

O uso de *softwares* educativos na aprendizagem de lógica de programação para crianças do ensino fundamental I

The use of educational software in the learning of logic programming for elementary school children

Isabela Duarte Moreno 

Fatec Praia Grande
duartemorenoisabela@gmail.com

Simone Maria Viana Romano 

Fatec Praia Grande
simone.viana@fatecpg.edu.br

RESUMO

É evidente que com a ascensão da tecnologia, torna-se crescente a frequência de seu uso nos mais variados ambientes. No âmbito educacional, nota-se o crescimento deste cenário de integração tecnológica, inclusive com sua utilização voltada à obtenção de conhecimentos através de jogos educacionais ou plataformas digitais. Estes dois fatores, em conjunto, resultam no aumento de iniciativas para o aprendizado da Tecnologia da Informação (TI) e consequentemente o ensino de algoritmos e lógica de programação, uma vez que trata-se da base para muitos conhecimentos em TI. O estudo apresentado tem foco em demonstrar como a lógica de programação pode ser uma importante ferramenta de aprendizado para crianças através dos principais *softwares* educativos e ferramentas para esta finalidade, iniciando com uma pesquisa bibliográfica tendo base em pesquisadores da educação e desenvolvimento do pensamento das crianças. Ademais, o artigo aborda propostas de inserção do Pensamento Computacional na educação, evidenciando também os resultados de um estudo de caso através de uma aula experimental com 17 alunos do 4º e 5º ano do Ensino Fundamental I. Por meio de uma análise descritiva dos resultados obtidos, foi possível avaliar o entendimento das crianças sobre os conceitos ensinados a partir do ponto de vista das mesmas, como também observar o desempenho satisfatório em unanimidade ao utilizarem ferramentas elucidadas no estudo, visto que todas concluíram as atividades propostas.

PALAVRAS-CHAVE: Lógica de Programação. Educação. Crianças. Tecnologia.

ABSTRACT

It is evident that with the advancement of technology, the frequency of its use in the most varied environments is increasing. In the educational field, we can see the growth of this scenario of technological integration, including its use aimed at obtaining knowledge through educational games or digital platforms. These two factors, together, result in the increase of initiatives for the learning of Information Technology (IT) and consequently the teaching of algorithms and logic programming, since it is the basis for a lot of knowledge in IT. The presented study focuses on demonstrating how logic programming can be an important learning tool for children through the main educational software and tools for this purpose, starting with a bibliographic research based on researchers in the education and development of children's thinking. In addition, the article approaches proposals for the insertion of Computational Thinking in education, also showing the results of a case study with an experimental class with 17 students from the 4th and 5th year of Elementary School. Through a descriptive analysis of the results obtained, it was possible to analyze the children's understanding of the concepts taught from their point of view, as well as observing the satisfactory performance in unanimity when using tools elucidated in the study, since all of them completed the proposed activities.

KEY-WORDS: Logic Programming. Education. Children. Technology.

INTRODUÇÃO

Atualmente, nota-se que a abrangência do uso da tecnologia nos mais diversos contextos vem tomando maiores proporções a cada dia, em decorrência do seu constante avanço. Dito isso, o convívio dos jovens com a Tecnologia da Informação (TI), com o passar dos anos, fez com que fossem perceptíveis as mudanças comportamentais em relação ao desenvolvimento do pensamento lógico desta geração.

O uso da *internet* para fins educacionais e recreativos faz-se gradativamente mais comum e, em alguns aspectos, pode-se dizer que indispensável. Sendo assim, percebe-se então a expansão de incentivos educacionais com foco em alcançar o público da educação básica e ampliar as oportunidades voltadas ao mesmo. Não obstante, as iniciativas baseadas na aprendizagem de fundamentos da área de desenvolvimento de *software* vêm apresentando considerável ascensão, tendo em vista o crescimento dos investimentos em TI.

Essa perspectiva não se restringe apenas aos dias atuais – o educador e matemático Seymour Papert (1980, p. 115, tradução nossa)¹, por exemplo, discorreu: “Em salas de aula tradicionais, os professores tentam trabalhar colaborativamente com crianças, mas geralmente o próprio material não gera problemas de pesquisa de forma espontânea”. Segundo ele, com o uso do computador, o professor e o estudante podem ter uma colaboração intelectual de forma real, compartilhando os problemas e experienciando como resolvê-los, fator este que colabora com a aprendizagem.

Desta forma, o presente estudo visa explorar o crescente surgimento de ferramentas, cursos e incentivos relacionados ao ensino de programação para crianças na faixa etária dos anos finais do Ensino Fundamental I, tendo em vista a integração entre tecnologia e educação, que pode resultar no despertar do interesse deste público nesta área, estimular sua criatividade e seu raciocínio lógico, principalmente se os conceitos forem abordados eficazmente, através de um currículo gradativo de aprendizagem e com o uso de *softwares* educacionais para tal fim, que serão demonstrados no decorrer deste estudo.

O objetivo geral do estudo baseia-se em evidenciar quais ferramentas podem ser utilizadas para o ensino de programação para crianças e os resultados desta abordagem. Os objetivos específicos foram definidos de forma a: descrever, com base em estudos anteriormente fundamentados, como a lógica de programação pode auxiliar nas habilidades cognitivas infantis; demonstrar como o Pensamento Computacional pode ser aplicado na

¹ No original: *In traditional schoolrooms, teachers do try to work collaboratively with children, but usually the material itself does not spontaneously generate research problems.*

educação das crianças através de um currículo gradativo proposto; conhecer os principais *softwares* educativos e ferramentas que podem ser utilizadas no ensino da programação para crianças; validar as informações anteriormente citadas com uma pesquisa com as crianças do fundamental I em que será avaliado o aprendizado e interesse deste público na programação.

Deste modo, de início, serão elucidados estudos de dois dos principais pensadores do desenvolvimento cognitivo infantil, através do procedimento de pesquisa bibliográfica que, segundo Lakatos e Marconi (2003), abrange toda bibliografia pública em relação ao tema de estudo, propiciando examinar um tema sob novo enfoque.

Além disso, a pesquisa está fundamentada na metodologia de estudo de caso que, segundo Gil (2002), possui diferentes propósitos, dentre estes, explorar situações da vida real onde os limites não estão definidos claramente, formulando hipóteses e teorias, proporcionando uma visão global do problema e fatores que o influenciam. À vista disso, serão evidenciados os resultados de uma aula experimental ministrada para alunos do 4º e 5º ano do Ensino Fundamental I, no qual conceitos introdutórios de lógica de programação foram ensinados e os alunos, ao final, responderam a uma pesquisa em formato de questionário.

Será elaborada uma análise descritiva das respostas, tanto das perguntas fechadas, ou seja, cuja escolha baseia-se entre opções “sim” ou “não”, como também das perguntas de avaliação, onde as respostas sugeridas são quantitativas por meio de uma escala com vários graus de intensidade, crescente ou decrescente, para um mesmo item (LAKATOS; MARCONI, 2003).

1. A LÓGICA DE PROGRAMAÇÃO COMO FERRAMENTA DE APRENDIZADO

Devido ao desenvolvimento acelerado da tecnologia nos dias atuais, observam-se as crescentes iniciativas voltadas à aprendizagem da área de TI. Principalmente quando há enfoque no desenvolvimento de *software*, é notório que um conceito indispensável para o início destes estudos é a lógica de programação, da qual torna-se base do aprendizado destes conceitos. Uma declaração do ex-presidente dos Estados Unidos, Barack Obama, evidencia estes incentivos: “Não só compre um *videogame* novo, faça um. Não só baixe um novo aplicativo, ajude a desenvolvê-lo. Não só jogue em seu celular, programe-o.” (OBAMA, 2013, tradução nossa)².

² No original: *Don't just buy a new videogame, make one. Don't just download the latest app, help design it. Don't just play on your phone, program it.*

Sendo assim, a existência de projetos como a organização sem fins lucrativos Code.org, por exemplo, que se dedica a expandir o acesso à ciência da computação nas escolas para todas as diferentes faixas etárias e seus cursos já são utilizados por um milhão de professores com alcance internacional (CODE.ORG, c2022), tende cada vez mais a expandir.

Contudo, os estudos sobre a utilização da programação como ferramenta de aprendizado para desenvolvimento do raciocínio lógico, especialmente voltado ao público infantil, não tiveram início somente nas últimas décadas – em 1980, Seymour Papert já abordava a computação como parte integrante do ensino nas escolas e suas possíveis aplicabilidades em sua obra “*Mindstorms: Children, computers and powerful ideas*”, considerando também conceitos estudados por Jean Piaget referentes ao desenvolvimento cognitivo infantil.

1.1 O RACIOCÍNIO LÓGICO NA INFÂNCIA E O CONCEITO DE “APRENDER FAZENDO”

Depreende-se da existência de diversos estudos voltados ao entendimento do desenvolvimento do raciocínio lógico na infância e a evolução do aprendizado das crianças em diferentes idades. Os estudos do epistemólogo suíço Jean Piaget, precursor do Construtivismo, eram baseados na construção do conhecimento e abordavam sobre os estágios de desenvolvimento cognitivo.

Segundo Piaget, o estágio operatório-concreto compreende dos 7 aos 11 anos, em que a criança constitui da lógica e estruturas operatórias concretas instrutivas para as operações lógicas, ao contrário do estágio operatório-formal, a partir dos 11 anos, onde as crianças passam a “versar sobre enunciados verbais (proposições), [...] não mais exclusivamente sobre objetos” (PIAGET, 1999, p. 107), e então sua lógica formal aplica-se a qualquer conteúdo.

Ou seja, o pensamento das crianças da faixa etária no contexto dos alunos pertencentes ao Ensino Fundamental I no Brasil, não se estrutura através de um entendimento verbal, mas sobre as classificações, relações e ações sobre objetos, bem como a compreensão de operações que possuem reversão, como a adição e subtração, seriação crescente e decrescente, etc.

Seymour Papert, no que lhe concerne, foi referência como um dos pioneiros no estudo da inteligência artificial, professor do Massachusetts Institute of Technology (MIT) e precursor do Construcionismo, teoria com alguns de seus ideais baseados no Construtivismo de Piaget, mas difere-se na utilização da tecnologia no ambiente educacional e não restringe a evolução das etapas do aprendizado por determinação biológica, mas também pela ideia de “aprender fazendo”. Segundo Papert (1980, p.21), Piaget conduz distinção entre o pensamento

operatório-concreto e o operatório-formal consoante a idade da criança. Já Papert acredita que o computador pode tornar-se uma ferramenta educacional que altere esta classificação.

[...] minha conjectura é que o computador pode concretizar (e personalizar) o formal. Visto sob esta luz, não é apenas uma poderosa ferramenta educacional. É único em nos fornecer os meios para abordar o que Piaget e muitos outros vêem como o obstáculo superado na passagem do pensamento de criança para adulto. Acredito que isso pode nos permitir mudar a fronteira que separa classificação concreta e formal. Conhecimento que era acessível apenas por processos formais pode agora ser abordado de forma concreta. (PAPERT, 1980, p. 21, tradução nossa).³

Ademais, Papert também foi o criador da linguagem de programação LOGO, voltada à educação com foco no desenvolvimento do pensamento lógico e aprendizagem infantil, cujas principais características serão abordadas neste estudo. Deste modo, o aprendizado torna-se mais eficiente quando o aluno pode experienciá-lo com seus erros e acertos.

Compartilhando o problema e a experiência de resolvê-lo permite que uma criança aprenda com um adulto, não "fazendo o que o professor diz", mas "fazendo o que o professor faz". E uma das coisas que o professor faz é perseguir um problema até que seja completamente compreendido (PAPERT, 1980, p. 115, tradução nossa).⁴

1.2 O PENSAMENTO COMPUTACIONAL NA BASE NACIONAL COMUM CURRICULAR

A Base Nacional Comum Curricular (BNCC) é o documento que define as “aprendizagens essenciais” a serem aplicadas aos alunos através de competências e habilidades. Nos últimos anos, vêm-se discutindo a inserção do Pensamento Computacional na educação. De acordo com a BNCC, o conceito “envolve as capacidades de compreender, analisar, definir, modelar, resolver, comparar e automatizar problemas e suas soluções, de forma metódica e sistemática, por meio do desenvolvimento de algoritmos” (BRASIL, 2018, p. 474).

O Centro de Inovação para a Educação Brasileira (CIEB) propôs, em 2018, um currículo de referência gradativo com orientações para as redes de ensino que pretendem incluir computação em suas bases curriculares. Separado por etapas da educação, eixos e conceitos,

³ No original: [...] my conjecture is that the computer can concretize (and personalize) the formal. Seen in this light, it is not just another powerful educational tool. It is unique in providing us with the means for addressing what Piaget and many others see as the obstacle which is overcome in the passage from child to adult thinking. I believe that it can allow us to shift the boundary separating concrete and formal. Knowledge that was accessible only through formal processes can now be approached concretely.

⁴ No original: Sharing the problem and the experience of solving it allows a child to learn from an adult not "by doing what teacher says" but "by doing what teacher does." And one of the things that the teacher does is pursue a problem until it is completely understood.

descreve os conteúdos propostos e as respectivas habilidades e competências da BNCC. No 4º ano do Ensino Fundamental I, no eixo de Pensamento Computacional, dentro do conceito de “algoritmos”, propõe-se a execução de algoritmos com operadores lógicos e relacionais. No 5º ano, é proposto em “algoritmos” a utilização de algoritmos de repetição e em “reconhecimento de padrões”, a introdução ao conceito de funções (RAABE; BRACKMANN; CAMPOS, 2018).

2. INTEGRAÇÃO ENTRE TECNOLOGIA E EDUCAÇÃO

No ambiente educacional, a integração com a tecnologia está em constante crescimento. Atualmente, o município de Praia Grande, no estado de São Paulo, por exemplo, demonstra este fato, visto que desde 2014 dispõe de *tablets* e lousas digitais nas escolas públicas municipais. Esta ação resultou em efeitos positivos para o ensino na cidade – segundo a Prefeitura de Praia Grande (2020), em 2017, o município obteve o maior crescimento no Índice de Desenvolvimento da Educação Básica (Ideb) entre as cidades da Baixada Santista, atingindo a nota 6,4 e ultrapassando assim a meta estabelecida pelo Ministério da Educação (MEC).

Notou-se uma ascensão desta integração desde o início da pandemia do Covid-19, visto que com as aulas presenciais suspensas, o ensino remoto tornou-se parte da rotina das crianças, jovens e adultos de todo o mundo. No início o isolamento social, o Núcleo de Informação e Coordenação do Ponto BR registrou por meio do projeto IX.br um volume de tráfego de rede inédito no Brasil – 13,5 *terabits* por segundo (NIC.BR. CETIC.BR. CGI.BR, 2020). O Estado de São Paulo, por exemplo, utilizou plataformas como o Centro de Mídias da Educação de São Paulo (CMSP) para adaptação à nova modalidade de ensino, no qual permite que estudantes acessem aulas ao vivo, videoaulas e conteúdos pedagógicos gratuitamente (SÃO PAULO, 2020), contribuindo para o aumento progressivo deste contato com a tecnologia.

Mesmo com esta crescente integração, a desigualdade social é destaque no contexto da educação brasileira. Segundo o Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP), o Censo Escolar de 2020 revelou que “quando se trata do ensino fundamental, a rede escolar dos municípios, maior ofertante também nessa etapa de ensino, é a que tem a menor capacidade tecnológica” (FERRARI, 2021). Ou seja, para a continuidade da inserção da tecnologia no ensino amplamente, é necessário maior investimento em recursos na educação, principalmente no ensino público.

3. FERRAMENTAS E SOFTWARES EDUCATIVOS

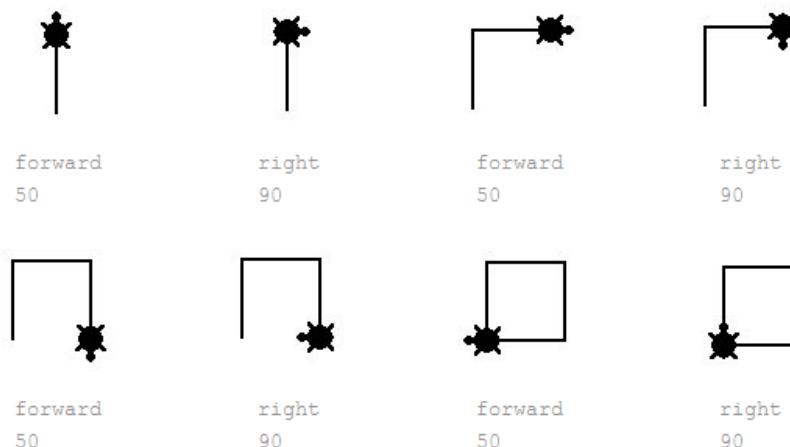
Ao longo do tempo, é evidente o surgimento crescente de ferramentas voltadas ao ensino da Tecnologia da Informação, uma vez que, conforme já abordado, a expansão deste mercado e suas aplicabilidades nos mais diversos contextos é visível. Com isso, podem ser abordados *softwares* educativos voltados para diferentes faixas etárias, mas principalmente visando alcançar e instigar o público que não possui conhecimento prévio sobre programação.

3.1 LOGO

A linguagem LOGO, projetada para utilização como ferramenta de aprendizagem, foi criada por Seymour Papert em 1967 a partir do ideal construtivista e sua principal característica é a figura de uma tartaruga que, inicialmente, tratava-se de um robô que seguia os comandos inseridos no computador e posteriormente passou a *interface* gráfica (LOGO FOUNDATION, c2014). Atualmente, é inserida em diferentes “ambientes LOGO” – versões da linguagem adaptadas para utilização em diferentes contextos ou países, no qual pode ser feita a tradução dos comandos ou inserção de novos, sem alterar os princípios fundamentais da linguagem.

Com uma linguagem interpretada, o ambiente tem foco no aprendizado por erros e acertos, retornando de imediato mensagens descritivas para o usuário em caso do não-reconhecimento de algum comando. Os comandos principais de movimento da tartaruga resumem-se em movimentos para frente, para trás e para os lados. O ambiente LOGO também permite comandos com operadores matemáticos, além da definição de procedimentos, que tratam-se de blocos de código que podem ser pré-definidos e salvos para utilização posterior. A Figura 1, abaixo, demonstra um exemplo detalhado dos comandos para desenhar um quadrado, no qual pode ser utilizado também o comando *repeat 4 [forward 50 right 90]*, aplicando estrutura de repetição.

Figura 1 – Comandos para a tartaruga desenhar um quadrado



Fonte: Logo Foundation, c2014

3.2 SCRATCH

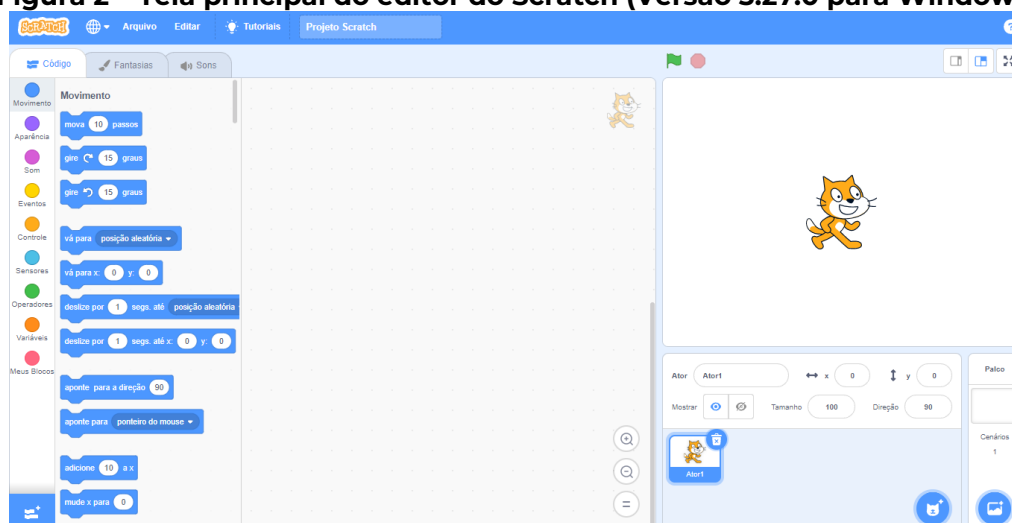
A linguagem de programação *Scratch* foi criada em 2007 pelo *Media Lab do Massachusetts Institute of Technology* (MIT) e projetada com foco no público com idade entre 8 e 16 anos, ainda que possa ser utilizada por todas as faixas etárias (MIT SCRATCH, s.d.).

Através da construção de comandos com peças que se encaixam formando blocos de código e resultam em ações sobre os personagens, permite o aprendizado de programação de forma lúdica e intuitiva, principalmente por sua extensa quantidade de diferentes atores, cenários e ações, possibilitando a criação de diversos projetos.

O ambiente de criação (editor), conforme Figura 2, possui ao lado esquerdo uma seção com as opções de Código, Fantasias e Sons que podem ser inseridos na estruturação dos comandos em blocos, construídos na parte central da *interface*.

Em “Código”, são exibidas estruturas para montar a sequência de ações dos atores. Os blocos são separados em categorias com blocos pré-definidos, de diferentes funcionalidades: Movimento; Aparência; Som; Eventos; Controle; Sensores; Operadores; Variáveis; e a opção “Meus Blocos”, que permite a criação de blocos customizados, simulando funções. Na opção “Fantasia”, podem ser personalizadas as características de cada ator – um mesmo ator pode ter várias “fantasias”, isto é, formas com o qual ele se apresenta na tela. Por fim, na opção “Sons”, é possível escolher sons pré-definidos ou ainda criá-los para aplicá-los na construção do projeto.

Figura 2 – Tela principal do editor do Scratch (Versão 3.27.0 para Windows)



Fonte: Mit Scratch, s.d.

O resultado das ações é exibido no quadro superior à direita e, abaixo dele, podemos escolher sobre qual ator aquela ação será executada – ou seja, para qual ator a sequência de código que o usuário construir será atribuído – ou selecionar novos atores e cenários. A ferramenta pode ser utilizada em sua plataforma *on-line*, ou ainda a versão para *Windows* que pode ser instalada na máquina local para ser acessada *off-line*.

3.3 BLOCKLY GAMES

O *Blockly Games* é um projeto de código-aberto da Google que dispõe de diversos jogos educacionais para o ensino de programação voltada para crianças sem conhecimento prévio sobre o assunto (*BLOCKLY GAMES*, s.d). A ferramenta é acessível *online* ou, ainda, é possível fazer o *download* para uso local. A construção do código segue também a ideia da programação visual, onde as linhas de comando são estruturadas em blocos que se encaixam e formam as ações. Durante os jogos, são introduzidos conceitos de repetições, estruturas condicionais, equações matemáticas e funções.

Cada jogo possui fases em que a complexidade da resolução dos desafios aumenta e, em alguns jogos, ao encontrar a solução correta, ela é exibida em forma textual na linguagem *JavaScript*. Nos jogos finais, já pode ser utilizado o editor de texto para escrever o código.

No exemplo da Figura 3, a captura à esquerda demonstra a estrutura do código da solução em blocos do nível 8 de um dos jogos disponíveis, o jogo *Maze*, no qual objetivo é levar o personagem até o final do percurso. Já à direita, a solução do nível em *JavaScript*.

Figura 3 – Blocos de código e solução em JavaScript no nível 8 do jogo Maze



Fonte: Blockly Games, 2022

3.4 PROGRAMAÇÃO EM JOGOS

Além das ferramentas acima, cujo principal objetivo é o ensino da lógica de programação, há a possibilidade de aprender estes conceitos através de outros jogos, em sua maioria já conhecidos pelo público infantil, mas muitas vezes não utilizados para esta finalidade.

O jogo *Minecraft*, em sua versão *Minecraft: Education Edition*, possui diversas ferramentas de ensino, dentre elas, atividades voltadas a STEM (*Science, Technology, Engineering and Mathematics*) e programação (MICROSOFT, c2021). Outro jogo, Roblox, através da plataforma Roblox Studio, utiliza a linguagem de programação Lua e dispõe de ferramentas para que o jogador possa desenvolver sua criatividade com diferentes graus de complexidade, abrangendo todos os níveis de experiência (ROBLOX CORPORATION, c2022).

3.5 COMPARANDO AS DIFERENTES FERRAMENTAS E SUAS ABORDAGENS

A partir de uma análise das ferramentas apresentadas, é possível verificar os pontos que se assemelham entre elas e avaliar as vantagens em aplicá-las em cada cenário. A linguagem LOGO apresenta uma *interface* simples e não dispõe de comandos em formato de “blocos” como as ferramentas *Scratch* e *Blockly*. O personagem tem suas ações a partir de comandos escritos pelo usuário, que podem ser “ensinados” ao personagem como procedimentos, o que a torna interessante para aplicação de princípios fundamentais da programação e primeira interação com linhas de código.

As ferramentas *Scratch* e *Blockly* têm em comum a utilização de blocos com comandos pré-definidos. A vantagem da ferramenta *Blockly* é apresentar desafios prontos a serem solucionados por uma lógica construída pelo usuário que, incentivado pela conclusão de cada nível dos jogos, pode utilizar a *interface* como um ambiente ideal para treinar os conhecimentos em lógica de programação.

A abordagem utilizada pelo *Scratch*, por sua vez, é vantajosa por ter seu ambiente de criação livre para que o usuário determine as ações do personagem e construa jogos utilizando livremente de sua criatividade. Assim, o ponto forte da ferramenta são as mais variadas possibilidades de estruturação dos jogos construídos com diversos temas e finalidades, ainda com a utilização dos blocos de código que se encaixam e resultam na aprendizagem de forma lúdica.

O uso da programação em jogos existentes torna-se uma abordagem interessante quando o aluno já conhece o jogo na visão de usuário, mas é apresentada a possibilidade de interagir com a ferramenta na visão de criador, fator este que pode instigar sua curiosidade ao universo da programação.

4. AULA EXPERIMENTAL COM ALUNOS DO FUNDAMENTAL I

Para validação das informações aqui descritas, foi realizada uma aula experimental presencial (Figura 4) com 17 crianças de faixa etária entre 9 e 11 anos, sendo 8 alunos matriculados no 4º ano e 9 alunos no 5º ano do Ensino Fundamental, em uma instituição privada de ensino no município de Praia Grande, estado de São Paulo. Com duração total de duas horas, o plano de aula foi elaborado a princípio considerando um público-alvo sem conhecimento prévio sobre lógica de programação.

Foi feita uma introdução com slides e figuras para estimular a interação das crianças e instigar a curiosidade sobre as informações apresentadas – no início foram exibidos conceitos de programação com palavras um pouco mais complexas, pedindo que cada aluno falasse o que acreditava ser o significado destes termos que não tinham tanta familiaridade. Posteriormente, demonstraram-se termos mais familiares (jogos, redes sociais, aplicativos), explicando como a tecnologia está presente no nosso cotidiano sem que se perceba e que a programação pode ser vista como um “esqueleto”, que sustenta e constrói os aplicativos utilizados por todos.

Assim, em duplas, os alunos foram desafiados a “construir sua própria diversão”, através do incentivo de que é possível produzirem seus próprios jogos – e isso, de forma simples, ainda

aprenderiam durante a aula – mas que para conseguirem, precisariam entender como as “peças do quebra-cabeça” se encaixam, propondo então o aprendizado da lógica de programação de forma lúdica.

Figura 4 – Aula ministrada aos alunos do ensino fundamental I

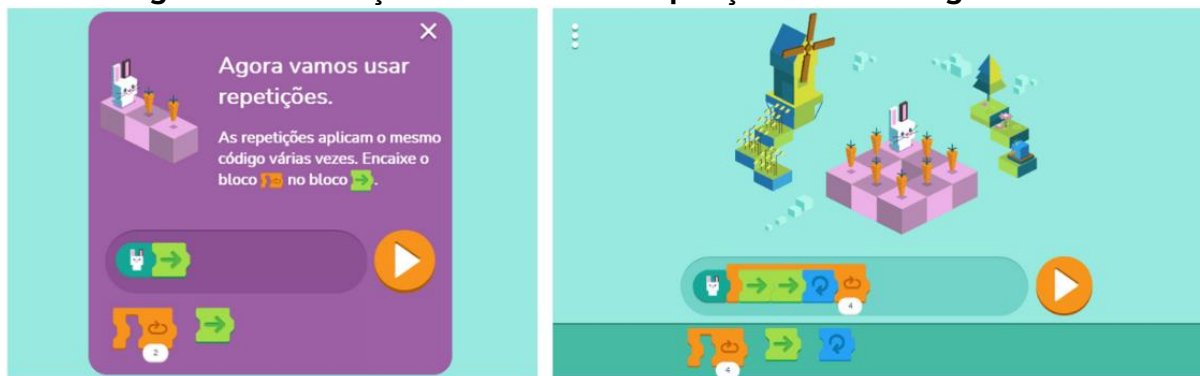


Fonte: Autoria própria, 2021

Para isso, foram utilizadas duas ferramentas. A primeira, plataforma *Google Doodle*, foi aplicada através de um jogo feito para a “Comemoração de 50 anos de programação para crianças” que consistia em 6 fases cujo objetivo era movimentar o personagem (Coelho) para colher todas as cenouras, com movimentos de linha reta e giros de 90°, no qual foi possível explicar os conceitos de *loops* (repetições).

Na Figura 5, à esquerda, uma captura do jogo demonstra a explicação sobre repetições, que antecede o início de uma das fases. Ao lado direito, é possível verificar o funcionamento do jogo, onde as peças do quebra-cabeça são encaixadas uma a uma para definir a forma de bloco que construirá a sequência de movimentos do personagem para obter todos os pontos.

Figura 5 – Introdução ao conceito de repetições com o Google Doodle



Fonte: Google, 2017

A segunda ferramenta utilizada foi o *Scratch* (MIT SCRATCH, s.d.), já elucidado em tópico anterior. O jogo escolhido para elaboração foi uma adaptação do Jogo Pong (SCRATCH BRASIL, 2014), com algumas alterações de modo a contemplar o conceito de variáveis, utilizadas para a pontuação, conforme Figura 6. O objetivo do jogo era utilizar a raquete para equilibrar a bola, sem deixá-la cair. A cada acerto, é contabilizado um ponto e os pontos são zerados caso a bola encoste no chão. Os alunos escolheram livremente as características dos atores e cenários.

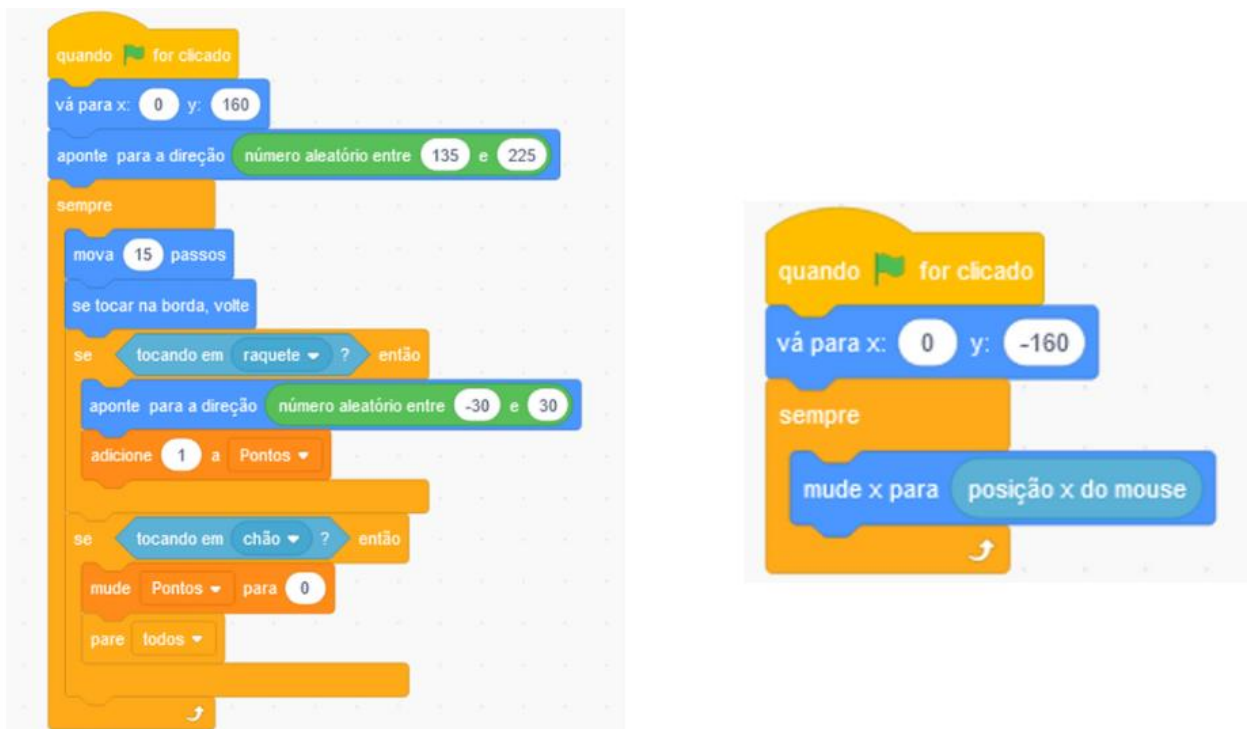
Figura 6 – Captura de tela do Jogo Pong em andamento



Fonte: Autoria própria, 2021.

Na Figura 7, é demonstrado o código ensinado aos alunos e por eles reproduzido durante a aula. Foi feito o passo a passo da montagem do jogo com os alunos, explicando as definições de estruturas condicionais, repetições, variáveis, coordenadas, ângulos e direções. A cada bloco adicionado, foram explicados os conceitos simples por trás da lógica aplicada com exemplos voltados às situações cotidianas, para melhor compreensão. Os alunos foram atendidos coletiva e individualmente em caso de eventuais dúvidas. Todos os alunos presentes conseguiram construir o jogo até o final e jogá-lo.

Figura 7 – À esquerda, código para o ator “Bola” e à direita, código para o ator “Raquete”



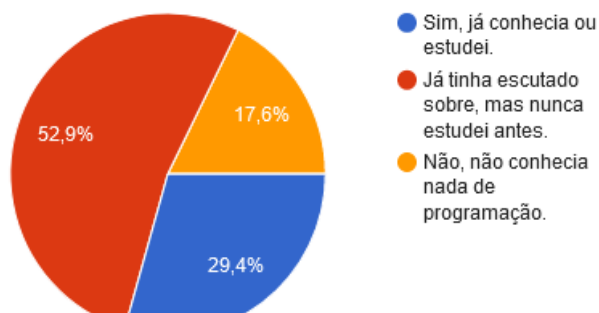
Fonte: Scratch Brasil, 2014; Adaptações de autoria própria, 2021

Após a conclusão do jogo, os alunos responderam um questionário proposto para verificação dos conhecimentos adquiridos e do entendimento da lógica de programação aplicada através dos *softwares* citados. Conforme evidenciado na Figura 8, apenas 29,4% dos alunos relataram já conhecer ou estudar programação anteriormente. Os demais, em sua maioria, somente escutaram sobre ou não tinham nenhum conhecimento sobre o assunto.

Figura 8 – Conhecimentos prévios de programação

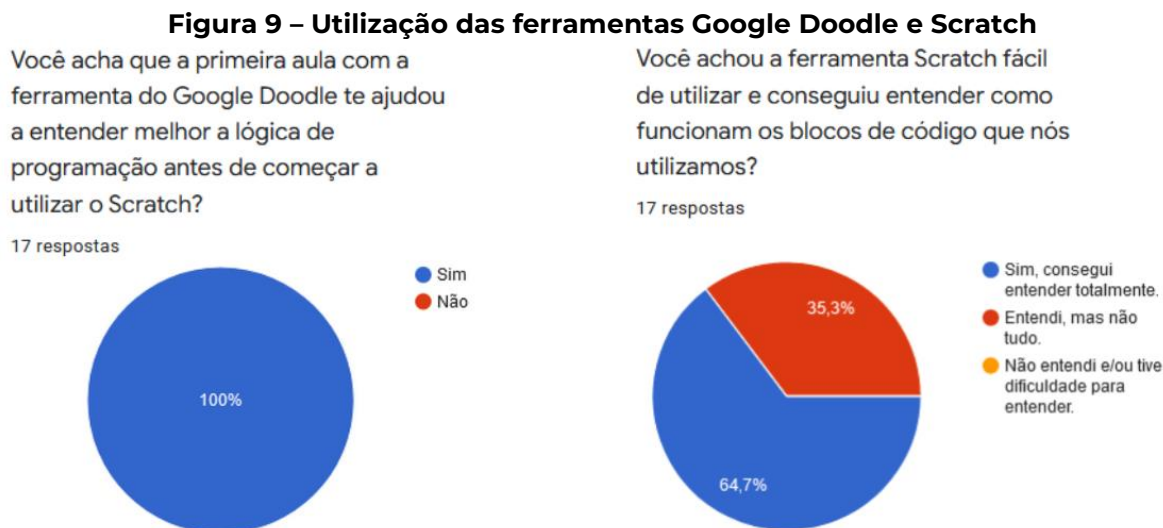
Você conhecia algo sobre programação antes desta aula?

17 respostas



Fonte: Autoria própria, 2021

Na Figura 9, os alunos responderam seus respectivos entendimentos sobre as ferramentas utilizadas, onde a utilização da ferramenta *Google Doodle* proporcionou melhor compreensão dos conceitos introdutórios para 100% dos alunos e 64,7% conseguiram entender totalmente o funcionamento dos blocos de código utilizados na ferramenta *Scratch*.



Fonte: Autoria própria, 2021

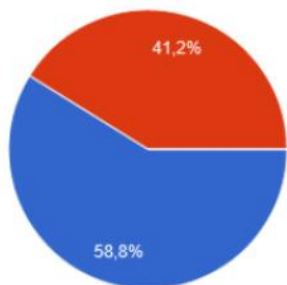
Em relação aos conceitos ensinados, foram disponibilizadas questões com 3 alternativas que se repetiam entre elas, a fim de verificar o nível de entendimento dos alunos sobre cada um dos tópicos, relembrando os exemplos demonstrados, entre parênteses.

Deste modo, nas figuras 10 e 11, a seguir, nota-se que a quantidade de respostas cujo houve entendimento total dos alunos sobre os conceitos ensinados foi superior a 50% em todos os casos, sendo o maior percentual de entendimento no conceito de variáveis, com 70,6%. Em nenhum quesito houve respostas referentes ao não entendimento ou dificuldades, o que foi notável durante a aula conforme a interação dos alunos e a conclusão das atividades.

Figura 10 – Estruturas condicionais e repetições

Você entendeu o que são "estruturas condicionais" (se - então / senão)

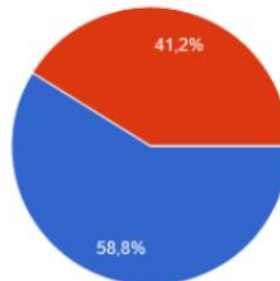
17 respostas



- Sim, consegui entender totalmente.
- Entendi, mas não tudo.
- Não entendi e/ou tive dificuldade para entender.

Você entendeu o que são "repetições/loops" (sempre...)

17 respostas



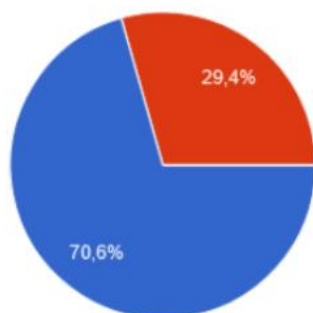
- Sim, consegui entender totalmente.
- Entendi, mas não tudo.
- Não entendi e/ou tive dificuldade para entender.

Fonte: Autoria própria, 2021

Figura 11 – Variáveis, coordenadas e ângulos

Você entendeu o que são variáveis? (utilizadas para as pontuações)

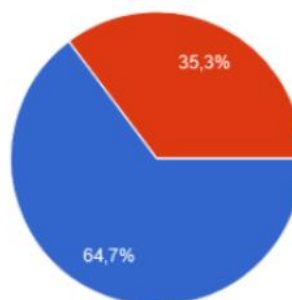
17 respostas



- Sim, consegui entender totalmente.
- Entendi, mas não tudo.
- Não entendi e/ou tive dificuldade para entender.

Você entendeu o que são coordenadas (posição X, posição Y dos personagens) e os ângulos/direções ("aponte para a direção entre -30° e 30°")?

17 respostas

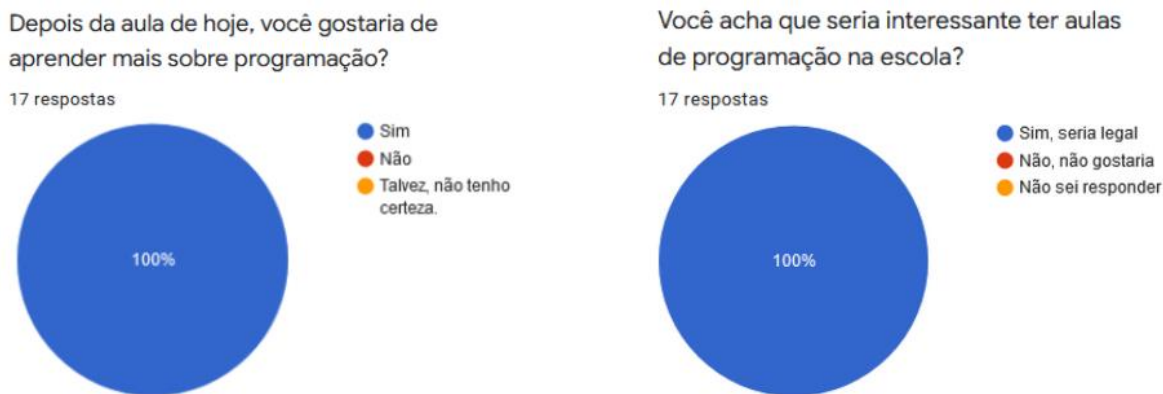


- Sim, consegui entender totalmente.
- Entendi, mas não tudo.
- Não entendi e/ou tive dificuldade para entender.

Fonte: Autoria própria, 2021

Por fim, na Figura 12, são evidenciadas as respostas de dois questionamentos referentes a continuidade de estudos sobre programação por parte dos alunos, qual se percebe, por unanimidade, o interesse das crianças em obter mais aprendizados sobre o assunto.

Figura 12 – Continuidade dos estudos em programação e aulas de programação nas escolas



Fonte: Autoria própria, 2021.

5. CONSIDERAÇÕES FINAIS

Com estudo aqui descrito, foi possível abordar a aplicabilidade de ferramentas educativas no ensino de lógica de programação para a faixa etária do Ensino Fundamental I no Brasil e demonstrar como este propósito teve seu avanço desde os primeiros ideais fundamentados com este intuito no século passado até os dias atuais, no qual diversos softwares, tendo ou não o ensino da tecnologia como objetivo final, proporcionam a possibilidade da obtenção destes conhecimentos para todos os públicos em diferentes contextos de aprendizagem.

Foi evidenciado que o uso de softwares educativos para esta finalidade oportuniza que os alunos aprendam conteúdos de extrema relevância para a área de tecnologia de uma forma lúdica e criativa, no qual o interesse das crianças tem seu despertar no momento em que é proposta não somente a visualização dos jogos como usuário, mas sim como criador.

Tendo em vista os objetivos propostos e seguindo os estudos de pesquisadores, os conceitos aqui abordados proporcionaram uma visão inicial dos possíveis resultados da implantação introdutória do Pensamento Computacional nas escolas brasileiras, levando em consideração a realização da aula experimental com os alunos da faixa etária escolhida no qual todos os alunos conseguiram realizar as atividades propostas, o que viabilizou um pré-entendimento das possíveis formas de estruturação das aulas e aplicação de um currículo gradativo de ensino.

Considerando-se os fatos mencionados, pode-se considerar a continuidade de pesquisas práticas nas escolas, uma vez que tenham infraestrutura adequada para tal fim – fator este que

é de suma importância que esteja em destaque para que a cada dia mais as instituições de ensino tenham acesso e suporte para a viabilidade de implantação destas aulas.

Com a aplicação de aulas em maior carga horária, acompanhamento da evolução do conhecimento dos alunos e análise da aprendizagem dos conceitos não só pelo seu próprio entendimento e visão, mas sim por avaliações pedagógicas recorrentes, torna-se possível validar como o uso destas ferramentas a longo prazo resultam no raciocínio lógico das crianças.

REFERÊNCIAS

BLOCKLY GAMES. Disponível em: <https://blockly.games>. Acesso em janeiro de 2022.

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular**. Brasília, 2018. Disponível em: <http://basenacionalcomum.mec.gov.br/>. Acesso em setembro de 2021.

CODE.ORG. c2022. Disponível em: <https://code.org/>. Acesso em janeiro de 2022.

FERRARI, Matheus. **Pesquisa revela dados sobre tecnologias nas escolas**. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP). 2021. Disponível em: <https://www.gov.br/inep/pt-br/assuntos/noticias/censo-escolar/pesquisa-revela-dados-sobre-tecnologias-nas-escolas>. Acesso em agosto de 2021.

GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. Vol. 4. São Paulo: Atlas, 2002.

GOOGLE. **Comemoração de 50 anos de programação para crianças**. 2017. Disponível em: <https://www.google.com/doodles/celebrating-50-years-of-kids-coding>. Acesso em novembro de 2021.

LOGO FOUNDATION. **What is Logo?** c2014. Disponível em: https://el.media.mit.edu/logo-foundation/what_is_logo/. Acesso em janeiro de 2022.

MARCONI, Marina de Andrade; LAKATOS, Eva Maria. **Fundamentos de metodologia científica**. 5 ed. São Paulo: Atlas, 2003.

MIT SCRATCH. Disponível em: <https://scratch.mit.edu/>. Acesso em novembro de 2021.

MICROSOFT. **What is Minecraft: Education Edition?** c2021. Disponível em: <https://education.minecraft.net/pt-pt/discover/what-is-minecraft>. Acesso em janeiro de 2022.

NIC.BR. CETIC.BR. CGI.BR. **Painel TIC COVID-19: Pesquisa sobre o uso da Internet no Brasil durante a pandemia do novo coronavírus – 3ª edição: Ensino remoto e teletrabalho**. 2020. Disponível em: <https://www.cetic.br/pt/publicacao/painel-tic-covid-19-pesquisa-sobre-o-uso-da-internet-no-brasil-durante-a-pandemia-do-novo-coronavirus-3-edicao/>. Acesso em agosto de 2021.

OBAMA, Barack H. **Computer Science Education Week 2013**. The Obama White House. Youtube, 2013. Disponível em: <https://www.youtube.com/watch?v=yE6IfCrqg3s>. Acesso em agosto de 2021.

PAPERT, Seymour. **Mindstorms: children, computers and powerful ideas**. Basic Books, Inc., 1980. New York, NY, USA. ISBN 0-465-04627-4.

PIAGET, Jean. **Seis Estudos de Psicologia**. Tradução de Maria Alice Magalhães D'Amorim e Paulo Sérgio Lima Silva. 24ª. Edição Revista, Rio de Janeiro: Forense Universitária, 1999.

PRAIA GRANDE, Prefeitura Municipal de. **Tecnologia na sala de aula eleva a qualidade do ensino em PG**. 2020. Disponível em: https://www.praiagrande.sp.gov.br/pgnoticias/noticias/noticia_01.asp?cod=50806. Acesso em setembro de 2021.

RAABE, André L. A.; BRACKMANN, Christian P.; CAMPOS, Flávio R. **Currículo de referência em tecnologia e computação: da educação infantil ao ensino fundamental**. São Paulo: CIEB, 2018. *E-book em pdf*. Disponível em: https://curriculo.cieb.net.br/assets/docs/Curriculo-de-referencia_EI-e-EF_2a-edicao_web.pdf. Acesso em setembro de 2021.

ROBLOX CORPORATION. **Roblox Studio**. c2022. Disponível em: <https://www.roblox.com/create>. Acesso em janeiro de 2022.

SÃO PAULO, Governo do Estado de. **Governo lança aulas em tempo real por TV aberta e celular a estudantes da rede estadual**. São Paulo, 2020. Disponível em: <https://www.saopaulo.sp.gov.br/spnoticias/governo-lanca-aulas-em-tempo-real-por-tv-aberta-e-celular-a-estudantes-da-rede-estadual/>. Acesso em agosto de 2021.

SCRATCH BRASIL. **Jogo Pong**. 2014. Disponível em: <http://www.scratchbrasil.net.br/index.php/atividades/126-jogo-pong.html>. Acesso em novembro de 2021.