, Blockly, Hour of Code e App Inventor (WEINTROP; WILENSKY, 2017).

Pensando nos objetivos da proposta de ensino e no público-alvo, utilizar a lógica de blocos mostrou-se uma alternativa satisfatória. Por não utilizar linguagem textual, essa lógica motivou-se pela complexidade da sintaxe dos comandos, pela falta de atratividade e por um possível impacto negativo que uma linguagem textual pode causar nos estudantes, em razão da utilização de caracteres especiais, linguagem dos comandos e palavras reservadas em língua inglesa (WEINTROP; WILENSKY, 2017).

Além desses fatores, а linguagem programação por blocos mostra-se positivamente intuitiva, pois não ocorrem erros de sintaxe. Os softwares que utilizam essa linguagem não permitem o encaixe de blocos que não se combinam. Dessa forma, restam apenas erros de lógica de programação aos estudantes, que serão solucionados e resolvidos com o desenvolvimento das atividades e do aprendizado dos conceitos e das técnicas de programação.

2.2.2 A Prática da Aprendizagem

A teoria de ensino-aprendizagem abordada e utilizada na construção da proposta de ensino da oficina de programação foi a de Jean Lave, sendo intitulada, traduzindo do original Understanding learning, aprendizagem situada. Essa teoria tem como base que as pessoas e o mundo social em que elas vivem e agem não podem ser separados (LAVE, 2013).

Nesse contexto, entende-se que atividades cotidianas são capazes de proporcionar aprendizado onipresente, em que várias situações, casos ou eventos levam a pequenas tomadas de decisões, tendo por base conhecimentos previamente adquiridos, como conceitos matemáticos, químicos, físicos, lógicos, históricos, sociais, ou seja, de qualquer ciência ou área (LAVE, 2013).

Abordagens socioculturais são a base da teoria da aprendizagem situada, explicando, assim, o processo e o fenômeno da aprendizagem (ALMEIDA, 2014). O processo de aprendizagem situada ocorre de forma informal, espontânea, não é imposto que os estudantes aprendam. Já o aprendizado espontâneo, implícito e não direcional, ocorre por meio de atividades que os aprendizes desenvolvem, vivenciam ou experimentam em determinado momento ou contexto (BORGES; PAEZ; BELTHER, 2016).

Pensamento, ação, sentimento e racionalidade são características das atividades dos aprendizes que

⁶ Disponível em: https://hourofcode.com/br.

não devem ser separadas. Essas atividades são relevantes, conflituosas e possuem um significado, compondo o contexto desses discentes (LAVE, 2013).

Situam-se os educandos para que alcancem o aprendizado e o conhecimento, sendo que o conhecimento sempre sofre construção transformação quando utilizado, pois o conhecer não se trata de absorver informação, e sim que ocorra reconceituação em produtos ou atividades culturais e/ou sociais. Assim, nota-se que "[...] a atividade situada sempre envolve mudanças no conhecimento e nas ações, e essas mudanças são fundamentais no conceito de aprendizagem" (LAVE, 2013).

Alguns itens são bastante necessários para a aprendizagem situada, como "atividades, compartilhamentos, relacionamentos, cooperação, dialética, observação, interação, negociação, aperfeiçoamento, significado e criatividade" (MOSER; SCHNEIDER; DE MEDEIROS, 2013). Contudo, não se trata de situar ou proporcionar atividades que compreendam todos esses itens, obrigatoriamente, mas que possam levar ao alcance desses itens. As atividades envolvendo contexto, proporcionam um espaço real ou uma situação simulada mais verídica aos estudantes, assim como uma intensa ação na resolução de um problema desse contexto. Com isso, adquirem-se novos conhecimentos, conhecimentos prévios são aplicados, há melhora nas habilidades técnicas, expande-se uma gama de ideias e agregamse novos conhecimentos (SOUZA; YONEZAWA, 2018).

3. Materiais e Métodos

Este estudo deu-se de forma qualitativa e quantitativa. O aspecto qualitativo visou discutir e compreender as características do processo e seu significado, o entendimento das relações entre a educação profissional, o pensamento computacional, o ensino de programação e a aprendizagem situada, como um fator contribuinte do processo de ensinoaprendizagem dos estudantes do Ensino Médio Técnico Integrado. O aspecto quantitativo foi analisado de forma descritiva para quantificar os dados.

Nesse projeto, buscou-se descobrir a opinião dos estudantes que participaram da oficina Instituto de Hackers, quanto ao aprendizado que tiveram, e se conseguiram perceber os benefícios do pensamento computacional em sua formação. A Oficina Instituto de Hackers foi assim denominada juntando o nome do Instituto Federal com o termo Hacker.

Os critérios metodológicos deste estudo foram de natureza descritiva e participante contemplando as etapas de revisão da literatura, delimitação da oficina, construção do plano de aula da oficina, realização da oficina, coleta dos dados, análise e interpretação dos dados e divulgação dos resultados.

Inicialmente, foi realizada uma pesquisa bibliográfica, dividida em duas temáticas, uma referente à educação profissional e tecnológica e a outra sobre o pensamento computacional. Foram consultadas bases como ResearchGate, Scielo, revistas científicas da área e Google Acadêmico, além de páginas de grupos de pesquisas disponíveis na internet livremente.

O estudo foi realizado no Instituto Federal do Paraná, Campus Pitanga, com estudantes do Curso Técnico em Cooperativismo, do 1.°, 2.°, 3.° e 4.° anos, no período do contraturno. O momento 1 do projeto ocorreu no segundo semestre de 2018, nos meses de setembro e outubro, às quintas-feiras. Já a segunda aplicação do produto educacional, momento 2, ocorreu no primeiro semestre de 2019, no mês de março, com aulas às segundas e terças-feiras, ambas nos horários de 13:30 às 17:00 horas, no laboratório de informática do

Para que o projeto pudesse ser aplicado com esses estudantes, ele foi submetido ao Comitê de Ética em Pesquisa do Instituto Federal do Paraná, tendo sido aprovado sobre número de CAAE: 87650418.2.0000.8156.

Para participar desse estudo, os estudantes foram convidados a fazer parte da oficina "Instituto de Hackers". A divulgação e o convite foram feitos via redes sociais do campus e em sala de aula. Foram realizados 6 encontros, totalizando 20 horas/aula.

O critério de seleção ocorreu por ordem de inscrição, e aos estudantes interessados a participar da oficina foi explicado o caráter e os objetivos da pesquisa na primeira aula.

Como o tema pensamento computacional é muito amplo, foi necessário delimitar a oficina, definindo uma carga horária de 20 horas/aula, contemplando tópicos iniciais de programação, estruturas sequenciais, estruturas de decisão e estruturas de repetição.

Após a delimitação dos tópicos a serem trabalhados, o plano de aulas foi organizado em atividades e problemas práticos que fizessem parte de situações vividas pelos estudantes, ou que tivessem relação com algum componente curricular da grade curricular do curso técnico do qual eles faziam parte. Os conteúdos trabalhados na oficina foram selecionados verificando as ementas das disciplinas que os estudantes já haviam cursado. Como havia participantes do 1.°, 2.°, 3.° e 4.° anos do curso de Cooperativismo,

Elisângela Valevein Rodrigues & Leandro Delgado de Souza

os conteúdos foram cuidadosamente escolhidos para que os estudantes das séries iniciais não se sentissem incapazes de resolver os problemas e as atividades propostas. Diante das ementas, buscaram-se tópicos que tivessem relação com alguma das estruturas a

serem trabalhadas na oficina: sequencial, decisão e repetição.

O quadro a seguir contém uma breve descrição da proposta de atividades das aulas da oficina.

Quadro 1 – Proposta de Atividades

Dia	Conteúdo	Atividades
1	Programação sequencial	 Explanação sobre programação; Apresentação dos objetivos da oficina; Atividade 1 (apresentação da programação por meio de lógica de blocos); Atividade 2 (apresentação do software Scratch).
2	Programação sequencial	 Atividade Movimento Retilíneo Uniforme (MRU) – conteúdo de física do 1.º ano do Ensino Médio; Atividade Misturas Químicas.
3	Estrutura de decisão	 Explicação de estruturas de decisão; Atividades Estrutura de Decisão (descontos em lojas).
4	Estrutura de decisão	1) Desenvolvimento de um jogo (cliques sobre objetos).
5	Estrutura de repetição	 Explicação sobre estruturas de repetição (caixa de supermercado, atividades rotineiras, relógio); Atividades da hora do código sobre estruturas de repetição.
6	Estrutura de repetição	1) Desenvolvimento de um jogo (estilo Guitar Hero).

Fonte: Os autores.

Os estudantes que concluíram a oficina com carga horária acima da mínima estabelecida responderam a um questionário de avaliação. O questionário foi adaptado do modelo de avaliação dETECT (VON WANGENHEIM et al., 2017) e buscou captar a percepção dos estudantes com relação à oficina, bem como da metodologia e da utilização da programação por blocos.

A avaliação da oficina ministrada ocorreu, no momento 1, por 14 estudantes, e no momento 2, por 10 estudantes. Os dados obtidos foram analisados e interpretados de forma qualitativa e quantitativa utilizando estatística descritiva.

4. Resultados e Discussão

O objetivo da avaliação da oficina foi explorar e compreender os aspectos relacionados com o ensino de programação por meio da linguagem de blocos, tendo como base da construção da oficina a teoria da aprendizagem situada de Jean Lave.

Durante a oficina, os estudantes apresentaram bastante interesse no desenvolvimento das atividades e um comprometimento na realização das atividades propostas. Todos relataram ter gostado da oficina, apontando pontos positivos, como a facilidade que tiveram na execução das atividades propostas, enfatizando em muitos casos a contribuição para o conhecimento e para sua formação. Como exemplo, sequem dois relatos:

> Gostei muito, pois tinha pouco conhecimento sobre o assunto, e o professor usou uma boa metodologia durante a oficina, o que ajudou muito na aprendizagem. Acho importante o ensino da computação em cursos que não sejam apenas de informática, pois nos proporciona um conhecimento mais amplo que pode ajudar a desenvolver trabalhos de outras disciplinas relacionadas ao curso que a pessoa está fazendo. (Estudante 1.1).

> Muito boa, aprendi muito e eu não fazia ideia de como programar, depois destas aulas consigo ter uma noção ótima de programação. (Estudante 2.1).

Outro aspecto relevante foi que eles perceberam, na metodologia proposta, baseada na aprendizagem situada, facilidade no entendimento das atividades e dos conteúdos oferecidos, mostrando o ponto diferencial para com outras metodologias de ensino.

Os estudantes responderam sobre a metodologia quando questionados sobre o que mais gostaram na oficina e também o que acharam dela, evidenciando-a conforme respostas a seguir:

> Achei excelente, pois teve uma metodologia boa, de fácil interpretação. (Estudante 3.1).

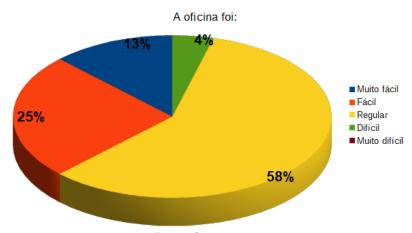
> Da metodologia utilizada pelo professor, a forma como foi apresentada a linguagem de blocos. (Estudante 4.1).

> Gostei de praticamente toda a oficina, desde a maneira que foi ensinado e o programa que utilizamos. (Estudante 5.1).

Com 24 questionários respondidos, 50% dos estudantes de alguma forma se referiram à metodologia como uma das coisas que mais gostaram na oficina. Como a questão era aberta, ocorreram diversas respostas, considerando que metade dos discentes apontou a metodologia, ou algo semelhante, como "a forma que o conteúdo foi passado", bastante significante.

O Gráfico 1 representa as respostas dos estudantes quando questionados como foi a oficina. As respostas da questão representam o nível de dificuldade que tiveram na realização da oficina. Praticamente 83% deles acharam que o nível ficou entre "regular e muito fácil", de acordo com as respostas.

Gráfico 1 – Respostas da questão "A oficina foi:" referente ao nível de dificuldade.



Fonte: Os autores.

O Gráfico 2 demonstra a opinião geral dos estudantes com relação à oficina. Para esse questionamento, 92% dos estudantes responderam que ela foi "Excelente ou Boa", representando, assim, um nível alto de satisfação.

Gráfico 2 - Respostas da questão "A oficina foi:" referente à opinião sobre a oficina.



Fonte: Os autores.

Elisângela Valevein Rodrigues & Leandro Delgado de Souza

Ao serem questionados sobre algo que menos gostaram na oficina, muitos estudantes relataram que o tempo de duração poderia ser maior, ou que as aulas poderiam ocorrer mais vezes por semana, ou que as atividades poderiam ter um nível de dificuldade maior, por exemplo:

> A curta duração das atividades e o conteúdo básico, que apesar de ter sido excelente poderia ter sido mais aprofundado. (Estudante 1.1).

Foi muito curto o tempo da oficina e poderia ter ocorrido mais de uma vez na semana. (Estudante

Não seria um não gostar, mas a oficina poderia abordar um conteúdo mais aprofundado sobre programação em blocos. (Estudante 2.2).

Gráfico 3 – Respostas com relação ao tempo de duração da oficina.



Fonte: Os autores.

O gráfico 3 mostra um contraste nas respostas dos participantes, a maioria dos estudantes, 75%, considera o tempo da oficina como bom. Porém, como já citado, esse tempo foi considerado curto por muitos deles. Esse contraste de informações pode ter relação com o que os estudantes gostaram da oficina e demonstraram bastante interesse, também reforçado por algumas respostas nas quais demonstraram querer se aprofundar mais nos conteúdos.

Quando questionados sobre a vontade de aprender mais a respeito de como criar ou programar softwares, apenas 8% responderam que não tinham interesse, sendo que o restante dos estudantes, 92%, responderam ter interesse em aprender mais e se aprofundar no assunto.

A respeito da experiência de participar da oficina, percebeu-se que para todos ela foi satisfatória. Os estudantes compreenderam os objetivos da oficina, a metodologia de ensino, a utilização da lógica de blocos e a contribuição que o saber programar ou ter os conceitos de programação pode fazer para a formação deles. Exemplifica-se isso por respostas como:

> A oficina denominada "Instituto de Hackers Fase I" foi uma atividade que reuniu conteúdos básicos da programação para estudantes que em sua maioria

nunca tiveram contato com a área, através de uma metodologia simples e muito eficaz. Fortalecendo o raciocínio lógico em cada um dos orientandos. (Estudante 1.1).

Ótima experiência, aprendi várias coisas como programas e básico e a utilizar melhor programas e aplicativos, linguagem de programação é ótima (simples, porém excelente e prática para iniciantes). (Estudante 1.3).

Foi excelente, a programação em blocos é mais didática, e rompe o clima de que a programação é algo difícil e estressante. (Estudante 2.2).

A linguagem de blocos também foi bastante citada quando os estudantes foram questionados sobre a experiência na realização da oficina. Nesse caso, ela mostrou-se bastante atrativa, tendo sido reconhecida como uma facilitadora do processo de aprendizagem, como citou um estudante:

> Bom, quanto à relação dos blocos achei muito fácil, é só colocar o bloco certo no lugar certo e acontece a mágica, forma fácil de aprender assim. (Estudante 2.6).

Minha experiência foi boa, aprender algo a mais é sempre bom, linguagem utilizada era bem simples, o que facilitou muito. (Estudante 1.9).

Em concordância com Coelho da Rocha (2016), o ambiente Scratch e a linguagem de blocos são de fácil domínio e utilização, os comandos e os blocos são bastante intuitivos, mesmo para pessoas que não têm muito contato com a programação de computadores.

Um dos objetivos da oficina foi o estímulo do pensamento computacional, que os estudantes compreendessem sobre o desenvolvimento de software, percebessem que as atividades desenvolvidas na oficina já eram consideradas softwares. Com isso, os estudantes foram questionados se já se consideravam aptos a desenvolver programas de computador ou a criar algum tipo de software. Nessa questão, 98% responderam afirmativamente, o que foi bastante satisfatório para este estudo.

O Gráfico 4 apresenta as respostas sobre a complexidade do desenvolvimento de software após a realização da oficina, que ficaram concentradas entre regular, fácil e muito fácil. Para 63% dos estudantes, é possível criar um software, mas compreendem que há certo nível de dificuldade. Já 37% dos estudantes consideraram programar como uma atividade fácil ou muito fácil de ser realizada.

Essas respostas reforçam a efetividade da oficina com relação à compreensão dos estudantes em conseguir produzir algum tipo de software.

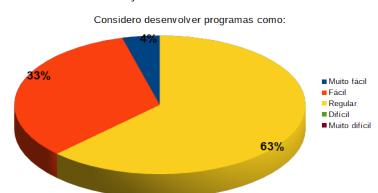


Gráfico 4 – Considerações sobre desenvolvimento de software.

Fonte: Os autores.

A oficina, trabalhada de modo contextualizado, utilizando problemas do cotidiano dos estudantes, buscou, além dos objetivos de aprendizagem, demonstrar a aplicação da computação nessas atividades. Com isso, eles foram questionados se no dia a dia a computação é importante e relevante, e 100% dos estudantes responderam afirmativamente. Eles também foram provocados a justificar essa resposta, com intuito de verificar como a computação poderia ser útil. Seguem algumas respostas dos estudantes:

> Sim, a computação é útil no dia a dia na medida em que todo nosso meio é digital. A compreensão lógica das coisas é essencial. (Estudante 1.1

> Cada vez mais a tecnologia está presente em nosso dia a dia, principalmente o uso de computadores e celulares, saber um pouco sobre computação ajuda muito e facilita seu uso. (Estudante 1.6)

> A computação pode facilitar e muito a vida das pessoas, pois podemos criar programas para executar as mais variadas funções. (Estudante 1.8)

Sim, em um mundo onde quase tudo têm ênfase tecnológica, a computação deve estar presente desde o ensino básico das pessoas, para que tenham um olhar lógico e analista das coisas. (Estudante 2.2)

Pois está presente em diversos formatos a nossa volta, desde os caixas eletrônicos nos bancos aos smartphones em nossas mãos. (Estudante 2.7)

Reforçando essa questão, a computação ou a fluência computacional hoje é um direito das pessoas, e demonstra grande relevância quanto à busca de oportunidades melhores de emprego ou educação, além de contribuir no processo de formação de cidadãos ativos e informados da sociedade (VAKIL, 2014). Com isso, percebe-se que este estudo contribuiu para a formação pessoal e profissional dos estudantes.

questionário respondido Além estudantes, pôde-se observar, durante a realização da oficina, que eles demonstraram bastante interesse na oficina em geral, eram muito participativos e o tempo todo tiravam dúvidas sobre a execução das atividades propostas. Durante a explicação da contextualização das atividades, além de participativos, os estudantes auxiliavam no processo de concepção dos conceitos necessários para a construção da solução dos problemas, contribuindo para que o objetivo da aprendizagem situada pudesse ser atendido.

O interesse e a participação dos estudantes também foram analisados por Medeiros, Wangenheim e Bergmann (2018), que verificaram "os impactos de práticas pedagógicas de desenvolvimento aplicativos usando ambiente de programação visual App Inventor", e uma de suas conclusões foi "[...] o uso de dispositivos móveis não representou nenhum problema na concentração dos alunos, possivelmente por conta das estratégias metodológicas. [...]".

Nesse sentido, ressalta-se que, apesar de todas as distrações que um laboratório de informática pode proporcionar, como acesso à internet, jogos, entre outras, os estudantes mantiveram-se focados e participantes das atividades.

A maior dificuldade dos estudantes foi na execução das atividades da aula 5, principalmente nas que envolviam a estrutura de repetição. Apesar de as primeiras atividades da aula serem feitas na hora do código, que se trata de uma atividade mais lúdica, os estudantes demonstravam falta de compreensão dos padrões de repetição da atividade, principalmente da "Frozen", a qual envolvia movimentação e ângulos, responsáveis pela ação de virar para os lados para que os desafios pudessem ser cumpridos. Foi percebido que os estudantes não liam as instruções da atividade, simplesmente viam na tela o desafio proposto e tentavam fazer, sem se atentar às instruções mostradas. Isso levava à tentativa e erro, principalmente em uma fase na qual a solução consistia em uma estrutura de repetição dentro da outra.

Nas atividades de gamificação, os estudantes puderam demonstrar criatividade e autonomia, e os resultados foram satisfatórios. Eles perguntavam se era possível fazer com outros atores ou personagens, se podiam fazer parecidos, mas usando as mesmas estruturas. Sabendo disso, criaram jogos variados, mas semelhantes à estrutura proposta do jogo "Duck Hunt".

O ambiente Scratch demonstrou ser uma excelente ferramenta despertar a criatividade para estudantes, também visto no trabalho de Von Wangenheim, Rodrigues Nunes e Dos Santos (2014). Nele, há uma gama de atores e personagens que já existem na aplicação, mas também é possível inserir novos atores a partir de outras imagens, podendo ser qualquer tipo de foto ou imagem, além da imensa gama de ações e eventos que o ambiente proporciona ao aprendiz.

Com relação à educação profissional técnica e tecnológica, o pensamento computacional, estimulado pelo ensino de programação com a utilização da linguagem de programação visual por blocos, mostrouse capaz de despertar a criatividade dos estudantes. Isso foi percebido, principalmente, na criação dos jogos propostos e no desenvolver do raciocínio lógicomatemático. O pensamento computacional também possibilitou aos estudantes desenvolver o espírito crítico, exemplificado pela compreensão que eles tiveram com a utilização da tecnologia nos dias atuais, e perceber que o aprendizado de programação é útil em sua formação, não apenas técnica, mas como ser atuante no meio tecnológico em que vivem.

Os estudantes terem conseguido criar soluções para os problemas propostos demonstra que eles podem ser autônomos, e não dependentes de soluções já prontas. Isso também leva os estudantes à emancipação tecnológica, pois possuem liberdade de escolha, aliada ao espírito crítico, em criar uma solução, se necessário, ou fazer aproveitamento de uma já existente. Segundo Fernandes (2017), o pensamento computacional corrobora com o desenvolvimento da emancipação tecnológica do ser, pois contém em sua essência proporcionar o conhecimento de quando e como utilizar as tecnologias, e não mais apenas usufruir do que já existe.

Finalizando, com relação à produção de unidades instrucionais ou iniciativas de ensino de programação, Medeiros, Wangenheim e Bergmann (2018) contribuem com a seguinte conclusão:

[...] Os resultados positivos nas pesquisas têm relação com as estratégias metodológicas escolhidas para a aplicação das práticas. Dentre os melhores resultados estão práticas voltadas para os interesses, os problemas do cotidiano dos alunos e a partir da formação de professores das instituições para participação e atuação na pesquisa [...].

Dessa nota-se forma, а relevância do desenvolvimento da proposta de ensino intitulada "Instituto de Hackers", uma vez que ela foi permeada por uma teoria de ensino-aprendizagem ativa, atividades contextualizadas condizente com vivenciadas pelos estudantes, em um contexto maior, que é a educação profissional e tecnológica contemporânea.

5. Conclusões

Com o intuito de estimular e avaliar o pensamento computacional em estudantes do Ensino Técnico Integrado ao Ensino Médio, de área não diretamente ligada à computação, por meio de uma oficina intitulada "Instituto de Hackers", notou-se que a proposta de ensino foi capaz de estimular o pensamento computacional nos estudantes de forma fácil e divertida, e que a metodologia oferecida foi agradável, de fácil entendimento e facilitou o processo de aprendizagem. Para muitos estudantes, esse foi o primeiro contato com a área de desenvolvimento de software, o que despertou o interesse e a curiosidade, visto que muitos relataram a necessidade de um aprofundamento maior da oficina ou um tempo maior de duração.

Quanto aos pilares do pensamento computacional, evidenciou-se, por meio de observação, uma grande importância da "abstração", em razão de sua relação direta com a aprendizagem situada, pois os estudantes apropriaram-se do problema, situaram-se com o contexto dele e, assim, puderam desenvolver melhor a solução necessária.

Apesar de o número de amostragem ser pequeno, 24 participantes, os resultados indicam que a proposta possui uma grande relevância. Ela mostrou-se também motivadora para novas pesquisas desenvolvimento e/ou aperfeiçoamento de novas unidades instrucionais, utilizando-se da linguagem de blocos e da teoria da aprendizagem situada.

Projetos futuros, relacionados com a pesquisa apresentada, tendem para a criação de produtos educacionais, com a utilização do pensamento computacional para o suporte ao ensino determinados assuntos em componentes curriculares específicos.

Além disso, a proposta de ensino criada pode ser aprimorada segmentando-se os conteúdos trabalhados e aprofundando mais em padrões utilizados, bem como gradualmente chegar a um nível de dificuldade maior dos exercícios propostos.

Em geral, a grande dificuldade dos estudantes era quanto à utilização do software, seja por representar uma novidade, seja pela falta de leitura das instruções da própria aplicação. Já a maior facilidade apresentada foi com relação à compreensão dos exemplos e das situações dos exercícios propostos, devido à teoria de aprendizagem utilizada.

Referências

ALMEIDA, E. G. de. Aprendizagem situada. Texto Livre: Linguagem e Tecnologia, v. 7, n. 1, 31 jul. 2014.

ARAUJO, R. M. de L.; FRIGOTTO, G. Práticas pedagógicas e ensino integrado. Revista Educação em Questão, v. 52, n. 38, p. 61-80, 2015.

ARMANDO, J. Integração do pensamento computacional no currículo da educação básica: diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. e-Curriculum, São Paulo, v. 14, n. 6, p. 34, 2016.

BATISTA, R. M.; HORTA, E. G.; FONSECA, A. R. Programação em blocos: impacto de um projeto de extensão executado em Escolas públicas de Diamantina/MG. Informática na educação: teoria & prática, v. 23, n. 2, 22 nov. 2020.

BORGES, L. G.; PAEZ, G. R.; BELTHER, J. M. Aprendizagem situada: análise de uma atividade situacional. In: ENCONTRO NACIONAL DE EDUCAÇÃO MATEMÁTICA, 12., 2016, São Paulo, p. 12. Disponível em:

http://www.sbem.com.br/enem2016/anais/pdf/4892_2 352_ID.pdf. Acesso em: 3 mar. 2021.

BRACKMANN, C. et al. Pensamento Computacional Desplugado: Ensino e Avaliação na Educação Primária Espanhola. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 6., 27 out. 2017, Recife, Pernambuco. Disponível em: http://www.brie.org/pub/index.php/wcbie/article/view/7487. Acesso em: 27 ago. 2018

BRASIL. Lei nº 11.892, de 29 de dezembro de 2008. Institui a Rede Federal de Educação Profissional, Científica e Tecnológica, cria os Institutos Federais de Educação, Ciência e Tecnologia, e dá outras providências. Diário Oficial da União, Brasília, DF: Presidência da República. Disponível em: http://www.planalto.gov.br/ccivil_03/_Ato2007-2010/2008/Lei/L11892.htm. Acesso em: 3 mar. 2021.

BRASIL. Ministério da Educação. Um novo modelo em educação profissional e tecnológica: concepções e diretrizes. Brasília, DF: Ministério da Educação; Instituto Federal de Educação Ciência e Tecnologia, 2010. Disponível em:

http://portal.mec.gov.br/index.php?option=com_docm an&view=download&alias=6691-ifconcepcaoediretrizes&category_slug=setembro-2010pdf&Itemid=30192. Acesso em: 3 mar. 2021.

CIAVATTA, M.; RAMOS, M. Ensino Médio e Educação Profissional no Brasil. Revista Retratos da Escola, v. 5, n. 8, p. 27-41, 2011.

DE JESUS, A.; BRITO, G. S. Concepção de ensinoaprendizagem de algoritmos e programação de

Elisângela Valevein Rodrigues & Leandro Delgado de Souza

computadores: a prática docente. In: ENINED -ENCONTRO NACIONAL DE INFORMÁTICA E EDUCAÇÃO, 1., 2009, p. 10. Disponível em: https://www.inf.unioeste.br/enined/2009/anais/enined/ A15.pdf. Acesso em: 3 mar. 2021.

FERNANDES, H. B.; SILVEIRA, I. F. A plataforma CODE.ORG online: desenvolvendo do pensamento computacional e a matemática. Revista Atlante: Cuadernos de Educación y Desarrollo, agosto 2017,p. 7. Disponível em:

https://www.eumed.net/rev/atlante/2017/08/matemati ca-pensamento-computacional.html. Acesso em: 3 mar. 2021.

FONTES, C. R.; SILVA, F. W. O. da. O ensino da disciplina linguagem de programação em escolas técnicas, v. 13, n. 2, 2008.

FORBELLONE, A. L. V.; EBERSPACHER, H. F. Lógica de programação: a construção de algoritmos e estrutura de dados. 3. ed. São Paulo: Prentice Hall, 2005.

FRIGOTTO, G. A relação da educação profissional e tecnológica com a universalização da educação básica. Educação & Sociedade, v. 28, n. 100, p. 1129-1152, out. 2007.

JUNQUEIRA, S. H. F. Ensino de lógica de programação com o uso de ferramenta visual voltada para dispositivos móveis com Android. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 36., João Pessoa, PB, de 3 a 6 de outubro de 2016, p. 16, 2016. Disponível em:

http://www.abepro.org.br/biblioteca/TN_STO_235_36 7_30052.pdf. Acesso em: 3 mar. 2021.

LAVE, J. A prática da aprendizagem. In: ILERIS, K. (org.). Teorias contemporâneas da aprendizagem. Porto Alegre: Penso, 2013. p. 235-245. MARIMON BOUCINHA, R. et al. Construção do pensamento computacional através do desenvolvimento de games. Renote, v. 15, n. 1, 28 jul. 2017.

MOSER, A.; SCHNEIDER, E. I.; MEDEIROS, L. F. de. A aprendizagem situada nas comunidades de prática: uma aproximação fenomenológica. Revista de Informática Aplicada, v. 9, p. 7, 2013.

PRETTO, N. Redes colaborativas, ética hacker e educação. Educação em Revista, v. 26, n. 3, p. 305-316, dez. 2010.

QUADROS, C. E. P. de; BICHO, A. de L.; ADAMATTI, D. F. A teoria das inteligências múltiplas contextualizada com educação, neurociência e pensamento computacional: uma revisão de literatura. Informática na Educação: Teoria & Prática, v. 23, n. 2, p. 21, maio/ago., 2020. Disponível em: https://seer.ufrgs.br/InfEducTeoriaPratica/article/view/1 03787. Acesso em: 3 mar. 2021.

RAMOS, J. L.; ESPADEIRO, R. G. Pensamento computacional na escola e práticas de avaliação das aprendizagens. uma revisão sistemática da literatura. 2015. p. 21. Disponível em: https://dspace.uevora.pt/rdpc/bitstream/10174/14227/

1/challenges%202015br.pdf. Acesso em: 3 mar. 2021.

RIBEIRO, L.; FOSS, L.; CAVALHEIRO, S. A. da. C.Entendendo o pensamento computacional. 2017. p. 19.

SAVIANI, D. Trabalho e educação: fundamentos ontológicos e históricos. Revista Brasileira de Educação, v. 12, n. 34, p. 152-165, abr. 2007.

SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. Diretrizes para ensino de computação na educação básica. Disponível em: https://www.sbc.org.br/educacao/diretrizes-para-

ensino-de-computacao-na-educacao-basica. Acesso em: 3 mar. 2021.

SOUZA, E. C. DE; YONEZAWA, W. M. Programação no ensino de matemáticautilizando Processing 2: um estudo dasrelações formalizadas por alunos do ensinofundamental com baixo rendimento emmatemática. Revista Brasileira de Ensino de Ciência e Tecnologia, v. 11, n. 1, p. 270-301, 2018.

VON WANGENHEIM, C. G. et al. dETECT: a model for the evaluation of instructional units for teaching computing in middle school. Informatics in Education, v. 16, n. 2, p. 301-318, 30 set. 2017.

VON WANGENHEIM, C.; RODRIGUES NUNES, V.; DOS SANTOS, G. D. Ensino de Computação com SCRATCH no Ensino Fundamental – um estudo de caso. Revista Brasileira de Informática na Educação, v. 22, n. 3, p. 115, 23 nov. 2014.

WEINTROP, D.; WILENSKY, U. Comparing block-based and text-based programming in high school computer science classrooms. ACM Transactions on Computing Education, v. 18, n. 1, p. 1-25, 27 out. 2017.

WING, J. Computational thinking. COMMUNICATIONS OF THE ACM, v. 49, n. 3, 2006.

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 de Ensino de Ciência e Tecnologia, v. 9, n. 2, 16 nov. 2016.