

ProgramChildren: Levando Tecnologia para Crianças de uma Escola Pública

Luis Filipe V. Braga¹, Daniel V. Braga¹, Amanda V. Gusmão², Caribe Z. Souza¹,
Neila M. Gualberto Leite¹

¹Instituto Federal de Educação, Ciência e Tecnologia do Norte de Minas Gerais (IFNMG)
Rua Dois, 300 – Village do Lago II – 39404-058 – Montes Claros – MG – Brazil

²Faculdades Integradas Pitágoras (FIP-MOC)
Montes Claros, MG – Brazil

luisfvbraga@gmail.com, dan02bragadb@gmail.com,
amandagusmao12@yahoo.com.br, caribe.souza@ifnmg.edu.br,
neila.gualberto@ifnmg.edu.br

Abstract. *Computer programming as a tool for development of computational thinking in elementary schools increases the student ability of understanding mathematical concepts and increase the chances of find new strategies to solve problems. Besides that, others abilities are also improved as creativity, autonomy and systematical thoughts. In this paper, we show an experiment report from an extension project developed for students in initial grades of the public elementary school of Montes Claros. The project was divided in two main steps: at first, it was developed group activities where programing logic issues were unplugged en plugged programming lessons. Then, the students receive programming and robotic classes and had the opportunity of implementing tools to help everyday tasks, using Arduino. The results show that our approach had low rates of quitting index with high rates of accuracy in the proposed activities.*

Resumo. *O uso da programação de computadores nas escolas como ferramenta para o desenvolvimento do pensamento computacional aumenta a habilidade do aluno para compreender os conceitos matemáticos e desenvolve novas estratégias de resolução de problemas, organização do pensamento, além de estimular a criatividade e desenvolver a autonomia da criança. Neste trabalho, apresentamos o relato de experiência de um projeto de extensão desenvolvido com alunos de 5º e 6º anos do ensino fundamental de uma escola pública de Montes Claros-MG. O projeto foi dividido em duas etapas: inicialmente foram trabalhadas atividades em grupo em que foram tratadas questões de lógica de programação plugadas e desplugadas. Na segunda etapa, os alunos tiveram contato direto com a programação e robótica e a oportunidade de implementar ferramentas que fazem parte do seu dia a dia, utilizando arduino. Com a metodologia aplicada obtivemos baixo índice de desistência e elevado índice de acertos nas atividades propostas por conteúdo trabalhado.*

1. Introdução

As novas tecnologias, emergentes nas últimas décadas, trazem inovação para diversas áreas do conhecimento e da sociedade. Devido à grande disponibilidade, o acesso das

crianças às novas tecnologias é facilitado, o que provoca uma inserção no mundo digital desde muito cedo, incitando mudança de paradigma no processo ensino-aprendizagem. Embora algumas instituições de ensino possuam dispositivos interativos como computadores, tablets e lousas digitais, a utilização desses recursos ainda é restrita, pois a interação entre o aluno e a máquina acontece de forma passiva, uma vez que o aluno utiliza os programas e jogos como uma ferramenta fechada, sem o entendimento da organização interna dos programas e computadores. Nesse contexto, a inserção do ensino de programação de computadores no sistema educacional possibilita uma nova visão dos alunos em relação ao desenvolvimento tecnológico, trazendo benefícios como a ampliação do raciocínio lógico e matemático [Matias et al. 2016].

O domínio de novas tecnologias, aliado à compreensão de como os computadores funcionam e de que modo podem ser utilizados para a solução de problemas, emergiu nos anos 2000 e é conhecido como pensamento computacional. Popularizado em 2006 por Jeannette Wing [Wing, 2006], é tido como uma forma de “desenvolver e empregar estratégias para entender e resolver problemas de forma a aproveitar o poder dos métodos tecnológicos para desenvolver e testar soluções” [IST 2017].

Para a Sociedade Brasileira de Computação [SBC 2017], o Pensamento Computacional é um dos três eixos do ensino de computação para a educação básica, juntamente com Mundo Digital e Cultura Digital. O pensamento computacional está relacionado à capacidade de sistematizar, representar, analisar e resolver problemas e hoje é considerado tão fundamental quanto a leitura, escrita e aritmética. Países como EUA, Alemanha, Argentina, Bélgica, Japão, entre outros, já adotam o ensino de Computação nas escolas de Educação Básica. Esta é uma chave para o futuro pois as profissões vindouras estão totalmente vinculadas à tecnologia. Infelizmente, no Brasil, esta ainda não é uma prática muito difundida, sendo aplicada apenas algumas escolas, sobretudo nos grandes centros urbanos.

A introdução de conteúdos da computação na educação básica é fundamental e estratégica pois estimula o pensamento computacional e o raciocínio lógico e constitui uma ferramenta importante para a construção do conhecimento, pois transforma o ato de jogar e divertir no ato de aprender e ensinar. O desenvolvimento de algoritmos e lógicas de programação transformam o conhecimento em técnicas e procedimentos, ou seja, descreve todos os passos necessários para atingir um certo objetivo e a resolução de um certo problema [Almeida 1999]. Além disso, incentiva a superação de desafios, estimula a criatividade e organiza o pensamento [Medeiros et al. 2013]. Ademais, garante maior empregabilidade, melhor compreensão do mundo através da transversalidade em outras áreas, além de auxiliar na alfabetização digital, melhoria de produtividade, organização do pensamento, entre outros [Brackmann 2017]

2. Trabalhos Correlatos

Diversos trabalhos que introduzem o pensamento computacional e a programação de computadores na educação básica têm sido desenvolvidos. Entre eles podemos citar os trabalhos de Fessakis e colaboradores [Fessakis et al. 2013], que apresentam um estudo no qual crianças tiveram oportunidade de desenvolver os conceitos matemáticos e aprimoraram a resolução de problemas através de novas estratégias utilizando uma linguagem de programação.

Barcelos e colaboradores [Barcelos et al. 2015] apresentam uma revisão sistemática da literatura de 48 artigos publicados entre 2006 e 2014 visando compreender o relacionamento entre o pensamento computacional e as disciplinas do currículo da educação básica para a identificação dos benefícios educacionais que estes proporcionam. Como resultados, identificaram um aumento significativo do interesse da academia em explorar as relações entre o pensamento computacional e a Matemática, embora ainda haja necessidade de maior desenvolvimento da modelagem matemática.

Fengfeng Ke [KE 2014] apresenta um estudo que examinou o potencial de atividades relacionadas a criação de jogos computacionais na aprendizagem de matemática para adolescentes. Suas pesquisas tiveram como resultado a constatação de que o desenvolvimento de jogos contribuiu significativamente para o desenvolvimento da matemática, além de propiciar um modo de pensar diferenciado sobre o cotidiano, ao elaborar o desenvolvimento dos jogos, incentivando a criatividade e o desenvolvimento do raciocínio abstrato.

Aono e colaboradores [Aono et al. 2017] apresentam uma proposta de ensinar o pensamento computacional a estudantes do ensino fundamental, valendo-se de aulas expositivas e da utilização do Scratch como um ambiente de aprendizado intuitivo e acessível. Apresentam uma explicação teórica aliada a atividades práticas de programação, a partir das quais os alunos construíram partes de um jogo, que, depois, formaram o projeto final. A partir da iniciativa apresentada, foi possível avaliar métodos para o ensino de programação além da utilização dos jogos como valioso incentivo pedagógico.

Neste trabalho, relatamos a experiência do desenvolvimento de um projeto de extensão universitária desenvolvido com crianças de uma escola pública de Montes Claros (MG), que tem como foco o desenvolvimento do pensamento computacional e, assim, do raciocínio lógico-matemático, através do ensino de programação e robótica para um grupo de alunos de 5º e 6º anos do ensino fundamental utilizando algumas plataformas digitais de livre acesso. Apresentamos ferramentas tecnológicas através de atividades e sequências didáticas, capazes de estimular o desenvolvimento do raciocínio lógico-matemático e de imergir os alunos num mundo tecnológico. Nosso principal objetivo foi proporcionar às crianças um novo tipo de contato com a tecnologia e, desta vez como agentes e não apenas como usuários, o que traz uma nova forma de organizar pensamento e colabora para o desenvolvimento do raciocínio lógico.

3. Metodologia

Para desenvolver esse projeto, lançamos mão de ferramentas e técnicas que vêm sendo analisadas por pesquisadores tais como atividades desplugadas e plugadas, acesso a plataformas e utilização de softwares livres. Buscando inovar e avançar nesta direção, associamos essas atividades à construção de robôs e, especificamente, acompanhamos o desenvolvimento de cada uma das crianças através de uma sala virtual, favorecendo o processo de avaliação. A seguir, apresentamos com mais detalhes a metodologia utilizada.

A equipe de trabalho para a execução do projeto foi composta por cinco discentes de ensino superior, sendo quatro graduandos da Ciência da Computação e um de Engenharia Elétrica orientados por um professor da área de Matemática e outro da área de Ciência da Computação.

O projeto foi desenvolvido numa escola pública com 20 (vinte) crianças de 5º e 6º ano do ensino fundamental durante os meses de agosto a dezembro de 2017. A seleção dos alunos foi feita considerando-se interesse, disponibilidade e custeio do material didático individual que seria utilizado. Como a demanda foi maior que o número de vagas, fizemos o sorteio de 20 alunos dentre os interessados. Para trabalhar os conteúdos planejados, foram realizados encontros semanais com duração de uma hora e trinta minutos. O desenvolvimento do projeto foi dividido em duas fases. Na primeira, além das atividades com o uso do computador (atividades plugadas), foram realizadas, principalmente, atividades sem a utilização de computadores ou de outras tecnologias (atividades desplugadas). Nas atividades desplugadas os alunos formavam equipes de 5 alunos orientados por um tutor. A aula consistia em atividades como jogos de tabuleiro, cartas, maratonas e brincadeiras de roda. Essas brincadeiras, embora já bastante utilizadas para outras finalidades, foram elaboradas de forma a ensinar os principais componentes de um algoritmo, como estruturas de controle, raciocínio lógico e álgebra booleana. Concomitante ao pensamento computacional, podemos listar também como aprendizado, o trabalho em equipe, divisão de funções e troca de experiências entre alunos e professores durante as atividades desplugadas. O material utilizado foi construído baseado em atividades ofertadas pela plataforma code.org, porém, boa parte delas passaram por readaptações para adequar-se ao contexto da turma (tempo de aula e número de alunos).

Após as atividades desplugadas, os alunos recebiam um bloco de atividades pré-definidas para serem resolvidas no decorrer da semana, através de uma sala virtual disponibilizada pela plataforma code.org. Nessa sala, os alunos aplicavam, individualmente, os conhecimentos adquiridos nos encontros presenciais através do desenvolvimento de algoritmos em linguagem de programação visual (blocos). A escolha deste tipo de linguagem justifica-se pela forma como é aplicada. Ao utilizá-la no desenvolvimento dos programas, anulam-se as chances de erro de sintaxe, de forma que o aluno concentre-se exclusivamente no raciocínio lógico que deverá ser aplicado no seu algoritmo. Além disso, a programação em blocos proporciona resultados rápidos, de forma que o discente percebe sua evolução em tempo real. Todas as ações executadas pelos alunos foram registradas na plataforma, e serão apresentadas na seção de resultados

Na segunda fase, passamos a aplicar o conhecimento adquirido utilizando a robótica. A partir dessa fase, os encontros presenciais passaram a acontecer no laboratório de informática, com atividades plugadas. Introduzimos a eletrônica e a integramos à programação utilizando o Arduino.

A robótica educacional ou pedagógica, como ficou conhecida a utilização de criação e manipulação de robôs como ferramenta de ensino, tem sido grande aliada para ensinar crianças e a desenvolver pensamento computacional [Torcato 2012].

Assistir robôs é bem interessante, mas para que programá-los também seja um processo atrativo, é necessário avaliar uma série de fatores. Por causa dessa percepção, buscamos algumas plataformas de desenvolvimento e estabelecemos alguns critérios: baixo custo da controladora utilizada pelo software, facilidade na manipulação dos códigos, depuração rápida e eficiente, a instalação do software sem ônus para Windows e Linux. Considerando essas restrições, o software que melhor atendeu às condições foi o Scratch4Arduino(S4A). Além de atender as características descritas, o S4A apresenta semelhanças com as atividades plugadas da primeira fase, atendendo nossos requisitos

pedagógicos. A linguagem de blocos utilizada pelo S4A, possibilitou um aproveitamento integral dos conhecimentos adquiridos nas atividades plugadas da primeira fase, devido a similaridade com linguagem das atividades anteriores.

Nesta fase, os acompanhamentos semanais passaram a ser feitos através de uma sala virtual no Google Classroom. Na sala, eram enviados, semanalmente, esquemas de circuito elétrico e suas respectivas instruções de desenvolvimento. As atividades executadas pelos alunos nesta fase consistiam em montar circuitos em casa, programá-los utilizando a sua controladora Arduino e publicar o resultado final para avaliação. Na sala também foram abertas discussões para cada atividade proposta com o objetivo de tirar dúvidas e propor soluções.

4. O curso

O curso foi desenvolvido no período de agosto a dezembro de 2017 em 15 encontros presenciais. As aulas desenvolvidas, assim como os conteúdos e as competências e habilidades envolvidas são apresentadas a seguir.

Movimento de um personagem

Conteúdo: algoritmos.

Objetivos: desenvolver o sentido de orientação e deslocamento no espaço; investigar e desenvolver estratégias; incentivar o trabalho em grupo.

Competências e habilidades desenvolvidas: dividir em etapas a solução de problemas relacionados ao movimento do corpo e trajetórias espaciais; utilizar linguagens visuais e língua nativa para representar dados e processos.

Loop

Conteúdo: estruturas de repetição; algoritmos; linguagens de programação visual.

Objetivos: reconhecer as estruturas de programação ciclos (loops), sequências e condições; registrar códigos a partir da análise de figuras.

Competências e habilidades: representar em experiências concretas as principais abstrações para descrever dados; identificar as principais abstrações para construir processos: escolha, composição e repetição, simulando e definindo algoritmos simples que representem situações do cotidiano infantil; utilizar linguagem lúdica visual para representar algoritmos.

Tartaruga

Conteúdo: linguagem de programação interpretada; plano cartesiano; ângulos; sintaxe de algoritmos.

Objetivos: reconhecer ângulos e plano de coordenadas cartesianas; desenvolver noções espaciais e de orientação.

Competências e habilidades: utilizar linguagem lúdica visual para representar algoritmos; criar passos para solução de problemas relacionados ao movimento do corpo e trajetórias espaciais.

Estruturas condicionais

Conteúdo: estruturas condicionais; operações aritméticas; operadores lógicos; operadores de comparação; indentação de códigos.

Objetivos: compreender o funcionamento das estruturas condicionais e os conceitos de comandos encadeados; avaliar afirmações lógicas para tomada de decisão.

Competências e Habilidades: compreender a técnica de decompor um problema para solucioná-lo; identificar as principais abstrações para construir processos: escolha, composição e repetição, simulando e definindo algoritmos simples que representem situações do cotidiano infantil.

Chaveiro

Conteúdo: função; parâmetros; recursividade.

Objetivos: identificar padrões; compreender o conceito de funções; simplificar processos.

Competências e Habilidades: empregar o conceito de recursão, para a compreensão mais profunda da técnica de solução através de decomposição de problemas; construir soluções de problemas usando a técnica de generalização, permitindo o reuso de soluções de problemas em outros contextos, aperfeiçoando e articulando saberes escolares.

ForRacing

Conteúdo: estruturas de repetição; contadores.

Objetivos: reconhecer a utilização do comando “for” e suas possibilidades; compreender condições de início, parada e salto.

Competências e Habilidades: identificar as principais abstrações para construir processos: escolha, composição e repetição, simulando e definindo algoritmos simples que representem situações do cotidiano infantil.

Maratona

Conteúdo: compiladores; linguagem de programação compilada; linguagem de programação visual.

Objetivos: reconhecer falhas no processo de construção de algoritmos; interpretar e reescrever algoritmo.

Competências e Habilidades: utilizar linguagem lúdica visual para representar algoritmos; relacionar um algoritmo descrito em uma linguagem visual com sua representação em uma linguagem de programação; estabelecer relação entre hardware e software (camadas/sistema operacional) em um nível elementar.

Quiz

Conteúdo: algoritmos; variáveis; estruturas condicionais; estruturas de repetição; compiladores.

Objetivos: trabalhar em equipe; avaliar o desempenho do grupo nas atividades plugadas; motivar a tomada de decisão.

Competências e Habilidades: identificar as principais abstrações para construir processos; escolha, composição e repetição, simulando e definindo algoritmos simples que representem situações do cotidiano infantil.

Conhecendo o Arduino

Conteúdo: corrente elétrica; resistores; sensores; arduino.

Objetivos: conhecer o arduino e seu funcionamento, conhecer o funcionamento de

módulos e sensores.

Competências e Habilidades: identificar os tipos de módulos e sensores e a placa Arduino e suas propriedades; diferenciar os objetos eletrônicos dos objetos não eletrônicos descrevendo seus usos e finalidades; criar passos para a solução de problemas.

Manipulação de LED's

Conteúdo: corrente elétrica; resistores; diodo emissor de luz; sinal digital; algoritmos.

Objetivos: entender o funcionamento de um LED, montar um circuito com LED e programar de forma dinâmica.

Competências e Habilidades: Reconhecer os elementos básicos e propriedades de circuitos elétricos; resolver problemas envolvendo a construção de circuitos com LED e seu controle a partir de programação com Arduino; identificar as principais abstrações para construir processos, simulando e definindo algoritmos simples que representem situações do cotidiano. Entender o conceito de informação, como armazená-la e codificá-la.

Semáforos

Conteúdo: diodo emissor de luz; sinal digital; interruptores; algoritmos.

Objetivos: aprender a manipular a chave tátil, simular um semáforo de carros e pedestres, estruturar um algoritmo de controle de semáforo.

Competências e Habilidades: reconhecer os elementos básicos de circuitos elétricos e a interconexão da chave tátil; identificar as regras de trânsito de carros e pedestres relacionadas aos semáforos; relações interpessoais entre colegas e tutores; identificar a presença da informática na vida das pessoas, bem como sua influência na sociedade atual.

Sensor de Iluminação

Conteúdo: fotoresistor; resistores; entrada de dados; algoritmos.

Objetivos: entender o funcionamento do fotoresistor e implementar um simulador de poste de luz, criando um algoritmo que apaga e acende um LED de acordo com a variação de luz ambiente.

Competências e Habilidades: identificar as propriedades e funcionamento dos fotoresistores; interconectar os resistores e fotoresistores; reconhecer as interconexões de postes de iluminação para o controle de ligar e desligar; identificar as principais abstrações para construir processos: escolha, composição e repetição, simulando e definindo algoritmos simples que representem situações do cotidiano

Controle de intensidade de som

Conteúdo: saída de dados; sinal analógico; buzzer; algoritmos.

Objetivos: aprender a manipular o sinal analógico em função do som de um buzzer, criando pequenas variações de som.

Competências e Habilidades: Identificar os tipos de sinais (digitais e analógico), reconhecendo o formato dos dados transmitidos; resolver problemas envolvendo a construção de circuitos com transmissão de dados analógicos e seu controle a partir de programação com Arduino; compreender a técnica de decompor um problema para solucioná-lo.

Sensor de Proximidade

Conteúdo: sensor ultrassônico; entrada de dados; buzzer; sinal digital; algoritmos.

Objetivos: aprender a manipular o sensor ultrassônico e criar uma simulação de sensor de ré, utilizando o buzzer na emissão dos sons.

Competências e Habilidades: identificar as propriedades do sensor ultrassônico; reconhecimento de conceitos de física; identificar as propriedades do sensor ultrassônico; resolver problemas envolvendo o controle de distância entre objetos e a programação da ação para restrições de distância; trabalho em equipe.

Potenciômetro

Conteúdo: variação; saída de dados; entrada de dados; sinal analógico; algoritmos.

Objetivos: aprender a utilizar a variação de resistência do potenciômetro como fonte de entrada de dados e conseqüentemente, utilizar esta entrada para variar a luminosidade do LED alto brilho.

Competências e Habilidades: identificar as propriedades do potenciômetro; reconhecimento de unidades de medida e variação; reconhecer a relação entre a variação e o fator de luminosidade do LED.

5. Resultados

A turma na qual o projeto foi desenvolvido foi composta por 20 (vinte) crianças, sendo 12 (doze) meninos e 8 (oito) meninas. Observou-se uma participação e o interesse de uma boa parte de meninas (40%), numa área em que a presença feminina é reduzida. Atividades como essa estimula e desperta o interesse por tecnologia, sendo muitas vezes importante na decisão pelas áreas de trabalho.

Com relação ao número total de interessados pelo curso, obtivemos um total de 40% de adesão entre todos os alunos de 5º e 6º ano, que atendiam os pré-requisitos: interesse, disponibilidade e investimento no material. O curso teve alto índice de envolvimento e interesse. Entre os selecionados, obtivemos uma única desistência que ocorreu no segundo mês de projeto sem justificativa, caracterizando 95% de permanência no curso.

Outro resultado interessante foi o desempenho nas atividades plugadas (Figura 1). Em todas elas, a média de desempenho da turma sempre esteve acima de 88%, com uma única exceção no conteúdo de estrutura de repetição FOR, cuja média de questões resolvidas foi de 71,79%. Para esta exceção, pontuamos duas dificuldades que podem ter acarretado esse desempenho inferior. A primeira foi a dificuldade em encontrar uma correspondência de atividade desplugada para trabalhar essa estrutura de repetição. A outra foi que a atividade escolhida para fixação do conteúdo apresentou uma quantidade de regras que desviou o objetivo do tema.

Fizemos também uma análise de rendimento por aluno nas atividades desplugadas e na robótica. Neste quesito, obtivemos uma eficiência de cem por cento. É importante destacar, que todos os alunos conseguiram resolver as atividades propostas nos encontros semanais e o fator determinante para este resultado foi a quantidade de tutores envolvidos nas atividades. Como as equipes eram compostas por cinco crianças e um tutor, o acompanhamento se mostrou muito eficiente, de forma que nenhum aluno deixasse de conseguir concluir as atividades propostas até o fim da aula. Ademais, os resultados individuais, além de apresentar as dificuldades de cada aluno, foram utilizados para definir

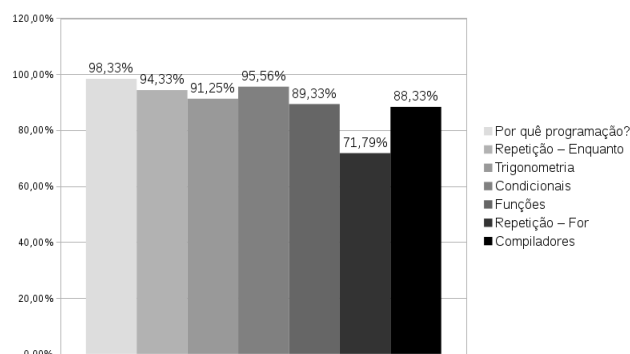


Figura 1. Média de exercícios resolvidos por tema

os discentes que iriam compor cada equipe de trabalho no encontro presencial da semana seguinte, de forma que podíamos criar grupos de trabalho com uma heterogeneidade de conhecimento e proporcionar paridade de força de trabalho entre as equipes e interação entre os alunos.

As atividades de robótica enviadas pela sala virtual também tiveram total aproveitamento. Após os encontros semanais, os alunos desenvolviam suas tarefas em casa e geralmente, além de apresentar o resultado na sala virtual, levavam para apresentar o seu trabalho para os colegas na aula da semana seguinte, o que promoveu interação e discussão entre os alunos e os tutores.

Com relação ao software, o S4A se mostrou bastante eficiente na maioria das atividades propostas, mas tivemos problemas principalmente com a quantidade limitada de recursos. O software não é dinâmico de forma a oferecer um suporte para novos dispositivos e apresenta limitações em algumas estruturas de algoritmo. Para utilizarmos alguns sensores, por exemplo, precisamos fazer algumas adaptações que comprometeram uma parte da estrutura do algoritmo desenvolvido pelos alunos. Algumas estruturas simples, como funções e repetição do tipo FOR não podem ser implementadas no S4A. Alguns sensores como o de presença, o de umidade e temperatura também não possuem suporte.

6. Considerações Finais

Neste trabalho, descrevemos o desenvolvimento de um projeto que trabalhou programação e robótica para crianças de 10 a 12 anos. Seguindo as novas tendências para o ensino, nosso projeto buscou associar habilidades e competências necessárias para essa geração através de atividades lúdicas que se concretizaram através da construção de pequenos robôs e do desenvolvimento de códigos. Observamos o comprometimento e o interesse das crianças durante as aulas e também na realização das atividades extraclasse, na plataforma online.

Como resultados, obtivemos excelentes envolvimento e rendimento das crianças, despertando a curiosidade e incentivando o trabalho em grupo. Observamos que alguns pontos da metodologia podem ser aperfeiçoados, inclusive a utilização do software escolhido. Ainda assim, podemos concluir que o objetivo foi atingido, uma vez que conseguimos proporcionar às crianças um novo contato com a tecnologia como agentes e não apenas como usuários, apresentando uma nova forma de organizar pensamento e colaborando para o desenvolvimento do raciocínio lógico.

Os bons resultados observados são resposta à reunião de uma diversidade de atividades que, juntamente com o acompanhamento individual, proporcionou a organização dos conceitos de forma incremental, favorecendo, assim, a estruturação e desenvolvimento do pensamento computacional.

Como trabalhos futuros pretendemos atender outras faixas etárias e apresentar atividades diferenciadas como desenvolvimento de jogos, para crianças de 7 a 9 anos e internet das coisas, para pré-adolescentes. Além de desenvolver o projeto em uma escola de ensino público, pretendemos atender crianças em situação de vulnerabilidade social no entorno da nossa instituição e observar o desempenho dos mesmos.

Referências

- (2017). ISTE standards for students. <http://www.iste.org/standards/standards/for-students#startstandards>. Acessado em: 18/04/2017.
- (2017). Sociedade Brasileira de Computação: Referenciais de Formação em Computação: Educação Básica. <http://www.sbc.org.br/noticias/10-slideshow-noticias/1996-referenciais-de-formacao-em-computacao-educacao-basica>. Acessado em: 02/05/2017.
- Almeida, A. R. S. (1999). A emoção na sala de aula. *Papirus*.
- Aono, A. H., Rody, H. V. S., Musa, D. L., Pereira, V. A., and Almeida, J. (2017). Modelos de ensino híbrido: Um mapeamento sistemático da literatura. *VI Congresso Brasileiro de Informática na Educação (CBIE 2017)*.
- Barcelos, T., Muños, R., Villarroel, R., and Silveira, I. (2015). Relações entre o pensamento computacional e a matemática: uma revisão sistemática da literatura. *Anais dos Workshops do IV Congresso Brasileiro de Informática na Educação (CBIE 2015)*, pages 1369–1378.
- Brackmann, C. P. (2017). Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica. *Tese(Doutorado) - Universidade do Rio Grande do Sul. Porto Alegre*.
- Fessakis, G., Gouli, E., and Mavroudi, E. (2013). *Computers Education*. Editorial Board, ed. 63, p. 87-89.
- KE, F. (2014). *Computers Education*. Editorial Board, ed. 73, p. 26-39.
- Matias, V., Nogueira, E. C., Oliveira, C. E. T., and Monteiro, E. F. (2016). Superpython: experimento de ensino de programação para crianças e adolescentes. *Anais dos workshops do V congresso Brasileiro de Informática na Educação (CBIE 2016)*, p. 647-654.
- Medeiros, T. J., da Silva, T. R., and da Silva Aranha, E. H. (2013). Ensino de programação utilizando jogos digitais: uma revisão sistemática da literatura. *Revista Novas Tecnologias na Educação, UFGRS, v.11 n.3*.
- Torcato, P. (2012). O robô ajuda? estudo de impacto do uso de robótica educativa como estratégia de aprendizagem na disciplina de aplicações informáticas b. *II Congresso Internacional TIC e Educação*, p. 2578-2583.