

Pensamento Computacional, robótica e educação

Rubens Lacerda Queiroz (PPGI/UFRJ)¹

Fábio Ferrentini Sampaio (NCE e PPGI/UFRJ)²

Mônica Pereira Dos Santos (FE/UFRJ)³

Resumo

Este artigo apresenta um estudo exploratório sobre o uso da robótica educacional com crianças do ensino fundamental I com o objetivo de investigar o desenvolvimento de habilidades relacionadas ao Pensamento Computacional.

Palavras-chave: Pensamento Computacional; Robótica Educacional; Linguagens Visuais; Tecnologias no Ensino.

Abstract

This article presents an exploratory study about the use of educational robotics with elementary school children with the objective of investigating the development of abilities related to Computational Thinking.

Keywords: Computational Thinking; Educational Robotics; Visual Computing Languages; Technologies in Education.

¹ Contato: rubensqueiroz@outlook.com

² Contato: ffs@nce.ufrj.br

³ Contato: monicapess@gmail.com

1. Introdução

Hoje, a questão do ensino de computação, já a partir das séries iniciais do ensino fundamental, tem sido discutida em fóruns acadêmicos a partir de diferentes perspectivas: a necessidade de todo cidadão ser produtor e não apenas consumidor de tecnologia digital; o frequente insucesso dos alunos de graduação nas disciplinas iniciais de programação como meio de desenvolver nos alunos competências e habilidades relacionadas ao Pensamento Computacional (Wing, 2006).

O trabalho aqui relatado vai ao encontro a este debate na medida em que busca investigar, a partir de um estudo exploratório, a possibilidade de desenvolvimento e exercício de habilidades relacionadas ao Pensamento Computacional em crianças do Ensino Fundamental I (pertencentes a comunidades de baixa renda) por meio do aprendizado de conceitos básicos de programação.

Para levar a cabo o presente estudo, o Laboratório LIVRE (Laboratório de Inovações em Robótica Educacional) lançou mão da Robótica Educacional (RE) de baixo custo, como forma de tornar mais concreta e atraente as atividades desenvolvidas pelos participantes; da linguagem de Programação Visual DuinoBlocksforKids (AUTORES), desenvolvida pela equipe como forma de minimizar problemas relatados pela literatura no aprendizado da sintaxe e semântica de linguagens de programação (Papadakis, Kalogiannakis et al, 2014; Good, Howland, 2016); da proposição e utilização de atividades didáticas, especificamente desenvolvidas para este público alvo, que exploram as possibilidades práticas da RE e da computação desplugada de forma lúdica⁴.

O presente trabalho está dividido nas seguintes seções: a seção 2 apresenta, de forma sucinta, a fundamentação teórica deste trabalho; a seção 3 trata dos procedimentos metodológicos; a seção 4 descreve os principais resultados alcançados e a seção 5 apresenta as conclusões e algumas considerações sobre o trabalho realizado.

2. Fundamentação Teórica

2.1. Pensamento Computacional

Jan Cuny, Larry Snyder e Janet Wing definem o Pensamento Computacional (*Computational Thinking*) como sendo:

⁴ Projeto parcialmente financiado pela FAPERJ (E-26/210.820/2014)

O processo de pensamento envolvido na formulação de problemas e de suas soluções de maneira que essas soluções estejam representadas de forma a poderem ser efetivamente executadas por um agente processador de informações (Cuny; Snyder; Wing, 2010, tradução autores).

Para Hemmendinger (2010 *apud* Barr; Stephenson, 2011), o objetivo do desenvolvimento do Pensamento Computacional (PC) não é o de fazer com que todos passem a pensar como cientistas da computação mas, sim, habilitar as pessoas a aplicarem esta maneira específica de raciocinar na busca por novos questionamentos e na solução de diversos tipos de problemas nas mais variadas áreas do conhecimento.

“Pensar Computacionalmente” é reformular um problema aparentemente difícil de maneira a “transformá-lo” em um que saibamos resolver, é prevenir erros e estar pronto para corrigi-los, revisando cada etapa realizada na busca pela solução de um problema. É saber planejar na presença de incertezas e entender que é possível trabalhar de forma segura com problemas complexos sem precisar conhecer todos os seus detalhes (Wing, 2006). Para autores como Wing (2011) os benefícios da capacidade de pensar computacionalmente podem ser transportados para qualquer domínio por meio do aumento e reforço de uma série de habilidades intelectuais e, embora não haja ainda consenso acerca do conjunto exato dessas habilidades, de acordo com Grover e Pea (2013), a maior parte dos pesquisadores e educadores da área de Ciências da Computação tem aceitado, de forma bastante ampla, que esta forma de pensamento pode ser compreendida pelos seguintes elementos:

- Abstração e generalização de padrões (incluindo modelagem e simulações);
- Processamento sistemático de informações;
- Sistemas simbólicos e representações;
- Noções algorítmicas de fluxo de controle;
- Decomposição estruturada de problemas;
- Pensamento paralelo, recursivo e interativo;
- Lógica condicional;
- Condicionantes de eficiência e performance;
- Depuração e detecção sistemática de erros.

Como se pode observar, o Pensamento Computacional envolve muitos elementos que, para serem trabalhados, necessitam do exercício e do desenvolvimento de uma ampla gama de habilidades. Para se abarcar todo esse conjunto de habilidades torna-se necessário a realização de um trabalho extenso e aprofundado de diversos aspectos relacionados às Ciências da Computação, ou seja: Pensar Computacionalmente, ao

contrário do que se pode imaginar em um primeiro momento, não se restringe à habilidade de programar computadores. No entanto, tem-se na programação (competência fundamental da Ciência da Computação) uma importante ferramenta de apoio ao desenvolvimento e exercício de algumas das habilidades cognitivas características do Pensamento Computacional (Grover; Pea, 2013).

Os debates sobre o desenvolvimento do PC nas escolas têm se apoiado, principalmente, em duas linhas de argumentação diferentes, a saber:

Formação para o mercado de trabalho

Alguns autores defendem o desenvolvimento do PC nas escolas como estratégia para formação de futuros profissionais de informática, uma vez que diferentes estudos vêm apontando uma carência na formação de pessoal qualificado na área, tanto no Brasil como no exterior (Barcelos; Silveira, 2012).

Formação STEAM

Diferentes pesquisadores, educadores e formuladores de políticas públicas argumentam que a computação deve ter um papel mais prominente no sistema *educacional*, a fim de desenvolver nos alunos habilidades ligadas às áreas de Ciências, Tecnologias, Engenharias, Artes e Matemática (*STEAM - Science, Technology, Engineering, Arts and Mathematics*). Neste sentido, nos EUA por exemplo, foi aprovado em 2015 o *STEM Education Act* que, entre outras providências, inclui esforços para expandir o ensino de computação nas escolas. A Inglaterra, por sua vez, alterou o currículo escolar (*National Curriculum*) desde o 1º ano do ensino fundamental (*Key stage 1*) a fim de "(...) preparar os estudantes para usarem o pensamento computacional e a criatividade para entender e modificar o mundo (...)" (National Curriculum, 2013). Este mesmo documento aponta cinco grandes metas que devem ser atingidas pelos estudantes:

- Entender e aplicar os princípios fundamentais e os conceitos da ciência da computação incluindo abstração, lógica, algoritmos e representação de dados;
- Desenvolver experiência prática na análise de problemas em termos computacionais;
- Avaliar e aplicar de forma analítica as tecnologias da informação na solução de problemas;
- Tornar-se um usuário responsável, competente, confiante e criativos no uso das tecnologias da informação e comunicação.

No Brasil, o Ministério da Educação (MEC), por considerar o letramento digital um direito a ser assegurado e de grande relevância para a cidadania e para a atuação crítica na vida social, definiu, no texto mais recente da Base Nacional Comum Curricular (BNCC), um tema integrador denominado "Culturas Digitais e Computação". Este tema está relacionado

ao "uso pedagógico das novas tecnologias da informação e comunicação e [à] exploração dessas novas tecnologias para a compreensão do mundo e para a atuação nele" (MEC, 2016, p.51). Ainda de acordo com o mesmo documento:

[...] em uma sociedade cada vez mais tecnologicamente organizada, torna-se imperativo que a escola passe a considerar as potencialidades do uso dos recursos tecnológicos para o alcance de suas metas (MEC, 2017, p. 37).

Também preocupada com a temática da introdução das Ciências da Computação no Ensino Básico, a Sociedade Brasileira de Computação (SBC) realizou no ano de 2016 duas reuniões com o MEC para discutir a definição de diretrizes curriculares nacionais específicas para a área da Computação, apresentando, durante esses encontros, diferentes justificativas sobre a importância da inclusão de conteúdos de computação na Educação Básica, tanto no Ensino Fundamental quanto no Ensino Médio (SBC, 2016a; SBC, 2016b).

Esta preocupação, tanto do Governo Federal quanto da SBC, em relação à introdução da computação na Educação Básica e de sua importância como ferramenta de atuação do aluno sobre o mundo, cria um ambiente favorável à utilização do aprendizado de programação como ferramenta de potencialização daquilo que o aluno pode criar e aprender com o computador.

2.2. Robótica Educacional

A palavra robô, mencionada pela primeira vez em 1920 na peça R.U.R (Rossum's Universal Robots) do escritor tcheco Karel Capek e lançada no Brasil pela editora Hedra com o título de "A Fábrica de Robôs" (Tchápek, 2012), tem origem na palavra tcheca *robot* que significa "trabalho compulsório". Na peça de Capek, um cientista descobre uma maneira de "dar vida" a máquinas com aparência humana, capazes de fazer tudo o que o homem faz. Em 1950, Isaac Asimov aborda novamente o tema em seu famoso livro de ficção científica "Eu, Robô" (Asimov, 2014), apresentando robôs humanoides altamente desenvolvidos. E é este robô, com aparência humana, que povoa o imaginário mundial (Alfaro, 2006; Souza; Duarte, 2015).

No entanto, robôs, de fato, são basicamente qualquer estrutura mecânica autômata multifuncional reprogramável, operada por meio de circuitos integrados, de controles eletro-hidráulicos e de engrenagens, projetada para movimentar, de diversas formas, uma série de materiais ou dispositivos especializados (Souza; Duarte, 2015; Kloc; Koscianski; Pilatti,

2009). A construção dessas estruturas e sua manipulação por meio de sistemas baseados em lógica de programação constitui um ramo da ciência denominado Robótica (Souza; Duarte, 2015). A robótica encontra, atualmente, aplicação em diversas áreas, como a indústria, a medicina, a exploração espacial e, também, a educação. Quando utilizada com propósitos educacionais a robótica recebe a denominação de Robótica Educacional (Salvin; Korsah; Norbakhsh, 2015).

A ideia do uso da robótica em educação se baseia fortemente na Teoria Construcionista de Seymour Papert, que une a Teoria Construtivista de Jean Piaget - com quem trabalhou entre 1958 e 1963 na Universidade de Genebra, Suíça - ao uso do computador na educação (Pinto, 2011). Papert (1993) adicionou à teoria de Piaget (1974) o fato de que a manipulação de objetos é chave para as crianças construírem seu próprio conhecimento; a ideia de que essa construção se dá de forma mais efetiva quando o aprendiz se engaja de maneira consciente na construção de algo tangível (Papert, 1993 apud Alimisis, 2013).

Nesse contexto, Papert criou o Logo, um software que permite aos usuários, através de linhas de código, movimentar uma “tartaruga”, um animal cibernético que pode ser tanto um objeto virtual (presente na tela do computador) quanto um objeto físico manipulável (Papert, 1993; Solomon; Papert, 1976). Essa tartaruga deixa um “rastro” (uma linha desenhada) por onde “anda”, permitindo ao usuário ter um *feedback* imediato dos comandos dados por ele ao computador, uma vez que, ao mesmo tempo em que move a “tartaruga robô”, o usuário constrói elementos gráficos por meio do “rastro” deixado por ela (Alimisis et al, 2007; Soares; Borges, 2011). É neste processo, de reflexão do usuário sobre os resultados concretos dos comandos dados por ele ao computador e, a partir da observação dos elementos gráficos construídos por meio dos movimentos da tartaruga, que se dá a construção do conhecimento.

O Logo, desde a sua criação por Papert até os dias de hoje, em todas as suas implementações, vem sendo a principal ferramenta de aplicação do “modo de pensar” construcionista, e traz em si uma semente do uso da robótica na educação por meio da utilização da sua tartaruga robô. No entanto, embora as primeiras versões do Logo fizessem uso de uma tartaruga robô real que se movimentava sobre uma folha de papel (Papert, 1993), as versões comumente utilizadas são puramente virtuais, o que oferece uma “arquitetura robótica” com algumas limitações, como a não possibilidade de se observar os possíveis efeitos do ambiente real sobre os experimentos realizados. Além disso, por se tratar de um ambiente 2D, o Logo impossibilita a realização de tarefas mais “elaboradas”

características da robótica propriamente dita que, deste modo, acaba por apresentar-se como sendo uma evolução natural do uso do ambiente de programação Logo como ferramenta de suporte ao Construcionismo (Alimisis et al, 2007).

O próprio Papert nos apresenta em seus estudos a proposta da Robótica Educacional como uma nova disciplina de cunho construcionista:

O esboço desta nova disciplina surgirá gradualmente, e o problema de situá-la no contexto da Escola e no ambiente de aprendizagem maior, será melhor apresentado quando a tivermos na nossa frente. Apresento aqui uma definição preliminar da disciplina – porém apenas como uma semente para discussão – como aquele grão de conhecimento necessário para que uma criança invente (e, evidentemente, construa) entidades com qualidades evocativamente semelhantes à vida dos mísseis inteligentes. Se este grão constituísse a disciplina inteira um nome adequado seria “engenharia de controle” ou até mesmo “robótica” (Papert, 1994, p.160).

No entanto, até pouco tempo, o uso da robótica na educação era inviabilizado pela falta de acesso a robôs, ou aos componentes necessários a sua construção devido, principalmente, ao alto custo desses materiais. Recentes avanços em relação ao preço, tamanho, performance e capacidade dos componentes utilizados para a construção e manipulação de robôs fizeram surgir no mercado uma série de Kits de Robótica Educacional, alguns dos quais proprietários, como o Lego Mindstorms (Mindstorms), e outros baseados em Hardware Livre, como o Arduino (2017), resultando em perceptível crescimento do uso da robótica no campo da educação.

Além dos ganhos para a aprendizagem relacionados ao caráter construcionista do uso da robótica na educação, já abordado, por tratar-se de uma disciplina de caráter multidisciplinar, a robótica permite também que os alunos trabalhem uma grande diversidade de competências e habilidades, à medida que engloba, em um único objeto de estudo, diferentes áreas do conhecimento como matemática, eletrônica, design, mecânica e programação de computadores. O uso da robótica no processo de ensino e aprendizagem também estimula a colaboração, a habilidade de trabalhar em grupo, a elaboração de estratégias para a solução de problemas e o desenvolvimento do pensamento formal⁵ (Souza; Duarte, 2015; López-Rodríguez; Cuesta, 2015; Perez et al, 2013; Rus, 2006; Denis; Hubert, 2001), além de promover uma melhora na capacidade de raciocínio lógico dos

⁵ [O pensamento formal] constitui um sistema complexo, mas coerente, relativamente diferente da lógica da criança: constitui a essência da lógica dos adultos cultivada, assim, como a base das formas elementares do pensamento científico (Piaget, 1972).

alunos, conforme apontam os estudos de Buechley, Eisenberg e Elumeze (2007) e Barbero e Demo (2011).

Robôs instigam naturalmente a curiosidade de crianças e adultos (Perez et al, 2013) motivo este, entre outros tantos, que faz com que:

A Robótica Educacional [...] [venha] emergindo como uma ferramenta de aprendizagem capaz de oferecer atividades práticas e divertidas em um ambiente de aprendizagem atrativo, alimentando o interesse e a curiosidade dos alunos (Eguchi, 2010 *apud* Alimisis, 2013, p.63, tradução autores).

No entanto, embora diferentes estudos apresentem resultados promissores em relação ao uso da robótica na educação, por ser esta uma prática bastante recente, existem ainda muitas áreas obscuras envolvendo esse campo de estudo, o que acaba por fazer com que o papel do uso de robôs no processo de ensino e aprendizagem não esteja perfeitamente claro (Salvini; Korsah; Nourbakhsh, 2015). Isto faz com que trabalhos como este se apresentem como sendo de significativa relevância para o auxílio do esclarecimento de questões ainda em aberto no tocante ao uso da robótica na educação.

2.3. Ensino de Programação para Crianças

Embora o tema “ensino de programação para crianças” tenha surgido com mais força nos últimos anos, este não é necessariamente recente. Como visto na seção anterior, já no final dos anos sessenta, Seymour Papert desenvolvia trabalhos nesse sentido por meio da utilização da linguagem Logo. Desde então, suas pesquisas em relação ao uso de informática na educação avançaram, dando origem a uma série de publicações sobre o tema, como os livros “*Mindstorms: Children, Computers and Powerful Ideas*”, lançado na década de 80 (Papert, 1993), e “*The Children's Machine: Rethinking School In The Age Of The Computer*”, publicado na década de 90 (Papert, 1994).

A revisão da bibliografia nos permite encontrar outros trabalhos das décadas de 80 e 90 (do século passado) dedicados ao estudo dos possíveis benefícios advindos do ensino de programação de computadores a crianças, como o trabalho de Clements e Gullo (1984), que apresenta uma análise dos efeitos da prática de programação de computadores na cognição de crianças com seis anos de idade. Os pós-testes desse trabalho mostraram um

ganho significativo em relação ao pensamento reflexivo⁶ e divergente⁷ das crianças que participaram das atividades de programação de computadores, enquanto que outro grupo, que usou o computador para atividades de Instrução Assistida por Computador (*Computer Assisted Instruction* - CAI), não apresentou ganhos em relação às avaliações de pré-teste.

Os estudos de Papert, e de Clements e Gullo, apontam para a existência de ganhos reais promovidos pelo ensino de programação, tanto na qualidade da aprendizagem de modo geral, quanto em relação a aspectos cognitivos específicos, o que vem a reforçar a ideia de ser esta uma boa prática a ser adotada na Educação Básica.

Além das características positivas apontadas por esses estudos, relacionadas à melhora na qualidade do aprendizado, a crescente - e cada vez mais acelerada - inserção das TIC no dia a dia de toda a sociedade - acabou por trazer uma nova dimensão ao aprendizado de programação por crianças: atualmente, saber programar tornou-se tão essencial quanto saber escrever (Resnick, 2014).

O entendimento de que o aprendizado da programação de computadores é algo essencial na sociedade atual tem feito surgir uma série de iniciativas no sentido de tornar esse aprendizado possível e acessível a qualquer pessoa, desde os primeiros anos da vida escolar até a vida adulta. Algumas dessas iniciativas são a Codecademy (2017), Code.org (2017), Programaê (2017), Code Club Brasil (2017) e o próprio Scratch (2017) de Mitchel Resnick.

Uma rápida pesquisa na internet nos permite encontrar também inúmeras reportagens que apontam o crescente número de escolas brasileiras, em sua grande maioria escolas da Rede Particular de Ensino, que vêm incluindo o ensino de programação no seu currículo, estando esta prática, muitas vezes, associada a atividades de Robótica Educacional.

Diante deste quadro, torna-se fundamental o desenvolvimento de iniciativas voltadas ao Ensino Público, buscando-se evitar com isso uma ampliação ainda maior do *gap* existente entre a qualidade do ensino público e a do ensino privado.

⁶ Pensamento reflexivo é a espécie de pensamento que consiste em examinar mentalmente o assunto e dar-lhe consideração séria e consecutiva (Dewey, 1959, p.13).

⁷ O pensamento divergente está relacionado à criatividade. Ele é responsável pela solução de problemas para os quais ainda não exista uma solução conhecida. Esse tipo de pensamento estaria associado ao levantamento de hipóteses, à capacidade de se imaginar uma variedade de soluções diferentes para um determinado problema (PELAES, 2010).

3. Procedimentos Metodológicos

O método de pesquisa adotado neste trabalho foi a realização de um estudo de caso único, exploratório, de caráter qualitativo com observação participante.

O caso estudado foi uma oficina de aprendizado de programação com robótica, composta por 14 aulas de 90 minutos, realizada no 2o. semestre de 2016, com 7 crianças dos terceiro e quarto anos do ensino fundamental (8 a 10 anos), pertencentes a uma comunidade de baixa renda da cidade do Rio de Janeiro. Todos os estudantes eram alunos de escolas públicas e não tinham conhecimento prévio de programação.

O estudo é exploratório procurando gerar mais informações sobre o fenômeno estudado e buscando levantar hipóteses que possam orientar estudos posteriores (Meirinhos; Osório 2010). A observação realizada é do tipo participante, pelo fato de o pesquisador ser ao mesmo tempo investigador e participante (Yin, 2001), tendo assumido o papel de professor na oficina realizada.

Como aporte teórico utilizado para nortear as observações e sustentar as inferências feitas a partir delas, assumimos o conceito de Pensamento Computacional delineado por Jeannette Wing (2006), a teoria de desenvolvimento cognitivo de Jean Piaget (Moreira, 1999) e o Construcionismo de Seymour Papert (1993).

Para a coleta de dados foram realizados registros das ações dos usuários (feitos com o uso de uma filmadora e das câmeras dos computadores utilizadas pelas crianças), anotações acerca das observações realizadas em campo e questionários aplicados aos participantes.

3.1. As Atividades Didáticas

Por meio de trabalhos práticos, aulas expositivas dialogadas, brincadeiras e narrativas, as atividades propostas na oficina visam - em associação com o ambiente de programação em blocos e com os materiais de robótica - introduzir e desenvolver as estruturas de programação contempladas pelo ambiente de programação visual DuinoBlocksforKids (DB4K).

As atividades práticas referem-se à montagem e à programação de pequenos circuitos, bem como à programação de um conjunto de materiais de robótica desenvolvidos pelos autores. As brincadeiras, inspiradas na técnica de Computação Desplugada (Bell et al. 2009), visam apresentar, de maneira lúdica e divertida, as ideias de sequenciamento e repetições de ações, bem como a ideia de tomada de decisão (utilizadas em estruturas condicionais), que são, posteriormente, trabalhados no DB4K em conjunto com os materiais

de robótica (ver 3.2). As narrativas, associadas aos robôs de sucata disponibilizados para os participantes, são utilizadas para contextualizar o uso dos dispositivos de robótica e das estruturas de programação que serão utilizadas.

3.2. Os Materiais de Robótica

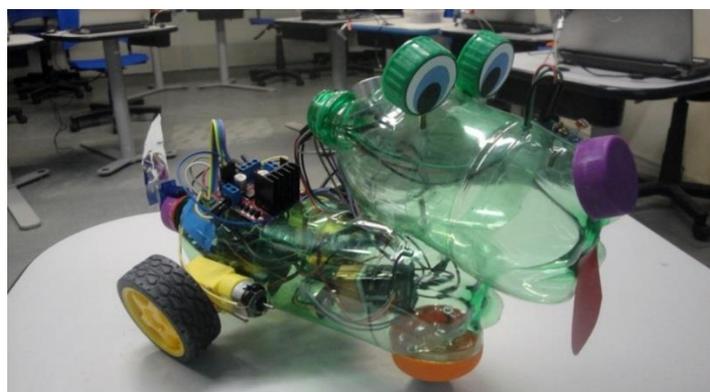
Com o objetivo de possibilitar às crianças o desenvolvimento de programas mais elaborados para o controle de circuitos com um grau de complexidade que seria de difícil construção para sua faixa etária, foram elaborados alguns materiais produzidos com potes plásticos e garrafas pet. Dentre estes materiais utilizamos uma caixinha, nomeada como "caixinha mágica", que permite às crianças programarem simultaneamente um conjunto de atuadores e sensores: um peixe-robô, um cachorro-robô e um morcego-robô, todos construídos com garrafas PET (Figuras 1 e 2). Além desses materiais, disponibilizou-se também kits para iniciantes de RE contendo uma placa Arduino e diferentes componentes eletroeletrônicos para o desenvolvimento dos projetos.

Figura 1 - A "caixinha mágica" e os robôs de garrafa PET



Fonte: Autores

Figura 2 – Cachorro-robô de garrafa PET



Fonte: Autores

4. Resultados

Devido à limitação de páginas do texto, apresentamos uma síntese dos resultados obtidos. A análise mais aprofundada dos acontecimentos do dia a dia da oficina e a evolução de cada participante ao longo do período de quatorze encontros foram publicadas na Dissertação do primeiro autor (Queiroz, 2017).

4.1. Uma visão geral do aproveitamento dos alunos

Dos sete alunos participantes da oficina, três: Daniel⁸ (10 anos, 4º ano), e José (9 anos, 4º ano) e Hilton (8 anos, 3º ano), apresentaram um aproveitamento bastante satisfatório de todo o conteúdo trabalhado. Os outros três alunos: Bruno (9 anos, 4º ano), Marcelo (8 anos, 3º ano) e Ivana (8 anos, 3º ano), apresentaram dificuldades para compreender esses conteúdos, em especial no que se refere às estruturas de repetição. Diana (9 anos, 4º ano), que deixou a oficina antes do término, também apresentou um bom aproveitamento dos conteúdos tratados nas aulas das quais participou.

Bruno, Marcelo e Ivana, ao final da oficina, conseguiram desenvolver programas simples, com poucos comandos, sem o uso de laços de repetição e nos quais a noção de atraso (*delay*) não era necessária (Figura 3).

Figura 3 - Programa sem laços de repetição e sem *delay*

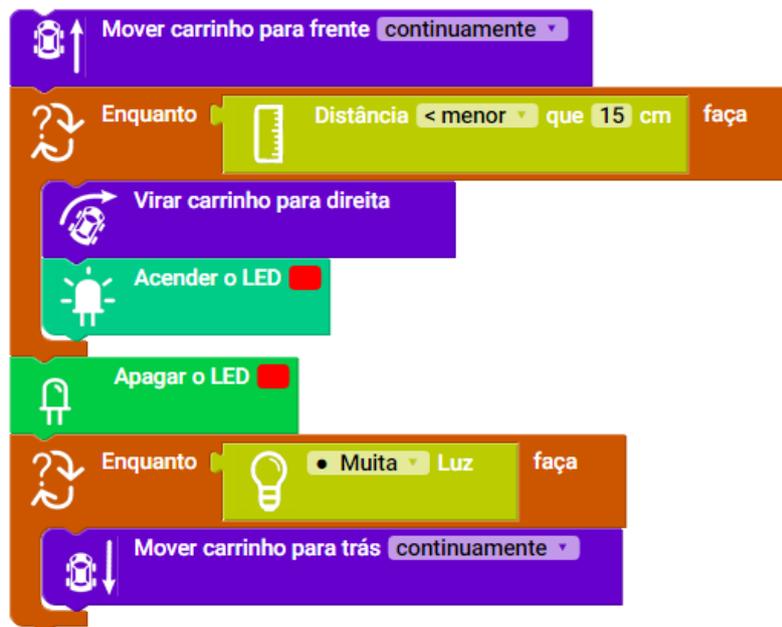


Fonte: Autores

⁸ Os nomes utilizados aqui são fictícios a fim de preservar a identidade dos participantes.

Por outro lado, Daniel, José e Hilton finalizaram a oficina demonstrando claramente a capacidade de construir programas com múltiplos laços de repetição condicional a partir da observação do "comportamento" apresentado pelos materiais de robótica utilizados (Figura 4).

Figura 4 - Programa com mais de 1 laço de repetição condicional



Fonte: Autores

4.2. Construção de programas puramente sequenciais

As respostas apresentadas pelas crianças às atividades desenvolvidas durante as três primeiras aulas, onde somente **programas sequenciais** foram trabalhados, já permitiram que se observasse, de forma bastante clara, a "divisão" da turma nos dois grupos citados anteriormente. Essa diferença na compreensão dos conteúdos acabou por fazer com que as crianças que apresentavam maior facilidade conduzissem em seu grupo a construção dos programas que, nos primeiros encontros, era feita em computadores compartilhados.

Essa dinâmica acabou por fazer com que as crianças que apresentavam maior dificuldade ficassem sem espaço para explorar - a sua maneira e ao seu tempo - os conteúdos trabalhados, o que, possivelmente, contribuiu para ampliar ainda mais a distância entre os resultados apresentados por elas em relação às demais crianças. A partir do sétimo encontro os participantes passaram a operar o ambiente DB4K no modo um computador por

aluno, mantendo-se, no entanto, o trabalho em duplas ou trios para o desenvolvimento das soluções dos problemas propostos.

As maiores dificuldades apresentadas pelas crianças nesse primeiro momento foram:

- Entendimento do efeito resultante do uso do tempo de atraso (*delay*). Essa dificuldade manteve-se presente durante toda a oficina, apresentando-se em diferentes níveis para cada um dos participantes;
- Compreensão do texto dos enunciados dos desafios propostos (leitura funcional⁹), o que levava ao desenvolvimento de programas equivocados oriundos da leitura fragmentada desses textos.

Em relação às habilidades relacionadas ao Pensamento Computacional foi possível observar, já nos três primeiros encontros, o exercício das seguintes habilidades:

- **Depuração e detecção sistemática de erros:** correção dos programas a partir da observação do resultado da sua execução nos materiais de robótica;
- **Capacidade de abstração:** construção mental da sequência de ações necessária para a obtenção de um determinado resultado e sua posterior materialização na forma de um programa de computador;
- **Noções algorítmicas de fluxo de controle:** sequenciamento adequado dos blocos (comandos) responsáveis por fazerem os materiais de robótica realizarem a sequência de ações esperada.

4.3. Uso da Estrutura de Repetição Contada

Ao final dos cinco encontros dedicados ao aprendizado da **estrutura de repetição contada**, Daniel, Hilton e José apresentavam-se claramente capazes de identificar *quando e como* utilizar este tipo de estrutura. O mesmo não se deu com Bruno, Marcelo e Ivana que, durante o tempo em que os computadores foram compartilhados, não conseguiam participar de forma mais ativa da tarefa de desenvolver os programas e, uma vez tendo sido disponibilizado um computador por aluno, embora apresentassem um empenho inicial na busca por construir os programas por conta própria, acabavam por finalizá-los olhando os programas dos colegas.

As maiores dificuldades apresentadas pelas crianças nessa segunda etapa foram:

⁹ Aquela que é utilizada para fins pragmáticos, em contextos cotidianos (RIBEIRO, 1997).

- Entendimento de que apenas os comandos posicionados dentro do bloco *repetir* seriam repetidos o número de vezes indicado como parâmetro. A tendência das crianças era acreditar que todos os comandos posicionados após o comando *repetir* (ou seja, a parte superior do bloco repetir) seriam repetidos, independente de esses comandos estarem posicionados dentro ou abaixo do bloco.
- Manteve-se a dificuldade com o uso do bloco *esperar (delay)*. Essa dificuldade era evidenciada pelo fato de as crianças, de modo geral, não conseguirem, sem auxílio do professor, montar a sequência de ações necessária para se observar um bloco piscar, ou seja: acender o LED → esperar um tempo (*delay*) → apagar o LED → esperar um tempo. Algumas colocavam apenas o bloco *acender LED*; outras apenas os blocos *acender e apagar LED*; outras ainda colocavam o bloco *esperar* apenas depois do bloco *acender LED*. No decorrer da oficina os alunos com melhor aproveitamento passaram a entender qual a sequência de comandos necessária para fazer o bloco piscar. As demais crianças finalizaram a oficina sem, aparentemente, demonstrar esse entendimento.

As habilidades relacionadas ao Pensamento Computacional que puderam ser observadas nessa segunda etapa foram:

- **Depuração e detecção sistemática de erros:** correção dos programas a partir da observação do resultado da sua execução nos materiais de robótica.
- **Capacidade de abstração:** abstrair um programa de computador a partir da observação de eventos do mundo real, ou seja, construir mentalmente um algoritmo (uma sequência de ações que representa um evento observado). Esse tipo de habilidade pode ser constatado nos exercícios nos quais os alunos assistiam a vídeos com a caixinha mágica apresentando um determinado "comportamento" e, apenas com base nessa observação, construíam um programa que fizesse a caixinha funcionar conforme o observado.
- **Noções algorítmicas de fluxo de controle:** reordenação correta dos comandos presentes no programa para que este viesse a refletir mudanças ocorridas na ordem das ações realizadas pelos materiais de robótica. Isso podia ser constatado com base no seguinte tipo de exercício: assistir um comportamento da caixinha mágica como, por exemplo: piscar o LED vermelho 2 vezes, girar o motor DC por 5 segundos, parar o motor DC e, então, piscar o LED amarelo 3 vezes e construir um programa que fizesse a caixinha funcionar conforme o observado. Ou ainda, observar a caixinha apresentando um segundo comportamento, no qual a diferença para o comportamento anterior é a ordem na realização das ações, como por exemplo: piscar o LED vermelho 2 vezes, piscar o LED amarelo 3 vezes, girar o motor DC por 5 segundos e depois parar o motor DC; reordenar os blocos no

programa para que ele refletisse a mudança de comportamento apresentada pela caixinha mágica.

4.4. Uso da Estrutura de Repetição Condicional

Concluídos os cinco encontros durante os quais foi trabalhada a **estrutura de repetição condicional**, Daniel, Hilton e José, conforme já mencionado em 4.1, demonstravam com clareza a capacidade de construir programas com múltiplos laços de repetição condicional. Já, Bruno, Ivana e Marcelo conseguiam desenvolver apenas programas puramente sequenciais bastante simples, que não necessitassem do entendimento da noção de atraso (*delay*).

A principal dificuldade apresentada nesta última etapa foi compreender o funcionamento "não determinístico" dos programas que fazem uso de uma estrutura condicional. Ou seja, os programas *puramente sequenciais* ou que utilizam apenas *estruturas de repetição contada* geram como resultado da sua execução um comportamento sempre idêntico ao do material de robótica programado (lembrando aqui que o Duinoblocks4Kids não possibilita o uso de variáveis e que o código nele construído é, por *default*, executado dentro de um "loop infinito"). Por exemplo, se for definido neste tipo de programa que o LED amarelo vai piscar 3 vezes e depois o LED azul 5 vezes, esses LEDs sempre piscarão esse número de vezes, nessa sequência, enquanto o programa permanecer rodando. No caso da *estrutura de repetição condicional*, quando uma determinada condição é satisfeita (um aumento na intensidade da luz, a aproximação de um objeto, etc.), o comportamento apresentado pelos materiais de robótica se modifica em tempo de execução. Assim, quando do uso do bloco *enquanto*, o número de repetições das ações associadas a uma determinada condição é definido por um fator externo, sendo essas ações repetidas tantas vezes quantas forem possíveis durante o tempo em que a condição dada se mantiver satisfeita e não de forma constante, como no caso da *estrutura de repetição contada*.

No que se refere ao Pensamento Computacional, as seguintes habilidades puderam ser observadas:

- **Depuração e detecção sistemática de erros:** correção dos programas a partir da observação do resultado da sua execução nos materiais de robótica.
- **Capacidade de abstração:** em adição às demonstrações do uso da capacidade de abstração anteriormente mencionadas e igualmente evidenciadas nesta última etapa, pôde-se também observar o exercício desta habilidade nas ocasiões em que algumas crianças testavam na caixinha mágica programas destinados a outro material de robótica, como, por

exemplo, o peixe-robô. Ou seja, elas percebiam que, embora os sensores e atuadores presentes nos dois materiais estivessem organizados de diferentes maneiras e por vezes apresentassem aspectos distintos, por se tratarem dos mesmos dispositivos, exibiriam o mesmo comportamento quando controlados pelo mesmo programa.

- **Noções algorítmicas de fluxo de controle:** soma-se às demonstrações anteriormente mencionadas (e também observadas durante esta etapa), a percepção das crianças (ainda que não necessariamente consciente) de que, com o uso da *estrutura de repetição condicional*, o fluxo do programa passa a ser determinado por uma condição externa variável, o que pode tornar esse fluxo "inconstante".

- **Lógica Condicional:** o uso da lógica condicional tornou-se evidente quando as crianças passaram a construir programas com múltiplos laços de repetição condicional a partir da observação do comportamento de materiais de robótica que faziam uso de mais de um sensor. Por exemplo: ao observarem o carrinho robótico se desviar de obstáculos ao aproximar-se cerca de 15cm deles (usando para isso um sensor de distância), andando para trás enquanto uma luz forte estivesse incidindo sobre ele (fazendo uso de um sensor de luz) e andando para frente enquanto nenhuma das condições anteriores era satisfeita, as três crianças com melhor aproveitamento eram capazes de, com base na observação desse comportamento, construir no DB4K um programa que fizesse o carrinho funcionar exatamente da maneira observada sendo necessário o uso de duas estruturas de repetição condicional (uma para cada sensor).

- **Decomposição estruturada de problemas:** algumas das práticas realizadas na oficina indicaram ser perfeitamente possível, mesmo em programas muito simples, trabalhar essa habilidade (ainda que de maneira bastante rudimentar). Por exemplo, ao mostrar para as crianças um material de robótica apresentando uma sequência de três ações distintas (piscar constantemente um LED; girar um motor DC lentamente quando um objeto se aproxima; girar o motor com muita velocidade quando a temperatura estiver alta), elas foram capazes de dividir o comportamento observado em partes e procuraram programar cada uma dessas partes separadamente, acrescentando pouco a pouco cada uma delas no código e testando o programa a cada nova etapa. No caso do programa acima apresentado, as crianças podiam primeiro, tentar fazer o LED piscar, depois fazer o motor DC girar lentamente enquanto algum objeto se mantivesse próximo e, por fim, adicionar ao programa o código responsável por fazer o motor girar rapidamente enquanto a temperatura está alta.

5. Conclusões

O presente trabalho apresentou um estudo exploratório envolvendo crianças do 3º e do 4º anos do Ensino Fundamental I trabalhando com robótica educacional e a linguagem visual de programação DuinoBlocksforKids, com o objetivo de investigar possibilidades de desenvolvimento de habilidades relacionadas ao Pensamento Computacional.

Os resultados obtidos sugerem que a partir de um bom planejamento pedagógico e utilizando ferramentas adequadas, é possível trabalhar algumas das habilidades do PC apresentadas na literatura com crianças nessa faixa etária. A mesma análise também indica que o *timing* de desenvolvimento das referidas habilidades está relacionado à idade das crianças envolvidas.

5.1. Considerações sobre o Trabalho Realizado

Apesar de o trabalho aqui apresentado ter sido desenvolvido em um contexto de "sala de aula", onde os participantes tinham dia e horário para comparecer, tarefas a serem realizadas e objetivos educacionais a serem atingidos, algumas diferenças em comparação com a realidade das nossas escolas podem ter interferido de forma positiva, nos resultados obtidos:

- O número de participantes da turma era muito menor do que aquele que encontramos nas salas de aula, o que permitiu uma maior atenção por parte do "professor" para cada um deles;
- A quase totalidade das tarefas propostas lançava mão do lúdico e do imaginário das crianças para envolvê-las, de forma verdadeira, na busca das soluções para os desafios propostos;
- O professor-pesquisador estava altamente motivado, sendo também coautor das atividades propostas aos participantes;
- O professor-pesquisador tem uma formação sólida na área de informática, bem como no campo da educação.

Tais aspectos da pesquisa realizada nos levam a uma reflexão sobre o papel das nossas Licenciaturas e das nossas salas de aula com tecnologias.

Independente dos objetivos educacionais ou de mercado a serem atingidos com o desenvolvimento do PC, o que percebemos na grande maioria das escolas hoje é uma ação desarticulada entre as disciplinas tradicionais do currículo e as atividades no

laboratório de informática. Mais ainda, como coloca Buckingham (2007, apud Valente, 2016, p. 866):

na maior parte do tempo estamos utilizando um potencial muito limitado dessas tecnologias, uma vez que nos restringimos ao uso de software de escritório como os processadores de texto e planilhas (Buckingham, 2007, apud Valente, 2016, p. 866)

Esta desarticulação e limitação no uso das tecnologias de informação e comunicação nas escolas têm suas raízes - ao menos no Brasil - na formação docente sob a responsabilidade das diferentes Licenciaturas.

Na grande maioria desses cursos o tema Informática na Educação ou não é abordado ou é tratado apenas uma vez, em uma única disciplina, ao longo dos quatro ou cinco anos da graduação. Mais ainda, muitas das nossas faculdades e universidades que oferecem Programas de Pós-Graduação em Educação não contam com linhas de pesquisa na área de tecnologias no ensino. Como querer então que os recém-formados professores exerçam sua criatividade quanto ao uso das TIC no ensino, se não foram preparados para isso?

Vale ainda ressaltar que o ensino de programação em tais cursos é praticamente inexistente e, quando existe, acontece apenas nas áreas de ciências exatas, e mesmo assim deixando ausentes debates sobre as ideias e propostas pedagógicas.

É preciso que o uso das tecnologias no ensino e uma reflexão crítica sobre suas possibilidades, limitações, casos de sucesso e de insucesso, permeiem todo o processo de formação dos futuros docentes para que possamos vislumbrar, no médio prazo, mudanças nas atitudes em sala de aula.

Precisamos também de mudanças nas políticas públicas em educação no sentido de valorizar a ação docente - de forma que os professores não necessitem atuar em dois ou mais estabelecimentos de ensino a fim de completarem uma renda mínima para o seu sustento - viabilizando assim a proposta de "professor de uma única escola".

6. Referências

ALFARO, S. A. Robôs em Projetos Tecnológicos. **Anais da 58ª Reunião Anual da SBPC**. Florianópolis: [s.n.]. 2006. Disponível em: <http://www.sbpnet.org.br/livro/58ra/atividades/TEXTOS/texto_884.html>. Acesso em: 12 out. 2017.

ALIMISIS, D. Educational Robotis: Open questions and new challenges. **Themes in Science and Technology Education**, v. 6, n. 1, p. 63-71, 2013. Disponível em: <<http://earthlab.uoi.gr/theste/index.php/theste/article/view/119>>. Acesso em: 12 out. 2017.

ALIMISIS, D.; MORO, M.; ARLEGUI, J.; PINA, A.; FRANGOU, S.; PAPANIKOLAOU, K. Robotics & Constructivism in Education: the TERECoP project. **Proceedings of the 11th European Logo Conference** (Eurologo 2007). Bratislava: [s.n.]. 2007. p. 19-24. Disponível em: <http://users.sch.gr/adamopou/docs/syn_eurologo2007_alimisis.pdf>. Acesso em: 12 out. 2017.

ARDUINO. **Página do Arduino**. Disponível em: <<https://www.arduino.cc/>>. Acesso em: 12 out. 2017.

ASIMOV, I. **Eu, robô**. Tradução de Aline Storto Pereira. São Paulo: Aleph, 2014. 320 p.

BARBERO, A.; DEMO, B. G. The art of programming in a Technical Institute after the Italian secondary school reform. **Proceedings ISSEP 2011**. Bratislava: [s.n.]. 2011. Disponível em: <<http://pubshop.bmbf.gv.at/download.aspx?id=573>>. Acesso em: 12 out. 2017.

BARCELOS, T. S.; SILVEIRA, I. F. Pensamento Computacional e Educação Matemática: Relações para o Ensino de Computação na Educação Básica. **Anais do XXXII CSBC**. Curitiba: Sociedade Brasileira de Computação. 2012.

BARR, V.; STEPHENSON, C. Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community? **ACM Inroads**, v. 2, n. 1, p. 48-54, Março 2011. Disponível em: <<http://dl.acm.org/citation.cfm?id=1929905>>. Acesso em: 12 out. 2017.

BELL, T.; ALEXANDER, J.; FREEMAN, I.; GRIMLEY, M. Computer science unplugged: School students doing real computing without computers. **The New Zealand Journal of Applied Computing and Information Technology**, v. 13, n. 1, p. 20-29, 2009. Disponível em: <<http://www.computingunplugged.org/sites/default/files/papers/Unplugged-JACIT2009submit.pdf>>. Acesso em: 12 out. 2017.

BUECHLEY, L.; EISENBERG, M.; ELUMEZE, N. Towards a curriculum for electronic textiles in the high school classroom. **Proceedings of the 12th annual SIGCSE conference on Innovation and technology in computer science education**. Dundee: [s.n.]. 2007. p. 28-32. Disponível em: <<http://dl.acm.org/citation.cfm?id=1268795>>. Acesso em: 12 out. 2017.

CLEMENTS, D. H.; GULLO, D. F. Effects of computer programming on young children's cognition. **Journal of Educational Psychology**, v. 76, n. 6, p. 1051-1058, 1984. Disponível em: <<http://psycnet.apa.org/journals/edu/76/6/1051/>>. Acesso em: 9 out. 2017.

CODE.ORG. **Página do Code.Org**. 2017. Disponível em: <<https://code.org/>>. Acesso em: 9 out 2017.

CODECADEMY. **Página do Code Academy**. 2017. Disponível em: <<https://www.codecademy.com/pt>>. Acesso em: 9 out 2017

CODECLUBBRASIL. **Página do Code Clube Brasil**. 2017. Disponível em: <<http://codeclubbrasil.org/>>. Acesso em: 9 out 2017

CUNY, J.; SNYDER, L.; WING, J. M. Demystifying Computational Thinking for Non-Computer Scientists . Manuscrito não publicado. 2010. Disponível em <<http://www.citeulike.org/user/jehicken/article/13256108>>. Acesso em: 12 out. 2017.

DENIS, B.; HUBERT, S. Collaborative learning in an educational roboTIC environment. **Computers in Human Behavior**, v. 17, n. 5, p. 465-480, set-nov 2001. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0747563201000188>>. Acesso em: 12 out. 2017.

GOOD, J.; HOWLAND, K. Programming language, natural language? Supporting the diverse computational activities of novice programmers. **Journal of Visual Languages & Computing**. Disponível em: <http://dx.doi.org/10.1016/j.jvlc.2016.10.008>. Acesso em: 12 out. 2017.

GROVER, S.; PEA, R. Computational Thinking in K-12: A Review of the State of the Field. **Educational Researcher**, Florida, v. 42, n. 1, p. 38–43, jan 2013. Disponível em: <<http://journals.sagepub.com/doi/abs/10.3102/0013189X12463051>>. Acesso em: 11 out. 2017.

HEMMENDINGER, D. A plea for modesty. **Acm Inroads**, New York, v. 12, p. 4-7, Jun 2010. Disponível em: <<http://dl.acm.org/citation.cfm?id=1805725>>. Acesso em: 11 out. 2017.

KLOC, A. E.; KOSCIANSKI, A.; PILATTI, L. A. Robótica: uma ferramenta pedagógica no campo da Computação. **Anais do I Simpósio Nacional de Ensino de Ciência e Tecnologia - UTFPR**. Ponta Grossa: [s.n.]. 2009. p. 1394-1403. Disponível em: <http://www.sinect.com.br/anais2009/artigos/11%20TICnoensinoaprendizagemdecienciaetecnologia/TICnoensinoaprendizagemdecienciaetecnologia_artigo15.pdf>. Acesso em: 12 out. 2017.

LÓPEZ-RODRÍGUEZ, F. M.; CUESTA, F. Andruino-A1: Low-Cost Educational Mobile Robot Based on Android and Arduino. **Journal of Intelligent & Robotic Systems**, v. 81, n. 1, p. 63-76, maio 2015. Disponível em: <<http://link.springer.com/article/10.1007/s10846-015-0227-x>>. Acesso em: 11 out. 2017.

MEC. **Base Nacional Comum Curricular**. Proposta Preliminar. Segunda Versão Revisada. Brasília: MEC. 2016.

MEIRINHOS, M.; OSÓRIO, A. O estudo de caso como estratégia de investigação em educação. **EduSer - Revista de Educação**, Bragança, v. 2, n. 2, p. 49-65, 2010. Disponível em: <<http://hdl.handle.net/10198/3961>>. Acesso em: 18 jan. 2017.

MINDSTORMS. Página da LEGO. Disponível em: <<https://www.lego.com/en-us/mindstorms>>. Acesso em: 12 out. 2017.

MOREIRA, M. A. **Teorias de aprendizagem**. São Paulo: EPU, 1999. 195 p.

NATIONAL CURRICULUM ENGLAND. 2013. Disponível em: <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study> Acesso em: 15 out. 2017

PAPADAKIS, S.; KALOGIANNAKIS, M.; ORFANAKIS, V.; ZARANIS, N. Novice Programming Environments. Scratch & App Inventor: a first comparison. In: **Proceedings of**

the 2014 Workshop on Interaction Design in Educational Environments (p. 1). ACM, 2014.

PAPERT, S. **Mindstorms: Children, Computers and Powerful Ideas**. 2. Ed. New York: BasicBooks, 1993. 252 p.

PAPERT, S. **The Children's Machine: Rethinking School In The Age Of The Computer**. New York: BasicBooks, 1994. 256 p.

PEREZ, A. L. F.; DAROS, R. R.; PUNTEL, F. M.; VARGAS, S. R. Uso da Plataforma Arduino para o Ensino e o Aprendizado de Robótica. **Anais International Conference on Interactive Computer Aided Blended Learning (ICBL)**. Florianópolis: IAOE. 2013. P. 230-232. Disponível em: <http://www.icbl-conference.org/proceedings/2013/papers/Contribution77_a.pdf>. Acesso em: 11 out. 2017.

PIAGET, J. **To understand is to invent**. New York: Grossman Publishers, 1972. 148 p.

PINTO, M. D. C. **Aplicação de Arquitetura Pedagógica em Curso de Robótica Educacional com Hardware Livre**. 2011. 158 f. Dissertação (Mestrado em Informática). Universidade Federal do Rio de Janeiro. Rio de Janeiro. 2011.

PROGRAMAE. **Página do Programaê**. 2017. Disponível em: <<http://www.fundacaolemann.org.br/programae/>>. Acesso em: 12 out. 2017.

QUEIROZ, R. L. **DuinoBlocks4Kids: utilizando Tecnologia Livre e materiais de baixo custo para o exercício do Pensamentos Computacional no Ensino Fundamental I por meio do aprendizado de programação aliado à Robótica Educacional**. 2017. 138 f. Dissertação (Mestrado em Informática) Programa de Pós-Graduação em Informática. Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2017.

RESNICK, M. Aprender a programar, programar para aprender. **Transformar 2014**, 2014. Disponível em: <<https://www.youtube.com/watch?v=hRGJUc2opQ4>>. Acesso em: 11 out. 2017.

RUS, D. Teaching robot everywhere. **IEEE RobotiC & Automation Magazine**, v. 13, n. 1, p. 15-94, mar 2006. Disponível em: <https://www.eng.yale.edu/grablab/roboticscourseware/Rus_RAM.pdf>. Acesso em: 11 out. 2017.

SALVINI, P.; KORSAH, A.; NOURBAKHSI, I. Special Issue on Educational Robotics: Call for Papers. **IEEE Robotics & Automation Magazine**, 2015. Disponível em: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7124590>>. Acesso em: 12 out. 2017.

SBC. **SBC realiza reunião para discutir a nova versão da Base Nacional Comum Curricular**. Sociedade Brasileira de Computação, 2016a. Disponível em: <<http://www.sbc.org.br/noticias/1693-sbc-realiza-reuniao-para-discutir-a-nova-versao-da-base-nacional-comum-curricular>>. Acesso em: 11 out. 2017.

SBC. **SBC realiza reunião com o Ministro da Educação**. Sociedade Brasileira de Computação, 2016b. Disponível em: <<http://www.sbc.org.br/noticias/1868-sbc-realiza-reuniao-com-o-ministro-da-educacao>>. Acesso em: 11 out. 2017.

SCRATCH. **Página do Scratch**. 2017. Disponível em: <<https://scratch.mit.edu/>>. Acesso em: 12 out. 2017.

SOARES, R. F.; BORGES, M. A. F. Robótica: aprendizado em informática de forma lúdica. **Anais do XXXI Congresso da Sociedade Brasileira de Computação**. Natal: [s.n.]. 2011. p. 1516-1519. Disponível em: <<http://www.dimap.ufrn.br/csbc2011/anais/eventos/contents/CADERNO.pdf>>. Acesso em: 12 out. 2017.

SOLOMON, C. J.; PAPERT, S. A case study of a young child doing Turtle Graphics in LOGO. **Proceedings of the June 7-10, 1976, National Computer Conference and exposition**. New York: ACM. 1976. p. 1049-1056. Disponível em: <<http://dl.acm.org/citation.cfm?id=1499945>>. Acesso em: 11 out. 2017.

SOUZA, M. A. M.; DUARTE, J. R. R. Low-cost educational robotics applied to physics teaching in Brazil. **Physics Education**, v. 50, n. 4, p. 482-488, jun 2015. Disponível em: <<http://iopscience.iop.org/article/10.1088/0031-9120/50/4/482/meta>>. Acesso em: 11 out. 2017.

TCHÁPEK, K. **A Fábrica de Robôs**. Tradução de Vera Machac. São Paulo: Hedra, 2012. 148 p.

VALENTE, J. A. Integração do Pensamento Computacional no Currículo da Educação Básica: Diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. **Revista e-Curriculum**. PUC-SP., v. 14, n. 3, p. 866, 2016.

WING, J. M. Computational thinking. **Communications of the ACM**, New York, v. 49, n. 3, p. 33-35, mar 2006. Disponível em: <<http://dl.acm.org/citation.cfm?id=1118178.1118215&coll=portal&dl=ACM>>. Acesso em: 11 out. 2017.

WING, M. Computational Thinking-What and Why? **The Link Magazine**, Pittsburgh, p. 20-23, Spring 2011. Disponível em: <http://www.cs.cmu.edu/sites/default/files/11-399_The_Link_Newsletter-3.pdf>. Acesso em: 12 out. 2017.

YIN, R. K. **Estudo de Caso: Planejamento e Métodos**. 2. ed. Porto Alegre: Bookman, 2001. 205 p.