

DuinoBlocks4Kids: Utilizando Tecnologia Livre e Materiais de Baixo Custo para o Exercício do Pensamento Computacional no Ensino Fundamental I por meio do Aprendizado de Programação Aliado à Robótica Educacional.

Rubens Lacerda Queiroz¹, Fábio Ferrentini Sampaio^{1,2}, Mônica Pereira dos Santos³

¹Programa de Pós-Graduação em Informática (PPGI/UFRJ)

²Instituto Tércio Pacitti de Aplicações e Pesquisas Computacionais (NCE/UFRJ)

³Programa de Pós-Graduação em Educação da UFRJ (PPGE/UFRJ)

rubensqueiroz@outlook.com, ffs@nce.ufrj.br, monicapes@gmail.com

***Abstract.** This paper presents an investigation about the practice of some Computational Thinking abilities with primary school children while learning some basic concepts of computer programming, using resources based on Free Technology and low-cost materials, combined with pedagogical strategies based on Educational Robotics. Hypotheses about a possible relationship between cognitive maturity of children aged between 8 and 10 and the learning of basic structures of computer programming are also raised. To carry out this study, a teaching kit including a Visual Programming Language for Arduino, a series of activities and a set of robotic materials have been developed.*

***Resumo.** Este artigo apresenta uma investigação acerca do exercício de habilidades do Pensamento Computacional em crianças do Ensino Fundamental I, a partir do aprendizado de programação, por meio de recursos baseados em Tecnologia Livre e materiais de baixo custo associados a estratégias pedagógicas alicerçadas na Robótica Educacional. São também levantadas hipóteses acerca da possível relação existente entre a maturidade cognitiva de crianças entre 8 e 10 anos e o aprendizado de estruturas básicas de programação. Foi desenvolvido, para a realização do estudo, um kit didático composto por um Ambiente de Programação Visual em Blocos, uma série de atividades e um conjunto de materiais de robótica.*

1. Introdução

O aprendizado de programação de computadores, já nos primeiros anos do ensino fundamental, tem sido visto como uma importante ferramenta para o desenvolvimento e exercício de algumas habilidades do Pensamento Computacional [Grover e Pea 2013]: uma forma de pensamento característica dos cientistas da computação, mas, universalmente aplicável, que envolve um conjunto de atitudes e competências, tais como o uso da depuração, recursividade, generalização e decomposição na solução de problemas tanto técnico-científicos quanto da vida cotidiana [Wing 2006].

Outro aspecto que tem recebido crescente atenção em relação a esse tema, e que acaba por apontar a programação de computadores como sendo uma nova habilidade básica, tal qual a escrita, a leitura e a aritmética, é a importância do seu desenvolvimento

como meio de possibilitar a todo cidadão ser produtor e não apenas consumidor de tecnologia digital, oportunizando assim, entre outras coisas, a mobilidade social em um mundo fortemente orientado por esse tipo de tecnologia [Whitehouse 2016].

Diante deste cenário, diferentes autores, como Mitchel Resnick e outros (2009), vêm defendendo a necessidade de se incorporar o ensino de programação já nos primeiros anos do Ensino Básico (K12 nos EUA), e, para que iniciativas nesse sentido sejam adotadas, torna-se necessário o desenvolvimento de recursos idealizados especificamente para o ensino de programação de computadores a crianças. Como resposta a essa demanda, uma série de estratégias vêm sendo pensadas e implementadas, entre elas, a associação da Robótica Educacional com Ambientes de Programação Visual em Blocos.

Para que os benefícios do aprendizado de programação com robótica possam ser levados a um número maior de crianças, especialmente aquelas pertencentes a comunidades de baixa renda e matriculadas em escolas públicas, torna-se fundamental o desenvolvimento de estratégias de aprendizado economicamente acessíveis, no que o uso de Tecnologias Livres e materiais de baixo custo tornam-se grandes aliados.

Dentre as tecnologias atualmente existentes para o desenvolvimento de atividades na área de Robótica Educacional, a plataforma de prototipagem eletrônica Arduino¹ vem se destacando devido ao seu baixo custo, qualidade, flexibilidade e facilidade de uso. No entanto, uma revisão recente da literatura na área de ambientes de programação visual voltados ao ensino de programação com robótica, realizada pela equipe do laboratório de Inovações em Robótica Educacional da UFRJ (LIVRE)², apontou para a inexistência de ambientes de programação voltados à programação de placas de prototipagem eletrônica Arduino, pensados e desenvolvidos especificamente para o ensino de programação a crianças.

Como resposta a essa demanda, surge a proposta do desenvolvimento do DuinoBlocks4Kids (DB4K), um kit didático baseado em Tecnologia Livre, voltado ao ensino de programação para crianças do Ensino Fundamental I, composto por um Ambiente de Programação Visual baseado em blocos, um conjunto de materiais de Robótica Educacional e uma série de atividades.

O objetivo geral do estudo aqui apresentado é, a partir do uso do kit DB4K, investigar a possibilidade de desenvolvimento e exercício de habilidades do Pensamento Computacional em crianças do Ensino Fundamental I a partir do aprendizado de conceitos básicos de programação por meio de recursos baseados em Tecnologia Livre e materiais de baixo custo, associados a estratégias pedagógicas alicerçadas na Robótica Educacional, pensadas e desenvolvidas especificamente para esse público.

Em paralelo, busca-se levantar hipóteses acerca da existência de uma relação direta entre certos aspectos da maturidade cognitiva de crianças entre 8 e 10 anos, tais como a habilidade de sequenciar eventos ou ideias e a habilidade de realizar operações mentais a partir de experiências concretas, dentre outras, e as habilidades necessárias para a realização de determinadas tarefas relacionadas à programação de computadores.

A realização deste tipo de investigação, especialmente no que se refere à capacidade de abstração, torna-se bastante importante no sentido de auxiliar os

¹ www.arduino.cc

² <http://www.nce.ufrj.br/ginape/livre/>

educadores quanto à escolha das atividades de programação a serem realizadas com as crianças, de modo que essas atividades sejam oferecidas no momento apropriado e de maneira adequada.

O presente artigo está organizado da seguinte forma: na próxima seção é apresentada a fundamentação teórica, na seção 3 são descritos os procedimentos metodológicos adotados, na seção 4 é feita uma breve descrição do kit DB4K, na seção 5 são apontados os principais resultados e, na última seção, são expostas as conclusões, as principais contribuições e sugestões de trabalhos futuros.

2. Fundamentação Teórica

2.1. Pensamento Computacional

“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 Hemmendinger (2010), o objetivo do desenvolvimento do Pensamento Computacional 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 [Barr e Stephenson 2011].

Dentre as habilidades comumente relacionadas ao Pensamento Computacional [Grover e Pea 2013], entendeu-se como sendo possíveis de serem trabalhadas por meio do uso do Kit DuinoBlocks4Kids (ver seção 4), as seguintes habilidades: capacidade de abstração, entendimento de fluxos de controle, uso da lógica condicional, decomposição de problemas e depuração e detecção sistemática de erros.

2.2. O período operatório concreto e o aprendizado de programação

Jean Piaget distingue quatro períodos gerais no que se refere ao desenvolvimento cognitivo, sendo eles: sensório motor (0 a 2 anos), pré-operatório (2 a 7/8 anos), operatório concreto (7/8 a 11/12 anos), operatório formal (11/12 anos em diante) [Moreira 1999].

No período operatório concreto, momento em que se encontravam os participantes do presente estudo, as crianças começam a realizar operações mentalmente e não apenas por meio de ações físicas como acontece no período pré-operatório [Terra 2010]. Em outras palavras: "o sujeito se torna capaz de reconstruir no plano da representação o que já havia construído no plano da ação" [Souza e Wechsler 2014, p.144]. Outro aspecto fundamental em relação a esse período é que, para realizar as operações, a criança recorre sempre a “objetos concretos” presentes ou já experimentados (daí a designação operatório concreto), sendo bastante limitada a realização de operações concretas em direção ao ausente [Furtado, Bock e Teixeira 2001], ou seja, a construção de abstrações.

A capacidade de abstração, considerada como sendo a "pedra fundamental" do Pensamento Computacional [Grover e Pea 2013], ainda é limitada neste período, tendo-

se o predomínio do uso da abstração empírica (ou simples), que consiste na capacidade da criança de focalizar em uma determinada propriedade de um objeto e ignorar as demais [Kamii 1992]. Embora limitado à realização de construções mentais a partir de referenciais concretos, esse tipo de abstração se encaixa perfeitamente em um dos aspectos relacionados à abstração apresentados por Kramer (2007) como sendo fundamentais para os cientistas da computação, qual seja: o processo de remover detalhes para simplificar e focar a atenção, seja “retirando alguma coisa” ou desconsiderando uma ou mais propriedades de um objeto complexo para prestar atenção em outros.

Além das questões ligadas à capacidade de abstração, uma análise das características cognitivas de crianças no período operatório concreto nos possibilita aventar a possibilidade de existência de uma relação direta entre algumas dessas competências e certas habilidades relacionadas à programação de computadores, como por exemplo: a capacidade de reaproveitar código estaria relacionada à reversibilidade; a percepção da ideia de processamento estaria relacionada à capacidade de estabelecer corretamente relações de causa e efeito, meio e fim; a construção de um trecho completo de código (a partir de um resultado concreto esperado) e a “visualização mental” do resultado da execução desse trecho de código estaria relacionada à habilidade de realizar operações mentais (a partir de experiências concretas) e de pensar simultaneamente no todo e nas suas partes; o entendimento de que os mesmos comandos organizados de diferentes maneiras podem levar a diferentes resultados e de que um mesmo problema pode apresentar diferentes soluções estaria relacionado à capacidade de trabalhar simultaneamente com mais de um ponto de vista sobre uma mesma ideia.

3 Procedimentos Metodológicos

O método de pesquisa adotado por este 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 com 7 crianças dos terceiro e quarto anos do ensino fundamental pertencentes a uma comunidade de baixa renda da cidade do Rio de Janeiro, matriculadas em escolas públicas, na qual foi adotado o kit didático DuinoBlocks4Kids.

O estudo é exploratório pois procura gerar mais informações sobre o fenômeno estudado, buscando levantar hipóteses que possam orientar estudos posteriores [Meirinhos e Osório 2010] e a observação realizada é participante pois o pesquisador foi 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, temos 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 usados pelas crianças), anotações acerca das observações realizadas em campo e questionários.

4. O Kit DuinoBlocks4Kids (DB4K)

É feita aqui uma breve apresentação do DB4K. Maiores detalhes sobre o ambiente de programação, a proposta metodológica, as atividades didáticas e os materiais de robótica que compõem o Kit podem ser encontrados nos trabalhos de Queiroz e Sampaio (2016) e Queiroz, Sampaio e dos Santos (2016).

4.1. O Ambiente de Programação Visual em Blocos

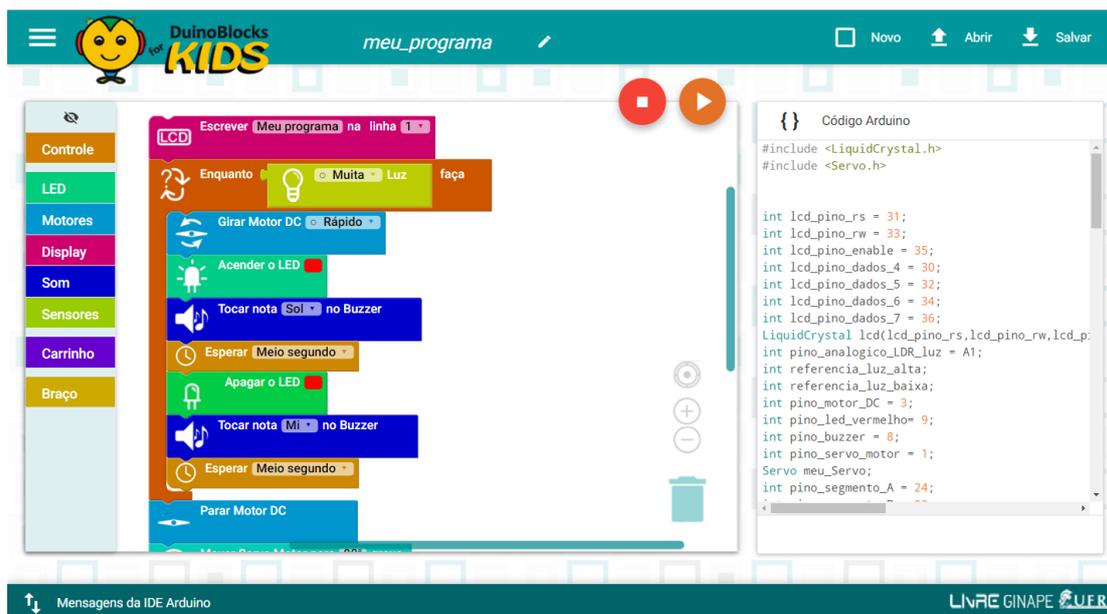


Figura 1. Visão Geral da Interface do DB4k.

O ambiente de programação visual DuinoBlock4Kids (Figura 1) possibilita o controle de um conjunto de LEDs, 1 LED RGB, 1 display de 7 segmentos, 1 display LCD, 1 servo motor, 1 motor DC, 1 buzzer, 1 sensor de distância, 1 sensor de luz e 1 sensor de temperatura, possuindo também alguns blocos específicos para o controle de um carro robô e um braço robótico. Além dos blocos diretamente relacionados com os dispositivos a serem manipulados, foram desenhados também blocos para as estruturas de controle utilizadas em programação, como repetição (contada e condicional) e decisão.

Algumas características do DB4K que o diferenciam dos ambientes de programação em blocos atualmente disponíveis para a plataforma Arduino - como o Ardublockly³, ambiente sobre o qual foi desenvolvido o DuinoBlocks4Kids - e que buscam tornar o seu uso mais acessível e aprazível a crianças do Ensino Fundamental I são: blocos de programação desenhados de modo a apresentarem uma semântica diretamente relacionada com os dispositivos sendo manipulados e com os efeitos por eles causados sobre estes dispositivos; supressão de detalhes relacionados ao *hardware*, como pinagens e valores de níveis de tensão; uso de linguagem icônica; conjunto "enxuto" de blocos; simplificação dos valores dos parâmetros utilizados nos blocos, como por exemplo: temperatura (alta ou baixa), luz (muita ou pouca), velocidade (rápida, média ou lenta).

³ <https://ardublockly.embeddedlog.com/>

4.2. As Atividades Didáticas

Por meio de trabalhos práticos, aulas expositivas dialogadas, brincadeiras e narrativas, as atividades propostas visam, em associação com o ambiente de programação em blocos e com os materiais de robótica, trabalhar as estruturas de programação contempladas pelo ambiente de programação visual DB4K.

As atividades práticas referem-se à montagem e programação de pequenos circuitos e também à programação de um conjunto de materiais de robótica (ver seção 4.3). 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 de tomada de decisão (utilizadas nas estruturas condicionais), conceitos esses que são posteriormente trabalhados no ambiente de programação em blocos em conjunto com os materiais de robótica. As narrativas, associadas aos robôs de garrafa pet (ver seção 4.3), são utilizadas para se contextualizar o uso dos dispositivos de robótica e estruturas de programação sendo trabalhadas.

4.3. Os Materiais de Robótica

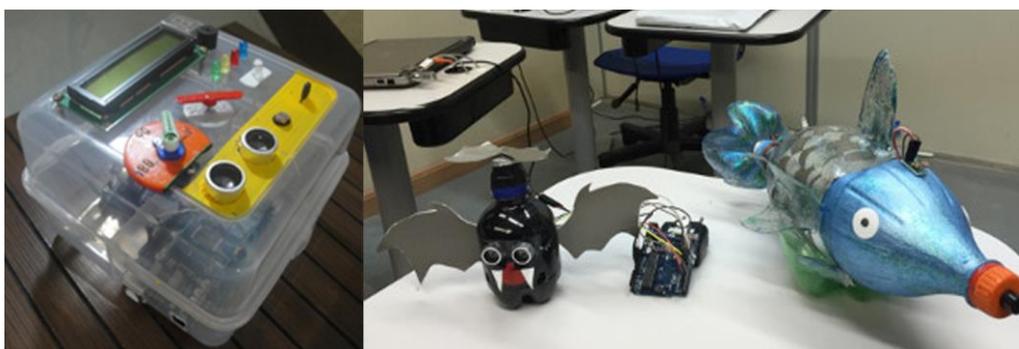


Figura 2. A "caixinha mágica" e os robôs de garrafa PET.

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. Um desses materiais é uma caixinha, batizada de "caixinha mágica", que permite às crianças programarem simultaneamente um conjunto de atuadores e sensores. Foram desenvolvidos também alguns materiais de garrafa PET utilizados nas atividades com narrativas, como o peixe-robô e o morcego-robô (Figura 2).

5. Resultados

Dos sete alunos participantes da oficina, três apresentaram um aproveitamento bastante satisfatório de todo o conteúdo trabalhado, e três apresentaram dificuldade em absorver esses conteúdos, em especial no que se refere às estruturas de repetição, e um deixou a oficina antes do término, tendo, no entanto, apresentado um bom aproveitamento dos conteúdos das aulas das quais participou.

Ao final do curso, as três crianças com menor aproveitamento conseguiam desenvolver apenas programas simples, com poucos comandos, sem o uso de laços de repetição e nos quais a noção de atraso (*delay*) não se fizesse necessária. Por outro lado, as três crianças com maior desenvoltura, apesar de terem apresentado uma dificuldade inicial em relação aos conteúdos abordados, 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 em aula.

Quadro 1. Síntese dos resultados relacionados ao aprendizado de programação

Estrutura Sequencial	Estrutura de Repetição Contada	Estrutura de Repetição Condicional
<p>Pensam. Computacional:</p> <ul style="list-style-type: none"> - Depuração e detecção sistemática de erros. - Capacidade de abstração. - Noções algorítmicas de fluxo de controle. <p>Dificuldades:</p> <ul style="list-style-type: none"> - Entendimento do efeito resultante do uso do tempo de atraso (<i>delay</i>). 	<p>Pensam. Computacional:</p> <ul style="list-style-type: none"> - Depuração e detecção sistemática de erros. - Capacidade de abstração. - Noções algorítmicas de fluxo de controle. <p>Dificuldades:</p> <ul style="list-style-type: none"> - A compreensão de que apenas os comandos posicionados dentro do bloco <i>repetir</i> seriam repetidos o número de vezes indicado como parâmetro. - Tempo de atraso (<i>delay</i>). 	<p>Pensam. Computacional:</p> <ul style="list-style-type: none"> - Depuração e detecção sistemática de erros. - Capacidade de abstração. - Noções algorítmicas de fluxo de controle. - Lógica condicional. - Decomposição estruturada de problemas. <p>Dificuldades:</p> <ul style="list-style-type: none"> - Compreender o funcionamento "não determinístico" dos programas que fazem uso de uma estrutura condicional.

A avaliação do desenvolvimento ou exercício das habilidades do Pensamento Computacional, bem como de outros aspectos relacionados à maturidade cognitiva das crianças, foi realizada a partir das competências que se entendia como sendo necessárias para a construção adequada dos programas propostos em aula. Para tal, lançou-se mão de uma escala Likert de 6 pontos, partindo de 0 = “Nem tentou” até 5 = “Aquisição ou realização completa do objetivo, tarefa ou conteúdo”. Essa escala era aplicada a itens referentes à aquisição ou exercício das habilidades pretendidas em cada aula, como por exemplo: “O aluno utiliza adequadamente o bloco enquanto <condição> faça (uso da lógica condicional) no programa desenvolvido, inserindo o bloco com o sensor correto como fator condicionante?”. O Quadro 1 apresenta uma síntese dos resultados observados por meio desse processo de avaliação no tocante às habilidades do Pensamento Computacional e às principais dificuldades encontradas pelas crianças durante a construção dos programas.

Além da demonstração das habilidades do Pensamento Computacional apresentadas no Quadro 1, os alunos também manifestaram outras competências durante a realização das atividades, tais como a construção mental das ações necessárias para a realização de uma determinada tarefa, a capacidade de reaproveitar código, a percepção da ideia de processamento, o entendimento de que os mesmos comandos organizados de diferentes maneiras podem levar a diferentes resultados e de que um mesmo problema pode apresentar diferentes soluções.

No que se refere à usabilidade do ambiente de programação visual em blocos, as crianças demonstraram bastante facilidade no seu manuseio. O seu uso, em conjunto com os materiais de robótica desenvolvidos, possibilitou às crianças assumirem uma maior autonomia em relação ao seu aprendizado a partir da depuração sistemática de erros realizada por meio do processo apresentado na Figura 3.

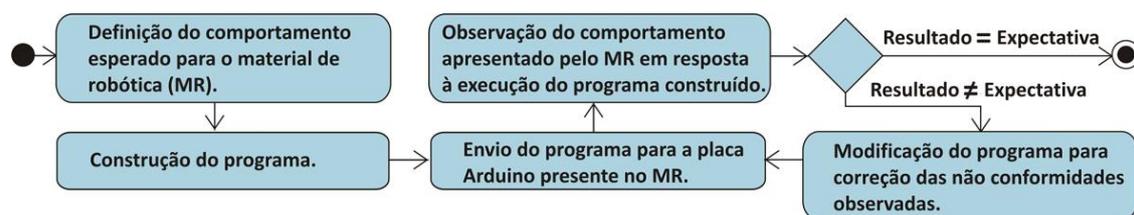


Figura 3. Diagrama de Atividades do processo de depuração de erros com o DB4K.

No que tange às estratégias pedagógicas adotadas, como a realização de brincadeiras baseadas na técnica de Computação Desplugada e o uso de narrativas, todas se apresentaram como sendo facilitadores lúdicos bastante apropriados para o aprendizado de programação com robótica, auxiliando na construção de um ambiente de aprendizagem divertido, afetuoso e acolhedor.

Os dados coletados possibilitaram uma avaliação bastante minuciosa de todos os aspectos investigados, tendo sido apresentada aqui apenas uma breve síntese dos resultados observados. O detalhamento dos dados, avaliações e inferências realizadas encontra-se disponível na dissertação de Mestrado de Queiroz (2017)⁴.

6. Conclusões

Os objetivos da pesquisa foram integralmente alcançados. O Ambiente de Programação Visual em Blocos DB4K foi desenvolvido e encontra-se plenamente funcional, foram também elaborados quatorze planos de aula com especificação detalhada dos objetivos, conteúdos, material necessário e desenvolvimento metodológico, e elaborados (com o uso de sucata e de materiais recicláveis ou de baixo custo) um conjunto de materiais de robótica para o uso com o ambiente de programação.

No que se refere ao objetivo geral da pesquisa, qual seja, a investigação do desenvolvimento e exercício de habilidades do Pensamento Computacional, os resultados apresentados indicam a viabilidade de se exercitar algumas destas habilidades, tais como a capacidade de abstração, o entendimento de fluxos de controle, o uso da lógica condicional, a decomposição de problemas e a depuração e detecção sistemática de erros, em crianças do Ensino Fundamental I, a partir do aprendizado de conceitos básicos de programação por meio de recursos baseados em Tecnologia Livre e materiais de baixo custo associados a estratégias pedagógicas alicerçadas na Robótica Educacional pensadas e desenvolvidas especificamente para esse público.

Além disso, a demonstração de certas habilidades durante a realização das atividades de programação, como a capacidade de reaproveitar código, o entendimento de que os mesmos comandos organizados de diferentes maneiras podem levar a diferentes resultados e de que um mesmo problema pode apresentar diferentes soluções, a percepção da ideia de processamento, entre outras, ajudam a reforçar as hipóteses aqui levantadas acerca da possível existência de uma relação direta entre determinadas características cognitivas de crianças no período operatório concreto e a realização de algumas atividades relacionadas à programação de computadores.

Em relação às dificuldades apresentadas por algumas crianças, como a utilização do tempo de atraso e dos laços de repetição, acredita-se que estas sejam decorrentes de

⁴ http://www.nce.ufrj.br/ginape/livre/paginas/dissertacoes/d_2017_rubens_queiroz.pdf

questões relativas à maturidade cognitiva, uma vez que os sujeitos observados se encontravam todos no estágio operatório concreto, no qual a capacidade de abstração ainda está em desenvolvimento. Além disso, uma vez que a maturidade cognitiva é dependente de fatores biológicos, educacionais e sociais [Furtado, Bock e Teixeira 2001], é possível que, em relação a certos aspectos, algumas das crianças observadas apresentassem uma maturidade pré-operatória, período no qual ainda não se consegue, de modo geral, realizar operações mentalmente

6.1. Principais Contribuições

A abstração é o processo mais importante do Pensamento Computacional [Wing 2011], e a programação de computadores uma importante ferramenta de apoio ao desenvolvimento e exercício de uma série de habilidades desta forma de pensamento, bem como uma competência fundamental da Ciência da Computação [Grover e Pea 2013].

Há hoje um crescente movimento no sentido de se introduzir o ensino de programação já nos primeiros anos do Ensino Fundamental e, até mesmo, a partir da Educação Infantil. Torna-se assim imprescindível a realização de pesquisas acerca da relação existente entre a capacidade de abstração de crianças nessa faixa etária e as habilidades necessárias ao aprendizado de conteúdos relacionados à programação de computadores, de modo que esses conteúdos sejam abordados no momento mais propício e do modo mais adequado. Uma importante contribuição deste trabalho é chamar a atenção para este fato e contribuir com a investigação de hipóteses acerca desse tema.

Outra contribuição significativa é o kit DB4K que, por ser baseado em Tecnologia Livre associada a materiais recicláveis ou de baixo custo, apresenta-se economicamente mais acessível do que as soluções proprietárias atualmente disponíveis no mercado, podendo contribuir assim para uma maior democratização do acesso ao aprendizado de programação de computadores nos primeiros anos do Ensino Fundamental.

6.2. Trabalhos Futuros

Como possíveis trabalhos futuros, derivados desta pesquisa, destacam-se: a realização de um estudo mais amplo acerca das teorias de desenvolvimento cognitivo, bem como das habilidades do Pensamento Computacional, de maneira a se produzir investigações mais detalhadas a respeito das hipóteses levantadas por essa pesquisa, o desenvolvimento de novas oficinas para a realização de investigações mais abrangentes acerca dos aspectos explorados por este trabalho e a realização de estudos do uso do kit DB4K envolvendo crianças com dificuldades cognitivas.

Referências

- Barr, V.; Stephenson, C. (2011) “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.
- Bell, T. et al. (2009) “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.
- Furtado, O.; Bock, A. M. B.; Teixeira, M. D. L. T. (2001) *Psicologias: uma introdução ao estudo de psicologia*. 13.ed. São Paulo: Saraiva. 365 p.

- Grover, S.; Pea, R. (2013) “Computational Thinking in K-12: A Review of the State of the Field”. *Educational Researcher*, Florida, v. 42, n. 1, p. 38–43.
- Hemendinger, D. (2010) “A plea for modesty.” *Acm Inroads*, New York, v. 12, p. 4-7.
- Kamii, C. (1992) *A criança e o número: implicações educacionais da teoria de Piaget para a atuação junto a escolares de 4 a 6 anos*. 36.ed. Campinas: PAPIRUS. 124 p.
- Kramer, J. (2007) “Is abstraction the key to computing?” *Communications of the ACM*, New York, v. 50, n. 4, p. 36–42.
- Meirinhos, M.; Osório, A. (2010) “O estudo de caso como estratégia de investigação em educação”. *EduSer - Revista de Educação*, Bragança, v. 2, n. 2, p. 49-65.
- Moreira, M. A. (1999) *Teorias de aprendizagem*. São Paulo: EPU. 195 p.
- Papert, S. (1993) *Mindstorms: Children, Computers and Powerful Ideas*. 2.ed. New York: BasicBooks. 252 p.
- Queiroz, L.; Sampaio, F. F.; dos Santos, M. P. (2016) “DuinoBlocks4Kids: Ensinando conceitos básicos de programação a crianças do Ensino Fundamental I por meio da Robótica Educacional. ” *Anais dos Workshops do V Congresso Brasileiro de Informática na Educação*. Uberlândia: [s.n.]. p. 1169-1178.
- Queiroz, R. L. (2017) *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*. Universidade Federal do Rio de Janeiro. Rio de Janeiro, 138 p.
- Queiroz, R. L.; Sampaio, F. F. (2016) “DuinoBlocks for Kids: Um ambiente de programação em blocos para o ensino de conceitos básicos de programação a crianças do Ensino Fundamental I por meio da Robótica Educacional”. *Anais do XXXVI CSBC*. Porto Alegre: PUC-RS. p. 2086-2095.
- Resnick, M. et al. (2009) “Scratch: programming for all.” *Communications of the ACM*, New York, v. 52, n. 11, p. 60-67.
- Souza, N. M. D.; Wechsler, A. M. (2014) “Reflexões sobre a teoria piagetiana: o estágio operatório concreto. ” *Cadernos de Educação: Ensino e Sociedade, Bebedouro*, v. 1, n. 1, p. 134-150.
- Terra, M. R. (2010) *O desenvolvimento humano na teoria de Piaget*. UNICAMP,. Disponível em: <<http://www.ceap.br/material/MAT31082010183131.doc>>.
- Whitehouse. (2016) Office of the Press Secretary - Fact Sheet. President Obama Announces Computer Science for All Initiative. Disponível em: <<https://www.whitehouse.gov/the-press-office/2016/01/30/fact-sheet-president-obama-announces-computer-science-all-initiative-0>>
- Wing, J. M. (2006) Computational thinking. *Communications of the ACM*, New York, v. 49, n. 3, p. 33-35.
- Wing, M. (2011) Computational Thinking-What and Why? *The Link Magazine*, Pittsburgh, p. 20-23, Spring 2011.
- Yin, R. K. (2001) *Estudo de Caso: Planejamento e Métodos*. 2.ed. Porto Alegre: Bookman. 205 p.