

INTRODUÇÃO AOS CONCEITOS DE LÓGICA DE PROGRAMAÇÃO COM A LINGUAGEM DE PROGRAMAÇÃO SCRATCH

Maria Clara dos Santos Vasconcelos¹
Maria Vitória dos Santos Vasconcelos²
Maria Amélia da Silva Costa³

RESUMO

A tecnologia vem ocupando mais espaço no dia a dia das pessoas, como uma forma de facilitar a vida e solucionar problemas, considerando tal fato, é posto que isolar o processo de ensino-aprendizagem destes avanços tecnológicos seria incoerente. Daqui a alguns anos, com o uso frequente de computadores, celulares, Smart TV's, e utensílios domésticos inteligentes, a demanda pela produção de softwares para estes dispositivos aumentará ainda mais e, por isto, saber programar será uma habilidade necessária, para não dizer básica, na educação, saindo do ensino superior e técnico e indo para o ensino básico. Visto isto, este trabalho tem como objetivo demonstrar a importância da introdução aos conceitos de lógica de programação com a linguagem de programação Scratch. Os principais referenciais teóricos para este artigo são a filosofia construtivista de Papert e alguns estudos sobre o uso da ferramenta Scratch no ensino de programação para jovens. Foram analisados os resultados de um minicurso, ministrado em três turmas dos anos finais do ensino fundamental, do 6º, 7º e 9º ano, sendo duas turmas de escola particular e uma de escola municipal, localizadas na cidade de Afogados da Ingazeira, no interior de Pernambuco. Foi realizado um estudo qualitativo durante as aulas do minicurso, e dentre os parâmetros analisados temos: o interesse e envolvimento com o tema proposto, e as dificuldades e soluções encontradas pelos alunos nas atividades práticas. Alguns dos resultados observados foram, em relação ao conteúdo, o entendimento de conceitos importantes de lógica de programação e a adaptação dos conteúdos apresentados para atender seus próprios objetivos, e em relação aos alunos, algumas variações nos níveis de interesse e produtividade entre as turmas.

Palavras-chave: Tecnologia, Scratch, Programação, Educação Básica, Construtivismo.

INTRODUÇÃO

A tecnologia vem ocupando mais espaço no dia a dia das pessoas, como uma forma de facilitar a vida e solucionar problemas. Acompanhando esse avanço, a demanda por profissionais com conhecimentos na área da computação só tende a aumentar, pois, quanto maior for o uso de computadores, celulares, Smart TV's, e utensílios domésticos inteligentes, maior será a necessidade da sociedade em recrutar profissionais com a capacidade de criar softwares para estes dispositivos.

1 Estudante do Curso de Licenciatura em Computação pelo IFPE campus Afogados da Ingazeira-PE. E-mail: mcsv@discente.ifpe.edu.br

2 Estudante do Curso de Licenciatura em Computação pelo IFPE campus Afogados da Ingazeira- PE. E-mail: mvsv@discente.ifpe.edu.br

3 Professora orientadora: Mestre em Ensino, Pedagoga, Bacharel em Comunicação Social e professora no IFPE campus Afogados da Ingazeira-PE. E-mail: maria.costa@afogados.ifpe.edu.br

Visto que, a educação possui entre outros objetivos, a missão de preparar o indivíduo para o mercado de trabalho e a vida em sociedade, não faz sentido isolar o processo educacional das mudanças e avanços desta, e por isso, saber programar e ter conhecimentos em informática, serão habilidades necessárias, para não dizer básicas, na educação, saindo do ensino superior e técnico e indo para o ensino básico.

Além disto, ensinar o pensamento computacional pode ajudar que o aprendiz desenvolva diversas habilidades importantes para a vida em sociedade, como resolução de problemas, abstração e criatividade, auto-expressão e colaboração. O pensamento computacional, trata-se, portanto, de uma estratégia usada para desenhar soluções e solucionar problemas de maneira eficaz tendo a tecnologia como base, e pode ser ensinado no estudo de lógica de programação desde as séries iniciais da Educação Básica.

O objetivo deste artigo é discutir sobre a importância do ensino de lógica de programação na Educação Básica e da utilização de metodologias ativas que possam contribuir para uma aprendizagem efetiva desses conceitos.

O trabalho foi desenvolvido com o intuito de trazer algumas contribuições acerca do ensino de programação para crianças e jovens, tendo com base as nossas experiências ao ministrar um minicurso de introdução à programação com a linguagem Scratch.

O referido minicurso foi realizado durante o período de Agosto de 2022, no laboratório de informática do Instituto Federal de Pernambuco – Campus Afogados da Ingazeira, contando com a presença de três turmas dos anos finais do ensino fundamental. Para cada turma foi ministrado 2h horas de aula do minicurso de Introdução à Programação, divididas entre parte teórica, apresentação da ferramenta Scratch e atividade prática.

Com base nos resultados observados durante o minicurso, foi realizada uma pesquisa descritiva, qualitativa, e do tipo relato de experiência, onde foram apresentados os aspectos gerais comuns às três turmas que participaram, como o desempenho, atenção, envolvimento com o tema, e depois foram descritos os aspectos particulares de cada turma, que apresentaram diferentes perspectivas entre si.

Como base teórica da nossa pesquisa, utilizamos como embasamento principal as ideias construtivistas de Seymour Papert (1980), criador da linguagem de programação LOGO, que assim como a linguagem Scratch, foi criada no MIT e possui uma filosofia muito semelhante. Também foi utilizada a pirâmide da aprendizagem de William Glasser (1925-2013), para demonstrar que a metodologia proposta no minicurso pode promover uma aprendizagem efetiva dos conceitos de lógica de programação e pensamento computacional.

Acreditamos que é de extrema importância discutir e pensar sobre a educação que devemos promover para futuro, a qual deve acompanhar as mudanças da sociedade e preparar os alunos para os novos desafios da era da tecnologia. A metodologia utilizada ao ministrar o minicurso e os resultados analisados podem contribuir como base para outros experimentos de aulas e cursos de introdução à programação, afim de promover uma aprendizagem focada no aprendiz e que seja realmente significativa.

METODOLOGIA

O presente estudo trata-se de uma pesquisa descritiva, qualitativa, do tipo relato de experiência. Pesquisas qualitativas, segundo Goldenberg (2005), apresentam descrições detalhadas sobre situações sociais e indivíduos, com o objetivo de compreendê-los pela perspectiva do autor.

O estudo foi realizado a partir da vivência docente em um minicurso sobre introdução à programação com a linguagem Scratch, realizado no Instituto Federal de Pernambuco, campus Afogados da Ingazeira – PE, Brasil, durante o período de Agosto de 2022. O minicurso foi realizado com três diferentes turmas dos anos finais do ensino fundamental, do 6º, 7º e 9º ano, sendo duas delas de escola particular e uma da rede municipal. Ao todo foram 90 alunos, com idades entre 11 e 14 anos.

Para cada turma, foram ministradas 2h de aula, divididas em três etapas: Introdução ao conceito de programação e linguagem de programação; Apresentação da ferramenta Scratch; Atividade prática.

Na primeira etapa foram trabalhados os conceitos teóricos sobre as linguagens de programação e onde elas podem ser utilizadas, ressaltando a importância de entender os conceitos de programação nos dias atuais, e também uma breve introdução sobre o que é a ferramenta Scratch e para que pode ser utilizada.

A segunda etapa foi focada na apresentação da ferramenta Scratch, em como ela funciona e quais recursos ela apresenta. Nesse momento a aula aconteceu através da intercalação entre teoria e prática, onde os blocos de comandos eram apresentados e explicados aos alunos e em seguida cada um aplicava o conteúdo apresentado em seu próprio projeto da maneira que preferisse.

A última etapa foi a construção de um jogo de pular sobre obstáculos, onde eram descritos os passos necessários e os alunos iam implementando em seus próprios projetos,



sendo auxiliados quando surgiam dúvidas e erros, e ao final da aula, cada um tinha o jogo completo em seu computador, com personagem, cenário e animações de sua preferência.

REFERENCIAL TEÓRICO

As tecnologias, em geral, são construídas com a finalidade de melhorar ou facilitar as atividades humanas, dessa forma, elas tendem a ser tornar cada vez mais presentes no dia a dia das pessoas. Os computadores, em particular, sejam desktops, notebooks, celulares, etc, estão presentes nos mais diversos contextos da nossa vida, seja no trabalho, nos estudos ou no lazer.

Para usufruir desses recursos, você não necessariamente precisa saber como um computador funciona internamente ou como programá-lo, porém, para que essas inovações possam acontecer e para que essas tecnologias evoluam, é necessário que existam profissionais capazes de trabalhar com elas, como demonstra uma pesquisa feita pelo INEP: “A estimativa é de que a demanda por profissionais qualificados na área de TI cresça exponencialmente entre 2021 e 2025, em 797 mil (INEP, 2019)”.

Ademais, um levantamento feito pela empresa de consultoria de recursos humanos, Robert Half, aponta a área de tecnologia como uma das com mais alta demanda para o mercado em 2023. (G1, 2022).

Dentro deste contexto, é posto que isolar o processo de ensino-aprendizagem destes avanços tecnológicos seria incoerente, pois, com o uso frequente de computadores, celulares, Smart TV's, e utensílios domésticos inteligentes, a demanda pela produção de softwares para estes dispositivos aumentará ainda mais e, por isto, saber programar será uma habilidade necessária, para não dizer básica, na educação, saindo do ensino superior e técnico e indo para o ensino básico.

O pensamento computacional, é entendido por Wing (2006) um conjunto de habilidades para resolução de problemas desenvolvidas ao estudar conteúdos provenientes da Ciência da Computação. Então, mesmo que o estudante não siga por uma carreira na área de tecnologia, o aprendizado de pensamento computacional, que está relacionado à programação, segundo Rodrigues, et. al (2015), pode contribuir para o desenvolvimento de habilidades como raciocínio lógico, capacidade de abstração e resolução de problemas, as quais são essenciais para a formação básica.



Em relação ao processo de ensino-aprendizagem, dentro do contexto da era da informática, é preciso pensar não apenas em o quão é importante inserir o ensino de computação e programação no ensino básico, como também pensar em como deve acontecer esse processo.

Segundo o matemático e professor do Instituto de Tecnologia de Massachusetts (MIT), Seymour Papert (1980), apesar dos grandes avanços das Tecnologias da Informação e Comunicação (TICs), e do seu grande potencial para ampliar as formas de ensinar e aprender, as escolas pouco exploram esses recursos, se limitando a reproduzir antigas metodologias de ensino, utilizando essas tecnologias apenas como forma de transmissão do conhecimento. O autor critica o uso destas tecnologias na educação onde não se alteram o modo atual de ensinar, um exemplo disso são professores que usam aparelhos como projetores apenas para substituir a lousa e o giz, e não exploram essa mídia com novas abordagens de ensino.

Papert, em seu livro “Mindstorms: Children, Computers and Powerful Ideas” (1980), pontua a importância do envolvimento e interesse do aprendiz ao objeto estudado, onde a paixão e interação com ele influencia diretamente no processo de aprendizagem. Com isso, ele critica a maneira com que muitas escolas utilizam os computadores em sala de aula, onde ele se limita apenas a um transmissor de informações, desperdiçando todo o seu potencial de se moldar e se adequar aos gostos e ritmo de aprendizagem da criança.

Seguindo esse raciocínio, Papert (1980) desenvolveu uma linguagem de programação chamada LOGO, voltada para crianças, jovens ou até adultos. Essa linguagem estimula uma forma de aprendizado onde o estudante age como sujeito ativo na construção do conhecimento. Ainda no livro “Mindstorms: Children, Computers and Powerful Ideas” (1980) o autor afirma que no ambiente LOGO a criança é colocada numa posição essencialmente ativa, programando o computador “E ao ensinar o computador a ‘pensar’, começa a explorar o seu próprio pensamento”. (1980, p.35).

A filosofia LOGO tem influência direta do Construtivismo de Jean Piaget, o qual Papert chegou a conhecer. O construtivismo é uma abordagem pedagógica que defende que o conhecimento é resultado da construção pessoal do aluno; e o professor age apenas como um mediador do processo ensino-aprendizagem. (FOSSILE, 2010).

Ainda sob uma visão construtivista de ensino, temos como tema principal deste artigo a linguagem de programação Scratch, criada em 2007, também no MIT, pelo Media Lab. O Scratch é uma linguagem de programação voltada para os jovens, que possui uma interface

visual simples, a qual permite que eles criem histórias, animações e até jogos, estimulando o desenvolvimento do raciocínio lógico e computacional.

Segundo o site do próprio Scratch (2022), a ferramenta “promove habilidades de pensamento computacional e resolução de problemas; ensino e aprendizado criativo; auto-expressão e colaboração; e equidade na computação (Tradução própria)”. A linguagem é bastante intuitiva, e incentiva o aprendiz a explorar seus recursos e construir projetos com base nos seus gostos e experiências, assim, eles aprendem com base na tentativa e erro, encaixando blocos de comandos em sequência e observando o comportamento dos personagens na cena.

Abordagens que envolvem o aluno de maneira ativa na construção do próprio conhecimento promovem uma aprendizagem mais significativa, onde, segundo a teoria de William Glasser (1925-2013), a construção do conhecimento não deve se resumir à memorização mecânica de conteúdo, a aprendizagem ativa aumenta as possibilidades de o aprendiz assimilá-lo.

Na figura 1 podemos observar que o modelo de pirâmide proposto por Glasser possui 7 níveis, onde de cima para baixo são representadas, em ordem crescente, a porcentagem de aprendizagem de um determinado conteúdo de acordo com a maneira que o indivíduo interage com ele.

Figura 1 - Pirâmide de Aprendizagem de Willian Glasser



Fonte 1 - Junior, Souza e Silva (2019, p. 147).

O topo da pirâmide representa a porcentagem de aproveitamento do conteúdo através da leitura, que é de apenas 10%; No segundo nível temos que escutar proporciona uma aprendizagem de 20%; No terceiro, ver ou observar proporciona 30% de aprendizagem; Logo após temos ver e escutar proporcionando 50% de aprendizagem; O próximo nível possui atividades relacionadas a interação entre os indivíduos, como conversar, perguntar, repetir, numerar, reproduzir, definir e debater, que correspondem a 70% de aprendizagem; O sexto nível diz respeito a atividades mais práticas, onde se aplica o conteúdo de alguma forma, essas atividades incluem escrever, interpretar, expressar, revisar, identificar, comunicar, ampliar, demonstrar, praticar e diferenciar, as quais correspondem a 80% de aprendizagem; O último nível é composto por atividades como explicar, resumir, estruturar, definir, generalizar, elaborar e ilustrar, que proporcionam 95% de aprendizagem do conteúdo.

Por isso, a ferramenta *Scratch*, se for utilizada da maneira correta no contexto de ensino/aprendizagem, pode promover uma melhor assimilação e interesse por conteúdos relacionados a programação, já que permite o aprendiz criar, estruturar, explicar, demonstrar, praticar, elaborar e ilustrar suas ideias utilizando conceitos de programação.

RESULTADOS E DISCUSSÃO

Os resultados analisados são de natureza empírica, obtidos a partir das nossas próprias experiências e observações enquanto ministrávamos o minicurso em três turmas dos anos finais do ensino fundamental. Inicialmente, essas observações serão divididas entre gerais e específicas de cada uma das turmas.

Em todas as turmas, foi observado que, a atenção dos alunos durante o minicurso foi aumentando gradativamente a medida que o conteúdo migrava da parte teórica para a parte prática; os alunos inicialmente apenas reproduziam em seus projetos o que era explicado por nós, mas logo começaram a adaptar e modificar os conteúdos para atender aos seus próprios interesses; Os estudantes começaram a explorar o Scratch em busca de novas funcionalidades.

Durante a primeira etapa do minicurso de introdução aos conceitos de lógica de programação com a linguagem Scratch, foi utilizada uma abordagem teórica para explicar os conceitos necessários, afim de contextualizar o aluno sobre o que são linguagens de programação, algoritmos e o Scratch, conhecimentos esses que eram necessários para que eles entendessem o que e para que estavam aprendendo aquilo.

Por ser uma introdução mais teórica, no início do minicurso, foi notado que, em todas as turmas, a atenção estava mais dispersa, muito mais voltada para o ambiente novo, os computadores do laboratório e as conversas com seus colegas. Ao chegar na segunda etapa do minicurso, a apresentação da ferramenta Scratch, foi observado que a atenção dos alunos foi aumentando, pois foram atraídos pela interface amigável e divertida da ferramenta, e a curiosidade sobre as possibilidades do que se podia criar foi aumentando cada vez mais. A última etapa do minicurso, que foi o desenvolvimento do jogo, foi a que mais chamou atenção de todos, pois, antes de começarem a programá-lo, mostramos já o resultado final do que alcançariam se seguissem todas as etapas, e, a partir daí, a atenção de todos passou a ser quase que exclusiva da explicação demonstrando o passo a passo e da ferramenta.

O segundo ponto observado foi que, inicialmente, como os alunos ainda não sabiam utilizar os recursos disponibilizados dentro da plataforma do Scratch, eles apenas adicionavam os blocos de código em seus projetos da mesma forma que explicávamos, mas a medida que eles foram entendendo como cada recurso funcionava, muitos passaram a tomar a liberdade de modificá-los para que se encaixassem melhor em seus projetos individuais. Como exemplo podemos citar que, durante a construção do jogo de pular obstáculos, foi apresentado um recurso que permitia que um objeto fosse iniciado no canto direito da tela e deslizasse em direção ao lado esquerdo, mas ao entender como esse comando funcionava, um aluno foi capaz de fazer com que o objeto que ele escolheu, que era uma maçã, inicialmente caísse de uma árvore e depois começasse a deslizar para a esquerda, adaptando um dos comandos apresentados para atender ao seu próprio estilo e objetivo.

O terceiro ponto observado foi que quando os alunos aprenderam como o Scratch funcionava e visualizaram os recursos apresentados durante a explicação, alguns já começaram a explorar funcionalidades diferentes de forma individual, buscando recursos interessantes que pudessem ser aplicados aos seus projetos.

A partir de agora iremos abordar os aspectos observados em cada turma de maneira isolada, apesar das turmas terem apresentado desempenhos muito semelhantes, algumas particularidades foram observadas e serão pontuadas a seguir em ordem cronológica das aulas ministradas:

1. A primeira turma foi do 9º ano do ensino fundamental, da rede municipal de educação, com 30 alunos, de idades entre 13 e 14 anos. Nesta turma, os alunos estavam mais agitados e um pouco menos interessados do que os das demais turmas. Devido a essa falta de interesse, principalmente na apresentação dos conceitos iniciais, eles foram os que apresentaram mais dificuldades em acompanhar as atividades práticas. Apesar das dificuldades, todos os alunos conseguiram concluir a etapa final, o jogo.
2. A segunda turma foi do 6º ano do ensino fundamental, de uma escola particular da cidade Afogados da Ingazeira, com 30 alunos, de idades entre 11 e 12 anos. Para esta turma, alguns ajustes na estrutura da aula e na explicação foram feitos, com base nas dificuldades da turma anterior. Os alunos dessa turma demonstraram muito mais interesse nas explicações e realizaram as atividades com mais facilidade, explorando mais a ferramenta e expressando mais criatividade em seus projetos, concluindo o jogo com maior facilidade. Alguns alunos, ao final do minicurso, demonstraram interesse em continuar explorando o Scratch em casa e no estudo de programação.
3. A terceira turma foi do 7º ano do ensino fundamental, de uma escola particular da cidade Afogados da Ingazeira, com 30 alunos, de idades entre 12 e 13 anos. Nessa turma não foi observado tanto interesse e empolgação como na turma anterior, mas no geral, prestaram atenção aos conceitos iniciais e apresentaram menos dificuldades que a primeira turma, concluindo o jogo normalmente.

Ao analisar os resultados do minicurso destacamos que as principais dificuldades encontradas pelos alunos aconteceram devido a não terem aprendido tão bem os conceitos iniciais, gerando dificuldades em entender como funcionavam alguns comandos mais complexos. Quanto a prática pedagógica identificamos que as dificuldades encontradas por nós na realização do minicurso foram principalmente em manter a turma focada durante as explicações, onde muitas vezes os alunos se distraíam com os recursos da ferramenta enquanto eram explicados conceitos importantes, e também tirar as dúvidas individuais de cada aluno, num espaço de tempo limitado.

Como proposta para solucionar os problemas descritos anteriormente, acreditamos que a maioria das dificuldades encontradas durante a realização dos minicursos poderiam ser resolvidas através do aumento da duração de cada minicurso e adição de um ou mais

monitores na turma, para explicar com mais detalhes cada função, dar mais liberdade para os alunos explorarem a ferramenta e seus recursos e dar mais assistência individual aos alunos com mais dificuldades.

CONSIDERAÇÕES FINAIS

Ao final deste artigo, podemos concluir, que a educação e a formação profissional devem acompanhar as mudanças da sociedade e os avanços tecnológicos. Também, introduzir o ensino de programação no ensino básico pode contribuir não apenas para os alunos acompanharem este mercado emergente como também que eles desenvolvam habilidades úteis para sua vida pessoal e profissional, como resolução de problemas, abstração e criatividade, auto-expressão e colaboração.

Além disso, com base nas observações feitas durante a aplicação do minicurso, inferimos que a ferramenta Scratch pode promover uma aprendizagem significativa dos conceitos básicos de lógica de programação, envolvendo o aluno numa abordagem atrativa, divertida, onde ele é o principal agente da construção do conhecimento, configurando ela como uma boa alternativa para ensinar pensamento computacional.

Em suma, os resultados apresentados nesse artigo foram baseados em nossas próprias experiências e conclusões a respeito do minicurso ministrado, e com base na nossa metodologia, existem outras muitas possibilidades de aplicações da introdução aos conceitos de lógica de programação em turmas do ensino fundamental, as quais podem levar mais tempo do que apenas um minicurso, e podem trazer outros resultados e conclusões significativas a respeito do ensino-aprendizagem de programação para crianças e adolescentes.

REFERÊNCIAS

BRASSCOM, BACEN, IDC, Relatórios Financeiros das Estatais, RAIS e Caged, Novo Caged, Censo do Ensino Superior (INEP, 2019).

CAVALLINI, Marta. Pesquisa mostra cargos em alta e média salarial em oito áreas em 2023. G1, 27 set. 2022. Disponível em: <<https://g1.globo.com/trabalho-e-carreira/noticia/2022/09/27/pesquisa-mostra-cargos-em-alta-e-media-salarial-em-oito-areas-em-2023.ghtml>> .Acesso em: 06 out. 2022.

CYSNEIROS, P. G. PAPERT, Seymour. A máquina das crianças: repensando a escola na era da informática. Revista Entreideias: Educação, Cultura E Sociedade. 12 ed. 2008.



FOSSILE, Dieysa K. Construtivismo versus sociointeracionismo: uma introdução às teorias cognitivas. Revista Alpha, Patos de Minas, 2010.

GOLDENBERG, M. A arte de pesquisar: como fazer pesquisa qualitativa em Ciências Sociais. 5a. ed. Rio de Janeiro: Record, P. 53, 2005.

JUNIOR, J. M. A.; SOUZA, L. P.; SILVA, N. L. C. (orgs). Metodologias ativas: práticas pedagógicas na contemporaneidade. Campo Grande: Editora Inovar, 2019.

PAPERT, S. Mindstorms. Children, Computers and Powerful Ideas. New York (EE. UU.), Basic Ideas, 1980.

RODRIGUES, R. S. et al. Análise dos efeitos do Pensamento Computacional nas habilidades de estudantes no ensino básico: um estudo sob a perspectiva da programação de computadores. SBIE, 2015.

SCRATCH. About Scratch. Scratch Documentation Site. Disponível em: <<https://scratch.mit.edu/about>>. Acesso em: 06 outubro 2022.

WING, J. M. Computational thinking. Communications of the ACM, V. 49, P. 33–35, 2006.