

# Ensino de programação para alunos nos anos escolares entre Ensino Fundamental II e Ensino médio: Um Mapeamento Sistemático

Erlon P. Bié, Eduardo Souto, David Braga, Elaine Oliveira, Leandro Carvalho

<sup>1</sup>Instituto de Computação (ICOMP) – Universidade Federal do Amazonas (UFAM)  
69080-900 – Manaus – AM – Brazil

{erlon.bie, esouto, david, elaine, leandro}@icompu.ufam.edu.br

**Resumo.** *O ensino de computação na educação básica tem sido amplamente discutido devido à rápida evolução tecnológica. Este estudo apresenta um mapeamento sistemático que investiga práticas, ferramentas e métodos para aprimorar o ensino de programação em escolas ao redor do mundo. Os resultados do mapeamento sistemático revelam diversas abordagens adotadas em relação aos métodos e ferramentas utilizados, bem como uma convergência de medidas recomendadas para o avanço do ensino de programação. Esse mapeamento oferece insights valiosos para educadores interessados em promover o ensino de programação de forma eficaz, e a necessidade de se pensar na importância de se incluir esse tema nos currículos.*

**Abstract.** *Teaching computing in basic education has been widely discussed due to rapid technological evolution. This study presents a systematic mapping that investigates practices, tools, and methods to improve programming teaching in schools around the world. The results of the systematic mapping reveal different approaches adopted concerning the methods and tools used, as well as a convergence of recommended measures for the advancement of programming teaching. This mapping offers valuable insights for educators interested in promoting the teaching of programming effectively, and the need to think about the importance of including this topic in the curricula.*

## 1. Introdução

Nos últimos anos tem havido um amplo debate sobre a inclusão do ensino de computação na educação básica devido à crescente presença da tecnologia no cotidiano das pessoas. Nesse contexto, o Pensamento Computacional (PC) surge como uma habilidade fundamental, essencial não apenas para cientistas da computação, mas também como parte integrante das habilidades analíticas que todas as crianças deveriam ter, ao lado da leitura, escrita e aritmética [Wing 2006]. Em resposta a essa discussão, diversos países, incluindo Finlândia, Grécia e Escócia, têm avançado na inclusão de tópicos básicos de computação no currículo do ensino fundamental e médio, com o objetivo de melhorar a competitividade econômica [SBC 2017]. Essa abordagem busca oferecer aos estudantes acesso precoce e abrangente ao conhecimento computacional, permitindo-lhes desenvolver habilidades essenciais desde cedo.

De acordo com a Sociedade Brasileira da Computação (SBC) [SBC 2017], o pensamento computacional é uma das três habilidades propostas para o ensino de computação

nas escolas, juntamente com Mundo Digital (MD) e a Cultura Digital (CD). Essas três habilidades são distribuídas ao longo dos anos escolares, com o ensino de programação introduzido geralmente nos anos finais do ensino fundamental ou no início do ensino médio. Isso ocorre quando os alunos estão na faixa etária entre 11 e 17 anos.

A programação, ou codificação, é uma atividade complexa que envolve conceitos que trazem benefícios significativos, como o desenvolvimento do pensamento computacional, que auxilia na reflexão sobre problemas e no aprendizado de conceitos como abstração, reconhecimento de padrões, composição e implementação prática de algoritmos. Quanto mais cedo os alunos são expostos à programação, mais facilmente assimilam esses conceitos, assim como acontece no aprendizado de uma língua estrangeira em relação à língua materna. Além disso, a programação também ajuda a desenvolver habilidades de comunicação e colaboração, conhecidas como “soft skills” [Ponsard 2019].

No entanto, apesar dos benefícios associados ao ensino de programação, é comum que um terço dos estudantes do ensino superior enfrentem dificuldades nas disciplinas de programação e acabem desistindo do aprendizado ao longo de suas trajetórias acadêmicas, sendo a falta de uma base sólida um dos fatores que contribuem para esse cenário [Watson and Li 2014]. Outras razões estão relacionadas à falta de motivação dos alunos e à dificuldade de criar um modelo mental que relacione programas com sistemas subjacentes, ou seja, compreender como os programas se relacionam entre si [Ben-Ari 1998].

Nesse contexto, diversos estudos vêm sendo realizados sobre o ensino de programação. Esta pesquisa tem como objetivo realizar um mapeamento sistemático (MS) para identificar avanços e iniciativas relacionados ao ensino de programação no contexto escolar, especificamente nos anos finais do ensino fundamental e no início do ensino médio. O propósito deste levantamento é compreender as abordagens adotadas por pesquisadores em diferentes partes do mundo. Os resultados deste estudo podem fornecer subsídios valiosos para o planejamento e desenvolvimento do ensino de computação na educação básica, tanto no Brasil quanto em outros contextos onde o ensino de computação seja relevante.

## **2. Método de Pesquisa**

Um Mapeamento Sistemático da Literatura (MSL) [Kitchenham and Charters 2007] foi realizado para identificar o panorama sobre os métodos, tecnologias e iniciativas adotadas para o ensino de programação de computadores no contexto escolar ao redor do mundo. Um protocolo foi elaborado e alguns dos seus elementos são descritos a seguir.

### **2.1. Questões de Pesquisa**

O primeiro estágio do trabalho consistiu na definição das questões de pesquisa, que derivam da pergunta central: *Qual é o estado atual do ensino e aprendizado de programação nas escolas para estudantes em anos escolares correspondentes ao ensino fundamental e médio?*

Para responder essa pergunta central, foram estabelecidas as seguintes questões específicas:

- **QP1:** Quais métodos, tecnologias ou abordagens têm sido utilizados para aprimorar o ensino de programação nas escolas?

- **QP2:** Por que é importante incorporar o ensino de programação no currículo escolar?
- **QP3:** Quais iniciativas têm sido desenvolvidas para fortalecer os programas de ensino de programação de computadores nas escolas de ensino médio?
- **QP4:** Como contribuir para o avanço e desenvolvimento do ensino de programação nas escolas?

## 2.2. Estratégia de Pesquisa

Considerando palavras no plural e singular, bem como sinônimos relacionados ao tema de interesse, a string de busca resultante, com pequenas variações dependendo da especificidade de cada fonte de busca, foi a seguinte:

(“teaching computer programming” OR “learning computer programming” OR “teaching computers programming” OR “learning computers programming” OR “programming instruction” OR “programming instructions” OR “coding education” OR “coding educations”) AND (“secondary school” OR “secondary schools” OR “secondary education” OR “secondary educations” OR “secondary student” OR “secondary students”)

Os estudos foram investigados em três bibliotecas digitais, sendo elas: *i) ACM Digital Library*, *ii) IEE Xplore* e *iii) Scopus*. As bibliotecas selecionadas foram escolhidas devido aos seus excelentes mecanismos de busca, à variedade de publicações qualificadas de diferentes áreas que elas abrangem, além de serem referências nas áreas de Computação e Informática [Dias and Seno Diaz 2021].

## 2.3. Seleção dos Estudos

Durante a condução do Mapeamento Sistemático da Literatura (MSL), foram incluídos trabalhos científicos completos que abordassem sobre o ensino ou aprendizado de programação para estudantes nos anos escolares investigados neste artigo e que respondessem a uma ou mais questões de pesquisa. O estudo buscou incluir artigos dos últimos 10 anos, levando em consideração a rápida evolução da área e concentrando-se principalmente em estudos recentes. Os estudos que atendiam a pelo menos um dos critérios de exclusão listados a seguir foram excluídos da análise: **CE1)** Artigos não acessíveis na íntegra; **CE2)** Tutoriais, discursos de *keynote*, relatórios de workshop, relatórios técnicos; **CE3)** Estudos secundários e terciários; **CE4)** Artigos que não estavam escritos em inglês ou português; **CE5)** Estudos duplicados: apenas o mais recente foi incluído; **CE6)** Estudos que não se referiam claramente ao processo de ensino ou aprendizagem de programação para estudantes nos anos escolares; **CE7)** Estudos que tratam do tema apenas como trabalhos futuros.

Na primeira etapa do processo de seleção (1º Filtro), foi realizada a leitura do título, palavras-chave e resumo de cada publicação. Nesse momento, foram selecionadas as publicações que atendiam pelo menos a um dos critérios de inclusão. Em caso de dúvida, a publicação era incluída para uma análise posterior. Na segunda etapa (2º Filtro), foi feita a leitura completa das publicações selecionadas no 1º Filtro, utilizando os critérios de inclusão e exclusão para decidir se a publicação seria selecionada ou não.

### 3. Resultados

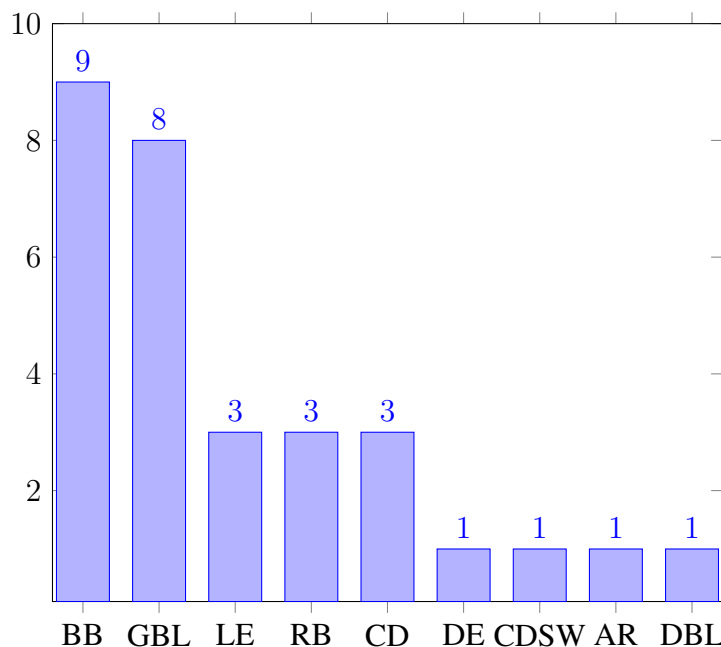
Após a leitura dos artigos e a aplicação dos filtros, um total de 37 artigos foram selecionados, sendo capazes de responder a pelo menos uma das questões de pesquisa. A Tabela 1 apresenta informações sobre cada biblioteca digital, incluindo o número total de documentos retornados pela string de busca e a quantidade de documentos que permaneceram após cada filtro.

**Tabela 1. Quantitativo das publicações retornadas.**

Bibliotecas Digitais	Documentos	1ºFiltro	2ºFiltro
IEEE Xlore	13	8	4
ACM Digital Library	86	35	16
Scopus	186	42	17
<b>Total</b>	<b>285</b>	<b>85</b>	<b>37</b>

#### 3.1. QP1: Quais métodos, tecnologias ou abordagens têm sido utilizados para aprimorar o ensino de programação nas escolas?

Com relação à *QP1*, a Figura 1 ilustra quais metodologias, tecnologias ou abordagens para o ensino de programação foram utilizadas pelos autores dos estudos selecionados para o ensino de programação em escolas. É importante ressaltar que alguns estudos empregaram mais de uma abordagem, e que portanto o total das utilizações não reflete o número absoluto de artigos selecionados nesta pesquisa.



**Figura 1. Abordagens utilizadas no ensino de programação.**

No total, foram identificadas nove tipos de abordagens nos estudos selecionados, a saber: *Block-Based* (BB), *Game-Based Learning* (GBL), Linguagens Estruturadas (LE), Robótica (RB), Computação Desplugada (CD), Dispositivos Embarcados (DE), Ciclo de

Desenvolvimento de Software (CDSW), Realidade Aumentada (AR) e *Design-Based Learning* (DBL). A seguir, destacam-se alguns dos estudos que exploram as principais abordagens utilizadas.

Com relação aos estudos que utilizam o ensino de programação em blocos (BB) e baseado em jogo (GBL), destaca-se o estudo de [Holenko and Hoic-Bozic 2021], que investigou o impacto do ensino de programação baseado em jogos educacionais em alunos do ensino fundamental e médio. O estudo discute diversas metodologias e ferramentas para aprimorar o aprendizado de programação, incluindo o uso de *game-based learning* (GBL), plataformas visuais e linguagens de programação especializadas como Scratch, Snap! e Alice, que permitem que os alunos criem programas manipulando elementos gráficos em vez de escrever linhas de código. A proposta de utilização do GBL foi de utilizar cenários com problemas reais e a criação de jogos para solucioná-los. Os resultados demonstraram um aumento significativo nas habilidades de programação dos alunos, e o estudo foi bem recebido tanto pelos alunos quanto pelos professores.

Outro estudo relevante, proposto por [Çalışkan 2020], investigou os efeitos da programação de robôs (RB) utilizando linguagem BB nas habilidades de resolução de problemas de estudantes do ensino médio. Os resultados revelaram que a programação de robôs auxiliou os alunos no desenvolvimento de habilidades de resolução de problemas, com um aumento significativo dessas habilidades após o experimento. Os resultados indicaram que tanto ambientes de programação baseados em blocos quanto atividades de robótica contribuem de forma positiva para o desenvolvimento de habilidades de resolução de problemas em estudantes do ensino médio.

No contexto do ensino de programação usando linguagens estruturadas (LE), [Bati 2022] descreve um programa de treinamento destinado a professores em formação na área de ciências, com o objetivo de incorporar o ensino de programação em suas aulas. O programa incluiu conceitos básicos de lógica de programação, algoritmos, uso do Google Colab<sup>1</sup>, variáveis, loops e funções em Python. O artigo destaca a importância de fornecer treinamentos adequados aos professores de ciências para ajudá-los a incorporar o ensino de programação em suas aulas, além da necessidade de desenvolver recursos educacionais e ferramentas de software para apoiar o ensino de programação nas escolas. Os resultados do programa demonstraram uma melhoria nas habilidades de programação e pensamento computacional dos professores em formação após a conclusão do treinamento.

### **3.2. QP2: Por que é importante incorporar o ensino de programação no currículo escolar?**

Os estudos selecionados apontam diversos motivos pelos quais o ensino de programação é importante no currículo escolar dos alunos do ensino médio. Esses motivos podem ser agrupados em categorias que refletem a importância da programação dentro do contexto altamente digitalizado da atualidade e suas consequências para os alunos:

1. Desenvolvimento de habilidades cognitivas: Muitos estudos (57.1% deles) relatam que a programação envolve a resolução de problemas complexos, o pensamento lógico e a criatividade. Ao aprender a programar, os alunos desenvolvem habilidades cognitivas fundamentais, como o raciocínio abstrato, a capacidade de decompor problemas

---

<sup>1</sup><https://colab.google/>

em etapas menores e a habilidade de encontrar soluções eficientes [Jeon and Song 2019] [Martínez López et al. 2017] [Merkouris et al. 2017] [Torres-Torres et al. 2019] [Portnoff 2018] [de la Hera et al. 2022] [Cuervo-Cely et al. 2022] [Benotti et al. 2014].

2. Preparação para o futuro profissional: Outros estudos (50% deles) apontam que a programação é uma habilidade cada vez mais requisitada em diversas áreas profissionais. Ao aprender programação desde o ensino médio, os alunos estarão se preparando para carreiras relacionadas à tecnologia, que oferecem amplas oportunidades de emprego e crescimento profissional [Martínez López et al. 2017] [Merkouris et al. 2017] [Jeon and Song 2019] [Benotti et al. 2014] [Goswami and Pal 2022] [Lee et al. 2020] [Torres-Torres et al. 2019].

3. Estímulo ao pensamento crítico e analítico: Alguns estudos afirmam que a programação requer análise cuidadosa, planejamento estratégico e tomada de decisões. Os alunos aprendem a pensar criticamente, a avaliar diferentes abordagens para um problema e a escolher a melhor solução possível [Malliarakis et al. 2013] [Wang et al. 2014].

4. Compreensão do mundo digital: Vivemos em uma sociedade cada vez mais dependente da tecnologia. O ensino de programação permite que os alunos compreendam o funcionamento dos dispositivos e sistemas digitais que fazem parte do seu cotidiano, capacitando-os a tomar decisões informadas e a se tornarem usuários responsáveis e críticos da tecnologia [de la Hera et al. 2022] [Torres-Torres et al. 2019] [Haden et al. 2016].

Esses são apenas alguns dos motivos pelos quais o ensino de programação é importante no currículo escolar dos alunos do ensino médio. A programação não apenas abre portas para oportunidades de carreira, mas também promove o desenvolvimento de habilidades essenciais para enfrentar os desafios do mundo atualmente digitalizado.

### **3.3. QP3: Quais iniciativas têm sido desenvolvidas para fortalecer os programas de ensino de programação de computadores nas escolas de ensino médio?**

Diversos países têm tomado iniciativas para fortalecer os programas de ensino de programação nas escolas de ensino médio. Um exemplo notável é o programa "Exploring Computer Science", que foi implementado em várias escolas do estado de Utah, nos Estados Unidos, inclusive em regiões rurais. Esse programa, originado em Los Angeles, foi posteriormente expandido para outros locais do país. Seu principal objetivo é capacitar professores sem experiência prévia na área de computação, por meio de workshops intensivos de 5 dias a até 6 meses. Os participantes adquirem competências fundamentais, tais como resolução de problemas, programação utilizando a plataforma Scratch, interação humano-computador, robótica e web design [Hu et al. 2016].

Outro país que tem se destacado nesse sentido é a Coreia do Sul, onde a educação em programação se tornou obrigatória nas escolas primárias e secundárias a partir de um esforço conjunto realizado em 2015, envolvendo o governo e a Universidade Gachon. Para apoiar essa iniciativa, a universidade implementou um programa de treinamento de professores e criou um fórum online de software, visando fornecer suporte contínuo aos educadores [Kim et al. 2017].

Na Croácia, a programação de computadores já está sendo ensinada nas escolas. O "Currículo Nacional para a Disciplina de Ciência da Computação/Informática", estabele-

cido pelo Ministério da Educação Croata, define objetivos de aprendizagem relacionados ao desenvolvimento de habilidades de programação, abrangendo diversas séries dentro dos cursos de informática nas escolas primárias e secundárias, conhecidos como ginásios [Holenko and Hoic-Bozic 2021].

Na Turquia, o ensino de programação foi gradualmente incluído no currículo do 5º e 6º ano entre 2017 e 2019, por meio de uma iniciativa do Ministério da Educação (MEB). Anteriormente, a educação em programação dependia da iniciativa dos professores, mas agora ela se tornou obrigatória no curso de "Tecnologias da Informação e Software", equivalente à disciplina de informática no currículo escolar brasileiro [Karaahmetoğlu and Korkmaz 2019].

Além dessas iniciativas, um grupo de trabalho internacional foi formado por especialistas de diversos países para desenvolver, testar, revisar e avaliar a validade e confiabilidade do instrumento METRECC (MEasuring TeacheR Enacted Computing Curriculum) na pesquisa com professores de escolas de ensino fundamental e médio sobre a implementação do currículo de computação. O objetivo é compreender a pedagogia, práticas, recursos e experiências em salas de aula ao redor do mundo [Falkner et al. 2019].

Esse grupo de trabalho é composto por profissionais dedicados da Austrália, Inglaterra, Irlanda, Itália, Malta, Escócia e Estados Unidos. Juntos, eles revisam e analisam os dados coletados por meio do METRECC, provenientes de 244 professores desses países. Essa colaboração internacional permite uma visão abrangente e diversificada da implementação dos currículos de computação nas escolas, enriquecendo as discussões e promovendo trocas de experiências entre diferentes contextos educacionais.

#### **3.4. QP4: Como contribuir para o avanço e desenvolvimento do ensino de programação nas escolas?**

O ensino de programação nas escolas desempenha um papel fundamental na preparação dos alunos para o mundo tecnológico em constante evolução. Para promover o avanço nessa área, os estudos selecionados neste trabalho afirmam que é necessário adotar uma abordagem abrangente que envolva pesquisas, adaptação curricular e incentivo à participação dos estudantes.

A realização de pesquisas é essencial para avaliar a eficácia de diferentes abordagens de ensino de programação e medir o impacto na aprendizagem dos alunos. Estudos como os de Makris et al. [Makris et al. 2013], Kolling et al. [Kölling et al. 2015], Fessard et al. [Fessard et al. 2019], Benotti [Benotti et al. 2014] e Xinogalos et al. [Xinogalos et al. 2015] têm fornecido *insights* valiosos sobre práticas pedagógicas eficazes no ensino de programação. Essas pesquisas contribuem para o amadurecimento dessa área, permitindo a identificação de abordagens que promovam o engajamento e o aprendizado dos alunos.

Além disso, de acordo com [Hu et al. 2016], a adaptação curricular desempenha um papel crucial no avanço do ensino de programação nas escolas. É necessário adaptar o currículo para atender às necessidades locais, considerando as demandas e realidades específicas de cada contexto educacional. A oferta de treinamento adequado para os professores também é fundamental. Os educadores precisam ser capacitados e atualizados para ministrar aulas de programação de forma eficaz, proporcionando aos alunos uma base sólida nesse campo.

Incentivar a participação ativa dos estudantes é outro aspecto importante para o avanço do ensino de programação. Competições e eventos relacionados à programação fornecem oportunidades para que os alunos aprimorem suas habilidades e interajam com outros entusiastas da área. Essas experiências extracurriculares não apenas motivam os estudantes, mas também consolidam seu interesse pela programação, estimulando um aprendizado mais significativo e duradouro.

É importante destacar que, para que esse avanço seja efetivo, é necessário acompanhar as tendências e os avanços na área de programação. Investir em formação contínua para os professores, fornecer recursos e materiais educacionais de qualidade e incentivar a diversidade na área de tecnologia são iniciativas que promovem o amadurecimento e o desenvolvimento do ensino de programação nas escolas [Hu et al. 2016], [Xambó et al. 2017].

### 3.5. Estudos Incluídos na Pesquisa

A Tabela 2 apresenta os estudos selecionados neste trabalho, juntamente com as questões de pesquisa que cada estudo aborda.

**Tabela 2. Autor(es) com relação a quais perguntas são respondidas**

Autor(es)	Q1	Q2	Q3	Q4
[AlHumoud et al. 2014]	X			X
[Wang et al. 2014]	X	X		
[Jeon and Song 2019]	X	X		
[Lee et al. 2020]	X	X		
[Busjahn and Schulte 2013]	X			
[Haden et al. 2016]		X		
[Xinogalos et al. 2015]	X			X
[Malliarakis et al. 2013]	X	X		
[Fessard et al. 2019]	X			X
[Benotti et al. 2014]	X	X		X
[Holenko and Hoic-Bozic 2021]	X		X	
[Namli and Aybek 2022]	X			
[Sigayret et al. 2022]	X			
[Huang et al. 2023]	X			
[Van Zadelhoff et al. 2021]	X			
[Bati 2022]	X			
[Wagner et al. 2013]	X			
[Sentance et al. 2019]	X			
[Wangenheim et al. 2019]	X			
[Attard and Busuttill 2020]	X			
[Çalışkan 2020]	X			
[Threekunprapa and Yasri 2020]	X			
[Martínez López et al. 2017]		X		
[Merkouris et al. 2017]		X		
[Torres-Torres et al. 2019]		X		X
[Portnoff 2018]		X		
[Kölling et al. 2015]		X		X
[Cuervo-Cely et al. 2022]		X		
[de la Hera et al. 2022]		X		
[Goswami and Pal 2022]		X		X
[Kim et al. 2017]			X	
[Falkner et al. 2019]			X	X
[Hu et al. 2016]			X	X
[Karaahmetoğlu and Korkmaz 2019]			X	X
[Park et al. 2015]				X
[Makris et al. 2013]				X
[Xambó et al. 2017]				X

Esses estudos selecionados abrangem uma variedade de questões de pesquisa, for-



necendo uma visão abrangente sobre os diversos aspectos do ensino de programação. Eles contribuem para o avanço do conhecimento nessa área, fornecendo informações valiosas que podem orientar práticas pedagógicas eficazes e melhorar a aprendizagem dos alunos.

#### **4. Resultados e Discussões**

Com base nos estudos analisados em relação à Questão de Pesquisa 1 (QP1), é evidente que existem diversos métodos e ferramentas para auxiliar no ensino de programação. Entre esses métodos, destacam-se o uso de jogos, robótica e programação baseada em blocos e linguagens estruturadas. Essas abordagens visam motivar os alunos, desenvolver habilidades de programação e criar um ambiente de aprendizado engajador. A ampla gama de métodos disponíveis proporciona aos educadores opções variadas para adaptar o ensino de programação às necessidades e preferências dos alunos.

Com relação à Questão de Pesquisa 2 (QP2), fica claro que incorporar o ensino de programação no currículo escolar é de suma importância. Essa incorporação traz benefícios significativos, preparando os alunos para futuras oportunidades de emprego, promovendo habilidades como pensamento lógico, resolução de problemas, criatividade e colaboração, e ajudando-os a compreender melhor a tecnologia ao seu redor. Ao introduzir a programação no currículo escolar, os alunos têm a oportunidade de desenvolver habilidades essenciais, como pensamento lógico, resolução de problemas e criatividade, ao mesmo tempo em que adquirem um melhor entendimento do mundo digital em que vivem.

No que diz respeito à Questão de Pesquisa 3 (QP3), observamos que várias iniciativas estão sendo implementadas globalmente para aprimorar o ensino de programação nas escolas. Países como Turquia, Taiwan e Coreia do Sul estão adotando abordagens obrigatórias e abrangentes, proporcionando aos alunos um aprendizado de programação desde as séries iniciais. Além disso, programas de formação de professores têm sido implementados em várias localidades, com o objetivo de capacitar os educadores para o ensino eficaz da programação. Essas iniciativas ressaltam a importância de um ensino aprimorado de programação nas escolas, visando desenvolver habilidades essenciais para que os alunos enfrentem os desafios do mundo digital em constante evolução.

Por fim, em relação à Questão de Pesquisa 4 (QP4), observamos que muitos estudos sugerem a necessidade de uma maior quantidade de pesquisas na área e de avaliações comparativas entre os métodos de ensino. Isso evidencia a importância de realizar mais estudos no campo do ensino de programação nas escolas a fim de responder de maneira assertiva a essa questão. Em geral, para o amadurecimento da área de ensino de programação nas escolas, é necessário adotar uma abordagem abrangente que envolva métodos de ensino eficazes, pesquisa contínua, formação de professores, incentivo à diversidade e oferta de oportunidades de carreira em tecnologia para os alunos.

Esses resultados destacam a importância de uma abordagem abrangente e contínua para o ensino de programação nas escolas, levando em consideração os diversos métodos disponíveis, a integração curricular, as iniciativas globais, a necessidade de pesquisas adicionais e a formação adequada dos educadores. Ao seguir essas diretrizes, podemos contribuir para o avanço e aprimoramento do ensino de programação, capacitando os alunos para um futuro digital cada vez mais exigente.

## 5. Considerações Finais

Este mapeamento sistemático teve como propósito fornecer uma visão abrangente das pesquisas e iniciativas em andamento no campo do ensino de programação no ensino fundamental e médio. O objetivo era oferecer aos educadores informações valiosas para incorporar o ensino de programação em suas práticas pedagógicas.

Ao explorar as questões de pesquisa abordadas neste estudo, podemos constatar que o ensino de programação é uma habilidade crucial nos dias atuais e sua inclusão no currículo escolar é essencial. Essa habilidade não apenas prepara os alunos para as demandas tecnológicas do mundo moderno, mas também desenvolve habilidades cognitivas, lógicas, criativas e de resolução de problemas.

Neste mapeamento, identificamos uma variedade de métodos e abordagens para o ensino de programação, como o uso de jogos, robótica e linguagens estruturadas ou baseadas em blocos. Essas abordagens engajadoras proporcionam aos alunos uma experiência motivadora, despertando seu interesse e estimulando o aprendizado ativo.

Além disso, observamos que vários países estão implementando iniciativas para aprimorar o ensino de programação, seja por meio de abordagens obrigatórias no currículo ou programas de formação de professores. Essas ações refletem a importância de fornecer aos alunos uma base sólida em programação desde as séries iniciais, preparando-os para um futuro digital.

No entanto, é necessário realizar mais pesquisas e avaliações comparativas para aprimorar ainda mais o ensino de programação. O aprofundamento desses estudos permitirá uma compreensão mais precisa das melhores práticas, da eficácia dos métodos de ensino e do impacto na aprendizagem dos alunos.

## 6. Agradecimentos

Esta pesquisa, realizada no âmbito do Projeto Samsung-UFAM de Ensino e Pesquisa (SUPER), de acordo com o Artigo 39 do Decreto nº10.521/2020, foi financiada pela Samsung Eletrônica da Amazônia Ltda, nos termos da Lei Federal nº8.387/1991, através do convênio 001/2020 firmado com a UFAM e FAEPI, Brasil.

## Referências

- AlHumoud, S., Al-Khalifa, H. S., Al-Razgan, M., and Alfaries, A. (2014). Using App Inventor and LEGO mindstorm NXT in a summer camp to attract high school girls to computing fields. In *2014 IEEE Global Engineering Education Conference (EDU-CON)*, pages 173–177.
- Attard, L. and Busuttil, L. (2020). Teacher Perspectives on Introducing Programming Constructs through Coding Mobile-Based Games to Secondary School Students. *Informatics in Education*, 19(4):543–568.
- Bati, K. (2022). Integration of Python into Science Teacher Education, Developing Computational Problem Solving and Using Information and Communication Technologies Competencies of Pre-service Science Teachers. *Informatics in Education*, 21(2):235–251.
- Ben-Ari, M. (1998). Constructivism in computer science education. In *Technical Symposium on Computer Science Education*.

- Benotti, L., Martínez, M. C., and Schapachnik, F. (2014). Engaging high school students using chatbots. In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education*, ITiCSE '14, pages 63–68, New York, NY, USA. Association for Computing Machinery.
- Busjahn, T. and Schulte, C. (2013). The use of code reading in teaching programming. In *Proceedings of the 13th Koli Calling International Conference on Computing Education Research*, Koli Calling '13, pages 3–11, New York, NY, USA. Association for Computing Machinery.
- Çalışkan, E. (2020). The effects of robotics programming on secondary school students' problem-solving skills. *World Journal on Educational Technology: Current Issues*, 12(4):217–230.
- Cuervo-Cely, K., Restrepo-Calle, F., and Ramírez-Echeverry, J. (2022). EFFECT OF GAMIFICATION ON THE MOTIVATION OF COMPUTER PROGRAMMING STUDENTS. *Journal of Information Technology Education: Research*, 21.
- de la Hera, D., Zanoni, M., Sigman, M., and Calero, C. (2022). Peer tutoring of computer programming increases exploratory behavior in children. *Journal of Experimental Child Psychology*, 216.
- Dias, J. and Seno Diaz, R. (2021). Ensino híbrido no processo de aprendizagem de programação: uma revisão sistemática. *Revista Novas Tecnologias na Educação*, 19(2):396–405.
- Falkner, K., Sentance, S., Vivian, R., Barksdale, S., Busuttil, L., Cole, E., Liebe, C., Maiorana, F., McGill, M. M., and Quille, K. (2019). An International Study Piloting the MEasuring TeacheR Enacted Computing Curriculum (METRECC) Instrument. In *Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education*, ITiCSE-WGR '19, pages 111–142, New York, NY, USA. Association for Computing Machinery.
- Fessard, G., Wang, P., and Renna, I. (2019). Are There Differences in Learning Gains When Programming a Tangible Object or a Simulation? In *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '19, pages 78–84, New York, NY, USA. Association for Computing Machinery.
- Goswami, B. and Pal, S. (2022). Introduction of two new programming tools in Bengali and measurement of their reception among high-school students in Purba Bardhaman, India with the prototypic inclusion of a vector-biology module. *Education and Information Technologies*, 27(2):1585–1607.
- Haden, P., Gasson, J., Wood, K., and Parsons, D. (2016). Can you learn to teach programming in two days? In *Proceedings of the Australasian Computer Science Week Multiconference*, ACSW '16, pages 1–7, New York, NY, USA. Association for Computing Machinery.
- Holenko, M. and Hoic-Bozic, N. (2021). Effectiveness of game development-based learning for acquiring programming skills in lower secondary education in Croatia. *Education and Information Technologies*, 26(4):4433–4456.

- Hu, H. H., Heiner, C., and McCarthy, J. (2016). Deploying Exploring Computer Science Statewide. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education, SIGCSE '16*, pages 72–77, New York, NY, USA. Association for Computing Machinery.
- Huang, S.-Y., Tarn, W., and Ou, K.-L. (2023). Effectiveness of AR Board Game on Computational Thinking and Programming Skills for Elementary School Students. *Systems*, 11(1).
- Jeon, I. and Song, K.-S. (2019). The Effect of Learning Analytics System towards Learner's Computational Thinking Capabilities. In *Proceedings of the 2019 11th International Conference on Computer and Automation Engineering, ICCAE 2019*, pages 12–16, New York, NY, USA. Association for Computing Machinery.
- Karaahmetoğlu, K. and Korkmaz, Ö. (2019). The effect of project-based arduino educational robot applications on students' computational thinking skills and their perception of basic stem skill levels. *Participatory Educational Research*, 6(2):1–14.
- Kim, W., Choi, A., Choi, J., Jeong, O.-R., Jung, Y. J., Kang, S., Lee, J., Lee, S.-W., Loh, W.-K., Yoo, J., Chung, S., Lee, H., Chon, S., Jeon, Y., Kim, H. S., Kim, J. W., Lee, J.-H., Min, Y., Song, G. S., and Yang, S. O. (2017). Korea's software education initiative. In *Proceedings of the 19th International Conference on Information Integration and Web-based Applications & Services, iiWAS '17*, pages 553–557, New York, NY, USA. Association for Computing Machinery.
- Kitchenham, B. A. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report.
- Kölling, M., Brown, N. C. C., and Altadmri, A. (2015). Frame-Based Editing: Easing the Transition from Blocks to Text-Based Programming. In *Proceedings of the Workshop in Primary and Secondary Computing Education, WiPSCE '15*, pages 29–38, New York, NY, USA. Association for Computing Machinery.
- Lee, B. Y., Liew, L. H., Bin Mohd Anas Khan, M. Y., and Narawi, A. (2020). The Effectiveness of Using mBot to Increase the Interest and Basic Knowledge in Programming and Robotic among Children of Age 13. In *Proceedings of the 2020 The 6th International Conference on E-Business and Applications, ICEBA 2020*, pages 105–110, New York, NY, USA. Association for Computing Machinery.
- Makris, D., Euaggelopoulos, K., Chorianopoulos, K., and Giannakos, M. N. (2013). Could you help me to change the variables? comparing instruction to encouragement for teaching programming. In *Proceedings of the 8th Workshop in Primary and Secondary Computing Education, WiPSE '13*, pages 79–82, New York, NY, USA. Association for Computing Machinery.
- Malliarakis, C., Satratzemi, M., and Xinogalos, S. (2013). Towards a new massive multi-player online role playing game for introductory programming. In *Proceedings of the 6th Balkan Conference in Informatics, BCI '13*, pages 156–163, New York, NY, USA. Association for Computing Machinery.
- Martínez López, P. E., Ciolek, D., Arévalo, G., and Pari, D. (2017). The GOBSTONES method for teaching computer programming. In *2017 XLIII Latin American Computer*

*Conference (CLEI)*, pages 1–9.

- Merkouris, A., Chorianopoulos, K., and Kameas, A. (2017). Teaching Programming in Secondary Education Through Embodied Computing Platforms: Robotics and Wearables. *ACM Transactions on Computing Education*, 17(2):9:1–9:22.
- Namli, N. and Aybek, B. (2022). An Investigation of The Effect of Block-Based Programming and Unplugged Coding Activities on Fifth Graders' Computational Thinking Skills, Self-Efficacy and Academic Performance. *Contemporary Educational Technology*, 14(1).
- Park, C. J., Hyun, J. S., and Heuilan, J. (2015). Effects of gender and abstract thinking factors on adolescents' computer program learning. In *2015 IEEE Frontiers in Education Conference (FIE)*, pages 1–7.
- Ponsard, C. (2019). Teaching computer programming to post-millennial kids: Overview of goals, activities and supporting tools. In Lane H., Zvacek S., and Uhomuibhi J., editors, *CSEDU - Proc. Int. Conf. Comput. Support. Educ.*, volume 2, pages 474–480. SciTePress.
- Portnoff, S. R. (2018). The introductory computer programming course is first and foremost a language course. *ACM Inroads*, 9(2):34–52.
- SBC (2017). Referenciais de formação em computação: Educação básica. Technical report.
- Sentance, S., Waite, J., and Kallia, M. (2019). Teaching computer programming with PRIMM: A sociocultural perspective. *Computer Science Education*, 29(2-3):136–176.
- Sigayret, K., Tricot, A., and Blanc, N. (2022). Unplugged or plugged-in programming learning: A comparative experimental study. *Computers and Education*, 184.
- Threekunprapa, A. and Yasri, P. (2020). Unplugged coding using flowblocks for promoting computational thinking and programming among secondary school students. *International Journal of Instruction*, 13(3):207–222.
- Torres-Torres, Y.-D., Román-González, M., and Pérez-González, J.-C. (2019). Implementation of unplugged teaching activities to foster computational thinking skills in primary school from a gender perspective. In *Proceedings of the Seventh International Conference on Technological Ecosystems for Enhancing Multiculturality*, TEEM'19, page 209–215, New York, NY, USA. Association for Computing Machinery.
- Van Zadelhoff, B., Rahimi, E., and Barendsen, E. (2021). Principles to facilitate design-based learning environments for programming in secondary education while making learning visible in an authentic way. In *ACM Int. Conf. Proc. Ser.* Association for Computing Machinery.
- Wagner, A., Gray, J., Corley, J., and Wolber, D. (2013). Using app inventor in a K-12 summer camp. In *SIGCSE - Proc. ACM Tech. Symp. Comput. Sci. Educ.*, pages 621–626. Association for Computing Machinery.
- Wang, H. Y., Huang, I., and Hwang, G. J. (2014). Effects of an Integrated Scratch and Project-Based Learning Approach on the Learning Achievements of Gifted Students in Computer Courses. In *2014 IIAI 3rd International Conference on Advanced Applied Informatics*, pages 382–387.

- Wangenheim, G., Medeiros, G., Filho, R., Petri, G., Pinheiro, F., Ferreira, M., and Hauck, J. (2019). Splash code - a board game for learning an understanding of algorithms in middle school. *Informatics in Education*, 18(2):259–280.
- Watson, C. and Li, F. W. B. (2014). Failure rates in introductory programming revisited. In ° Asa Cajander, Daniels, M., Clear, T., and Pears, A., editors, *Proceedings of the 2014 conference on Innovation & technology in computer science education (ITiCSE '14)*., pages 39–44. Association for Computing Machinery (ACM), New York. Best Full Paper Award (Top downloaded paper on ACM SIGCSE over six weeks), updated on 01/10/2014.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3):33–35.
- Xambó, A., Drozda, B., Weisling, A., Magerko, B., Huet, M., Gasque, T., and Freeman, J. (2017). Experience and ownership with a tangible computational music installation for informal learning. *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction*.
- Xinogalos, S., Malliarakis, C., Tsompanoudi, D., and Satratzemi, M. (2015). Microworlds, Games and Collaboration: Three effective approaches to support novices in learning programming. In *Proceedings of the 7th Balkan Conference on Informatics Conference, BCI '15*, pages 1–8, New York, NY, USA. Association for Computing Machinery.