

Jogos Digitais no Ensino de Conceitos de Programação para Crianças

Tancicleide C. S. Gomes¹, Jeane C. B. de Melo², Patricia C. de A. R. Tedesco¹

¹Centro de Informática - Universidade Federal de Pernambuco (UFPE)- Recife - PE, Brasil

²Departamento de Estatística e Informática - Universidade Federal de Pernambuco (UFRPE) - Recife - PE, Brasil

{tancicleide.gomes, jeane.ufrpe}@gmail.com, pcart@cin.ufpe.br

Abstract. *This paper describes a programming teaching experience based on digital games for children aged between five and six years in a private school. The students demonstrated understand the concepts presented, and keep themselves engaged. Initial analyzes suggest that digital games favor the presentation of content and manifest themselves as a possibility relevant to the programming teaching this age group. However, it was observed that some elements of the analyzed games interface need to be improved, considering the specificity of learning this audience, to provide a more understandable representation of certain programming concepts.*

Resumo. *O presente trabalho relata uma experiência de ensino de programação baseada em jogos digitais para crianças com idades entre cinco e seis anos em uma escola da rede privada. Os alunos demonstraram compreender os conceitos apresentados, além de manterem-se engajados. As análises iniciais sugerem que os jogos digitais favorecem a apresentação dos conteúdos e se manifestam como uma possibilidade relevante para o ensino de programação nesta faixa etária. No entanto, observou-se que alguns elementos da interface dos jogos analisados precisam ser aperfeiçoados considerando as especificidades de aprendizagem deste público-alvo, para prover uma representação mais compreensível de determinados conceitos de programação.*

1. Introdução

No decorrer dos últimos anos, o pensamento computacional tem sido apresentado como uma habilidade necessária não apenas para cientistas da computação, mas para toda a sociedade, sendo estimado como imprescindível aos profissionais do século XXI [CAS 2012, IFTF 2009; Stephenson *et al.* 2012, Wing 2006; Wing 2008].

Equiparado à escrita e à leitura, o pensamento computacional remete a estratégias para resolução de problemas através de raciocínio lógico e formal, em vários níveis de abstração [IFTF 2009, Falcão e Barbosa 2015, Wing 2006]. A institucionalização do ensino do pensamento computacional desponta como uma possibilidade para fornecer a base necessária para que os cidadãos de um novo mundo, programável, resolvam problemas, entendam e projetem sistemas bem como compreendam o comportamento humano, a partir de conceitos fundamentais de computação [Stephenson *et al.* 2012, Wing 2006, Wing 2008].

Algumas das questões relacionadas a disseminação do pensamento computacional são quando e de que maneira se deve apresentá-lo. Há pesquisadores que argumentam que desde a infância [IFTF 2009, Mittermeir 2013], porém desta proposição surgem as seguintes indagações:

1. Quais os meios eficazes de ensino (aprendizagem) do pensamento computacional para (pelas) crianças?
2. Quais os conceitos elementares do pensamento computacional?
3. Qual seria um encadeamento eficaz de conceitos para ensinar as crianças à medida que sua capacidade de

aprendizagem evolui ao longo dos anos? 4. Qual a melhor forma de integrar ferramentas para o ensino destes conceitos? [Wing 2008]

Neste sentido, algumas abordagens têm sido propostas, tais como: a apresentação do pensamento computacional de maneira interdisciplinar ou mesmo a inclusão dos fundamentos de ciência da computação enquanto disciplina do currículo básico desde a educação infantil [CAS 2012, IFTF 2009, Stephenson et al. 2012]. Além disso, existem estratégias de ensino que incorporam jogos digitais ao longo do processo de ensino e aprendizagem de programação [Karadeniz, Samur e Ozden 2014]. Observando-se o mercado de jogos digitais, nota-se uma quantidade crescente de aplicativos voltados ao ensino de programação para crianças [Falcão, Gomes e Albuquerque 2015; Gibson e Bell 2013]. Comumente, estes jogos envolvem sequenciamento de instruções, repetição, condicionais, funções, parâmetros e depuração, englobando quase todos os tópicos de programação propostos por currículos como o *CSTA/ACM* e *Computing at School* para esta faixa etária [Gibson e Bell 2013, Gomes e Alencar 2015]. No entanto, foram encontrados poucos relatos de iniciativas utilizando jogos digitais voltadas ao ensino fundamental e a educação infantil no cenário nacional.

O presente artigo emerge desse contexto e relata uma experiência de ensino de conceitos de programação para crianças em fase de letramento utilizando jogos digitais. Este trabalho está organizado da seguinte forma: a Seção 2 introduz um breve panorama sobre uso de jogos digitais no ensino de programação para crianças. A Seção 3 apresenta a metodologia aplicada em uma experiência de ensino de programação utilizando jogos digitais e a Seção 4 discorre sobre os resultados obtidos e as discussões relacionadas. A Seção 5 expõe as considerações finais.

2. Trabalhos Relacionados

O ensino do pensamento computacional, e em especial o ensino de programação, é um assunto recorrente em conferências e periódicos da área de computação. Analisando as publicações relacionadas, considerando o panorama nacional e internacional, observa-se um esforço expressivo em pesquisa voltado a propor metodologias, técnicas, novas abordagens e ferramentas para apresentar estes conteúdos [Aureliano e Tedesco 2012, Silva et al. 2014, Ramos et al. 2015]. No entanto, é possível observar que estas iniciativas são predominantemente voltadas para o ensino médio e superior.

Em uma revisão introdutória da literatura considerando os anais do Simpósio Brasileiro de Informática na Educação (SBIE) e do Workshop de Informática na Escola (WIE) nos anos de 2011 a 2015, foram encontrados poucos relatos descrevam experiências de ensino de programação nas seguintes faixas escolares:

- na educação infantil [Torezani, Chagas, e Tavares 2013; Raabe *et al.* 2015]
- no ensino fundamental [Andrade et al. 2013; Melo, Lima e Batista 2013; Campos *et al.* 2014; Friedrich *et al.* 2012; Schäfer, Sperb e Fagundes 2011; Oro *et al.* 2015]

Conforme é possível observar, esse público alvo é pouco comum na cenário nacional. Ao restringir esta pesquisa aos trabalhos que envolvem jogos digitais, os resultados são ainda mais escassos. Neste caso, apenas dois artigos foram encontrados: um relata uma experiência de ensino de programação e o outro propõe um jogo sobre conceitos de bloqueio e roteamento de redes.

O cenário internacional, por sua vez, apresenta resultados mais expressivos, inclusive porque alguns países incorporaram programação ao currículo básico, tendo como principais abordagens o uso de robótica, tecnologias tangíveis ou abordagens híbridas [Lehat *et al.* 2014, Mittermeir 2013]. No entanto, foram encontrados poucos relatos utilizando jogos digitais [Lehat *et al.* 2014, Mittermeir 2013], embora, em um estudo conduzido por Lehat *et al.* (2014), as crianças tenham demonstrado preferência por jogos digitais, considerando-lhes mais divertidos do que os artefatos tangíveis.

Alguns estudos delinearão dificuldades encontradas no uso alguns destes jogos, dentre as quais se destaca a compreensão no conceito de repetição. Karadeniz, Samur e Ozden (2014), relatam uma experiência híbrida com crianças de 5 anos, envolvendo atividades desplugadas e uso do jogo *Kodable*¹, em que as crianças demonstraram compreender os conteúdos apresentados e completaram as atividades solicitadas com sucesso, exceto pela notória e limitante dificuldade em compreender a lógica do conceito de repetição. Uma situação similar é relatada por Falcão, Gomes e Albuquerque (2015), na utilização do jogo *The Foos* com crianças entre cinco e seis anos de idade. Os participantes demonstraram engajamento e facilidade em executar as atividades propostas, mas quando era necessário usar blocos de repetição, eles usavam sem compreender o que estavam fazendo, obtendo resultados muito diferentes do que eles pareciam esperar.

Analogamente, Torezani, Chagas e Tavares (2013) relatam que as crianças, com idades entre cinco e oito anos, demonstraram entusiasmo e motivação na realização das atividades. Um dos diferenciais do jogo utilizado (*NewProg*) era apresentar o conceito de repetição implícito, sem o uso de parâmetros, tornando-o menos abstrato. As crianças deveriam mover personagens até um local específico ao longo de um labirinto e as ações eram divididas em: andar uma vez, andar duas vezes e andar três vezes, contudo ainda assim as crianças demonstraram dificuldades, com elevadas taxas de erro.

Outros desafios no uso de jogos digitais para ensino de programação para crianças são: 1. As interfaces de alguns jogos exigem muita leitura, limitando o uso com as crianças em fase de letramento [Falcão, Gomes e Albuquerque 2015]; 2. O *feedback* apresentado por alguns jogos não é suficientemente claro para o público alvo [Falcão e Barbosa 2015; Gomes *et al.* 2015]; 3. Nenhum dos jogos pesquisados é adaptativo, 4. Poucos jogos disponibilizam recursos que permitam ao professor o acompanhamento individual de cada aluno, por exemplo: mensurar a quantidade de erros, o tempo despendido em cada fase, dentre outros [Falcão e Barbosa 2015; Gomes *et al.* 2015].

Diante deste panorama, percebe-se que os estudos voltados ao ensino de tópicos de programação utilizando jogos digitais na educação infantil e ensino fundamental ainda são incipientes.

3. Metodologia

As experiências iniciais relatadas neste trabalho foram realizadas em uma escola da rede privada de ensino no 1º semestre letivo do ano de 2016 e são parte de uma iniciativa piloto que visa introduzir o ensino de conceitos de programação na escola. O método de pesquisa adotado foi a observação participante, posto que as atividades descritas foram realizadas durante as aulas de informática, ministradas pela primeira autora do presente trabalho.

Participaram destas atividades 20 crianças de duas turmas, sendo nove alunos da Turma 1 e 11 alunos da Turma 2, todos com idades entre cinco e seis anos de idade. Os participantes, em sua maioria, haviam experimentado jogos com fins similares e mesmo o próprio *The Foos* no ano anterior. No entanto, os alunos estavam na pré-escola (Infantil III) e eram cognitivamente menos maduros, além de apresentarem maior dificuldade em manusear o *mouse* devido à coordenação motora menos desenvolvida, dentre outras questões [Gomes e Alencar 2015, Falcão, Gomes e Albuquerque 2015]. O laboratório da escola é composto por computadores de mesa com acesso à internet e cada criança utilizou individualmente o computador, as aulas têm periodicidade semanal e duração de quarenta minutos.

As experiências de ensino foram realizadas em seis sessões de experimentação de jogos digitais e podem classificadas a partir de três temáticas principais: (i) Sequências de Instruções, (ii) Repetição e (iii) Letramento. Nas experiências voltadas a aprendizagem de sequências de instruções e desenvolvimento do raciocínio lógico foram utilizados os jogos *Ninja Shape*, *The Foos*

¹ <https://www.kodable.com/>

(níveis 1 ao 8) e *Code Baymax*, totalizando três sessões. As experiências focadas no ensino do conceito de repetição abrangeram os jogos *Lightbot* e novamente *The Foos* (a partir do nível 9 ao 24), envolvendo duas sessões. Finalmente, os jogos *Code Monkey* e Laboratório (*Code.Org*) envolveram práticas de letramento no aprendizado de sequências de instrução e conceitos de repetição, desenvolvida ao longo de uma sessão. Além disso, buscou-se ainda avaliar o quanto divertido os alunos acharam os jogos utilizados.

3.1. Jogos Digitais Selecionados

Os jogos digitais utilizados ao longo do presente trabalho foram selecionados considerando aspectos como: linguagem utilizada (visual ou textual), gratuidade e que pudessem ser utilizados em computadores de mesa, não havendo restrições se o jogo era *online* ou não. Os jogos selecionados podem ser classificados a partir de duas categorias, baseados em: (i) linguagem visual, (ii) linguagem textual.

Os jogos baseados em linguagens visuais têm como principal característica o uso de blocos cujos símbolos representam instruções de código, comumente associadas às ações de controle sobre um personagem e/ou objeto, tais como os jogos *Code Baymax*, *Code.Org*, *Lightbot* e *The Foos*. Os jogos baseados em linguagem textual exigem que os jogadores sejam capazes de ler e/ou escrever. Assim, foram selecionados dois jogos que permitissem aos alunos, ainda que de forma limitada, ler e escrever. Nesta categoria se encontram os jogos *Code Monkey* e a atividade Laboratório do site *Code.Org*, pois ambos utilizavam linguagem textual e envolviam conceitos de sequências de instruções e repetição. Todos os jogos são apresentados a seguir, em ordem alfabética.

3.1.1. Code Baymax

A narrativa do *Code Baymax*² envolve personagens do filme de animação *Operação Big Hero*. Originalmente destinado à crianças com idades entre oito e 12 anos, o jogo possui 24 fases e consiste em controlar o personagem Baymax utilizando blocos visuais para lutar contra um inimigo e resgatar seu amigo. Está disponível gratuitamente no site brasileiro de jogos da Disney®.

3.1.2. Code Monkey

No *Code Monkey*³, o jogador precisa controlar um macaco através de instruções de código em um painel interativo. O objetivo é capturar bananas ao mesmo tempo em que se desvia de obstáculos escrevendo sequências de código simples como: *turn left*, *turn right*, *step* (Figura 01). O jogo, que está disponível apenas na *web*, possui uma versão *trial* com 30 níveis e uma versão mais enxuta denominada Hora do Código, dedicada ao movimento homônimo e que contém apenas oito níveis. Nesse trabalho foi utilizada a versão Hora do Código.



² jogos.disney.com.br/operacao-big-hero-code-baymax

³ <https://www.playcodemonkey.com/>

Figura 01. Code Monkey - Nível 01

3.1.3. Code Studio

O site *Code Studio*⁴ compila uma série de atividades lúdicas que buscam apresentar conceitos de programação para crianças a partir dos quatro anos de idade. A maioria das atividades utiliza blocos de código encaixáveis similares aos encontrados na ferramenta *Scratch*. Nessa experiência foi utilizada a atividade Laboratório.

3.1.4. Lightbot

O *Lightbot*⁵ tem como personagem principal um robô que deve percorrer trajetos de ladrilhos e iluminar apenas os ladrilhos azuis (Figura 02). Destinado à crianças a partir dos quatro anos, o jogo está disponível gratuitamente na *web* e em lojas de aplicativos *Android* e *iOS*.

3.1.5. Ninja Shape

O *Ninja Shape*⁶ é um jogo do tipo *puzzle* que permite desenvolver o raciocínio lógico e abrange conceitos como sequências de instruções e depuração. O objetivo do jogo consiste em alcançar o vilão através de diversas interações com os botões e objetos do cenário do jogo utilizando apenas o mouse. A dificuldade é alternar entre as formas que o protagonista deve assumir considerando a disposição dos objetos auxiliares à sua volta (como bolas, molas e propulsores), além de considerar o momento exato em que as ações necessitam ser executadas. O jogo está disponível gratuitamente na *web*, sem público alvo especificado, porém é disponibilizado em sites de jogos voltados ao público infantil.

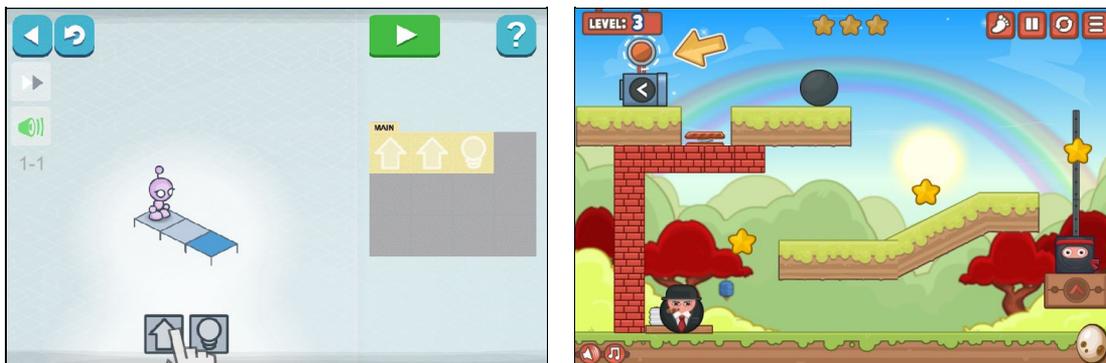


Figura 02. Lightbot - Nível 01; Ninja Shape - Nível 03

3.1.6. The Foos

*The Foos*⁷ é um jogo digital voltado ao ensino de programação para crianças a partir dos cinco anos de idade. O jogo está disponível em duas versões gratuitas: na *web*, em uma versão limitada que contém 24 níveis; e outra com novos níveis acrescentados semanalmente que está disponível para as plataformas *Android* e *iOS*. Para o experimento foi utilizada a versão *web*.

Tabela 1. Jogos Selecionados versus Conceitos de Programação Abrangidos

	Sequência de Instruções	Repetição	Condicionais	Procedimentos	Parâmetros	Depuração
Code Baymax						
Code Monkey (versão						

⁴ <https://code.org>

⁵ <https://lightbot.com/hocflash.html>

⁶ <http://www.bigdino.com/game/1511/Ninja-Shape>

⁷ thefoos.com/webgl/

Hora do Código)						
Code.Org (Laboratório)						
Lightbot (versão <i>web</i>)						
Ninja Shape						
The Foos (versão <i>web</i>)						

A distribuição dos conteúdos presentes em cada jogo utilizado encontra-se na Tabela 1. Convém mencionar que a tabela acima contempla apenas as versões utilizadas nesse trabalho, outras versões dos jogos podem abranger mais conteúdos.

4. Resultados Obtidos e Discussões

As sessões de experimentação dos jogos digitais visando o aprendizado de conceitos de sequências de instruções e repetição, assim como as práticas de letramento envolvendo esses conceitos são descritas nas subseções a seguir.

4.1. Sequências de Instruções

Inicialmente, em uma primeira sessão, o jogo Ninja Shape foi utilizado de modo a permitir o desenvolvimento do raciocínio lógico por parte dos alunos, trabalhando a resolução de problemas. A principal dificuldade consiste em controlar tanto o personagem, quanto interagir com objetos do cenário que auxiliam o personagem a derrotar o vilão. O jogo induz à determinação de uma sequência de passos, os quais devem ser seguidos estritamente, levando os alunos a "prever" quais passos devem ser realizados, em que sequência, e quais as suas consequências, considerando ainda o tempo de execução, o que exige algumas tentativas. De um modo geral, os alunos consideraram o jogo difícil, sendo o nível máximo alcançado o de número seis, atingido apenas por 30% da turma. Os demais atingiram apenas os níveis 4 e 5.

Em uma segunda sessão, os alunos experimentaram o jogo *The Foos* (níveis 1 ao 8) sem a intervenção docente, ou seja, foi proposto que os alunos tentassem resolver as fases do jogo sem pedir a ajuda da professora, podendo consultar os colegas. Os alunos haviam experimentado em 2015 uma versão anterior do *The Foos*. No experimento atual, usando uma versão com elementos de usabilidade melhorados, buscou-se observar impactos no desempenho e na autonomia dos alunos na resolução dos desafios. Ao longo da sessão, 50% dos alunos resolveu todos os níveis (1-8) e os demais alcançaram pelo menos o nível 5. Todos alcançaram pontuação máxima (3 estrelas) em todos os níveis, o que sugere que apesar dos ritmos diferentes de aprendizagem, os desafios propostos foram compreendidos, embora o jogo não ofereça elementos de ajuda.

Na terceira sessão, os alunos utilizaram o jogo *Code Baymax*. Não houve uma explanação preliminar sobre o conteúdo abordado, nem mesmo sobre as regras e objetivos do jogo. Esperava-se observar o quão longe os alunos conseguiriam avançar sem a intervenção da professora, sobretudo considerando a vivência anterior com o jogo *The Foos*. Um dos primeiros aspectos observados foi o envolvimento dos alunos, visto que o *Code Baymax* incorpora personagens do filme de animação *Operação Big Hero*. A narrativa é um outro diferencial desse jogo, e que apenas o *CodeMonkey* e *The Foos* apresentam, porém de maneira menos imersiva. Os alunos resolvem desafios com níveis de dificuldade crescentes e necessitam capturar objetos que fornecem novas habilidades e recursos e que auxiliam a derrotar os inimigos ao longo do jogo.

A linguagem utilizada no jogo é predominantemente visual e os símbolos escolhidos para representar as ações que controlam o personagem demonstraram ser de fácil compreensão para os alunos. Além disso, os elementos de ajuda oferecidos pela interface tornaram a intervenção

docente pouco requisitada e os alunos mantiveram-se engajados jogando até o fim da sessão. Os elementos de ajuda do *Code Baymax* são um dos seus grandes diferenciais, dentre eles se pode destacar: (i) A trajetória a ser realizada pelo robô é projetada por alguns instantes no tabuleiro; (ii) Um ícone de uma lâmpada permite que o jogador ‘peça uma dica’, essa dica resolve parcialmente o desafio proposto e pode ser acionada após um tempo limite; (iii) Um botão de ajuda simbolizado por uma silhueta de um olho e permite visualizar o trajeto a qualquer momento. O jogo não possui um sistema de pontuação por níveis, tal qual o *The Foos*, mas todos os alunos conseguiram resolver os desafios propostos (níveis 1-6). Além disso, os alunos classificaram o jogo como muito divertido, afirmando que gostariam de jogá-lo novamente.

4.2. Repetição

No intuito de apresentar o conceito de repetição aos alunos foi utilizado o jogo *Lighbot*. Os gráficos do *Lighbot* são isométricos (comumente chamados de falso 3D ou 2.5D) e demonstraram ser uma das principais fontes de dificuldades para os alunos, que nessa idade ainda estão desenvolvendo a orientação espacial. Apesar do apelo de controlar um robô - que foi algo pelo qual os alunos demonstraram entusiasmo -, os símbolos escolhidos para representar as ações, tais como girar a esquerda, girar a direita e pular, não foram bem compreendidos pelos alunos.

Os alunos demonstraram dificuldades em reconhecer para qual direção as setas direcionais girariam o personagem e erravam diversas vezes, mesmo compreendendo qual o desafio proposto pelo nível, o que os deixava frustrados. Os resultados obtidos sugerem que a combinação entre os gráficos isométricos e a simbologia designada para representar o controle do robô foram limitantes para a experiência de aprendizado, pois os alunos despendiam mais tempo tentando compreender os elementos de interação do que tentando compreender o desafio em si.

Dentre os assuntos abordados, observou-se dificuldades na compreensão sobre como utilizar os elementos de interação para resolver o problema proposto, devido, em parte, à representação simbólica do conceito de repetição adotada no jogo. O *Lightbot* não possui um símbolo específico que represente esse conceito, utilizando o conceito de procedimento para a resolução de problemas que necessitam de *loops*.

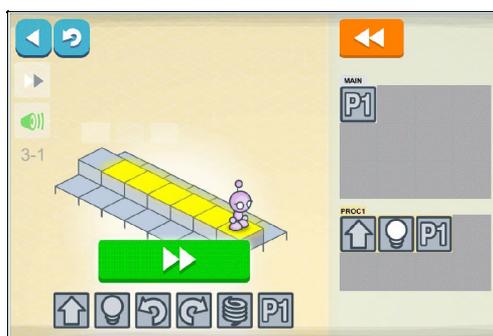


Figura 03. Nível 01 - Laços, Jogo Lightbot

O painel principal (MAIN, Figura 03) em que o jogador insere instruções para controlar o robô delimita a quantidade de blocos de código a ser utilizada. Aparentemente, o objetivo é apresentar ao jogador a possibilidade de economizar código através da ‘chamada’ de um ‘procedimento’ (PROC1) no painel principal utilizando um único bloco de código (P1) e que engloba todas as ações que necessitam ser repetidas, inclusive a chamada recursiva ao procedimento. No entanto, essa representação demonstrou ser inadequada para o público alvo a que se destina, possivelmente por ser uma concepção abstrata demais para a maturidade cognitiva dos participantes, ocasionando desinteresse demonstrado por falas como: “Ah, tia não tô

entendendo nada não (sic)”; *“Tia, muda esse jogo, não tô entendendo não.(sic)”*. Questionados sobre o quão divertido eles acharam o jogo, os alunos afirmaram que achavam o robô ‘legal’, mas que controlá-lo era ‘muito difícil’.

Dada a experiência pouco proveitosa no uso do jogo *Lighbot*, o conceito repetição foi retomado em mais uma sessão. O conceito foi apresentado através de uma conversa dirigida, em que os alunos foram levados a refletir sobre a equivalência de usar três blocos ‘para frente’ ou de usar um ‘bloco mágico’ repetir que pudesse fazer com o que o personagem controlado repetisse tantas vezes quanto necessário uma determinada ação.

Para prover uma experimentação foi proposta uma brincadeira em sala de aula em que os alunos deveriam auxiliar um colega ‘robô’ a atravessar a sala. O colega robô deveria atender somente às instruções: anda, pega objeto, empurra a cadeira, e apenas um aluno por vez poderia controlar o robô. Objetos foram distribuídos no chão da sala e cadeiras foram colocadas como obstáculos. Os alunos se voluntariaram para serem programadores e robôs e houve três rodadas de ‘programação’. Os discursos dos alunos indicaram a compreensão do conceito: em uma das rodadas, por exemplo, um aluno interrompeu o outro que insistia em dizer a palavra ‘anda’ repetidas vezes ao invés de usar o comando repetir: *“Não, para de dizer anda. Diz logo repete 12 vezes anda que ele vai logo tudo e chega lá na porta.(sic)”*.

Por fim, objetivando a aplicação dos conceitos recém ensinados foi proposto que os alunos resolvessem os níveis (9-16) do jogo *The Fools*, pois ele contém um bloco de repetição cujo controle é dado por parâmetros (1-4 vezes). O bloco de repetição foi explicado e associado à atividade recém-realizada. Um terço dos alunos alcançou o nível 16 e os dois terços restantes pelo menos entre o nível 12 e 15. Alguns alunos demonstraram dificuldades em utilizar o bloco de repetição, mesmo após a atividade e as explicações.

4.3. Letramento

Com idades entre cinco e seis anos muitas crianças estão aprendendo a ler e escrever. Deste modo, buscou-se combinar o aprendizado de programação com práticas de letramento, que são oportunidades de fomentar a leitura e a escrita em contextos que tenham sentido e que façam sentido para o aluno [Soares 2003]. Além da prática de letramento em si, buscou-se observar a transição de uma linguagem puramente visual e simbólica para uma linguagem textual em duas sessões de experimentação, considerando as seguintes questões: (i) Os alunos compreendem suficientemente bem os desafios a ponto de os resolverem utilizando outro tipo de linguagem? (ii) O quão impactante foi para os alunos utilizarem outra linguagem, em termos de desempenho e autonomia?

No *Code Monkey*, os alunos necessitam escrever código em inglês, usando instruções simples que indicam se o personagem deve girar à esquerda ou à direita e quantos passos o personagem deveria andar. Para medir a quantidade de passos entre o personagem e qualquer objeto, o jogo disponibilizava uma régua que mostrava a distância. Por exemplo, ao clicar no personagem e em uma banana usando a régua, o display exibe o número 10, então o aluno precisa escrever o comando *‘step 10’*, fazendo com que o personagem se mova 10 passos.

Na atividade Laboratório, os desafios propostos envolviam a escrita de maneira mais criativa como diálogos entre personagens, e ao invés de controlar os personagens através de instruções de código escritas, os alunos conectavam blocos de código. No entanto, os longos textos que indicavam o desafio proposto a cada nível acabavam por serem sempre ignorados, pois os alunos não estavam dispostos a ler e o objetivo do nível nem sempre era possível de compreender sem que o enunciado fosse lido.

Diferentemente do esperado, os alunos demonstraram melhor desenvoltura na experimentação do *Code Monkey*, de modo que todos os participantes alcançaram o último nível da versão Hora do Código sem a intervenção da professora. Os alunos demonstraram dificuldades em escrever em inglês por não conhecerem a grafia das palavras, mas a lógica da

resolução era descrita corretamente. Quando indagados à respeito, consideraram mais divertido escrever curtas instruções de código em inglês do que a experiência com a atividade Laboratório. Possivelmente, o fato dos alunos também terem aulas de inglês tornou a ação de escrever em outro idioma menos impactante.

Os resultados obtidos sugerem que os alunos nessa faixa etária não são propensos a ler textos com enunciados longos, sobretudo quando inseridos em atividades de jogo. Além disso, alguns elementos de interação da atividade Laboratório foram mal projetados para o público alvo a que se destina e esse possivelmente foi um dos fatores que mais influenciou na má compreensão dos desafios, e não o uso de uma outra linguagem. A construção textual utilizada, tanto nos enunciados quanto nos blocos, demonstrou não ser adequada para essa faixa etária, apesar de estarem no idioma nativo.

Outro aspecto problemático foi a unidade de medida escolhida: pixels. Os alunos precisavam, por exemplo, fazer o personagem mover-se em uma determinada direção, porém a unidade de medida além de não ser comum ao cotidiano dos alunos era de difícil mensuração e os alunos necessitaram testar as possibilidades disponíveis até encontrar a resposta adequada. No mais, os alunos consideraram divertida a possibilidade de escrever frases curtas e ver os personagens 'falarem' ou 'conversarem' entre si, mas os desafios só foram resolvidos mediante a condução das turmas pela professora.

5. Considerações Finais

A disseminação do pensamento computacional para crianças, especialmente o ensino de programação na educação infantil e nos anos iniciais do ensino fundamental, tem recebido notoriedade ao longo dos últimos anos.

A inclusão destes conteúdos no currículo básico traz consigo questionamentos que envolvem quais os conteúdos e em qual ordem estes conteúdos devem ser ensinados, de modo que os currículos de referência surgiram visando responder essas indagações. No entanto, ainda persistem as questões sobre quais os meios mais eficazes de ensino e aprendizagem de programação para esse público alvo e qual a melhor forma de integrar ferramentas para auxiliar na aprendizagem desses conceitos.

Neste sentido, o presente trabalho visou explorar os jogos digitais enquanto ferramenta para auxiliar no ensino de programação à partir de uma experiência realizada em uma escola da rede privada, como parte de uma iniciativa piloto de introduzir o pensamento computacional para crianças com idades entre cinco e seis anos.

Os resultados obtidos sugerem que os jogos digitais são uma possibilidade relevante para apoiar o ensino de programação para essa faixa etária, por proverem uma abordagem lúdica que mantém os alunos engajados enquanto aprendem. No entanto, as análises exploratórias iniciais indicam que alguns elementos de interface dos jogos utilizados parecem não considerar as especificidades do público ao qual se destinam e nem sempre proveem uma adequada apresentação dos conteúdos de programação, tanto usando linguagens visuais e simbólicas, quanto em linguagens textuais, asseverando desafios indicados nos trabalhos relacionados [Falcão e Barbosa 2015; Falcão, Gomes e Albuquerque 2015; Gomes *et al.* 2015].

Referências

- Andrade, D.; Carvalho, T.; Silveira, J.; Cavalheiro, S.; Foss, L. Fleischmann, A. M.; Aguiar, M.; Reiser, R. Proposta de Atividades para o Desenvolvimento do Pensamento Computacional no Ensino Fundamental. In: Anais do Workshop de Informática na Escola. 2013.
- Aureliano, V. C. O., e Tedesco, P. C. A. R. Ensino-aprendizagem de Programação para Iniciantes: uma Revisão Sistemática da Literatura focada no SBIE e WIE. In: Anais do Simpósio Brasileiro de Informática na Educação. 2012.

- Campos, G. M. de; Cavalheiro, S.; Foss, L.; Pernas, A. M.; Piana, C. F. de B.; Aguiar, M.; Reiser, R. Organização de Informações via Pensamento Computacional: Relato de Atividade Aplicada no Ensino Fundamental. In: Anais do Workshop de Informática na Escola. 2014.
- CAS - Computing At School Working Group (2012). Computer Science: A curriculum for schools. Disponível em: <http://bit.ly/1MNe9Mi>, Maio.
- Falcão, T. P., e Barbosa, R. "Aperta o Play!" Análise da Interação Exploratória em um Jogo Baseado em Pensamento Computacional. In: Anais do Simpósio Brasileiro de Informática na Educação. 2015. p. 419.
- Falcão, T. P., Gomes, T. e Albuquerque, I. R. (2015) O Pensamento Computacional Através de Jogos Infantis: uma Análise de Elementos de Interação. In: Anais do Simpósio de Fatores Humanos em Sistemas Computacionais.
- Friedrich, R. V.; dos Santos, D. S.; dos Santos Keller, R.; Puntel, M. D. e Biasoli, D. Proposta Metodológica para a Inserção ao Ensino de Lógica de Programação com Logo e Lego Mindstorms. In: Anais do Simpósio Brasileiro de Informática na Educação. 2012.
- Gibson, B., e Bell, T. Evaluation of games for teaching computer science. In: Proceedings of the 8th Workshop in Primary and Secondary Computing Education. ACM, 2013. p. 51-60.
- Gomes, T., Alencar, A. Análise Empírica de Jogos Educativos para Dispositivos Móveis voltados a Disseminação do Pensamento Computacional na Educação Básica. In: Anais dos Workshops do Congresso Brasileiro de Informática na Educação. 2015. p. 731.
- IFTF - Institute for the Future. Everyone is a programmer: Making the World a Control System. When everything is programmable: Life in a Computational Age. 2009.
- Karadeniz, S.; Samur, Y.; Ozden, M. Y. Playing with Algorithms to Learn Programming A Study on 5 Years Old Children. In: International Conference on Information Technology and Applications, 9, 2014, Sydney – Australia. Annals... Sydney: ICITA, 2014
- Lehat, M. L., Mokhtar, R., Sokman, Y., Ismail, M. I., e Basir, N. M. Games: An approach to introduce computer programming for upper primary school students. In: User Science and Engineering (i-USER), 2014 3rd International Conference on. IEEE, 2014. p. 85-90.
- Melo, L. de A., Costa, T. K. de L., e Batista, A. C. D. Pense bem: proposta e desenvolvimento de jogo digital para ensino de computação na educação básica. In: Anais do Simpósio Brasileiro de Informática na Educação. 2013.
- Mittermeir, R. T. Algorithmics for Preschoolers—A Contradiction? Creative Education, v. 4, n. 09, p. 557, 2013.
- Oro, N., Pazinato, A., Teixeira, A., e Gross, Á. Olimpíada de Programação de Computadores para Estudantes do Ensino Fundamental: A interdisciplinaridade por meio do Software Scratch. In: Anais do Workshop de Informática na Escola. 2015. p. 103.
- Raabe, A., Rodrigues, A. J., Santana, A. M., Vieira, M. V., Rosário, T., e Carneiro, A. C. R.. Brinquedos de Programar na Educação Infantil: Um estudo de Caso. In: Anais do Workshop de Informática na Escola. 2015. p. 42.
- Ramos, V., Wazlawick, R., Galimberti, M., Freitas, M., e Mariani, A. C. A Comparação da Realidade Mundial do Ensino de Programação para Iniciantes com a Realidade Nacional: Revisão sistemática da literatura em eventos brasileiros. In: Anais do Simpósio Brasileiro de Informática na Educação. 2015. p. 318.
- Schäfer, P. B.; Sperb, B. F.; Fagundes, L. da C. Squeak Etoys na modalidade 1 para 1: programação e autoria multimídia no desenvolvimento da conceituação. In: Anais do Workshop de Informática na Escola. 2011. p. 1226-1235.
- Silva, T. R. da, Medeiros, T. J., e Aranha, E. H. D. S. Jogos Digitais para Ensino e Aprendizagem de Programação: uma Revisão Sistemática da Literatura. In: Anais do Simpósio Brasileiro de Informática na Educação. 2014. p. 692-701.
- Stephenson, C. et al. The new CSTA K--12 computer science standards. In: Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education. ACM, 2012. p. 363- 364.
- Torezani, C.; Chagas, L. B. da C.; Tavares, O. de L. NewProg-um ambiente online para crianças aprenderem programação de computadores. In: Anais do Workshop de Informática na Escola. 2013.
- Von Wangenheim, C. G. e Von Wangenheim, A. "Ensinando Computação com Jogos". Bookess.2012.
- Wing, J. M. Computational thinking and thinking about computing. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, v. 366, n. 1881, p. 3717- 3725, 2008.
- Wing, J. M. Computational thinking. Communications of the ACM, v. 49, n. 3, p. 33-35, 2006.
- Soares, M. B. (2003). "O que é letramento". Diário na Escola - Santo André. Diário do Grande ABC.