

ENSINO DE PROGRAMAÇÃO: UMA REVISÃO SISTEMÁTICA E AS APLICAÇÕES AO ENSINO PROFISSIONAL

Daniel Lima Gomes Júnior¹
daniellima@ifma.edu.br

Diego Ted Rodrigues Boguea²
diego.bogea@ifma.edu.br

RESUMO

Devido ao grande avanço das tecnologias computacionais, a programação tem se tornado uma habilidade necessária no mercado de trabalho em áreas especializadas. Nesse sentido, o ensino de disciplinas introdutórias de programação constitui-se um desafio tanto para educadores quanto para alunos, visto que se trata de uma mudança de paradigma e maneira de análise de um determinado problema. Pode-se dizer que essa mudança tem aspectos na produção autoral e, do ponto de vista curricular, possui ensinamentos que são base para todo o curso nas áreas de informática. Este trabalho apresenta uma revisão sistemática da literatura relacionada ao Ensino de Programação e faz uma associação ao contexto da educação profissional. Essa investigação visa contribuir para a aplicações ao contexto brasileiro, elencando os principais desafios nessa área, através da consulta de 91 publicações acadêmicas nas bases IEEE Xplore, ACM, Science Direct e Springer. Os resultados indicam a importância da disciplina de programação em diversos países, com abordagens metodológicas para combate aos desafios inerentes dessa atividade de ensino, além da análise voltada à educação profissional.

Palavras-chave: Ensino; Programação; Educação Profissional.

ABSTRACT

Due to the great advance of computing technologies, programming has become a necessary skill in the job market in several specialized areas. Thus, the teaching of introductory programming is a challenge for both educators and students, since it is considered a new paradigm and a different way of problem analysis. This paradigm change has aspects in the authorial production and, from the curriculum change, it has lessons that are the basis for some entire course in the areas of information technology. This work presents a systematic review of the literature related to the Teaching of Programming and makes an association with the professional education context. This investigation aims to contribute with applications for the Brazilian scenario, listing the main challenges, through the review of 91 academic publications in the IEEE Xplore, ACM, Science Direct, and Springer academic databases. The results indicate the importance of the programming discipline in several countries, with some methodological approaches for surpassing the inherent challenges of the teaching activity, in addition to the analysis aimed at professional education.

Keywords: Teaching. Programming. Professional Education.

¹ Doutor. Professor do Instituto Federal do Maranhão – Campus Monte Castelo.

² Mestre. Professor do Instituto Federal do Maranhão – Campus Imperatriz.

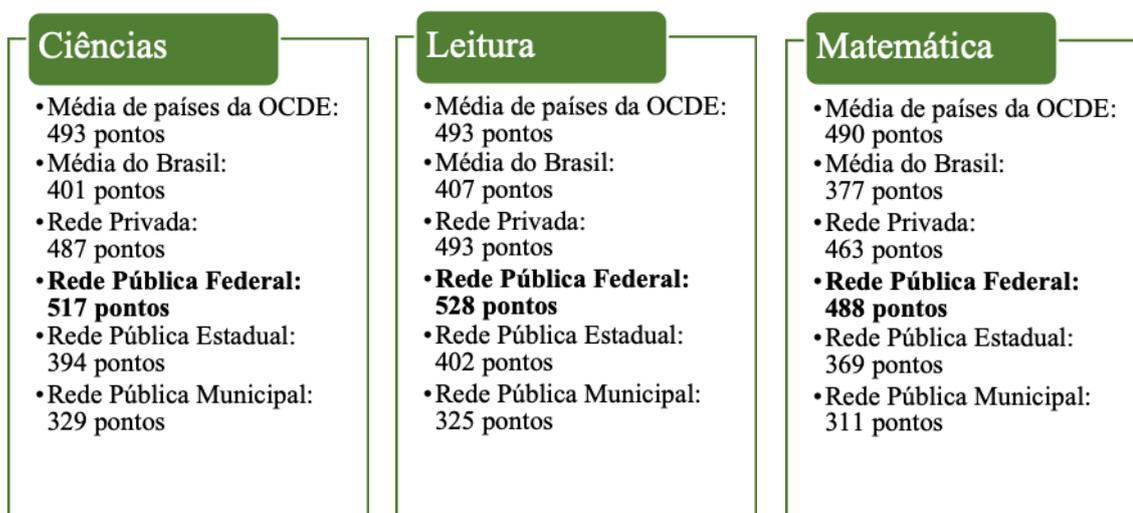
1. Introdução

A área da informática tem se tornado, ao longo dos anos, um pré-requisito em áreas interdisciplinares. Essa afirmação ganhou ainda mais atenção com a decisão de países como os Estados Unidos e Reino Unido tornaram a disciplina de programação obrigatória no currículo de escolas primárias (HUSSAIN, FERGUS, AL-JUMEILY, PICH, & HIND, 2015). Essa mudança de direcionamento possibilita aos alunos e professores envolvidos o uso de tecnologias que apareceram e muitos tem à disposição no contexto americano.

No Brasil, os desafios – já existentes e outros colocados à mostra durante a pandemia do covid-19 – são ainda maiores. Pode-se afirmar que estão ligados ao contexto social de nosso país. Dificuldades de acesso à internet e às tecnologias que poderiam auxiliar nas disciplinas ligadas à programação são evidentes obstáculos à melhoria do processo de ensino-aprendizagem. No entanto, compreender qual a linha de ação está sendo utilizada em outros lugares pode nos ajudar a definir um escopo de atuação voltado ao contexto nacional.

No contexto brasileiro, os dados do Programa Internacional de Avaliação de Estudantes (PISA) de 2015 (INEP, 2016) mostram que os Institutos Federais tiveram um resultado superior à média dos estudantes no Brasil nas três áreas avaliadas. E, de acordo com (MATSUMOTO, ALBUQUERQUE, RITA, & PINTO, 2019), superaram a média de países do OCDE, com exceção em matemática, como mostrado na Figura 1.

Figura 1: Performance das Instituições Federais no PISA/2015



Fonte: Adaptado de (MATSUMOTO, ALBUQUERQUE, RITA, & PINTO, 2019)

Em uma suposição preliminar como professor da rede federal, esse resultado poderia estar associado à utilização da integração entre ensino, pesquisa e extensão, realizada dentro das instituições federais. Além de pesquisa e uso de metodologias ativas, defendida por diversos professores da rede em abordagens reais. O presente resultado motivou esta revisão sistemática na área de programação, visto que uma revisão sistemática possibilita o levantamento de informações e melhor compreensão do cenário em um determinado contexto.

De acordo com (SANTOS, et al., 2018), é essencial acompanhar a constante evolução da informática na Educação para que os pesquisadores possam conhecer e aplicar essas inovações tecnológicas em sala de aula. Essas inovações no âmbito da informática podem ser aplicada tanto a várias disciplinas e também reaplicadas na própria informática, como em disciplinas introdutórias dos cursos de computação.

Este trabalho pretende contribuir, nesse contexto, com a temática do ensino de programação para guiar docentes e pesquisadores da área de informática em relação aos desafios no ensino de programação, respondendo as seguintes questões:

- Q1: Quais as principais linguagens de programação utilizadas no ensino de programação?
 - Hipótese: Python, devido à facilidade de uso e ao crescimento nos últimos anos.
- Q2: Quais as principais estratégias e/ou tecnologias aplicadas ao ensino de disciplinas introdutórias em programação?
 - Hipótese: uso de metodologias ativas associada a novas tecnologias.
- Q3: Quais os principais desafios relacionados ao ensino de programação?
 - Hipótese: Evasão escolar (ou desistência) e a mudança de paradigma.

A principal motivação desse trabalho está relacionada a mais de uma década de ensino e o desafio vivenciado por alunos de nível médio, profissional e superior. Mesmo dentro das universidades e institutos de ensino superior, há um sentimento de angústia relacionado ao baixo rendimento dos alunos em disciplinas de programação. E os docentes sentem-se, muitas vezes, incapazes de mudar o cenário e ajudar discentes nessas disciplinas. Ademais, estamos inserido em um contexto das Tecnologias Digitais e, dessa forma, Informática e Programação

se fazem necessárias do ponto de vista logístico para sustentação de toda a base do desenvolvimento tecnológico.

O restante deste trabalho está estruturado em Metodologia (Seção 2), que trata da abordagem escolhida para a Revisão Sistemática. Em seguida, na seção 3, os Resultados e Discussões dos dados obtidos na pesquisa. Por fim, a Seção 4 apresenta as Considerações Finais do presente trabalho, destacando os aspectos associados ao Ensino de Programação.

2. Desafios, abordagens e ensino profissional

De acordo com a análise realizada em (BERGER FILHO, 1999), conciliar os objetivos da preparação aos estudos, ao mercado de trabalho e ao desenvolvimento pessoal nos cânones contemporâneos é a grande questão dos sistemas educacionais. A afirmação já possui mais de duas décadas, mas continua atual. E torna-se cada vez mais desafiadora, devido à velocidade das mudanças nas últimas décadas.

Apesar do desafio apresentado ser imenso, diversas tecnologias e metodologias têm sido apresentadas para auxílio à mudança de paradigma vigente na educação em todo o planeta. Modelos mais ativos de ensino, de acordo com (BOTTENTUIT JUNIOR, 2019), tem se tornado a grande preocupação no cenário educacional, visto que diversas características corroboram para este caminho, como a competência exigida dos alunos, demandas do mercado e competências próprias no século XXI. Esta última característica é o foco do presente trabalho, tendo como objetivo fazer uma avaliação dos desafios, tecnologias e abordagens de ensino, especialmente voltadas à disciplina de programação.

Segundo (COUTINHO, 2007), a concepção de tecnologia educativa vem se consolidando há mais de um século, sendo compreendido como uma forma sistemática de concepção, gestão e avaliação do processo de ensino aprendizagem. A autora faz ainda a correlação entre Tecnologia Educativa e Desenvolvimento Curricular, que é um desafio importante, principalmente ao panorama pedagógico ocidental.

Além dos desafios de integração e a adequação a um paradigma educacional consistente, a utilização de tecnologias nas modalidades presencial e a distância agravam o cenário educacional, ainda mais no Brasil, um país de extensa faixa territorial e com grandes extremos de acesso às tecnologias.

No contexto do ensino profissional, apesar dos vários significados associados ao formato, há uma preocupação da educação omnilateral que supõe integrar tanto conhecimentos gerais quanto específicos – em nosso caso, a programação – e pressupõe uma

medida que integre trabalho, ciência e cultura de forma associada à educação dos discentes (MOURA, 2016, p.68).

Em uma análise bastante interessante e relacionada ao ensino profissional, feito em (KUENZER & GRABOWSKI, 2006), é levantada a questão da formação para o trabalho. E da nova exigência do domínio do trabalho intelectual aos que vivem do trabalho, que anteriormente era prerrogativa da burguesia. A autora afirma ainda que esse contexto aplicado à educação profissional está associado ao modo de produção capitalista, sintetizando contradições do próprio sistema e constituindo-se, simultaneamente, em espaço de possibilidades e impossibilidades.

Em relação às tecnologias de ensino, o trabalho de (KENSKI, 2003, p.31) afirma que no campo da linguagem digital não é mais possível identificar todas as novas tecnologias, suas finalidades e os graus de complexidade. Ainda de acordo com Kenski (2003), a tecnologia digital rompe com as narrativas contínuas e com textos sequenciais, podendo ser caracterizada como um fenômeno descontínuo, além de abordar novos modos de compreender os assuntos nesse contexto diversificado.

Nesse contexto, que trata do ensino em si, e aborda diversas áreas de conhecimento, podemos afirmar que o ensino de programação se desenvolve em outras ramificações diversas, devido a esse avanço de tecnologias e linguagens de programação, como afirmado por Kenski. Na verdade, se avaliarmos a variedade de linguagens de programação disponíveis, podemos fazer a analogia que o desenvolvimento dessas linguagens se dá por esse fenômeno descontínuo. Cada problema específico, cada análise diferenciada leva a uma solução ao mesmo tempo similar e diferente em diversos aspectos.

A abordagem utilizada no ensino de programação é parte fundamental desse processo. De acordo com (ARAÚJO, 2018), a replicação e identificação de abordagens por professores e pesquisadores que investigam o processo de ensino-aprendizagem permite a construção de materiais pedagógicos e propostas que melhoram esse processo. O autor apresenta uma abordagem ao contexto da educação profissional e levanta a dificuldade do ensino de programação, ressaltando que, no contexto da educação profissional, somam-se problemas históricos existentes nessa modalidade.

É possível que a dificuldade apresentada pelos alunos na área de programação, esteja relacionada ao próprio pensamento computacional que, de acordo com a Sociedade Brasileira

de Computação (SBC), é um processo cognitivo usado pelos seres humanos para resolver problemas através do uso de abstração, refinamento, modularização, recursão e metacognição. Nesse sentido, essa mudança de perspectiva, modifica também o relacionamento do indivíduo com o mundo (GARLET, BIGOLIN, & SILVEIRA, 2018).

Com base nessa discussão, o trabalho proposto e realizado nessa revisão sistemática foi compilar os resultados, dentro das principais bases ligadas à área de computação, de abordagens metodológicas e aplicações ligadas ao ensino de programação.

3. Metodologia

Uma revisão sistemática da literatura constitui-se uma técnica importante para avaliação das pesquisas acadêmicas e abordagens realizadas em uma área específica de conhecimento (KHANGURA, 2012).

Uma revisão sistemática é um trabalho que deve ser realizado de forma periódica e cada levantamento pode destacar uma nuance específica. Na área de educação, existem diversas análises como desafios dos alunos, currículo, legislação vigente, etc. Nesse sentido, a realização de revisão sistemática possui fundamentação e importância nítida para análise de um determinado tema. Na área de informática, essa pesquisa torna-se ainda mais importante, visto que essa prática poderia ser melhor aproveitada.

Para o desenvolvimento deste trabalho, foram realizados as seguintes ações:

- 1) Levantamento das questões de análise e as principais hipóteses;
- 2) Consulta de trabalhos relacionados às questões levantadas em bases acadêmicas voltadas a computação;
- 3) Compilação dos resultados e aplicação dos critérios de inclusão/exclusão;
- 4) Divisão em categorias de análise diferentes, sendo essas: linguagem de programação, tecnologias inovadoras, abordagens de ensino, e desafios no ensino de programação.
- 5) Contabilização dos resultados discretos;
- 6) Análise dos resumos e validação dos desafios apresentados nos trabalhos.

Obviamente, existem diversas bases acadêmicas para realização de uma revisão sistemática. De acordo com (BRERETON, KITCHENHAM, BUDGEN, TURNER, & KHALIL, 2007), sugere-se a realização das consultas em quatro bases eletrônicas: IEEE

Xplore³, ACM Digital⁴, Springer⁵, e ScienceDirect⁶. Essa abordagem foi utilizada no presente trabalho, sendo que as consultas realizadas nessas bases usaram o conjunto de palavras-chave nos títulos dos trabalhos e foram analisados os resumos (abstracts) de cada um.

Para realização de uma consulta ampla relacionada ao tema, foram consideradas nas buscas os termos em português e inglês, visto que muitas pesquisas realizadas são traduzidas para publicação nos veículos mencionados. Dessa forma, as palavras-chave utilizadas na construção da *string* de busca foram ***“Ensino AND Programação” OR “Teaching AND Programming”***.

Com base nos resultados encontrados, foram definidos um conjunto de critérios de inclusão e critérios de exclusão para obtenção de trabalhos relacionados às questões Q1, Q2 e Q3, definidos de acordo com as subseções seguintes.

3.1 Critérios de inclusão

Para contabilização dos resultados obtidos, foram utilizados os seguintes critérios de inclusão:

- a) Trabalhos publicados no período de 2018 a 2020;
- b) Trabalhos com foco em abordagens de ensino/aprendizagem de programação;
- c) Com aplicação de tecnologias emergentes para auxílio no aprendizado;
- d) Trabalhos que apresentam desafios inerentes ao ensino de programação;
- e) Trabalhos com uso de linguagens de programação tradicionais para ensino de programação.

3.2 Critérios de exclusão

Devido à grande quantidade de resultados obtidos, o presente trabalho definiu os seguintes critérios de exclusão:

³ <http://ieeexplore.ieee.org>

⁴ <http://dl.acm.org>

⁵ <http://www.springer.com>

⁶ <http://www.sciencedirect.com>

- a) Visto que nosso foco é educativo, os trabalhos com uma abordagem tão somente técnica, sem relação alguma com o Ensino, foram excluídos;
- b) Investigações que sejam educacionais, mas de áreas diferentes do foco de análise, a saber: a programação;
- c) Short-papers foram excluídos por conta de sua densidade e contribuições relevantes;
- d) Como nossa análise está centrada em um revisão sistemática, os estudos com a mesma finalidade foram excluídos.

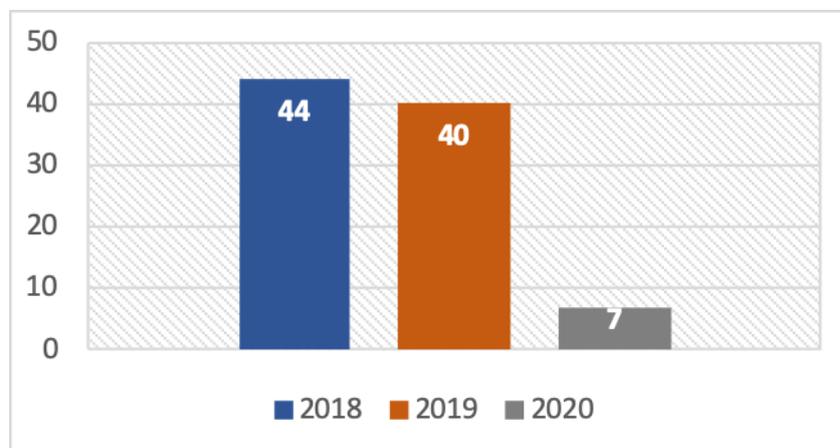
Na próxima seção apresenta-se o cenário do ensino de programação no Brasil e a contextualização dos desafios e oportunidades existentes na disciplina que se mostra fundamental às nossas profissões e necessidades de formação interdisciplinar, tendo como base a disciplina de programação. Essa perspectiva interdisciplinar se apresenta ainda como uma possibilidade de inclusão para melhorar o desempenho na área de exatas nos alunos das redes de ensino no Brasil.

4. Resultados e discussão

A pesquisa foi conduzida dentre os meses de junho a agosto de 2020. Foram obtidos, com base na consulta apresentada o total de 277 artigos científicos, obtidos das bases mencionadas IEEE Xplore (113), ACM Digital (99), Springer (10), e Science Direct (55).

Com esse quantitativo de artigos, foram aplicados os critérios de inclusão e exclusão apresentados na seção 3. Esse processo resultou na seleção de 91 artigos, de 2018 a 2020, com quantitativos apresentados na Figura 2.

Figura 2: Quantidade de artigos por ano

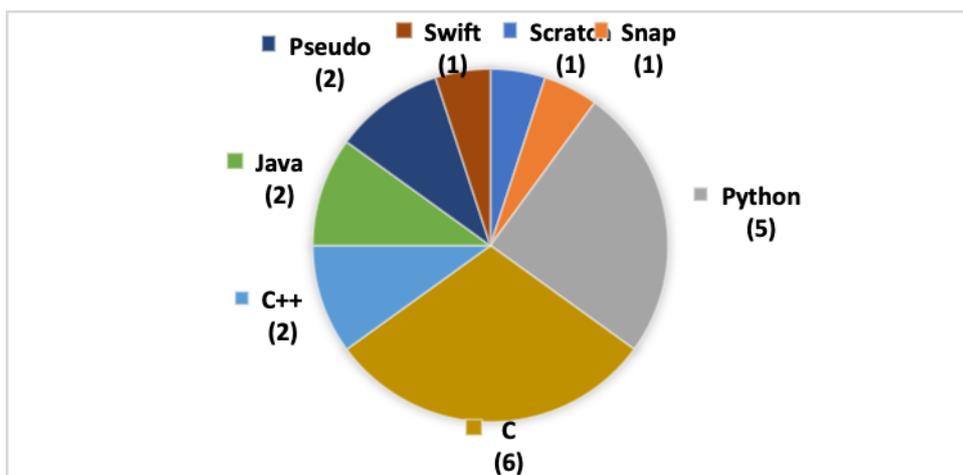


Fonte: Dos autores.

Dos trabalhos avaliados, alguns se enquadraram em mais de uma das categorias mencionadas, sendo 19 trabalhos que tratam do uso de linguagens de programação (Figura 3); 14 trabalhos abordam tecnologias inovadoras; 42 trabalhos que mostram abordagens específicas de ensino; e, por fim, 23 trabalhos que tratam dos principais desafios encontrados no ensino de programação.

Pode-se inferir que há uma certa possibilidade de atuação interdisciplinar, tendo a linguagem de programação uma potencialidade nesse sentido. Além disso, a Figura 3 apresenta linguagens como Scratch e Snap, que possibilitam uma mudança de contextualização e facilitação ao ensino dos alunos, mesmo não sendo essas linguagens tradicionais de programação.

Figura 3: Linguagens de programação em ensino de programação (Q1)



Fonte: Dos autores.

Outro ponto importante dos trabalhos analisados foi o uso de abordagens e metodologias de ensino como a Aprendizagem Baseada em Projetos (do inglês *Project Based Learning* - PBL) associadas ao ensino de programação.

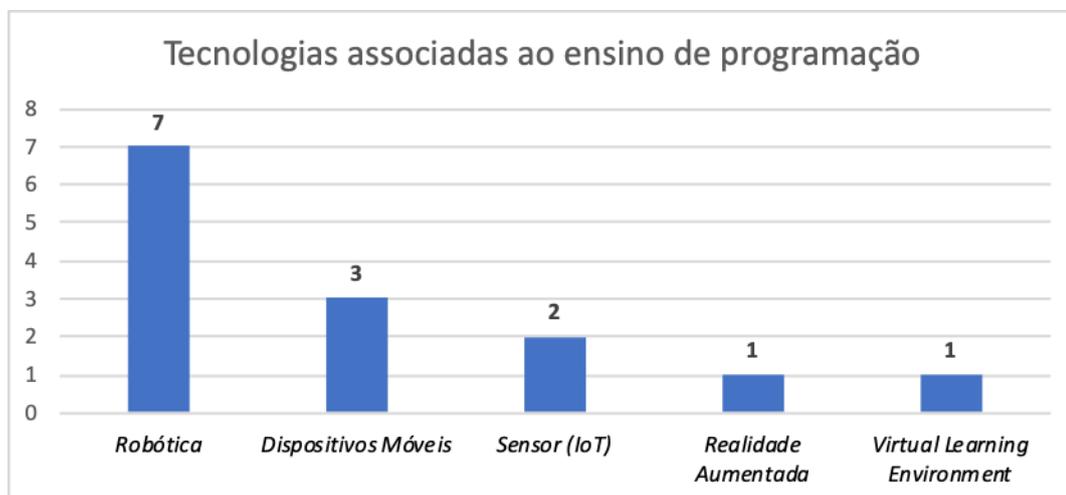
O trabalho de (dos SANTOS, et al., 2018) apresenta uma avaliação voltada ao contexto brasileiro e da abordagem utilizando PBL com alunos do programa de educação tutorial (PET), existente dentro das universidades públicas. Os autores trabalham o ensino de

programação voltado aos alunos dos anos iniciais com diferentes níveis de conhecimento e domínio, objetivando a diminuição da evasão escolar.

Em (BRANDÃO, FELIX, PEREIRA, & BRANDÃO, 2018), obtém-se a afirmação que as habilidades relacionadas ao pensamento computacional tem ganho importância e aplicada, muitas vezes, ao estágio inicial da educação formal. Esse trabalho vai de encontro à mudança mencionada anteriormente em relação ao currículo dos EUA. Vale ressaltar que nos Institutos Federais, há a ocorrência de diversas atividades indiretamente ligadas a esse contexto e aplicadas de forma interdisciplinar, como maratonas de programação, eventos acadêmicos de nível médio e superior, produção de jogos, dentre outros.

Além disso, identificamos tecnologias emergentes sendo utilizadas para facilitar ou avaliar o ensino de programação (Figura 4). Tecnologias como Robótica, Realidade Aumentada ou Sensores utilizado na perspectiva da Internet das Coisas (IoT) apresentaram abordagens interessantes associadas ao ensino de programação. Como dito anteriormente, já que o ensino possui diversos obstáculos, o uso dessas abordagens e tecnologias podem auxiliar e facilitar o entendimento dos alunos.

Figura 4: Principais tecnologias inovadoras associadas ao ensino de programação (Q2)



Fonte: Dos autores.

Dentre os resultados obtidos, os trabalhos que tratam de desafios e análises das perspectivas da disciplina de programação foram compilados e apresentados na Tabela 1. Esses trabalhos mostram as realidades em diferentes países e deixam clara a preocupação com a importância do ensino de programação de forma interdisciplinar.

Tabela 1: Principais desafios relacionados ao ensino de programação (Q3)

#	Artigo	Desafio abordado	Proposta apresentada	Nível escolar abordado	Aspectos relevante do trabalho
1	(SANTACRUZ-VALENCIA & RIAL-FERNÁNDEZ, 2019)	Aprendizado de alunos	Novo vocabulário para auxiliar a compreensão de conceitos	Educação Infantil (5 anos de idade)	Ressalta a necessidade de inclusão no currículo
2	(BUTHELEZI, 2018)	Aprendizado de alunos	Estudo exploratório que obteve melhoria após a intervenção do estudo	Educação Superior	Análise pedagógica
3	(FÖRSTER, FÖRSTER, & LÖWE, 2018)	Aprendizado de alunos	Construção de apps com Scratch	Ensino Fundamenta (alunos de 6º ano), com aprox. 50 estudantes	Integração de programação ao Currículo em matemática
4	(KAILA, LAAKSO, & KURVINEN, 2018)	Aprendizado de alunos	Ensino de programação associado à matemática para possibilitar uma mudança na forma de pensamento	Nível fundamental	Ressalta a importância da programação na sociedade moderna
5	(dos SANTOS, et al., 2018)	Minimização da evasão escolar	Abordagem PBL através do programa PET	Graduação	Contexto Brasileiro
6	(BLAS, GOLOBISKY, & GARCÍA LOZANO, 2018)	Acessibilidade	Apresenta ferramentas indicadas especificamente para auxiliar as pessoas cegas	Graduação	Objetivo secundário de minimizar evasão na Argentina
7	(SOOSAI RAJ, PATEL, & HALVERSON, 2018)	Aprendizado de alunos	Abordagem proposta para melhoria do aprendizado	Não se aplica	Tempo de aprendizado
8	(MARCOLINO, SANTOS, SCHAEFER, & BARBOSA, 2018)	Aprendizado de alunos	Sugere um catálogo de gestos associados a blocos de comandos visando facilitar o processo de ensino-aprendizado	Graduação	Orienta o uso de dispositivos móveis para auxílio no processo
9	(NICULESCU, SERBAN, & VESCAN, 2019)	Proposição curricular	Aplicação da estratégia <i>Cyclic Learning</i> para o aprendizado de programação	Graduação	Trata-se de um estudo de caso com resultados que demonstram o caráter interdisciplinar da programação
10	(ARIZA, 2018)	Proposição curricular	Propõe uma metodologia que integra hardware e software para o ensino de programação	Técnico	Induz os alunos a aprenderem uma nova linguagem de programação por conta própria
11	(LOYO, 2018)	Ensino com novas tecnologias	Proposta de Didática com uso da matemática para aperfeiçoamento do pensamento computacional (lógica)	Nível fundamental (10 a 13 anos)	Exemplos obtidos de cursos do México, Suíça e Colômbia
12	(McDONALD, 2018)	Aprendizado de alunos	Incluir no um exame ou pré-requisito de conhecimento em	Graduação	Aprendizado do primeiro ano de

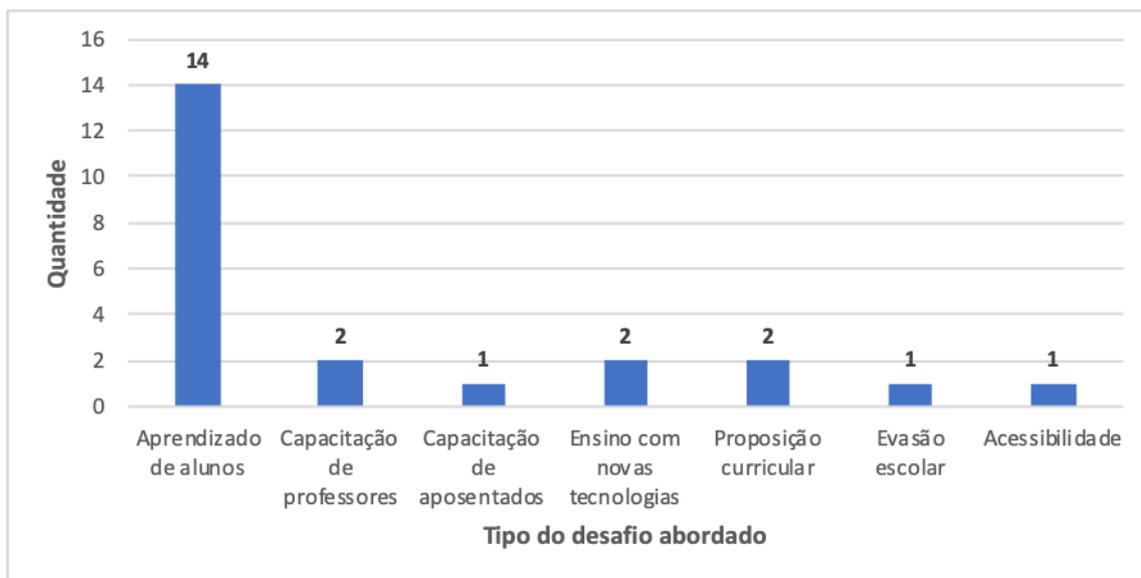
Ensino de programação: uma revisão sistemática

			Computação, forçando um estudo prévio nessa área		graduação em todas as universidades
13	(SHEN, DONGHEE YVETTE, & LEE, 2019)	Aprendizado de alunos	Propõe o uso de Ensino de programação híbrido, com ações presenciais e com sistemas baseados em tutores	Não se aplica	Alunos elogiam o ensino presencial e a vantagem dos trabalhos de codificação presenciais
14	(ARAUJO, BITTENCOURT, & SANTOS, An Analysis of a Media-Based Approach to Teach Programming to Middle School Students, 2018)	Ensino com novas tecnologias	Apresenta o conceito de uso de manipulação de imagens com programação em Python	Alunos de Ensino médio	Contexto brasileiro
15	(GELDREICH & TALBOT, 2018)	Capacitação de professores	Proposta de conteúdo em formato de workshop de 3 dias para treinamento com professores sem experiência anterior em programação	Educação infantil	Aborda o crescimento da demanda para computação na educação primária
16	(DONG, et al., 2019)	Capacitação a professores	Treinamento de 116 professores de Ensino médio e superior	Ensino médio e superior	Professores de disciplinas diferentes geram produtos interdisciplinares e distintos
17	(KO, et al., 2019)	Aprendizado de alunos	Uso de ferramentas como a depuração na codificação obtém melhorias nos resultados de aprendizagem	Estudantes de 15 a 18 anos	O ensino das estratégias de programação requer maior detalhamento aos alunos
18	(AIVALOGLOU & HERMANS, 2019)	Aprendizado de alunos	Uso de clubes de programação no período pós-escola (Code Clubs)	Alunos de Ensino médio	Apesar de pouca experiência em ensino, os Code Clubs apresentam maior interesse dos estudantes
19	(ANYANGO & SULEMAN, 2018)	Aprendizado de alunos	Análise aborda as dificuldades adicionais em países mais pobres. Estudo identificou que as maiores dificuldades dos alunos são: entendimento de recursão, vetores e tipos abstratos de dados.	Educação superior	O Ensino de programação apresenta desafios aos inexperientes
20	(GARCÍA-PEÑALVO & JOSÉ, 2019)	Aprendizado de alunos	Proposto um modelo preditivo para identificar alunos que precisam de mais ajuda através de exercícios	Graduação	Monitoramento dos estudantes é necessário
21	(HONG, YAO, MICHAEL, & PHILLIPS, 2018)	Aprendizado de alunos	Aborda o ensino de 2 linguagens de programação simultaneamente e obteve melhores resultados	Alunos de Ensino médio	Trabalho aborda o aspecto da linguagem
22	(OHASHI, YAMACHI, MUROKOSHI, KUMENO,	Capacitação de aposentados	Currículo de um curso promovido em projeto de extensão acadêmica	Aposentados e Estudantes nível	Possibilidade dos aposentados auxiliarem

	& TSUJIMURA, 2020)		no Japão com cidadãos aposentados	fundamental	no ensino de alunos de nível fundamental
23	(MARTÍNEZ-VALDÉS, GARCÍA-PEÑALVO, & VELÁZQUEZ-ITURBIDE, 2019)	Aprendizado de alunos	Integração da matemática com programação através do AppInventor	Não se aplica	Abordagem que evita erros iniciais cometidos em linguagens tradicionais (C++, Python, etc.)

Dentre os trabalhos analisados, em decorrência da identificação de desafios inerentes ao ensino de programação, a Figura 5 apresenta um resumo dos principais desafios abordados nos trabalhos dessa revisão sistemática.

Figura 5: Principais desafios identificados (Q3)



Fonte: Dos autores.

O trabalho de (SANTACRUZ-VALENCIA & RIAL-FERNÁNDEZ, 2019) mostra que países como Estados Unidos, Singapura, Estônia e Espanha estão realizando adaptações para inclusão da programação como área estratégica. Isso ressalta a afirmação prévia da preocupação com a formação dos alunos ao mercado de trabalho do século XXI.

Por outro lado, o trabalho de (FÖRSTER, FÖRSTER, & LÖWE, 2018) apresenta um resultado qualitativo interessante ao alinhar a programação com diferentes áreas de

conhecimento, tendo uma comparação dos resultados desenvolvidos em um *dataset* de 250.000 aplicativos desenvolvidos com Scratch.

De acordo com (McDONALD, 2018), as dificuldades encontradas no ensino de programação são independentes da linguagem de programação utilizada ou da Universidade em si. O trabalho ressalta ainda que nos cursos da área de computação, exige-se um alto conhecimento em matemática, mas nenhum conhecimento prévio em lógica de programação. Esse indicador pode apontar a indicador que potencializa a evasão.

Em (GARCÍA-PEÑALVO & JOSÉ, 2019), afirma-se que ensinar programação é difícil. Desde que a primeira linguagem de programação criada, estuda-se a melhor forma de ensinar programação. E essa característica pode ter resultado em tantas linguagens de programação inventadas, em uma tentativa de suprir deficiências das já existentes.

Por fim, o trabalho de (HONG, YAO, MICHAEL, & PHILLIPS, 2018) apresenta uma discussão interessante ao fazer a analogia do estudo de programação como a de um idioma. A análise sugere que mostrar mais de uma linguagem de programação pode ser benéfico aos alunos.

A área de programação, de fato, apresenta um novo paradigma ao aluno, ao professor ou ainda ao aposentado que nunca escreveu um código computacional. É necessário organizar o pensamento para que a máquina possa seguir a sequência lógica de comandos e realizar os cálculos e operações necessárias para a obtenção do resultado desejado.

5. Considerações finais

Este trabalho apresentou uma revisão sistemática na temática do ensino de programação. Dentre os principais resultados do levantamento, podemos citar a grande preocupação com a temática do aprendizado dos alunos por parte dos educadores.

Além disso, foram identificadas as linguagens Python (bastante utilizada atualmente em diversas áreas na computação) e a linguagem C (linguagem de programação clássica), que continua bastante ativa na abordagem de ensino. Foram identificados também trabalhos com o uso de linguagens novas como Scratch e Snap, que visam facilitar o entendimento dos alunos.

Por outro lado, dentre as tecnologias identificadas, a robótica foi a tecnologia emergente mais utilizada nos trabalhos, provavelmente devido ao baixo custo de placas como Arduino e outros sensores que podem ser usados para exemplificação e abordagens no ensino de programação.

Os resultados desta revisão ressaltam ainda mais a importância da inclusão do ensino de programação no currículo do Brasil não apenas nos cursos da área de informática, mas integra-la com outras áreas. Vale ressaltar que dentro dos Institutos Federais, existem uma série de possibilidades ao aluno – mesmo de cursos diferentes de informática ou computação – e que aproximam esses alunos dessa matéria específica, tais como envolvimento com projetos de Jogos eletrônicos, Maratonas de Programação, Eventos Científicos, dentre outras.

Dentro do ensino profissional, essa discussão ganha ainda mais relevância e a adaptação curricular, apesar do bom desempenho de 2015, se realizada de forma incorreta poderia provocar um retrocesso aos resultados obtidos na rede federal. Ao invés disso, é um modelo que deveria ser espelhado aos contextos estaduais e municipais.

Pretende-se ainda, a título de trabalho futuro, realizar uma abordagem de ensino aplicando os resultados encontrados, como as linguagens de programação ou robótica, para que seja possível avaliar os resultados dessas abordagens identificadas e a possível melhoria em turmas de introdução a programação. E também realizar parceria para uma abordagem interdisciplinar, vinculando a lógica de programação com outras áreas do conhecimento.

REFERÊNCIAS

AIVALOGLU, E., & HERMANS, F. (2019). How is programming taught in code clubs? Exploring the experiences and gender perceptions of code club teachers. In: **Proceedings of the 19th Koli Calling International Conference on Computing Education Research (Koli Calling '19)**, pp. 1-10.

ANYANGO, J., & SULEMAN, H. (2018). Teaching Programming in Kenya and South Africa: What is difficult and is it universal? In: **Proceedings of Koli Calling International Conference on Computing Education Research**, pp. 22-25.

ARAÚJO, L. G. (2018). (D. d. mestrado, Ed.) Feira de Santana.

ARAÚJO, L. G., BITTENCOURT, R. A., & SANTOS, D. M. (2018). An Analysis of a Media-Based Approach to Teach Programming to Middle School Students. In: **Proceedings of the 49th ACM Technical Symposium on Computer Science Education (SIGCSE '18)**, pp. 1005-1010.

ARIZA, J. (2018). Towards education alternatives to teaching and learning of programming: A course experience using open hardware tools. In: **2018 IEEE Frontiers in Education Conference (FIE)**, pp. 1-8.

BERGER FILHO, R. L. (1999). Educação profissional no Brasil: novos rumos. In: **Revista Iberoamericana de Educación**, 20.

BLAS, M. J., GOLOBISKY, M. F., & GARCÍA LOZANO, D. A. (2018). Accessibility at the University: An Experience on How Teaching Programming to Blind Students. In: **XIII Latin American Conference on Learning Technologies (LACLO)**, pp. 165-172.

BOTTENTUIT JUNIOR, J. B. (2019). Sala de Aula Invertida: Recomendações e Tecnologias Digitais para sua Implementação na Educação. In: **Novas Tecnologias na Educação**, V. 17 N° 2, pp. 11-21.

BRANDÃO, L. d., FELIX, I. M., PEREIRA, P. A., & BRANDÃO, A. A. (2018). Evolving technology to better support teaching introductory programming inside Moodle. In: **2018 XIII Latin American Conference on Learning Technologies (LACLO)**, pp. 436-443.

BRERETON, P., KITCHENHAM, B. A., BUDGEN, D., TURNER, M., & KHALIL, M. (2007). Lessons from applying the systematic literature review process within the software engineering domain. In: **Journal of Systems and Software**, 80 (4), pp. 571-583.

BUTHELEZI, M. (2018). Improving Student Understanding of Programming Concepts: The Case of a Scenario Based Approach to Teaching Programming Online. In: **2018 International Conference on Computational Science and Computational Intelligence (CSCI)**.

COUTINHO, C. P. (2007). TECNOLOGIA EDUCATIVA E CURRÍCULO: CAMINHOS QUE SE CRUZAM OU SE BIFURCAM? In: **VII Colóquio sobre Questões Curriculares**, 8(15-16), p. 16.

DONG, Y., CATETÉ, V., LYTLE, N., ISVIK, A., BARNES, T., JOCIUS, R., . . . ANDREWS, A. (2019). Infusing Computing: Analyzing Teacher Programming Products in K-12 Computational Thinking Professional Development. In: **Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '19)**, pp. 278–284.

dos SANTOS, S. C., SANTANA, É., SANTANA, L., ROSSI, P., CARDOSO, L., FERNANDES, U., . . . TÔRRES, P. (2018). Applying PBL in Teaching Programming: an Experience Report. In: **2018 IEEE Frontiers in Education Conference (FIE)**, pp. 1-8.

FÖRSTER, E., FÖRSTER, K., & LÖWE, T. (2018). Teaching programming skills in primary school mathematics classes: An evaluation using game programming. In: **2018 IEEE Global Engineering Education Conference (EDUCON)**, pp. 1504-1513.

GARCÍA-PEÑALVO, J. F., & JOSÉ, F. (2019). Teaching and learning strategies of programming for university courses. In: **Proceedings of the Seventh International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'19)**, pp. 1020–1027.

GARLET, D., BIGOLIN, N. M., & SILVEIRA, S. R. (2018). Ensino de Programação de Computadores na Educação Básica: um estudo de caso. In: **Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica**, 9(2), pp. 135-160.

GELDREICH, K., & TALBOT, M. H. (2018). Off to new shores: preparing primary school teachers for teaching algorithmics and programming. In: **Proceedings of the 13th Workshop in Primary and Secondary Computing Education (WiPSCE '18)**, pp. 1-6.

HONG, G., YAO, J., MICHAEL, C., & PHILLIPS, L. (2018). A multilingual and comparative approach to teaching introductory computer programming. In: **Journal of Computing Sciences in Colleges**, pp. 4-12.

HUSSAIN, A. J., FERGUS, P., AL-JUMEILY, D., PICH, A., & HIND, J. (2015). Teaching Primary School Children the Concept of Computer Programming. In: **2015 International Conference on Developments of E-Systems Engineering (DeSE)**, pp. 180-184.

INEP. (2016). *Brasil no PISA 2015: análises e reflexões sobre o desempenho dos estudantes brasileiros / OCDE-Organização para a Cooperação e Desenvolvimento Econômico*. São Paulo: Fundação Santillana. Fonte: Disponível em: Acesso em 15 jan. 2018.

KAILA, E., LAAKSO, M., & KURVINEN, E. (2018). Teaching future teachers to code — Programming and computational thinking for teacher students. In: **41st International Convention on Information and Communication Technology, Electronics and Microelectronics**, pp. 677-682.

KENSKI, V. M. (2003). **TECNOLOGIAS E ENSINO PRESENCIAL E A DISTÂNCIA**. PAPIRUS.

KHANGURA, S. K. (2012). Evidence summaries: the evolution of a rapid review approach. In: **Systematic Reviews**, *1(10)*, pp. 1-10.

KO, A. J., LATOZA, T. D., HULL, S., KO, E. A., KWOK, W., QUICHOCHO, J., . . . PANDIT, R. (2019). Teaching Explicit Programming Strategies to Adolescents. In: **Proceedings of the 50th ACM Technical Symposium on Computer Science Education**, pp. 469–475.

KUENZER, A. Z., & GRABOWSKI, G. (2006). Educação Profissional: desafios para a construção de um projeto para os que vivem do trabalho. In: **PERSPECTIVA**, *24(1)*, pp. 297-318.

LOYO, A. H. (2018). Effects on the School Performance of Teaching Programming in Elementary and Secondary Schools. In: **Informatics in Schools. Fundamentals of Computer Science and Software Engineering. ISSEP 2018.**, *11169*(Lecture Notes in Computer Science), pp. 30-41.

MARCOLINO, A., SANTOS, A., SCHAEFER, M., & BARBOSA, E. (2018). Towards a Catalog of Gestures for M-learning Applications for the Teaching of Programming. In: **2018 IEEE Frontiers in Education Conference (FIE)**, pp. 1-9.

MARTÍNEZ-VALDÉS, J., GARCÍA-PEÑALVO, F. J., & VELÁZQUEZ-ITURBIDE, J. Á. (2019). The role of basic mathematics concepts in programming teaching and learning. In:

Proceedings of the Seventh International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'19), pp. 1046–1054.

MATSUMOTO, M. S., ALBUQUERQUE, S. S., RITA, L. P., & PINTO, I. S. (2019). Academic performance and investments in federal institutes of education in the Brazilian Northeast. In: **REBRAE**, 12(2), pp. 154-170.

McDONALD, C. (2018). Why Is Teaching Programming Difficult? In: **Higher Education Computer Science**, pp. 75-93.

MOURA, D. H. (2016). *Educação Profissional: Desafios Teórico-Metodológicos e Políticas Públicas*. Natal, RN, Brasil: IFRN.

NICULESCU, V., SERBAN, C., & VESCAN, A. (2019). Does Cyclic Learning have Positive Impact on Teaching Object-Oriented Programming? In: **2019 IEEE Frontiers in Education Conference (FIE)**, pp. 1-9.

OHASHI, Y., YAMACHI, H., MUROKOSHI, Y., KUMENO, F., & TSUJIMURA, Y. (2020). Development of a Programming Course for Senior Citizens Taught by Senior Citizens. In: **Proceedings of the 2020 8th International Conference on Information and Education Technology (ICIET 2020)**, pp. 18–23.

SANTACRUZ-VALENCIA, B., & RIAL-FERNÁNDEZ, L. P. (2019). The Teaching of Programming is not the Future but the Present. In: **2019 International Symposium on Computers in Education (SIIE)**, pp. 1-6.

SANTOS, F. E., PEREIRA, D. S., GODIN, J. M., LIMA, J. V., ZARO, M. A., & CANTO FILHO, A. B. (2018). A Robótica Educativa no Ensino de Lógica de Programação: uma revisão sistemática da literatura. In: **Novas Tecnologias na Educação**, 16(1), pp. 1-10.

SCHERER, R., SIDDIQ, F., & VIVEROS, B. S. (2020). A meta-analysis of teaching and learning computer programming: Effective instructional approaches and conditions. In: **Computers in Human Behavior**, pp. 1-18.

SHEN, R., DONGHEE YVETTE, W., & LEE, M. J. (2019). Comparison of Learning Programming Between Interactive Computer Tutors and Human Teachers. In: **Proceedings of the ACM Conference on Global Computing Education (CompEd '19)**, pp. 2-8.

SOOSAI RAJ, A. G., PATEL, J., & HALVERSON, R. (2018). Is More Active Always Better for Teaching Introductory Programming? In: **2018 International Conference on Learning and Teaching in Computing and Engineering (LaTICE)**, pp. 103-109.

SUNDIN, L., & CUTTS, Q. (2019). Is it feasible to teach query programming in three different languages in a single session? A study on a pattern-oriented tutorial and cheat sheets. In: **Proceedings of the 1st UK & Ireland Computing Education Research**, pp. 1-7.