

Ensino de algoritmos, programação e matemática: panorama e estudo de caso com estudantes de escolas públicas brasileiras

Leônidas de Oliveira Brandão¹, Paulo Viniccus Vieira¹,
Priscila da Silva Neves Lima²

¹Instituto de Matemática e Estatística – Universidade de São Paulo
São Paulo – São Paulo – Brasil

²Instituto de Informática – Universidade Federal de Goiás
Goiânia – Goiás – Brasil

{leo, paulov}@ime.usp.br, priscila_lima@ufg.br

Abstract. *The literature indicates that algorithms and computer programming is an increasing tendency in elementary education. However, as indicated here, the official curriculum in Brazilian schools do not reflect this trend yet. Besides, this paper also presents the results from the first year of a project to teach programming exploring curricular math concepts with high school students, in a Brazilian public school. This course was organized in two semesters, during the first was used the visual programming paradigm, to have a smooth introduction. In the second semester, a commercial language (Python) was used. Regardless some difficulties faced, the students made good progress.*

Resumo. *A literatura indica uma crescente preocupação com a apresentação dos conceitos de programação no ensino fundamental. Entretanto, a partir de dois levantamentos realizados, verificamos que esse movimento ainda não chegou aos currículos oficiais das escolas públicas brasileiras. Além disso apresentamos os resultados de um projeto no qual relacionamos programação e matemática, com turmas de primeiro ano do ensino médio de uma escola pública de periferia da cidade São Paulo. Nesse projeto foram utilizadas ferramentas integradas ao ambiente web, em dois semestres. No primeiro os conceitos foram apresentados utilizando o paradigma da programação visual e no segundo semestre utilizamos a linguagem Python. Relatamos alguns dos resultados obtidos, em particular o bom aproveitamento de aprendizado e de motivação por parte do alunos, além de algumas dificuldades enfrentadas.*

1. Introdução

A computação está presente em vários setores da sociedade. Essa área de conhecimento permeia todas as atividades humanas, de tal modo que fica difícil hoje imaginar a sociedade sem computadores e suas tecnologias associadas [França e Amaral 2013]. Tal relevância tem provocado mudanças nas escolas, existindo várias iniciativas para incorporar os conceitos de programação de computadores (que por simplicidade nos referiremos apenas como programação) [CSTA 2005]. Por exemplo, [Kenski 2008] defende que o ensino de tecnologia é essencial para a educação, considerando indissociável no contexto atual da sociedade.

Apesar dos grandes benefícios que a tecnologia trouxe para a educação, ainda há muito a se fazer. O sistema educacional ainda está bastante defasado, principalmente no Brasil. Grandes investimentos têm sido feitos mundialmente a fim de integrar a Internet aos processos educacionais. Contudo, o foco das mudanças quase sempre recai sobre a infraestrutura ou, mais geralmente, sem uma visão integrada de todos os elementos do processo de aprendizado, não criando o embasamento ideal para que a inovação educacional ocorra. Mesmo com todos os recursos provenientes do mundo tecnológico, a educação ainda é limitada, seja em aspectos físicos, como em salas de aula, em aspectos culturais ou em aspectos econômicos [Santana et al. 2012].

Para [Wing 2006], estudantes devem desenvolver, desde o ensino básico, um subconjunto de competências e habilidades básicas relacionadas à computação. Esse subconjunto pode ser denominado como pensamento computacional, envolvendo as competências relacionadas à abstração e decomposição de problemas de forma a permitir sua resolução usando recursos computacionais e estratégias algorítmicas. No Brasil, a Base Nacional Comum Curricular (BNCC) estabelece conhecimentos, competências e habilidades que se espera que todos os estudantes desenvolvam ao longo da escolaridade básica. E é orientada pelos princípios éticos, políticos e estéticos traçados pelas Diretrizes Curriculares Nacionais da Educação Básica.

A BNCC é um documento de caráter normativo que define o conjunto orgânico e progressivo de aprendizagens essenciais que todos os alunos devem desenvolver ao longo das etapas e modalidades da educação básica [Brasil 2018]. Conforme definido na Lei de Diretrizes e Bases da Educação Nacional (LDB, Lei nº 9.394/1996), a BNCC deve nortear os currículos dos sistemas e redes de ensino das Unidades Federativas, como também as propostas pedagógicas de todas as escolas públicas e privadas de educação infantil, ensino fundamental e ensino médio, em todo o Brasil.

Nos Parâmetros Curriculares Nacionais (PCN) encontramos a indicação do ensino de informática, porém não com o viés de programação, mas focado na alfabetização digital. O PCN aponta para a necessidade de saber usar recursos básicos de um computador. Na terceira versão da BNCC, já vemos o pensamento computacional sendo explicitado e relacionado com o ensino de matemática:

(...) a aprendizagem de Álgebra pode contribuir para o desenvolvimento do pensamento computacional dos alunos, tendo em vista que eles precisam ser capazes de traduzir uma situação dada em outras linguagens, como transformar situações-problema, apresentadas em língua materna, em fórmulas, tabelas e gráficos e vice-versa [Brasil 2017].

A BNCC explicita também a necessidade e a importância dos algoritmos e a linguagem algorítmica

(...) Associado ao pensamento computacional, cumpre salientar a importância dos algoritmos e de seus fluxogramas, que podem ser objetos de estudo nas aulas de Matemática. Um algoritmo é uma sequência finita de procedimentos que permite resolver um determinado problema. Assim, o algoritmo é a decomposição de um procedimento complexo em suas partes mais simples, relacionando-as e ordenando-as, e pode ser representado graficamente por um fluxograma. A linguagem algorítmica tem pontos

em comum com a linguagem algébrica, sobretudo em relação ao conceito de variável. Outra habilidade relativa à álgebra que mantém estreita relação com o pensamento computacional é a identificação de padrões para se estabelecer generalizações, propriedades e algoritmos [Brasil 2017].

Contudo, em outubro de 2018 a Sociedade Brasileira de Computação (SBC) emitiu nota técnica manifestando discordância e críticas quanto as versões BNCC-EF e BNCC-EM homologadas [SBC 2018]. As notas técnicas trazem uma análise crítica das habilidades relacionadas à Computação, demonstrando sua inadequação e solicitando providências que, se não realizadas, pode prejudicar o ensino na Educação básica no Brasil. A SBC é a principal sociedade da área de computação no Brasil, sua atuação é extremamente ativa, tem secretarias regionais em todos os estados brasileiros, possui competência técnica necessária para discordar, criticar e propor diretrizes da área de computação.

A partir desse contexto e visando contribuir para o debate sobre a elaboração de Diretrizes Curriculares no âmbito da computação, apresentamos um estudo de caso relacionando o aprendizado de algoritmos, programação e matemática em escolas públicas brasileiras. O impacto desta pesquisa abrange diversos atores do contexto educacional. Para os gestores, na tomada de decisões estratégicas no que tange à formação dos estudantes e também na implantação de políticas públicas voltadas para a educação básica.

Esse estudo está estruturado em 4 seções. Primeiramente, relatamos na seção 2 um panorama sobre o ensino de algoritmo e programação em escolas públicas brasileiras. Em seguida, na seção 3 relatamos os resultados do primeiro ano do projeto em que apresentamos os conceitos de programação para grupos de alunos de uma escola pública da cidade CID. Por fim, na seção 4 apresentamos as considerações finais do trabalho.

2. Um panorama do ensino de algoritmos e programação nas escolas brasileiras

Apesar da BNCC apresentar trechos demonstrando a necessidade e importância da linguagem algorítmica, apenas 10% das habilidades propostas mencionam explicitamente tecnologia [CiEB 2019a]. Enquanto isso, países como Grã-Bretanha, Finlândia, Noruega, Suécia, Polônia e Estados Unidos possuem disciplinas específicas em programação e ciência da computação desde a alfabetização até o fim do ensino médio [Heintz et al. 2016].

Com o objetivo de verificar como o ensino de programação está inserido nas escolas brasileiras, buscamos referências ao ensino de programação nas orientações curriculares de municípios brasileiros. Vale destacar que foram considerados apenas municípios com mais de um milhão de habitantes, usando como base os dados do Instituto Brasileiro de Geografia e Estatística (IBGE) [IBGE 2019].

De acordo com esse critério foram considerados 17 municípios: São Paulo (SP); Rio de Janeiro (RJ); Salvador (BA); Fortaleza (CE); Belo Horizonte (MG); Manaus (AM); Curitiba (PR); Recife (PE); Brasília (DF); Porto Alegre (RS); Belém (PA); Goiânia (GO); Guarulhos (SP); São Luís (MA); São Gonçalo (RJ); Campinas (SP); e Maceió (AL).

A questão de pesquisa que norteou esse levantamento foi:

- Qual o estado atual da inserção de Algoritmos, Programação ou Pensamento Computacional nos currículos da educação básica no Brasil?

Para delimitar melhor o contexto da pesquisa, usamos como questão secundária:

- Quais os trabalhos que relacionam a programação com a matemática?

O processo utilizado para localizar as orientações curriculares foi o de busca manual nos sítios *web* das prefeituras dos municípios selecionados. Esta busca foi executada em duas etapas:

- A primeira etapa consistiu na verificação dos sítios *web* em busca de orientações curriculares. Nessa etapa, foram localizadas as orientações curriculares de 12 municípios: São Paulo (SP), Rio de Janeiro (RJ), Salvador (BA), Fortaleza (CE), Belo Horizonte (MG), Manaus (AM), Curitiba (PR), Recife (PE), Porto Alegre (RS), Belém (PA), Campinas (SP) e Maceió (AL).
- A segunda etapa foi realizada a partir da análise dos documentos selecionados na etapa anterior, onde se buscou referências ao ensino de programação ou pensamento computacional no ensino básico. A partir dessa análise, notou-se que apenas dois municípios apresentavam itens relacionados ao ensino de programação: São Paulo (SP) e Belo Horizonte (MG) (Figura 1).



Figura 1. Relação de cidades pesquisadas e presença de algoritmos no currículo.

Nas recomendações curriculares da cidade de São Paulo, é apresentada uma discussão significativa sobre o ensino de programação e pensamento computacional. Propõe-se a promoção do pensamento computacional, em uma abordagem construcionista, a partir de três eixos: Programação, Tecnologia de Informação e Comunicação e Letramento Digital. São apresentadas orientações e sugestões sobre a inserção de cada um desses eixos, desde os anos iniciais do ensino básico. Além disso, propõe-se o ensino de programação como processo de construção do conhecimento, promovendo a articulação com as diferentes áreas do conhecimento [São Paulo 2019b]. No que se refere à articulação entre programação e matemática, propõe-se o desenvolvimento de softwares/aplicativos para resolver equações, construir gráficos, entre outras tarefas [São Paulo 2019a].

Nos documentos das escolas municipais de Belo Horizonte, destaca-se a ideia de algoritmo relacionado ao ensino de álgebra. Entretanto, nada de programação é explicitamente mencionado.

Do ponto de vista normativo, percebe-se que o Brasil ainda deixa a desejar quanto ao ensino de programação e algoritmos. Isso não significa que não haja esforços para a inclusão dessa habilidade nos currículos. Como exemplo, podemos citar o Currículo de Referência em Tecnologia e Computação, disponibilizado pela CiEB (Centro de Inovação

para a Educação Brasileira), que oferece orientações para apoiar escolas a incluir os temas tecnologia e computação em suas propostas curriculares. [CiEB 2019b].

No que tange os projetos pedagógicos da educação pública, vale ressaltar os Institutos Federais de Educação, Ciência e Tecnologia (IF). Essas instituições são especializadas na oferta de educação profissional e tecnológica, com currículos integrados que permitem a inserção de algoritmos e programação em cursos técnicos integrados ao ensino médio [Brasil 2015].

Devido às dificuldades enfrentadas para inserir os conteúdos básicos da área de computação nos anos escolares, o que tem-se feito são trabalhos, muitas vezes de extensão, realizados por pesquisadores e alunos de graduação de cursos de Computação de diversas regiões do país [Oliveira et al. 2014].

Iniciativas como Pernambucoders¹, Computação na Escola², Projeto Rope³, *CodifiKIDS*⁴, Codifique⁵, Letramento em Programação⁶, Code.org⁷, Programaê⁸, Escola de Hackers⁹ e Computação na Escola¹⁰, buscam disseminar o ensino de computação no ensino fundamental e médio levando aos alunos conceitos de programação. Os alunos são desafiados a construir artefatos com o auxílio das linguagens de programação trabalhando sozinhos ou em times para a criação de jogos, aplicativos e outras mídias.

Esses projetos já estão implementados em algumas escolas públicas de diferentes estados e, além de incentivar os alunos a aprender programação, também visam desenvolver e aprimorar o raciocínio lógico e matemático, bem como a criatividade.

Porém, para participar, é necessário que as escolas ofereçam laboratórios de informática com computadores em bom estado e em quantidade suficiente para atender todos os alunos. Embora presentes em 81% das escolas públicas, o estudo realizado por [NIC.BR 2017] demonstra que os laboratórios de informática são efetivamente utilizados em pouco mais da metade delas, cerca de 59%. Uma outra dificuldade enfrentada por esses projetos é abranger todos os alunos, visto que a maioria deles são limitados a um determinado número de vagas.

O ideal para minimizar essas dificuldades é que haja a inserção do ensino de programação e algoritmos na BNCC. Assim, todas as escolas públicas deverão introduzir conceitos básicos de lógica computacional. Essa ação fomentará, inclusive, novas pesquisas voltadas para esse tipo de abordagem pedagógica.

¹Disponível em: <https://www.cesar.org.br/index.php/portfolio/pernambucoders> Acessado em: 17/01/2019.

²Disponível em: <http://www.computacaonaescola.ufsc.br/> Acessado em: 17/01/2019.

³Disponível em: <http://lite.acad.univali.br/pt/rope-brinquedo-de-programar/> Acessado em: 17/01/2019.

⁴Disponível em: <https://vitorgt.typeform.com/to/VS1ZXX> Acessado em: 09/07/2019.

⁵Disponível em: <https://cursocodifique.weebly.com/> Acessado em: 09/07/2019.

⁶Disponível em: <https://www.institutoayrtonsenna.org.br/pt-br/Atuacao/letramento-em-programacao.html> Acessado em: 09/07/2019.

⁷Disponível em: <https://code.org/international/about> Acessado em: 09/07/2019.

⁸Disponível em: <http://programae.org.br/> Acessado em: 09/07/2019.

⁹Disponível em: <https://www.udemy.com/user/escola-de-hackers/> Acessado em: 09/07/2019.

¹⁰Disponível em: <http://www.computacaonaescola.ufsc.br> Acessado em: 09/07/2019.

3. Materiais e Métodos

Com o objetivo de compreender o processo de apresentação dos conceitos de algoritmos e de programação em escolas de ensino básico, desenhamos um estudo de caso para, de um lado, explorar o ensino de algoritmos com matemática e, de outro lado, revisar os conceitos de matemática a partir de algoritmos.

O estudo de caso foi realizado em uma escolar pública estadual periférica, no laboratório de informática da escola, sempre no contraturno escolar. O curso foi dividido em duas partes, na primeira os conceitos de programação foram apresentados a partir de um ambiente de programação visual, enquanto a segunda etapa introduziu a linguagem de programação de propósito geral *Python*¹¹.

Durante o curso foram utilizadas as ferramentas descritas a seguir, sendo todas elas de licença *open source*: (i) o *iVProg*¹². foi utilizado para a programação visual e (ii) o pacote VPL (Virtual Programming Lab)¹³ foi usado na programação em *Python*. Todas as ferramentas foram usadas de modo integrado ao gerenciador de conteúdo *Moodle*. Além disso foram utilizados os recursos para avaliação automática baseada em casos de teste, disponíveis tanto no *iVProg* quanto no *VPL/Moodle*.

A escolha por essas ferramentas também se deve ao fato de que elas podem ser executadas *online*. Essa característica pode facilitar a experiência dos alunos e promover uma melhor supervisão do professor, já que podem ser utilizadas fora do período de aula. Outra característica importante das ferramentas utilizadas é que elas permitem retroação imediata, a partir de recursos de avaliação automática. Com isso o aprendiz pode saber se está ou não no caminho certo, bastando clicar em um botão que dispara o avaliador.

O projeto tinha como objetivo principal motivar os alunos para aprenderem matemática e conceitos básicos de programação. Uma vez que o aprendizado de computação costuma ser estimulante para jovens aprendizes, o projeto esperava que seus participantes ficassem também mais estimulados a aprender matemática. Para isso o projeto buscava apresentar os conceitos de programação e estimular o pensamento computacional a partir dos conteúdos de matemática já estudado previamente como, por exemplo, progressões aritméticas e geométricas. De outro lado, o projeto visava reforçar o aprendizado de matemática, revisitando diferentes tópicos, a partir de desafios de programação.

A participação dos alunos foi voluntária e todos os alunos do primeiro ano do ensino médio da escola podiam se inscrever. Dos inscritos foram sorteados 24 alunos (12 alunos por turno). As turmas participantes foram limitadas a 12 alunos, pois o laboratório de informática da escola dispunha de 13 computadores (um deles utilizado pelos instrutores). Todos os computadores estavam conectados à internet. Desse modo, as duas turmas que participaram do estudo foram distribuídas contra o turno em que os alunos estudavam, onde os alunos da manhã fizeram o curso à tarde e vice-versa.

O método utilizado no projeto foi de realizar atividades semanais, com 2 aulas por semana e duração de 1h40m cada. Na primeira aula apresentava-se conceitos teóricos e discutia-se as atividades propostas. Os alunos iniciavam a realização das atividades, de

¹¹Disponível em: <https://docs.python.org> Acessado em: 15/07/2019.

¹²Disponível em: <https://github.com/LInE-IME-USP/ivprog-html> Acessado em: 15/07/2019.

¹³Disponível em: <https://vpl.dis.ulpgc.es/> Acessado em: 15/07/2019.

modo individual, e essas atividades ficavam abertas para serem realizadas no período de uma semana. Com isso, o aluno tinha um tempo extraclasse para praticar e resolver os exercícios propostos, além do período da segunda aula semanal, onde podiam também tirar dúvidas com o professor. No início da semana seguinte as atividades eram finalizadas.

O curso foi dividido em dois módulos. No primeiro, os estudantes aprenderam os conceitos iniciais de algoritmos utilizando o ambiente de programação visual *iVProg* (Figura 3). No segundo módulo, foi apresentada a linguagem de programação *Python*. Para isso foi utilizado como ambiente de programação o sistema VPL (Figura 4).

- Exercício**
- 1.1 Introdução - ler um número inteiro (int) e imprimir este valor.
 - 1.2 Introdução - gerar o quadrado do valor de um número dado
 - 1.3 Introdução - ler dois números inteiros (int) e imprimir sua soma
 - 1.4 Introdução - ler dois número inteiros (int) e imprimir a média aritmética dos valores.
 - 1.5 Introdução - obter o maior valor entre 2 números
 - 1.6 Introdução - dado um inteiro, decidir se é par ou impar

Figura 2. Imagem com enunciado das primeiras atividades no ambiente usado

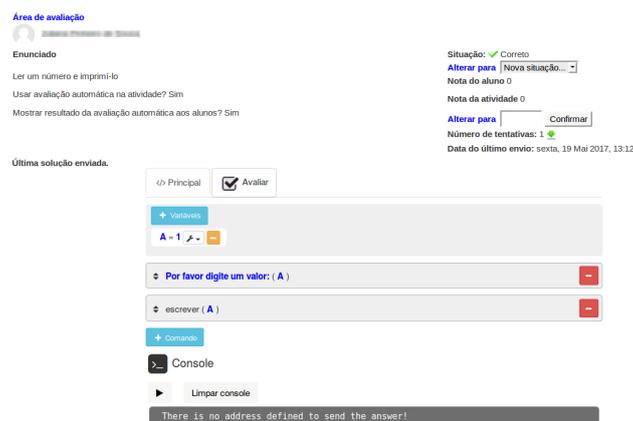


Figura 3. Solução enviada via *iVProg* por um aluno para a atividade 1.1

3.1. Estudo de caso em uma escola pública estadual

Inicialmente criamos dois módulos, cada um com duas turmas, todos hospedados em um servidor com o *Moodle* e os pacotes *iVProg* e *VPL*. As aulas foram ministradas pelo professor da escola com orientação da equipe responsável pela pesquisa, e tinham duração de uma hora e quarenta minutos. Sendo turmas de primeiro ano do ensino médio, a proposta foi ensinar algoritmos e programação e relacionar com o conteúdo da disciplina de matemática, tais como conjuntos numéricos, funções, progressão aritmética e geométrica.

No primeiro módulo, ministrado no primeiro semestre de 2017, os alunos são apresentados ao ambiente gerenciador de curso. A seguir foi feita uma apresentação dos conceitos elementares de algoritmos e do sistema *iVProg*. Nesse módulo realizou-se 7 aulas, onde em cada aula foram sugeridos cerca de 6 exercícios práticos, 1 exercício problema (exercício mais complexos e com tempo limite de entrega), e 1 projeto final totali-

zando 44 atividades nesse módulo, todos baseados no conteúdo proposto de matemática para o primeiro ano do ensino médio.

No segundo módulo, após cerca de seis semanas de atividades do módulo 1, os alunos foram apresentados a uma linguagem de programação de uso geral, o *Python*, sempre voltada com exercícios e explicações baseadas no conteúdo proposto de Matemática. Nesse módulo realizou-se 4 aulas, onde em cada aula foram sugeridos cerca de 6 exercícios práticos, 1 exercício problema (exercício mais complexos e com tempo limite de entrega), e 1 projeto final totalizando 26 atividades nesse módulo.

A dinâmica de aula ocorreu da seguinte forma: a aula se iniciava com a leitura de aula, onde os alunos aprendiam sobre o *iVProg* e faziam um exemplo junto ao professor. Logo após, o professor tirava as dúvidas dos alunos e com isso, podiam partir para os exercícios. Nesse momento, os alunos atuam como protagonistas do processo de aprendizagem, no qual o professor é apenas um observador.

Na Figura 2, é possível observar uma tela do ambiente do curso, com os títulos (e acessos) para as atividades utilizadas na primeira aula. Nesse bloco, as atividades são focadas no aprendizado do ambiente e no recurso para avaliação automática disponível na ferramenta utilizada, o *iVProg*. Para completar a Figura 4 mostra a solução enviada por um aluno para a atividade 1.1, presente na Figura 2. Nessa atividade o aluno precisa criar uma variável, inserir um comando para leitura/entrada de dados (a ser digitado pelo “usuário” de seu programa) e depois um comando para impressão/saída do valor digitado.

Sobre as dificuldade enfrentadas, devemos destacar as dificuldades do professor da escola que dirigia as atividades. O professor teve um treinamento intensivo de cerca de 30 horas e relatou algumas dificuldades com algoritmos e programação. Além disso, houve problemas de paralisações que gerou adiamentos e cancelamento de algumas aulas. Outro problema significativo foi a dificuldade para obter dados sobre o desempenho dos alunos para podermos analisar eventuais mudanças a partir do envolvimento com o projeto.



```
Le1*.py
1 a = raw_input('digite um numero\')
2 print a
```

Figura 4. Solução enviada para uma atividade equivalente à 1.1 usando o VPL

4. Considerações finais

Aprender uma linguagem de programação atualmente é tão importante quanto aprender a ler e escrever [Rushkiff e Silva 2012]. O ensino de programação é importante porque estimula a criatividade e a autonomia. Além de que, desenvolve o raciocínio lógico, a capacidade de resolução de problemas e o trabalho em equipe.

No desenvolver das atividades do projeto relatado na seção 3, foi possível notar nos alunos o desenvolvimento da abstração em relação aos exercícios e a lidar com variáveis propostas. Outro ponto notado, foi a reconstrução dos conceitos que os alunos

fizeram sobre o conteúdo que eles haviam aprendido, como por exemplo um aluno que citou que após a realização do curso conseguia ver sequências de outra forma e estava enxergando a resolução do problema como se fosse uma sequência de passos.

Na atividade final do projeto, cabia ao aluno observar um problema do cotidiano e propor uma solução que eventualmente poderia ser implementada usando o *software iVProg* ou a linguagem de programação *Python*. Todas as propostas apresentadas foram bastante coerentes com o cotidiano dos alunos e suas aspirações, principalmente quanto ao futuro profissional e carreira. Na apresentação desses projetos, todos os programas desenvolvidos puderam ser executados e estavam coerentes com as propostas iniciais. Isso demonstra que os alunos conseguiram compreender os conceitos de programação e aprenderam a utilizar e aplicar uma linguagem de programação para a solução de problemas.

Com o objetivo de analisar a motivação do aluno em aprender matemática, utilizamos como métrica a Escala de Motivação Matemática [Gontijo e Fleith 2009], aplicando questionário no início e no fim da pesquisa. Com isso, pudemos comparar suas respostas e fazer a inferência de que houve um aumento na motivação em aprender matemática, bem como um aumento na interação dos alunos com o professor da disciplina.

Além disso, observamos que os alunos compreenderam os conceitos de algoritmo e conseguiram dominar a linguagem de programação *Python*. Isso ficou evidente na apresentação do projeto final, devido ao nível de maturidade das propostas e execução dos programas desenvolvidos. Por fim, notamos que as notas dos alunos na disciplina de matemática tiveram um aumento significativo nos dois últimos bimestres observados.

Apesar das dificuldades enfrentadas no desenvolver do projeto, este nos trouxe reflexões importantes acerca do ensino de algoritmos e programação no ensino básico. Neste estudo de caso o experimento foi voltado para a matemática, mas essa abordagem pedagógica pode ser empregada para crianças que não têm afinidade com a área de exatas. Possibilitando observar o desempenho das mesmas em outras áreas a partir do ensino de programação. O que esse estudo de caso nos mostra é que, pelo menos em matemática, há melhora em diversas vertentes (cognitiva, social, desempenho escolar) para os alunos.

Portanto, com base não apenas nesse estudo, mas no panorama apresentado na seção 2 e em trabalhos como o de [Geraldes 2014], reafirmamos a importância da inserção do ensino de programação no ensino básico brasileiro por meio da BNCC. Para que o Brasil atinja níveis cada vez mais altos nos *rankings* mundiais de educação, melhorando a qualidade da educação e aumentando o número de profissionais nas áreas de tecnologia.

5. Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Referências

- Brasil (2015). Projeto pedagógico do curso técnico em informática integrado ao ensino médio. *IFSP Disponível em* <<http://www2.ifsp.edu.br/index.php/arquivos/category/305-documentos.html>>. Acesso em 09 jul. 2019.
- Brasil (2017). Base nacional comum curricular. *Ministério da Educação e Cultura. Disponível em* <<http://basenacionalcomum.mec.gov.br/>>. Acesso em 09 jul. 2019.

- Brasil (2018). Avaliação dos Cursos de Graduação. *Ministério da Educação e Cultura*. Disponível em <<http://basenacionalcomum.mec.gov.br/>>. Acesso em 09 jul. 2019.
- CiEB (2019a). A importância de políticas nacionais e centros de inovação em educação. Disponível em <<http://cieb.net.br/cieb-notas-tecnicas-1-a-importancia-de-politicas-nacionais-e-centros-de-inovacao-em-educacao/>>. Acesso em 09 jul. 2019.
- CiEB (2019b). Referências para construção do seu currículo em tecnologia e computação da educação infantil ao ensino fundamental. Disponível em <<http://curriculo.cieb.net.br/>>. Acesso em 09 jul. 2019.
- CSTA (2005). The new educational imperative: Improving high school computer science education. Computer Science Teachers Association.
- França, R. e Amaral, H. (2013). Ensino de computação na educação básica no Brasil: Um mapeamento sistemático.
- Geraldes, W. (2014). Programar é bom para as crianças? uma visão crítica sobre o ensino de programação nas escolas. *Texto Livre: Linguagem e Tecnologia*, 7(2):105–117.
- Gontijo, C. H. e Fleith, D. d. S. (2009). Motivação e criatividade em matemática: um estudo comparativo entre alunas e alunos de ensino médio. *ETD - Educação Temática Digital*, 10:147–167.
- Heintz, F., Mannila, L., e Färnqvist, T. (2016). A review of models for introducing computational thinking, computer science and computing in k-12 education. In *2016 IEEE Frontiers in Education Conference (FIE)*, páginas 1–9.
- IBGE (2019). *Conheça Cidades e Estados do Brasil*. Instituto Brasileiro de Geografia e Estatística. Disponível em: <https://cidades.ibge.gov.br/>. Acessado em 10 Junho 2019.
- Kenski, V. M. (2008). Educação e tecnologias: o novo ritmo da informação. *Práxis Educativa*.
- NIC.BR (2017). TIC Educação. *CETIC.br* Disponível em <<https://cetic.br/pesquisa/educacao/>>. Acesso em 09 jul. 2019.
- Oliveira, M. L. S., Souza, A., Barbosa, A. F., e Barreiros, F. S. (2014). Ensino de lógica de programação no ensino fundamental utilizando o scratch: um relato de experiência. In *Congresso da Sociedade Brasileira de Computação (CSBC)*, páginas 1525–1534.
- Rushkiff, D. e Silva, C. (2012). *AS 10 QUESTOES ESSENCIAIS DA ERA DIGITAL*.
- Santana, B., Rossini, C., De, N., e Pretto, L. (2012). *Recursos Educacionais Abertos: Práticas Colaborativas e Políticas Públicas*. São Paulo, Salvador, 1º edição edition.
- SBC (2018). Nota Técnica da Sociedade Brasileira de Computação sobre a BNCC-EF e a BNCC-EM. Disponível em <<http://sbc.org.br/institucional-3/cartas-abertas/category/93-cartas-abertas>>. Acesso em 10 jul. 2019.
- São Paulo (2019a). Currículo da cidade: Ensino fundamental: componente curricular: Matemática.
- São Paulo (2019b). Currículo da cidade: Ensino fundamental: componente curricular: Tecnologias para aprendizagem.
- Wing, J. M. (2006). Computational thinking. *Commun. ACM*, 49(3):33–35.