



UEPB

**UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS I - CAMPINA GRANDE
CENTRO DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO
CURSO DE GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO**

JOSÉ KELVYN DOS SANTOS SILVA

**ANÁLISE DA PROGRAMAÇÃO COMPUTACIONAL COMO UMA FERRAMENTA
DE ENSINO NA EDUCAÇÃO INFANTIL**

**CAMPINA GRANDE - PB
2023**

JOSÉ KELVYN DOS SANTOS SILVA

**ANÁLISE DA PROGRAMAÇÃO COMPUTACIONAL COMO UMA FERRAMENTA
DE ENSINO NA EDUCAÇÃO INFANTIL**

Trabalho de Conclusão de Curso apresentado ao Programa de Graduação em Bacharelado em Ciência da Computação da Universidade Estadual da Paraíba, como requisito parcial à obtenção do título de Bacharel em Ciências da Computação.

Área de concentração: Informática na Educação.

Orientador: Prof. Dr. Wellington Candeia de Araújo.

**CAMPINA GRANDE - PB
2023**

JOSÉ KELVYN DOS SANTOS SILVA

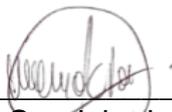
ANÁLISE DA PROGRAMAÇÃO COMPUTACIONAL COMO UMA FERRAMENTA
DE ENSINO NA EDUCAÇÃO INFANTIL

Trabalho de Conclusão de Curso
apresentado ao Programa de Graduação
em Ciências da Computação da
Universidade Estadual da Paraíba, UEPB,
como requisito parcial à obtenção do título
de Bacharel em Ciências da Computação.

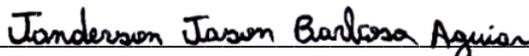
Área de concentração: Informática na
Educação.

Aprovada em 16 de Junho de 2023.

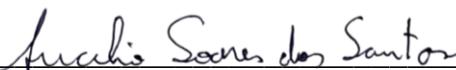
BANCA EXAMINADORA



Prof. Dr. Wellington Candeia de Araújo (Orientador)
Universidade Estadual da Paraíba (UEPB)



Prof. Dr. Janderson Jason Barbosa Aguiar
Universidade Estadual da Paraíba (UEPB)



Prof. Dr. Jucelio Soares dos Santos
Universidade Estadual da Paraíba (UEPB)

É expressamente proibido a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano do trabalho.

S586a Silva, Jose Kelvyn dos Santos.
Análise da programação computacional como uma ferramenta de ensino na educação infantil [manuscrito] / Jose Kelvyn dos Santos Silva. - 2023.
56 p. : il. colorido.

Digitado.

Trabalho de Conclusão de Curso (Graduação em Computação) - Universidade Estadual da Paraíba, Centro de Ciências e Tecnologia, 2023.

"Orientação : Prof. Dr. Wellington Candeia de Araújo, Coordenação do Curso de Computação - CCT. "

1. Ensino infantil. 2. Pensamento computacional. 3. Programação computacional. I. Título

21. ed. CDD 372.358

Dedico este trabalho aos meus familiares e amigos.

AGRADECIMENTOS

Gostaria de expressar minha gratidão a todas as pessoas que me ajudaram, diretamente ou indiretamente, a alcançar este marco tão importante em minha vida acadêmica, durante todo esse ciclo.

Primeiramente, gostaria de enfatizar meu agradecimento ao meu orientador, Wellington Candeia, por sua orientação, apoio e disponibilidade durante todo o desenvolvimento do trabalho. Seu conhecimento e suporte foram cruciais para a sua conclusão. Agradeço também aos meus familiares, em especial meus pais, José Alves e Maria José pelo total apoio e suporte durante toda a realização da formação, sem os seus auxílios inestimáveis, eu não teria sido capaz de chegar aonde estou.

Não posso deixar de agradecer também aos amigos, que estiveram comigo durante toda a formação, Camila Sabino, Gustavo Barros e Maiara da Paz, companheiros de curso que foram fundamentais para minha formação, que sempre me apoiaram e encorajaram em cada etapa do caminho. Seus sorrisos, abraços e palavras de incentivo me fizeram não desistir no decorrer do caminho.

Ao Cristhian Filgueira Dias, pela sua parceria, apoio e resiliência durante meus últimos anos na graduação. Ao Luan Franklin e Lucas Kayan que sempre me apoiaram e estiveram ao meu lado durante toda a minha jornada.

Por último, mas não menos importante, gostaria de agradecer a Deus, essa força superior que está comigo em todos os momentos, atuando como um pilar fundamental em minha vida.

"Programar é pensar de forma lógica, criativa e estruturada. Ensinar programação é ensinar a pensar."

(Mitchel Resnick)

RESUMO

Com a crescente popularização e avanço da programação em diversas áreas da sociedade, surge o incentivo no ensino dessa habilidade às crianças, pois oferece vantagens como o desenvolvimento do pensamento computacional, estímulo à criatividade, aprimoramento das capacidades de computação e estímulo ao pensamento criativo. Estimular a computação desde cedo tem um impacto significativo no desenvolvimento infantil, já que o seu cérebro é altamente adaptável e capaz de aprender de forma rápida. Entretanto, é crucial enfatizar que cada indivíduo tem diferentes ritmos de aprendizado, sendo essencial adaptar o ensino à idade e ao desenvolvimento cognitivo de cada um. Além disso, o ensino deve ser divertido e lúdico para que as crianças possam aprender enquanto se divertem. Para tornar o ensino de programação mais divertido, foram desenvolvidos alguns métodos como também software para que as crianças pudessem aprender brincando e de forma mais descontraída. O trabalho também destaca alguns possíveis impactos negativos, como a sobrecarga cognitiva, a dependência tecnológica e a falta de habilidade social. O objetivo deste trabalho acadêmico é refletir sobre o ensino da programação na infância, seu impacto na vida das pessoas e suas vantagens. A metodologia utilizada para explorar o ensino da programação na infância foi uma pesquisa bibliográfica, fornecendo insights técnicos e sociais, embora não apresente soluções diretas. No entanto, ao discutir argumentos relacionados ao ensino precoce da computação, seu desenvolvimento e seus benefícios, este trabalho contribuirá para o debate sobre o ensino nos primeiros anos de vida.

Palavras-Chave: ensino infantil; pensamento computacional; programação.

ABSTRACT

With the growing popularization and advancement of programming in various areas of society, there is an incentive to teach this skill to children, as it offers advantages such as the development of computational thinking, stimulating creativity, improving computing skills and stimulating creative thinking. Stimulating computing from an early age has a significant impact on child development, as their brain is highly adaptable and able to learn quickly. However, it is crucial to emphasize that each individual has different learning rhythms, and it is essential to adapt teaching to each person's age and cognitive development. In addition, teaching should be fun and playful so that children can learn while having fun. To make teaching programming more fun, some methods were developed as well as software so that children could learn by playing and in a more relaxed way. The work also highlights some possible negative impacts, such as cognitive overload, technological dependence and lack of social skills. The objective of this academic work is to reflect on the teaching of programming in childhood, its impact on people's lives and its advantages. The methodology used to explore the teaching of programming in childhood was a bibliographical research, providing technical and social insights, although it does not present direct solutions. However, by discussing arguments related to the early teaching of computing, its development and its benefits, this work will contribute to the debate about teaching in the first years of life.

Keywords: child education; computational thinking; schedule.

LISTA DE ILUSTRAÇÕES

Figura 1	Áreas de conhecimento no ensino de computação	20
Figura 2	Pilares do Pensamento Computacional	21
Figura 3	Forma geométrica usando Logo	25
Figura 4	Resolução de desafio utilizando Scratch	27
Figura 5	Tela Inicial do Website CODE.ORG	29
Figura 6	Sala de estudo da Happy Code na cidade de São Paulo	30
Figura 7	Patinho Feio	31

LISTA DE QUADROS

Quadro 1 – Pontos positivos e pontos negativos do ensino da programação computacional na infância 43

Quadro 2 – Plataformas de ensino de programação computacional para crianças 44

LISTA DE ABREVIATURAS E SIGLAS

2D	Duas dimensões
BNCC	Base Nacional Comum Curricular
Capre	Comissão de Coordenação das Atividades de Processamento Eletrônico
CC	Ciência da Computação
EUA	Estados unidos da América
IBGE	Instituto Brasileiro de Geografia e Estatística
INCoD	Instituto Nacional de Convergência Digital
INE	Departamento de Informática e Estatística
MG	Minas Gerais
MIT	Instituto de Tecnologia de Massachusetts
PC	Pensamento Computacional
PET	Programa de Educação Tutorial
SBC	Sociedade Brasileira de Computação
SC	Santa Catarina
SP	São Paulo
TI	Tecnologia da Informação
UFSC	Universidade Federal de Santa Catarina
UIT	União Internacional de Telecomunicações
USP	Universidade de São Paulo

LISTA DE SÍMBOLOS

- % Porcentagem
- ® Marca Registrada

SUMÁRIO

1 INTRODUÇÃO	13
2 REFERENCIAL TEÓRICO	17
2.1 A importância do estímulo na infância	17
2.1.1 Estágios de desenvolvimento da criança	18
2.2 Lógica de programação para crianças	18
2.2.1 Pensamento computacional	19
2.2.2 Fundamentos do pensamento computacional	21
2.2.3 Como explicar um Algoritmo a uma criança	22
2.2.4 Atividade de programação desplugada	22
2.3 Plataformas de ensino de programação infantil	24
2.3.1 A linguagem Logo	24
2.3.2 Scratch: A plataforma para crianças	25
2.3.2.1 SCRATCH no Ensino Fundamental – Um Estudo de Caso	27
2.3.3 Code.org	28
2.3.4 Happy Code	29
2.4 Avanço do ensino de programação no Brasil	30
2.4.1 SuperGeeks	32
2.4.2 Programaê	33
2.5 Pensamentos críticos do ensino de programação precoce	34
3 OBJETIVOS	37
4 METODOLOGIA	38
5 RESULTADOS E DISCUSSÕES	42
6 CONSIDERAÇÕES FINAIS	48
REFERÊNCIAS BIBLIOGRÁFICAS	50

1 INTRODUÇÃO

A programação é um processo de escrita, testes e manutenção de programas de computadores. Esses programas, por sua vez, são compostos por conjuntos de instruções determinados pelo programador que descrevem tarefas a serem realizadas pela máquina e atendem diversas finalidades. A arte de programar consiste na arte de organizar e dominar a complexidade (DIKSTRA, 2009). Algumas pessoas acreditam que a Ciência da Computação é a arte dos gênios, mas, na verdade, é o contrário; é um conjunto de pessoas construindo coisas com base no trabalho de outras, assim como uma parede feita de pequenas rochas.

Durante um longo período, a programação era vista como algo reservado e específico para um determinado grupo de pessoas que possuíam algumas habilidades e uma determinada facilidade para dominar a codificação. Entretanto, alguns estudiosos sempre acreditaram que a programação de computadores deveria ser uma prática acessível a todas as pessoas, independente da competência, e que, se possível, poderia ser ensinada desde cedo a crianças e adolescentes.

Mitchel Resnick (2008) afirmou que: “Escrever códigos não é só para os magos dos computadores”, defendendo a programação como algo para todos. Em uma palestra com o tema “Vamos ensinar as crianças a escrever códigos”, Resnick exhibe as vantagens de incentivar as crianças a codificar, para que eles passem de usuários para criadores de tecnologias. Resnick argumenta que nem todos os alunos se tornam escritores, mas todos aprendem a escrever para expressar suas ideias, se comunicar e organizar pensamentos. O mesmo acontece quando as crianças aprendem a codificar. “Eles aprendem a organizar, expressar e compartilhar ideias de novas maneiras, em um novo meio”, afirma a educadora Daniela Lyra Cardoso (2013).

A Lógica de Programação deveria estar como qualquer outra disciplina do ensino primário (PEREIRA, 2013). O ensinamento da programação na infância poderia evoluir o pensamento lógico e computacional para a solução de problemas (KAFAI; BURKE, 2013). Segundo a Model Curriculum for K-12 Computer Science (2012), a maioria das profissões deste século têm de alguma forma relação e/ou necessidade de conhecimento na área da informática e computação, o que reforça a importância do ensino de Lógica de Programação desde a Educação Básica.

O desenvolvimento do raciocínio lógico das crianças que obtêm contato com a programação é notório, já que no mundo de desenvolvimento o usuário lida com

resoluções de problemas lógicos, necessitando uma análise e tomada de decisões, como também um olhar mais crítico às situações e ao lidar com a tecnologia.

Alguns benefícios estão comprovados que o ensino da linguagem de programação pode trazer para os alunos. Tendo o potencial de ajudar no aprendizado de outras disciplinas, como também no estímulo do raciocínio lógico, na criatividade e na organização de pensamentos. O fundador da escola de programação e robótica Happy Code, Rodrigo Santos (2015) afirmou que: “Os alunos aprendem a programar através de brincadeiras e projetos educativos, o que desperta o entusiasmo e promove o desenvolvimento da criança”.

De acordo com Mitchel Resnick (2014), às pessoas que não aprenderem a programar serão programadas. Resnick afirmou que a programação deveria ser tão crucial como escrever ou ler, definida como uma das habilidades do século XXI. O uso da tecnologia no ensino para crianças sempre foi um desafio, assim como refletir sobre o seu uso efetivo, e a relação do estudante com a tecnologia. A estimulação para passarem de consumidores passivos para desenvolvedores de tecnologia e programas, incentivando e aprimorando o desenvolvimento do pensamento computacional, é algo gradativo e constante. Mesmo com os desafios e dificuldades, o aprendizado na infância ainda é mais efetivo, quando o aluno está em fase de desenvolvimento e adaptação e se envolve na descoberta de um novo mundo.

Mesmo com alguns estudos e comprovações dos benefícios e vantagens do ensino a programação a crianças, vale destacar alguns educadores e profissionais que se opõem a essa prática. Alguns críticos afirmam que o estímulo a essa prática pode ser prejudicial ao desenvolvimento das crianças, diminuindo o seu convívio social e das experiências necessárias durante a fase da criança, como brincar e sair ao ar livre.

Setzer (1988), afirma que entregar um computador para uma criança é uma forma certa de uma criança perder sua infantilidade, tornando-o senil no ponto de vista mental e emocional, obrigando a uma especialização mental precoce.

Em entrevista ao site Business Insider, Linus Torvalds (2014), o responsável pelo kernel, disse: “Acho que é algo especializado, e ninguém espera que a maioria das pessoas façam isso. Não é como aprender a ler e escrever ou fazer contas básicas de matemática”. O cientista também afirma: “Eu acho que as pessoas devem ter alguma forma de obter mais contato, mas apenas para que possam descobrir por si mesmas se têm aptidão para isso”. O convívio social é extremamente importante

para o desenvolvimento de crianças e adolescentes, e como tal não pode ser deixado de lado, mas talvez isso possa acontecer tendo algo que divida sua atenção, como é o caso do ensino da programação (GERALDES, 2014, p.2).

Em uma entrevista para o site iG, o pediatra Antônio Carlos de Souza Aranha (2013) considera a importância da criança desenvolver primeiramente o raciocínio e a criatividade para depois utilizar eletrônicos livremente, sem se tornar dependente da tecnologia. Segundo o pediatra, hoje em dia as crianças são cada vez mais consumidoras e menos criativas em todos os níveis.

A implementação da programação computacional no ensino infantil pode apresentar possíveis desafios e limitações, como a falta de computadores e eletrônicos adequados e de educadores preparados para o ensino. Além da representatividade limitada e inclusão no setor tecnológico. A falta de representatividade e diversidade dentro do mundo da tecnologia pode gerar um certo afastamento de determinados grupos no setor. Mesmo que as coisas estejam mudando aos poucos e o mundo se adaptando ao ambiente tecnológico, a diversidade ainda não é uma realidade na maioria das empresas. Nos EUA, 95% dos trabalhadores de tecnologia são brancos e 75% são homens, de acordo com uma pesquisa da Atlassian (2021). Apenas 5% dos CEOs das empresas listadas na Fortune 500, a lista anual das 500 maiores empresas dos Estados Unidos, são mulheres. No Brasil, apenas 20% dos 580 mil profissionais que atuam no mercado de TI são mulheres, segundo pesquisa do IBGE de 2016.

Segundo o IBGE, a população negra do Brasil representa 54,9% da população, mas apenas 4,7% atuam nos conselhos executivos dessas organizações. Nesse sentido, a pesquisa Potências Negras Tec, realizada pela Potências Negras Journeys, com foco na evolução da população negra e parda e na experiência de compra, envolveu 2.693 pessoas online (1.528 negras e 1.165 outras partidas), entre junho e julho de 2021.

Os resultados mostraram que 59% dos negros não trabalham com tecnologia, mas têm interesse em trabalhar na área. Entre os negros que estudam ou cursam tecnologia, 41% trabalham na área. De acordo com o levantamento, 29% dos negros em tecnologia trabalham em cargos de gestão ou liderança (divididos por gênero, cai para 28% para as mulheres), em comparação com 46% dos brancos. Entre a amostra pesquisada, 67% dos negros eram altamente educados, em comparação com 82% dos brancos.

Barreiras de carreira no mercado de tecnologia também se reflete naqueles que se identificam como LGBTQIAPN+, que são menos propensos a prosperar no campo e mais propensos a abandonar suas carreiras e passar para outros campos devido ao assédio frequente e derrogação de carreira, de acordo com cientistas da Universidade de Michigan e a Universidade de Temple, nos Estados Unidos (BARROS, 2021).

Este trabalho tem como principal objetivo discutir sobre o ensino da programação computacional na infância, abordando benefícios do estímulo do mundo computacional para as crianças a partir de análises de dados e de opiniões de professores e cientistas. Bem como, refletir sobre a dificuldade de implementação do ensino da computação e de possíveis problemas que podem ocorrer sem um monitoramento devido no seu ensino.

2 REFERENCIAL TEÓRICO

Incentivar uma pessoa a realizar determinada atividade nos primeiros anos de vida pode ser fundamental para sua assimilação e domínio, já que ela se encontra na fase de desenvolvimento humano. Incentivar o ensino de programação de computadores na infância pode levar ao desenvolvimento no raciocínio e na resolução de problemas, entre outras áreas da vida. Essa temática será discutida nas seções a seguir, como alguns métodos utilizados para lecionar a programação computacional e o seu avanço.

2.1 A importância do estímulo na infância

O período que vai da gestação até os 6 anos de idade é o momento mais importante, vital e único no desenvolvimento humano, conhecido como a primeira infância, onde todos os sentidos de uma criança são desenvolvidos. Este é um momento de desenvolvimento cognitivo, social, emocional e de linguagem.

Cada criança tem sua singularidade desde o nascimento, e essas diferenças individuais impactam a maneira como elas aprendem. Além disso, a forma como são cuidadas e interagem com seus familiares e cuidadores nos primeiros anos de vida desempenha um papel significativo na formação de sua identidade e no desenvolvimento de suas habilidades. (ENGLE e LUCAS, 2012, p. 9).

A contribuição na interação e no desenvolvimento é fundamental para uma influência de dedicação, cuidado, atenção e qualidade nesse período, como também realizar atividades que promovam o desenvolvimento da criança.

As interações e experiências com outras crianças, adultos, professores e irmãos, bem como o envolvimento com o ambiente escolar e familiar, desempenham um papel fundamental no desenvolvimento intelectual, racional, moral e linguístico. (NOGUEIRA e LEAL, 2015, p.130).

Segundo Piaget (1964), a criança se desenvolve através de um contínuo processo de equilíbrio e desequilíbrio. O equilíbrio corresponde àquilo que a criança já sabe, é quando as situações são conhecidas por ela e o desequilíbrio acontece quando ela se depara com uma situação desconhecida, que exige com que busque soluções mobilizando a modificação ou criação de novos esquemas.

A primeira etapa da educação básica é a educação infantil, sendo crucial proporcionar estímulos apropriados às crianças no seu desenvolvimento psicológico,

emocional, sensorial e cognitivo. Para o seu aprimoramento, a criança necessita de um ambiente que forneça a possibilidade de experimentar diferentes sensações.

2.1.1 Estágios de desenvolvimento da criança

Segundo PIAGET (1994), o desenvolvimento da criança respeita certas fases, que a permitem avançar desde os conceitos básicos aos mais complexos, de acordo com a sua etapa de evolução. São elas: Sensório-Motor (0 a 2 anos), Pré-Operatório (2 a 7 anos), Operatório-Concreto (7 a 11 anos) e Operatório-Formal (11 anos em diante).

O primeiro período é o Sensório-Motor, geralmente ocorre entre zero aos dois anos de idade, aqui a criança explora o mundo através dos sentidos, interagindo com os objetos. As ações da criança nesta etapa de desenvolvimento ocorrem por meio de reflexos. O segundo período é o Pré-Operatório, geralmente ocorre entre os dois aos sete anos de idade, este é o período em que a criança inicia sua vida pré-escolar. Nesta fase, aparece o desenvolvimento da linguagem e da função simbólica, onde objetos são fantasiados na imaginação da criança. Também é a fase do egocentrismo, onde a criança quer ser o centro das atenções. Nesta etapa, a criança também começa a adquirir noções de espaço e de tempo, mostrando-se bastante curiosa.

O terceiro período é o Operatório-Concreto, geralmente ocorre entre os sete aos onze anos de idade, fase esta em que a criança tem facilidade para lidar com a lógica e encontrar soluções por meio do concreto, não conseguindo solucionar problemas abstratos. Etapa esta que marca a transição da infância para a puberdade.

O quarto período é o Operatório-Formal, geralmente ocorre entre os onze aos quinze anos de idade, onde a maioria das crianças resolve seus problemas por meio da lógica, dependendo nesta fase de ambiente estimulador, também desenvolve suas potencialidades conseguindo resolver operações mentais, podendo encontrar várias alternativas para resolver um problema.

Todos os indivíduos passam por essas quatro fases seguindo a mesma sequência, o que muda de um para outro, é o período de duração de cada fase, dependendo do estímulo recebido. Tendo a oportunidade de ser motivado por meio de pessoas e ambientes, toda criança tem possibilidade de se desenvolver.

2.2 Lógica de programação para crianças

Desde os primórdios, o homem tenta resolver seus problemas o mais rápido possível, com o mínimo de esforço permitido e tentando automatizar as tarefas

cotidianas. Isso possibilitou instrumentos como rodas, alavancas, carros e até computadores.

No entanto, tarefas que devem ser automatizadas por um computador devem ser pensadas e colocadas em uma sequência de ações antes de serem plugadas na máquina. Essa sequência de ações é chamada de programa, e a etapa de formulação de uma solução que gera o programa é chamada de lógica de programação (XAVIER, 2018, p. 11).

A meta principal de qualquer programador é resolver problemas por meio de soluções lógicas para obter resultados eficientes. Porém, se as soluções não forem bem planejadas, com certeza essa meta não será atingida (XAVIER, 2018, p. 12).

A lógica desempenha um papel fundamental no aprendizado da programação, pois vai além da simples habilidade de escrever código, envolvendo uma ampla gama de habilidades cognitivas. Portanto, é essencial que as crianças pratiquem a lógica desde cedo, pois isso contribui significativamente para o desenvolvimento do raciocínio lógico, estimula a criatividade e fortalece as habilidades de resolução de problemas. Essa prática precoce proporciona às crianças uma base sólida para enfrentar desafios futuros relacionados à programação e promove o desenvolvimento integral de suas capacidades intelectuais.

No entanto, alguns dos desafios de ensinar esta lição básica para crianças é torná-la um assunto alegre e emocionante que realmente desperte sua curiosidade sobre o assunto de uma maneira apropriada para a idade.

Aprender a programar é muito mais do que entender e escrever linhas de código e criar jogos e aplicativos. A codificação já está presente na vida de crianças em países como os Estados Unidos, Brasil, Austrália, etc. e pode estimular o aprendizado nos primeiros anos de vida ao estimular algo denominado pensamento computacional.

2.2.1 Pensamento computacional

O método de assimilar os aspectos da computação no mundo e aplicar técnicas e ferramentas para facilitar processos e sistemas é denominado de pensamento computacional. Por exemplo, quando uma criança resolve um problema dividindo em etapas e usando a lógica. Usando fundamentos de computação em diversos domínios, desenvolvendo competências criativas, críticas e estratégicas. Essa metodologia é caracterizada pela ligação entre os fundamentos da computação e do pensamento crítico para resolução de problemas. Sendo esse método, um dos fatores

de sucesso para o ensino da programação. A computação apresenta algumas áreas de conhecimento, visto na Figura 1. Esta visão defende que a cultura do computador ajuda a sociedade não somente a aprender, mas especialmente oferece uma nova maneira de aprender a aprender. (LU e FLETCHER, 2009, p.260-264).

O Pensamento Computacional pode ser colocado como uma das habilidades intelectuais básicas de um ser humano, comparada a ler, escrever, falar e fazer operações aritméticas. Habilidades estas que servem para descrever e explicar situações complexas. Nesta linha de raciocínio, o PC é mais uma linguagem que pode ser utilizada para as pessoas falarem sobre o Universo e seus processos complexos (WING, 2006).

Figura 1 — Áreas de conhecimento no ensino de computação



Fonte: WANGENHEIM, Christiane; NUNES, Vinícius; SANTOS, Giovane, p.116, 2014.

O aprendizado de programação envolve o contato com novas linguagens e também com um novo modo de pensar, como o exemplo das habilidades de abstração e análise. Quando as crianças começam a aprender programação, um novo olhar, mais participativo e inovador, começa a se formar, e os aparelhos tecnológicos ganham um novo significado na vida delas.

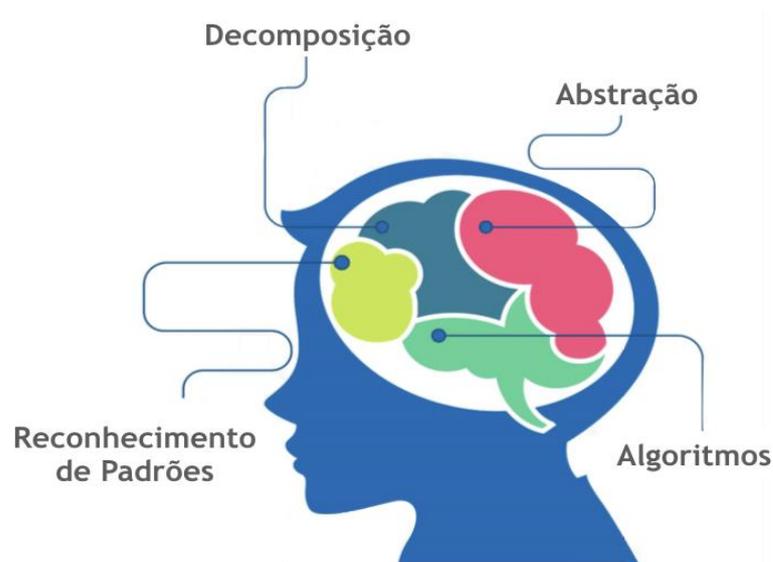
A programação é uma atividade que consegue trabalhar algumas áreas do Pensamento Computacional. Assim, é possível começar a aprender a programar a partir dos 4 anos de idade, sem que as crianças precisem ter contato com computadores, através da utilização de brinquedos e brincadeiras que estimulem, por

exemplo, a capacidade do raciocínio lógico. Com atividades desplugadas, as crianças transmitem conhecimentos sem a utilização de equipamentos eletrônicos.

2.2.2 Fundamentos do pensamento computacional

O pensamento computacional consiste em alguns princípios, como mostrado na Figura 2, para a resolução de problemas mais complexos. O princípio da abstração analisa elementos importantes, separando daqueles que podem ser relevados. A decomposição, divide um problema maior em partes menores. O terceiro pilar é o reconhecimento de padrões, que ajuda a identificar aspectos comuns no processo e, com isso, a criança consegue identificar tendências e semelhanças de comportamento. Inspire o pensamento de novas soluções, aproveite a inovação e a criatividade. O quarto pilar, pensamento algorítmico, cria passos e soluções para alcançar objetivos. É o uso da lógica e da razão para resolver problemas.

Figura 2 — Pilares do Pensamento Computacional



Fonte: Adaptado de Shoop *et al.* (2016).

O ensino do pensamento computacional é caracterizado pela construção do raciocínio lógico, sendo uma das principais habilidades adquiridas. Nas crianças, o PC começa nas primeiras fases do ensino, quando percebem que alguns padrões definem determinados comportamentos. Passando a resolver problemas de forma lógica, sempre focando no pensamento racional para determinar respostas aos diferentes estímulos das atividades cotidianas. Por meio da exposição precoce a jogos e outras atividades relacionadas ao ambiente digital, os indivíduos podem resolver problemas de maneira mais eficaz e estratégica. Preparando a criança para conseguir assimilar novidades com maior facilidade.

O pensamento computacional tem como base os algoritmos e a abstração, que abrem espaço para a autonomia das crianças, passando elas de consumidores da tecnologia para criadores e desenvolvedores de recursos digitais.

2.2.3 Como explicar um Algoritmo a uma criança

Um algoritmo é uma sequência de instruções ou comandos realizados de maneira sistemática com o objetivo de resolver um problema ou executar uma tarefa, como uma instrução detalhada, com diferentes passos, ou a uma fórmula de resolver um problema ou de completar uma tarefa. O algoritmo pode ser abordado em várias situações, como em uma receita ou um cronograma de atividades. Na programação, os programadores escrevem algoritmos que dão ordens ao computador para que este realize uma determinada tarefa. A estimulação do uso dos algoritmos nas crianças pode ocorrer em simples tarefas do dia a dia, como escovar os dentes mantendo sempre as mesmas repetições e ordens de execução.

Pensar com recurso a algoritmos pode também ser definido como o desenho mental dos passos necessários para resolver um problema. Esta capacidade é crucial para o sucesso em áreas como Matemática e Ciências. As crianças usam algoritmos a toda a hora e sem se aperceberem, particularmente em Matemática, para resolver um problema que inclua uma divisão longa, as crianças automaticamente aplicam um algoritmo previamente ensinado, que lhes permite iterar os dígitos da resposta do problema. Para cada dígito do dividendo, a criança tem que dividir, multiplicar e subtrair. Em suma, utilizar algoritmos permite que as crianças consigam resolver problemas complexos numa soma de problemas menores e mais fáceis (HAPPY CODE, 2018). Também permite que estes organizem soluções de problemas em termos de passos claros e bem definidos, parte de um procedimento ou protocolo.

As crianças podem fortalecer o seu raciocínio de algoritmos ao completar uma série de atividades de criação de linhas simples de código ou quando desenharam algoritmos simples baseados em repetições, sequências e lógica condicional aplicada aos problemas do dia a dia. Melhorando o seu raciocínio sobre algoritmos se exercitarem constantemente e realizarem projetos criativos para aplicar os conteúdos absorvidos.

2.2.4 Atividade de programação desplugada

O educador Alberto Cunha (2020), define as atividades de programação desplugada como aquelas que não dependem de equipamentos eletrônicos, tais como celulares, tablets ou computadores. São atividades que podem ser desenvolvidas com

materiais físicos, normalmente baratos e reutilizados, como papéis, papelão, canetas, lápis coloridos, caixas, palitos, cola, régua, garrafas pet, tampinhas, tecidos, linhas, ou mesmo ingredientes culinários, entre outros. Trazendo conceitos e problemas do mundo da computação para a educação básica.

Alberto (2020) foi fellow do Desafio Brasileiro de Aprendizagem Criativa da Rede Brasileira de Aprendizagem Criativa com o projeto “Aprender fazendo: programação e robótica”. O professor afirma que “Trabalhar atividades desplugadas com crianças e adolescentes traz vários benefícios. Elas exigem que se coloque as mãos na massa, trabalhando a organização do pensamento lógico e reflexivo”. Ele defende que a atividade desplugada proporciona o desenvolvimento de várias habilidades pouco exercitadas, desafiando todos os envolvidos. Promovendo diversos aprendizados de forma lúdica e prazerosa para todos os envolvidos.

O professor Alberto Cunha, que divide suas experiências no canal do Youtube Aprendiz 21, afirma que ao trabalhar com atividades desplugadas, crianças e jovens deparam-se com o novo o tempo todo. A construção de objetos concretos para se brincar, jogar e contar histórias desafia os envolvidos a materializar ideias que estão em níveis abstratos. A aprendizagem vai ser mais significativa quando ela extrapolar o nível apenas do abstrato. Os *feedbacks* e as avaliações chegam tanto dos construtores, quanto dos que estão de fora. As críticas podem promover ajustes tanto no desenvolvimento do processo de construção quanto no produto, o que eleva o nível do desenvolvimento das ideias, promovendo a aprendizagem a um patamar mais significativo para todos.

A formadora de professores, Carla Arena, que desenvolve metodologias para treinamento em letramento digital, destaca a acessibilidade como um ponto essencial na educação. Arena (2017) afirma que quando pensamos em atividades desplugadas, estamos falando do desenvolvimento de uma mentalidade e atitude para desvendarmos os “segredos da computação”. Quando falamos de acesso, o mais importante para nossos jovens é desenvolver essa mentalidade e atitude de questionadores, que testem, falhem, compreendam e resolvam os problemas a que são submetidos em seus contextos.

Quando desenvolvida e estimulada a literacia computacional, é dada a oportunidade de obter uma nova forma de raciocinar e de resolver problemas. Logo, quando conseguirem mais recursos, estarão prontos para progredir em termos de competência e desenvolvimento de habilidades em informática. Quando falamos de

acesso, o mais importante para nossos jovens é desenvolver uma mentalidade e uma atitude questionadora, eles testam, falham, entendem e resolvem os problemas que apresentam em seu ambiente.

2.3 Plataformas de ensino de programação infantil

Na atualidade, o uso de plataformas com linguagem específicas para o ensino de programação tem sido um exemplo de introdução para o mundo da codificação, esse meio também vem sendo utilizado para a imersão de crianças na programação de computadores. A interface de fácil entendimento e interatividade atrai os seus usuários, facilitando a compreensão.

2.3.1 A linguagem Logo

Uma plataforma pioneira de ensino e aprendizagem de lógica e linguagens de programação é a *Logo Language*, criada no final dos anos 60, especialmente para uso das crianças a partir dos 4-5 anos. A linguagem consiste em uma tartaruga, como mostra na Figura 3, gráfica que pode ser movimentada por meio de comandos como avançar e virar a direita, permitindo a criação de formas geométricas.

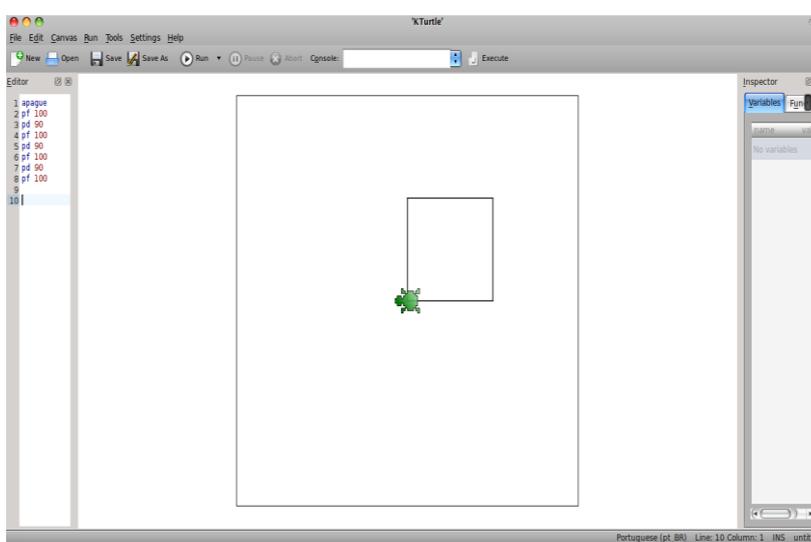
Criada por Cynthia Solomon, Wally Feurzeig e Seymour Papert, a linguagem LOGO é baseada em uma notação que faz sentido para um sujeito sem nenhum conhecimento de informática, ao contrário do que ocorre na maioria das outras linguagens de programação. Tornando-a torná-la simples, mas poderosa, podendo ser utilizada por indivíduos ainda nos primeiros anos da escola, permitindo o desenvolvimento de habilidades e de capacidade de resolução de problemas. É simples à primeira vista, mas ensina os conceitos de comandos, parâmetros e procedimentos que formam a base de funcionamento de todas as linguagens de programação.

Ao considerar os ambientes intelectuais disponíveis para as crianças na sociedade atual, observa-se uma carência de recursos que as estimulem a pensar sobre o próprio pensamento, aprender a expressá-lo e testar suas ideias por meio da sua exteriorização. No entanto, a introdução dos computadores pode transformar completamente essa situação. Até mesmo atividades simples, como trabalhar com uma tartaruga programável, podem proporcionar novas oportunidades para aprimorar nossa capacidade de refletir sobre o ato de pensar. Ao programar a tartaruga, somos levados a refletir sobre como executar as ações que gostaríamos que ela realizasse. Dessa forma, ensinar a tartaruga a agir ou pensar pode nos levar a uma reflexão sobre

nossas próprias ações e pensamentos, ampliando nossa compreensão e aprofundando nossa capacidade de autorreflexão. (PAPERT, 1985, p. 45).

De acordo com Larivée e Michaud (1980), os princípios fundamentais subjacentes a essa linguagem de programação podem ser resumidos em seis: os conceitos de estado, procedimento, nomeação, recursão, bug e depuração. O Logo foi incentivado desde o início por uma perspectiva tipo Robin Hood de roubar a programação dos tecnologicamente privilegiados[...] e dá-lo às crianças. (PAPERT, 2008, p. 170).

Figura 3 — Forma geométrica usando Logo



Fonte: KLEINSORGEN. Educar Para Liberdade. 2011.

O LOGO é considerado mais como uma linguagem com a qual se pode aprender do que uma linguagem que se aprende, visando mais a aprendizagem de um processo (modos de pensar) do que a aprendizagem de conteúdo (o que pensar) (MIRANDA, 1990).

2.3.2 Scratch: A plataforma para crianças

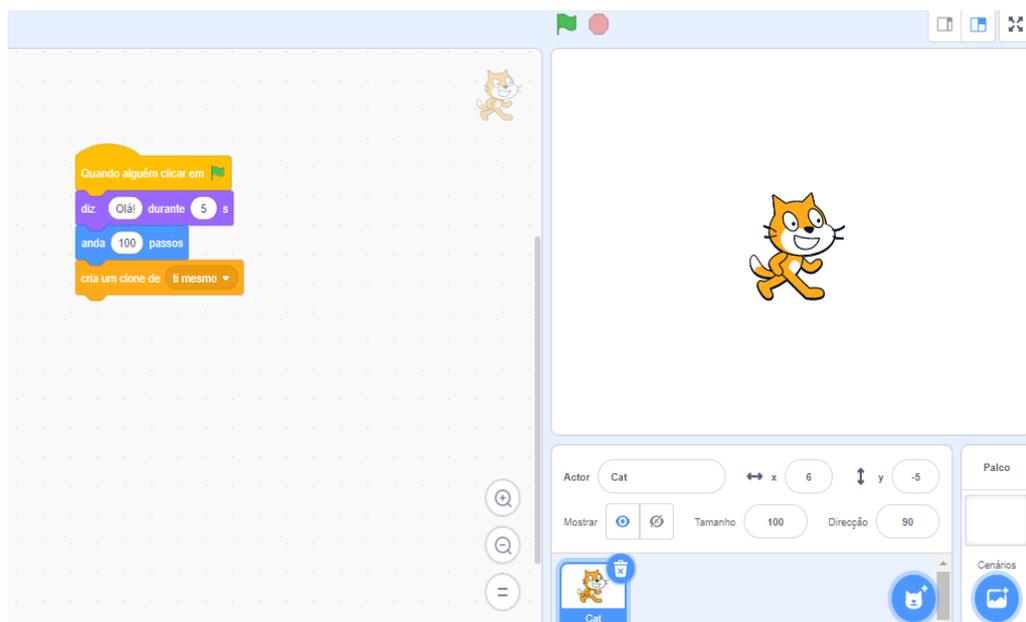
Uma das primeiras plataformas a evoluir a linguagem Logo foi a Scratch em 2007, com o objetivo de ensinar lógica de programação para crianças de forma mais visual, sem a necessidade de escrita.

No ambiente multimídia Scratch, os alunos programam suas criações por junção de blocos gráficos, sem que nenhum dos sinais de pontuação obscuros ou sintaxe das linguagens de programação tradicionais. Desta forma, ele torna a programação acessível a um público muito mais amplo, em uma faixa etária mais jovem. (RESNICK, 2007, p. 78).

O Scratch foi desenvolvido pelo Lifelong Kindergarten Group no MIT Media Lab, com o apoio financeiro da National Science Foundation, Microsoft, Intel Foundation, Fundação MacArthur, Google, Iomega e MIT consórcios de pesquisa do Media Lab. O projeto iniciou em 2003 e, em 2007, o ambiente multimídia Scratch e o primeiro site foram lançados, sendo uma linguagem de programação gráfica que pode ser usada para criar programas interativos, jogos e animações. É um espaço de programação que tem blocos como base, como mostra na Figura 4, ideal para quem está iniciando no mundo da programação por não necessitar de um conhecimento introdutório em linguagens de programação. Essa linguagem foi desenvolvida para auxiliar crianças acima de 8 anos a aprender noções da computação e da matemática, possibilitando que os usuários desenvolvam histórias e animações. Permitindo também a criação de personagens que interagem, dançam e cantam inspirados nos princípios da linguagem do Logo. Com o foco em ajudar no aprendizado de programação de uma maneira criativa e divertida para o usuário, podendo ser utilizado por crianças a partir de 8 anos e pessoas sem conhecimentos de programação.

O ensino da lógica de programação na plataforma, um ambiente voltado para computação criativa, é mais intuitiva e mais estimulante visualmente. Divididos em oito categorias: Variáveis, Operadores, Sensores, Movimento, Som, Controle, Aparência e Caneta. Assim o usuário necessita apenas construir o algoritmo. Atualmente, o SCRATCH está presente em mais de 150 países com mais de 11 milhões de usuários e quase 4 milhões de projetos compartilhados.

No artigo “Ensino de Computação com SCRATCH no Ensino Fundamental – Um Estudo de Caso” publicado na revista Brasileira De Informática na Educação é apontado que algumas pesquisas mostram que o uso do SCRATCH contribui positivamente no ensino de computação em escolas. Criando programas de software com o SCRATCH, crianças aprendem a pensar criativamente, a trabalhar de forma colaborativa e a pensar de forma sistemática na solução de problemas.

Figura 4 — Resolução de desafio utilizando Scratch

Fonte: Captura de tela, 2022.

2.3.2.1 SCRATCH no Ensino Fundamental – Um Estudo de Caso

Em Florianópolis-SC foi desenvolvida uma atividade interdisciplinar para alunos do Ensino Fundamental, com o objetivo de analisar um meio para o ensino de computação e programação. Coordenado pelo Instituto Nacional de Convergência Digital (INCoD) do Departamento de Informática e Estatística (INE) da Universidade Federal de Santa Catarina (UFSC), em cooperação com o Programa de Educação Tutorial (PET) do Departamento de Informática e Estatística (INE) da Universidade Federal de Santa Catarina (UFSC) e o Instituto Federal de Santa Catarina (IFSC). O projeto foi realizado durante 6 semanas em uma escola privada da cidade, com crianças entre 6 e 7 anos de idade. Durante a realização do projeto os alunos desenvolveram uma história interativa usando o SCRATCH, com conteúdo de Literatura e Artes. Com um total de 24 alunos, a avaliação foi feita por meio de observações e com base na análise dos resultados produzidos pelos alunos. Em geral, todos conseguiram criar a história com facilidade atingindo os objetivos de aprendizagem. Os alunos passaram a reconhecer que o software pode controlar operações do computador de forma interativa. Demonstraram competência em trabalhar de forma cooperativa e colaborativa. Para concluir os autores acreditam que, por meio desta experiência, é possível perceber que o ensino da computação pode ser integrado ao currículo do Ensino Fundamental de forma harmônica e

interdisciplinar. Além disso, observaram que as aulas motivaram os alunos a aprender mais sobre programação, percebendo a participação ativa de todos os alunos, causando a impressão de que as crianças estavam aprendendo programação sem perceber.

2.3.3 Code.org

Code.Org é uma plataforma que fornece aulas de programação visual, como mostra na Figura 5, baseadas em blocos para crianças de 4 anos ou mais. São cursos completos de ensino de programação que, além de fomentar o raciocínio lógico, o pensamento computacional e ensinar conceitos de programação, permitem a construção de histórias e jogos 2D nas atividades gratuitas do curso, que podem ser compartilhadas entre os usuários.

Como uma instituição que visa divulgar e ensinar programação para pessoas de todas as idades. A Code.org visa fazer com que as pessoas, mais precisamente as crianças, se interessem pela programação de computadores. Auxiliado por vários tecnólogos influentes e com alguns cursos ministrados por grandes nomes da área, como Bill Gates e Mark Zuckerberg. Sendo apoiado por marcas de grande influência, incluindo Google, Facebook e a Microsoft.

Os irmãos Ali Partovi e Hadi lançaram o Code.org, em 2013, com um vídeo impulsionando a ideia. Após o sucesso do vídeo, mais de 10.000 escolas pediram ajuda aos irmãos para o auxílio da programação nas escolas. Necessitando o aumento da sua equipe de voluntários, construindo uma organização completa de apoio ao movimento global.

Acreditamos que uma educação de qualidade em ciência da computação deve estar disponível para todas as crianças, não apenas para alguns poucos privilegiados. A visão é que todos os alunos em todas as escolas tenham a oportunidade de estudar ciência da computação tanto quanto estudam biologia, química ou álgebra (Code.org, 2022. About Us. Disponível em: <https://code.org/about>. Acesso em: 24 de abril de 2023).

O Code.Org apresentam alguns objetivos, dentre eles se destacam:

- Aumentar a diversidade cultural e de gêneros na Ciência da Computação (CC);
- Inspirar estudantes;
- Criar cursos voltados para a CC;
- Levar a CC para as salas de aula;
- Formar professores;
- Mudar o currículo em distritos escolares;

- Auxiliar/sugerir na mudança de leis estaduais para inclusão da CC no currículo;
- Possibilitar que estudantes de todo o mundo tenham acesso ao material.

Figura 5 — Tela Inicial do Website CODE.ORG



Fonte: CODE.ORG, 2023.

2.3.4 Happy Code

A Happy Code é uma rede de franquias na área educacional fundada em 2015. Sendo referência global no ensino das ciências, tecnologia, engenharia e matemática. A instituição está presente em várias regiões do Brasil, Figura 6, como também em Angola, Espanha, Portugal e Estados Unidos. Com a metodologia de ensino baseada no raciocínio lógico, na criatividade, pensamento crítico e na comunicação.

Com a missão de formar futuros nativos digitais através do letramento digital. Baseada nos novos desafios do século XXI, o objetivo é que as crianças sejam capacitadas para lidar com a era digital, sem deixar de praticar os valores humanos. Uma criança nativa em tecnologia, de acordo com a proprietária da Happy Code Americana, Denise Zerbeto, é um ser humano pensante, capaz de desenvolver um projeto que envolve diferentes áreas do conhecimento.

Figura 6 — Sala de estudo da Happy Code na cidade de São Paulo



Fonte: MARCELO DARGELIO. *NB Notícias*, 2020.

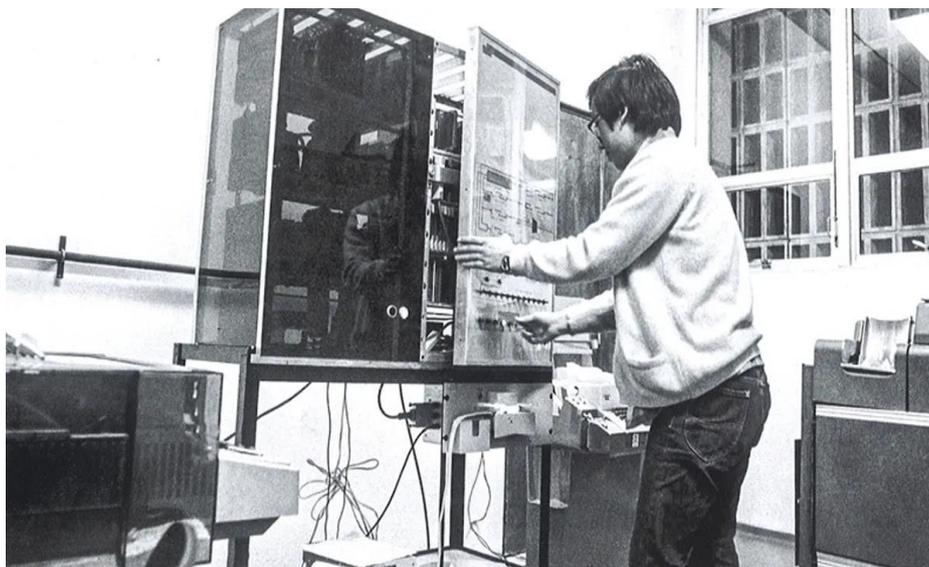
Nós da Happy Code escolhemos não criar uma metodologia própria, optamos em buscar nos EUA parcerias com instituições que se dedicam apenas no desenvolvimento de metodologias de ensino. Atualmente contamos com quatro plataformas de ensino desenvolvidas nos EUA. Destas parcerias, uma delas é com a empresa americana Make Wonder criadora dos robôs Dash e Dot ganhadores de vários prêmios de inovação e ensino nos EUA, onde somos a única empresa oficial e com exclusividade em distribuição no Brasil (Happy Code, 2017).

Romoaldo Rodrigues, um dos professores de programação da Happy Code, afirma que ao usar a diversão como ferramenta de conhecimento na instituição, a programação e seus conceitos ficam introduzidos nessa metodologia de ensino de modo que as crianças aprendam brincando, sem perceber. Quando os alunos menos percebem, já estão tomando decisões lógicas em seu dia a dia, graças a esse treino de raciocínio muito utilizado na aprendizagem de programação.

2.4 Avanço do ensino de programação no Brasil

Entre 1958 a 1975, a informática brasileira foi marcada pela importação de tecnologia de vários países, principalmente dos Estados Unidos, composto por computadores de grande porte, situados em universidades e grandes empresas.

Apenas em 1972, foi construído na USP o “Patinho Feio”, Figura 7, o primeiro computador nacional, no mesmo ano foi criado a Capre (Comissão de Coordenação das Atividades de Processamento Eletrônico), com o objetivo de propor uma política governamental de desenvolvimento do setor.

Figura 7 — Patinho Feio

Fonte: Kenji/AE, 1972.

No ano de 1976 o desenvolvimento da informática brasileira foi caracterizado pelo crescimento de uma indústria nacional. Iniciando a reestruturação da Capri e a criação de uma reserva de mercado na faixa de minicomputadores, para empresas nacionais, além da instituição do controle das importações.

Em 1984 foi sancionada a lei nº 7232, fixou a Política Nacional de Informática, que visava mudar o atraso tecnológico do Brasil frente aos outros países, o que possibilitou um crescimento de 30% ano da taxa de crescimento da informática e com a qual se oficializou a reserva para alguns segmentos do mercado, inclusive software, com duração limitada de oito anos.

Com o avanço da tecnologia no Brasil ocorre o aumento de pesquisadores que acreditavam no poder da programação no desenvolvimento das crianças, logo alguns projetos e iniciativas são realizadas em várias partes do país.

É claramente visível que a demanda de profissionais de computação é maior que a oferta, porém o ensino da computação na Educação Básica ainda não faz parte da realidade no Brasil ou no mundo. Para um país é extremamente estratégico formar bons profissionais em computação, pois é um ramo que está presente em todas as atividades econômicas e produtivas (BEZERRA; DIAS, 2014).

Para a Sociedade Brasileira de Computação (SBC), o conteúdo de ensino da educação básica deve ser realizado de forma universal, por isso são necessárias

mais discussões para mudar as Bases Curriculares Comuns Nacionais (BNCC), pois são mudanças sérias e vão refletir no futuro da educação brasileira. A SBC afirma que a computação deve ser ensinada desde o ensino fundamental como outras ciências para que a sociedade tenha cidadãos competentes (SBC, 2016). Ainda assim, conforme discutido na SBC, evidências científicas mostram que crianças e adolescentes que aprendem a resolver problemas computacionalmente melhoram seu desempenho em outros domínios. Existem inúmeros estudos e iniciativas no Brasil voltados para o ensino do pensamento computacional para escolares. A SBC acredita firmemente que a computação desencadeará uma nova revolução industrial e social nas próximas décadas, e os alunos devem estar preparados para isso. Portanto, é necessário implementar conceitos de computação na BNCC como um novo campo, como um componente do curso, como um tópico integrado abrangendo diferentes áreas, ou como unidades curriculares em diferentes áreas.

2.4.1 SuperGeeks

Em uma entrevista realizada ao portal de notícias Economia SP Drops, a CEO da SuperGeeks®, Cássia Akemi Ban enfatiza que a instituição é a primeira escola e primeira rede do segmento de Tecnologia, Inovação, Programação, Robótica, Ciência da Computação e Empreendedorismo para Kids e Teens do Brasil. A SuperGeeks foi idealizada e criada pelo casal Marco Giroto e Vanessa Ban, entre 2012 e 2013, no Vale do Silício, Califórnia, Estados Unidos. Baseando-se no desenvolvimento de aplicativos e na criação de jogos, na língua inglesa e no empreendedorismo e língua inglesa. A escola está se espalhando por todo o país, através de franquias e unidades próprias, totalizando 31 unidades em 2016.

Marco Giroto trabalhou em grandes empresas brasileiras e multinacionais, e criou sua primeira empresa de tecnologia aos 23 anos. Vanessa Ban lecionou para crianças da pré-escola ao ensino médio e especializou-se em língua, literatura e semiótica.

Com a união dos seus “superpoderes” criaram uma metodologia de ensino de Ciência da Computação infantojuvenil que levou mais de 1 ano de pesquisa e desenvolvimento, para ser concluída em um modelo ideal para ensinar Ciência da Computação para crianças e adolescentes. A SuperGeeks utiliza quatro metodologias de ensino:

- **Aprendizado de jogos:** Utilizando jogos para facilitar o aprendizado de conceitos de programação e ciência da computação;
- **Gamificação:** Usasse a mecânica do jogo, como pontos, recompensas e desafios de maneiras divertidas para envolver os alunos no aprendizado. Os alunos ganham ou perdem pontos como em um jogo;
- **Empreendedorismo:** A partir da terceira fase, os alunos serão incentivados, individualmente ou em grupos, a criar sua própria empresa startup e trazer seu produto para o mercado, seja um jogo, aplicativo, aplicativo web, hardware ou qualquer outro tipo de produto de tecnologia;
- **Contação de histórias:** insira histórias em animações, livros e histórias em quadrinhos para torná-las mais agradáveis para os alunos absorverem. Histórias de aventura são inseridas no conteúdo.

2.4.2 Programaê

Com o objetivo de guiar os professores, tendo ou não um conhecimento na área de informática, no ensino de programação para crianças, o projeto Programaê foi formulado, em setembro de 2014, pela Fundação Telefônica Vivo e a Fundação Lemann. Baseado no Code.org e no Scratch, o site apresenta cursos básicos que visam ensinar o básico da programação, como lógica, com cursos mais avançados que ensinam a desenvolver projetos na linguagem Scratch.

Todos os materiais criados para o "Programaê!" Aborda os temas de letramento digital, pensamento computacional, programação e robótica para professores e educadores de forma leve, divertida e perceptiva contínua no cotidiano das escolas públicas brasileiras. Esses workshops tentaram abordar duas questões: Computação desconectada e Computação conectada. Desta forma, realmente permite que todos experimentem a lógica de programação.

Em 2017, o programa desenvolveu estratégias para adotar a linguagem de programação e pensamento computacional nas práticas pedagógicas, auxiliando Redes de Ensino e escolas a desenvolverem um Currículo que valorizem o Pensamento Computacional sugerido na BNCC. Construindo o "Guia Programaê!: um guia para construção do pensamento computacional".

Durante o ano de 2018, foi iniciada a oficina nomeada de "Onde as girafas encostam o pescoço para dormir? Uma imersão no pensamento criativo e

computacional”. Com o objetivo de auxiliar o professor em uma reflexão sobre em que consiste a cultura digital, o pensamento criativo e o pensamento computacional e como a linguagem de programação pode ser integrada ao currículo. No decorrer da pandemia do COVID19 o Programaê! continuou com a criação dos cursos em EAD, onde atualmente já são compostos por cinco cursos disponíveis na plataforma das Escolas Conectadas. São eles:

- “Se meu computador pensasse” – Uma correlação entre a lógica computacional e os problemas do dia a dia – Pensamento Computacional;
- “Ensinando o computador: da lógica da programação para a lógica da aprendizagem” – Programação desplugada e em blocos;
- “Alô Mundo!” Lógica de Programação e autoria – Programação;
- “Eu Robô!” A Robótica Criativa e o engajamento social – Robótica Sustentável;
- “O que um gato pode ensinar para o computador?” Gamificação com Scratch – Programação em Blocos.

No projeto, foram desenvolvidos materiais sobre práticas pedagógicas orientadas por conteúdos e atividades de pensamento computacional, programação ligada e desligada, robótica e narrativas digitais oferecidas por professores de escolas públicas.

2.5 Pensamentos críticos do ensino de programação precoce

Vera Rita da Costa (2014), informa que os críticos dessa prática acreditam que o incentivo ao ensino na programação nas escolas pode atrapalhar as experiências comuns das crianças e adolescentes, do convívio social e do seu desenvolvimento. O argumento de alguns especialistas da área de Tecnologia da Informação é o mesmo apresentado por Torvalds. Para eles, a programação de computadores é algo que requer raciocínio lógico apurado e capacidade de resolver problemas com alto grau de complexidade, características que nem todas as pessoas possuem.

A preocupação na falta de desenvolvimento no convívio social em decorrência ao uso da computação precocemente é algo que incomoda alguns pesquisadores. Questionam como vão aprender a se comportar e conviver socialmente e reconhecer

seus 'limites sociais' se não terão isso estimulado desde cedo, em brincadeiras e jogos espontâneos e interação com outras crianças, muito mais do que com os adultos.

Vera Rita (2014), afirma que as crianças já se encontram privadas em suas casas do convívio social ideal com outras crianças e até com adultos, e encontram-se vivendo em um contexto em que a escola se tornou praticamente o único espaço para a brincadeira e a socialização. Mesmo que se argumente que isso será feito de forma lúdica e adaptada à idade, precisamos pensar se isso se justifica tirarmos delas mais desse tempo e espaço, introduzindo em seu lugar mais atividades intelectuais ou virtuais.

Da parte da psicologia cognitiva também se tem uma contribuição interessante para esse debate. Com base em resultados de pesquisa, sabemos que a forma de pensamento da criança mais nova é diferente da do adolescente. Diante de todo esse conhecimento acumulado pela psicologia do desenvolvimento, é de se questionar, portanto, até que ponto introduzir o ensino de programação computacional tão cedo fará realmente a educação avançar ou novamente empacar, levando-nos a embarcar em mais um modismo sem fundamentação científica.

Creemos que a introdução precoce de computadores no ensino faz parte dessa lamentável aceleração, não se deixando crianças e jovens comportarem-se de maneira infantil e juvenil. Estamos criando senis precoces, com mentalidade de adultos. Será que o raciocínio algorítmico, lógico-simbólico seco, morto, alienado da realidade, imposto pelos computadores em crianças e jovens, não prejudica a criatividade dos futuros adultos no âmbito profissional e social, onde nada há de bem definido, ao contrário do espaço apresentado por aquelas máquinas? Será que com isso esses pobres seres não terão muito mais dificuldade para reverter a tendência antissocial de nosso mundo moderno? Talvez eles venham a ser ainda mais antissociais, pois terão aprendido a admirar as máquinas numa idade em que não podiam encará-las objetivamente. Não terão aprendido que atrás de cada ser humano existe uma individualidade sagrada, que deve ser respeitada e venerada, e não os circuitos de uma máquina que aparentemente pensa. Sim, cremos que a mais terrível consequência do uso precoce de computadores na educação é fazer com que os futuros adultos adquiram uma veneração pelas máquinas, passando a crer que elas sejam superiores aos seres humanos e estas sejam máquinas... imperfeitas! As atrocidades dos nazistas e as que estamos presenciando na América, na África e na Europa serão insignificantes perto do que advirá, pois não há ética no trato com uma máquina, tudo será permitido e tolerado (V.W. SETZER, 2002, p. 85-134)

Kernel do sistema operacional Linux, afirma que não acredita que todos devam necessariamente tentar aprender a codificar. Para ele, acho que é algo razoavelmente especializado, e ninguém realmente espera que a maioria das pessoas tenha que fazer isso. Não é como saber ler e escrever e fazer contas básicas. As pessoas deveriam ter alguma maneira de se expor a isso, apenas para que os que acham que

gostam e têm aptidão saibam sobre a possibilidade. Não porque todo mundo vai querer ou precisar aprender, mas apenas porque é uma grande vocação, e pode haver muitas pessoas que nunca perceberam que podem realmente gostar de dizer aos computadores o que fazer.

Segundo Setzer (1988), dar um computador ou outro tipo de dispositivo eletrônico é uma forma de fazer uma criança perder a sua infantilidade. Fazendo-o perder sua juventude e o tornando “velho” tendo em vista o mental e o emocional. O computador desenvolve uma evolução mental precoce.

Valdemar W. Setzer (2002), do departamento de Ciências da Computação do Instituto de Matemática e Estatística da USP afirma que “qualquer uso de computadores antes do ensino médio, isto é, mais ou menos aos quinze anos, é prejudicial à criança ou jovem”. O cientista também declara que “ao usar um computador, a criança é obrigada a exercer um tipo de pensamento que deveria empregar somente em idade bem mais avançada. Com isso podemos dizer que os computadores roubam das crianças sua necessária infantilidade. Elas são obrigadas a pensar e usar uma linguagem que deveria ser dominada exclusivamente por adultos” (2002).

Waldemar Setzer (2002) em defesa do ensino computacional após a adolescência. Sezer afirma que o computador obriga a criança a usar uma linguagem formal e, portanto, é absolutamente estranho, totalmente morto, desprovido de vida ou emoção. Exige um raciocínio lógico que a criança não deveria ter. Exige um raciocínio algorítmico, que as crianças não devem praticar de forma formal. Waldemar acredita que a prática exige uma experiência de causas e efeitos exatos, que não existe natureza, experimentar causalidade exata, não existe natureza, torna a máquina misteriosa porque a criança não sabe como ela funciona: torna-se um grande mistério.

Para Setzer (2002), o incentivo à tecnologia e o mundo computacional rouba as energias e forças que deveriam ser dedicadas a seu desenvolvimento normal. Afirmando como um instrumento “antiarte”, pois elimina completamente o elemento subjetivo e inconsciente que toda obra de arte deveria ter, obrigando o “artista” a formalizar completamente seus sentimentos e pensamentos.

3 OBJETIVOS

Objetivo geral

- O objetivo geral do presente trabalho consiste em apresentar a programação computacional como uma ferramenta de ensino na infância, através de estudos e análises.

Objetivos específicos

- Analisar a relevância dos estímulos realizados desde a infância;
- Identificar os benefícios do ensino da programação na infância;
- Relatar a relevância do Pensamento computacional na infância;
- Especificar softwares direcionados ao ensino de programação para crianças;
- Expor opiniões em oposição ao incentivo do ensino da programação precoce;
- Apresentar o desenvolvimento da programação para crianças no Brasil.

4 METODOLOGIA

Com o intuito de obter uma quantidade considerável de informações e dados acerca do ensino da programação na infância e todas as suas diretrizes até a atualidade, foi feito um levantamento bibliográfico. A pesquisa bibliográfica é realizada com base em material já elaborado (principalmente livros e artigos científicos). Embora quase todas as pesquisas envolvam algum tipo de trabalho dessa natureza, também existem estudos que são desenvolvidos inteiramente a partir de fontes bibliográficas. Uma grande parte da pesquisa exploratória pode ser definida como pesquisa bibliográfica. Os estudos de ideologia, e aqueles que apresentam análises de diferentes posições sobre as questões, também costumam ser conduzidos quase que exclusivamente por meio de fontes bibliográficas (GIL, 2002). Neste trabalho foi realizada uma pesquisa referente ao ensino da programação computacional na infância, sobre os meios utilizados para a realização dessa prática e dados que comprovem seus benefícios e possíveis dificuldades para a realização desse ensino, com o intuito de avaliar e analisar a prática e o tema.

Ao realizar uma pesquisa científica, é importante que o pesquisador tenha clareza sobre o problema a ser estudado, tenha curiosidade e paciência para aprofundar o entendimento da disciplina exigida e saiba quais recursos financeiros, materiais e humanos são necessários para a sua conclusão. a pesquisa. objetivo desejado. É prudente entender e se aprofundar nos procedimentos técnicos necessários para a realização de pesquisas (GIL, 2002).

As revisões de literatura são divididas em diferentes tipos, cada uma delas possui critérios de coleta e análise distintos. Em referência às revisões sistemáticas, pode-se dizer que estas são caracterizadas por sua rigorosidade durante a busca e a análise dos dados. O objetivo é “levantar, reunir, avaliar criticamente a metodologia da pesquisa e sintetizar os resultados de diversos estudos primários” (CORDEIRO, OLIVEIRA, RENTERÍA, 2007, p. 429). Esse tipo de revisão foca em questões específicas e bem definidas e é muito utilizado para tomadas de decisões em gestões públicas, práticas clínicas e em evidências científicas (GALVÃO e PEREIRA, 2014). A revisão bibliométrica está relacionada aos métodos quantitativos das pesquisas científicas, ou seja, é um método voltado para as análises dos dados estatísticos. Por meio da revisão bibliométrica os pesquisadores são capazes de mensurar as contribuições do conhecimento que vem sendo realizado nas pesquisas nos diferentes

campos de aprendizado, e pode ser utilizada na representação e identificação de temáticas para novas pesquisas, a partir de uma análise estatística descritiva (SANTOS, 2015).

Para o desenvolvimento deste trabalho foi utilizado a revisão narrativa para abordar o assunto de forma mais ampla, sem muita especificidade, são basicamente resenhas de revistas, livros, artigos baseados na interpretação do autor, ou seja, é uma abordagem subjetiva que pode se basear na experiência do leitor varia. Dessa forma, as resenhas narrativas não buscam dados segundo critérios rígidos, cuja análise pode sofrer interferência dos intérpretes da obra. Costuma abordar assuntos mais amplos, sendo muito comum em fundamentações teóricas de trabalhos de conclusão de curso, teses e dissertações (CORDEIRO, OLIVEIRA e RENTERÍA, 2007; MATTOS, 2015). As revisões narrativas são classificadas como uma análise de literatura que irá fornecer sínteses narrativas e compreensivas das informações que já foram publicadas. Esse método de pesquisa se constitui em um instrumento de ensino, e é muito útil devido a construção e sistematização das informações, sendo muito utilizada para a discussão e descrição de diferentes assuntos e em diferentes campos de conhecimento. Contudo, esse tipo de revisão não deixa explícito quais são as fontes de informações utilizadas, nem de como foi realizado o método de busca para chegar nos resultados (RIBEIRO, 2014).

O tema de uma pesquisa é o assunto que se deseja provar ou desenvolver; "é uma dificuldade, ainda sem solução, que é mister determinar com precisão, para intentar, em seguida, seu exame, avaliação crítica e solução" (Asti Vera, 1976:97). Determinar com precisão significa enunciar um problema, isto é, determinar o objetivo central da indagação. Assim, enquanto o tema de uma pesquisa é uma proposição até certo ponto abrangente, a formulação do problema é mais específica: indica exatamente qual a dificuldade que se pretende resolver (Marconi, 2003, p.126).

A pesquisa é desenvolvida mediante o concurso dos conhecimentos disponíveis e a utilização cuidadosa de métodos, técnicas e outros procedimentos científicos. Na realidade, a pesquisa desenvolve-se ao longo de um processo que envolve inúmeras fases, desde a adequada formulação do problema até a satisfatória apresentação dos resultados (GIL, 2002).

Durante os meses de novembro de 2022 e maio de 2023. As buscas foram realizadas no Google Acadêmico®, utilizando como palavra-chave: Ensino de programação na infância, pensamento computacional na infância, benefícios da programação na infância e opiniões sobre o ensino da programação computacional.

Os artigos utilizados disponíveis na íntegra nas bases de dados tiveram critérios de inclusão: materiais disponíveis no idioma português e inglês. Os livros utilizados foram adquiridos por meio de download de forma gratuita. Artigos e notícias foram desfrutados, como também vídeos científicos e palestras disponibilizados de forma gratuita no Youtube.

O trabalho apresenta uma abordagem qualitativa com base em critérios de abstração, complexidade e relevância prática. A análise qualitativa depende de muitos fatores, tais como a natureza dos dados coletados, a extensão da amostra, os instrumentos de pesquisa e os pressupostos teóricos que nortearam a investigação. Pode-se, no entanto, definir esse processo como uma sequência de atividades, que envolve a redução dos dados, a categorização desses dados, sua interpretação e a redação do relatório (GIL, 2002, p. 133).

Desenvolveu-se uma revisão bibliográfica sobre o ensino da programação para crianças, com o foco em permitir ao pesquisador compreender com maior amplitude os fenômenos e seus mecanismos. A pesquisa bibliográfica é desenvolvida com base em material já elaborado, constituído principalmente de livros e artigos científicos. Os livros de referência, também denominados livros de consulta, são aqueles que têm por objetivo possibilitar a rápida obtenção das informações requeridas, ou, então, a localização das obras que as contêm (GIL, 2002). Embora em quase todos os estudos seja exigido algum tipo de trabalho dessa natureza, há pesquisas desenvolvidas exclusivamente a partir de fontes bibliográficas (GIL, 2002).

A principal vantagem da pesquisa bibliográfica reside no fato de permitir ao investigador a cobertura de uma gama de fenômenos muito mais ampla do que aquela que poderia pesquisar diretamente. Essa vantagem torna-se particularmente importante quando o problema de pesquisa requer dados muito dispersos pelo espaço (GIL, 2002).

O trabalho apresenta uma pesquisa exploratória, apresentando uma visão do ensino da programação computacional nos primeiros anos de vida, identificando suas principais características, realizando coleta de dados de diferentes fontes, como entrevistas e análise documental. Uma procura de tais fontes, documentais ou bibliográficas, torna-se imprescindível para a não duplicação de esforços, a não "descoberta" de ideias já expressas, a não-inclusão de "lugares-comuns" no trabalho (LAKATOS, 2003). Pesquisa documental, fonte de coleta de dados está restrita a documentos, escritos ou não, constituindo o que se denomina de fontes primárias.

Estas podem ser feitas no momento em que o fato ou fenômeno ocorre, ou depois (LAKATOS, 2003).

5 RESULTADOS E DISCUSSÕES

A relevância do estímulo de determinadas atividades logo nos primeiros anos de vida de um indivíduo pode afetar diretamente o seu desenvolvimento. Incentivos na infância é de extrema importância na evolução emocional, física, social e cognitiva da criança. Nos primeiros anos de vida, o cérebro da criança passa por um período de desenvolvimento em que ocorrem mudanças significativas na estrutura e função do cérebro. É nesse período em que as experiências vividas pela criança terão um impacto significativo e duradouro no seu desenvolvimento. Essa fase é caracterizada pela sensibilidade das influências ambientais do cérebro. Os estímulos podem ser de diferentes tipos, incluindo intelectuais, em destaque o incentivo ao pensamento computacional, que vem crescendo com o passar do tempo devido às suas vantagens.

Com isso, é fundamental que pais, responsáveis e educadores proporcionem um ambiente rico em estímulos para as crianças durante a infância, de forma a garantir um desenvolvimento saudável e bem-sucedido. Possibilitando uma variedade de estímulos positivos para as crianças em seus primeiros anos de vida.

Por outro lado, a carência de estímulos adequados durante a infância pode causar consequências negativas no desenvolvimento da criança, incluindo atrasos cognitivos e emocionais, problemas de comportamento e dificuldades de aprendizagem. Como também, é de suma importância uma atenção ao que será estimulado na infância, levando em conta que algumas atividades podem prejudicar a evolução saudável da criança.

O Pensamento Computacional é uma distinta capacidade criativa, crítica e estratégica humana de saber utilizar os fundamentos da Computação, nas mais diversas áreas do conhecimento, com a finalidade de identificar e resolver problemas, de maneira individual ou colaborativa, através de passos claros, de tal forma que uma pessoa ou uma máquina possam executá-los eficazmente (BRACKMANN, 2017, p. 29).

O pensamento computacional é formado por quatro pilares: decomposição, reconhecimento de padrões, abstração e algoritmos. Cada um desses pilares auxilia no desenvolvimento da criança e podem ser introduzidos de maneiras distintas. A decomposição envolve a habilidade de quebrar um problema complexo em partes menores, sendo uma habilidade importante em muitas áreas da vida. O reconhecimento de padrões auxilia a identificar semelhanças e diferenças em um

conjunto de informações, sendo importante na análise de dados e na tomada de decisões.

A abstração é a capacidade de generalizar a partir de exemplos, onde essa generalização pode ser usada para resolver outros problemas. O algoritmo é a capacidade de criar um conjunto de passos lógico para resolver um problema, é fundamental para a programação, pois o algoritmo é usado para traduzir uma solução para um problema de um programa de computador funcional.

O diferencial na introdução da lógica de programação para crianças é abordar de uma forma leve, divertida e criativa, ajustando o ensino de acordo com a sua idade e o seu grau de compreensão, por exemplo, a utilização de jogos que envolvam resolver problemas lógicos. O Scratch é um espaço de programação que tem blocos como base, uma maneira popular de ensinar lógica de programação para crianças, pela sua forma mais intuitiva de começar a entender a lógica de programação.

O ensino de programação computacional na infância é marcado pelas suas vantagens, como listado no Quadro 1, entre as mais evidentes temos o desenvolvimento de habilidades na resolução de problemas, já que a programação envolve a habilidade de identificar problemas e encontrar soluções para eles. Estimulando as crianças a aprenderem e pensar de maneira lógica e sistemática. O estímulo à criatividade, já que a programação possibilita a criação de algo novo. A familiaridade com a tecnologia, tendo em vista o seu avanço e sua importância na atualidade. Além da autoconfiança, desenvolvimento na habilidade de colaboração e o estímulo ao pensamento crítico. O uso de softwares e plataformas, como listado no Quadro 2, para introduzir as crianças ao mundo da programação computacional tem sido um dos meios mais utilizados para o seu ensino.

Quadro 1 - Pontos positivos e pontos negativos do ensino da programação computacional na infância

Pontos Positivos	Pontos Negativos
Pensamento lógico e resolução de problemas Criatividade e pensamento crítico Habilidades matemáticas e de raciocínio abstrato Colaboração e trabalho em equipe Autonomia e iniciativa	Sobrecarga cognitiva A falta de diversidade na indústria de tecnologia Dependência da tecnologia Falta de habilidades sociais

Fonte: Elaborado pelo autor, 2023.

Quadro 2 - Plataformas de ensino de programação computacional para crianças

Plataforma	Ano	Autor
Bee-Bot	2002	TTS Group Ltd
Scratch	2007	Lifelong Kindergarten Group
Code.Org	2013	Hadi Partovi e Ali Partovi
CodeSpark Academy	2013	Grant Hosford, Joe Shochet e Surya Vanka
CodaKid	2014	David Dodge

Fonte: Elaborado pelo autor, 2023.

Embora haja várias vantagens, alguns pontos devem ser considerados no que se refere ao ensino da programação computacional ainda na infância. Estimular o seu ensino sem o cuidado e atenção necessária pode gerar sobrecarga cognitiva, aprender a programar pode ser desafiador e exigir muita atenção e concentração. Podendo sobrecarregar as crianças e causar estresse, especialmente se elas não estiverem preparadas para lidar com isso. Alguns conseguirão absorver melhor os conceitos, pois nem todos possuem a habilidade natural para a programação, sendo necessário respeitar a individualidade de cada um. Algumas crianças podem demonstrar total aptidão pela programação e outros podem apresentar rejeição e isso precisa ser muito bem observado para intervir em determinados casos. Cada ser humano tem o seu limite e as suas preferências, então impor a programação não deve ser algo considerado. Silvana Bahia, diretora de projetos do Olabi e coordenadora do PretaLab aborda a relevância da questão tecnológica para a população negra, especialmente mulheres, afirma:

As tecnologias estão carregadas com as visões políticas, econômicas e culturais de quem as cria – e esse poder hoje está centrado nas mãos de homens, brancos, heterossexuais, classe média/ricos. Isso já potencializa uma grande desigualdade, em um mundo cada vez mais digital (PRETALAB, 2020).

A falta de diversidade na indústria de tecnologia é outro fator relevante, devido a área de TI ainda ser dominada por homens brancos. Isso pode fazer com que crianças de grupos minoritários se sintam desencorajadas a aprender programação, já que não se veem representadas no setor. Além disso, as crianças podem apresentar uma maior dependência de tecnologia: Se as crianças forem expostas

apenas à programação computacional, elas podem acabar se tornando excessivamente dependentes da tecnologia. Sendo crucial ensiná-las a equilibrar o uso da tecnologia com outras atividades saudáveis e enriquecedoras.

As crianças podem apresentar uma falta de habilidades sociais, já que a programação pode ser uma atividade solitária, criando uma maior dependência de tecnologia, se as crianças forem expostas apenas à programação computacional, elas podem acabar se tornando excessivamente dependentes da tecnologia. É importante ensiná-las a equilibrar o uso da tecnologia com outras atividades saudáveis e enriquecedoras para que não ocorra um distanciamento do mundo real, das pessoas e de atividades que deveriam ser realizadas na infância. Sendo assim, é importante ensiná-las a desenvolver habilidades sociais e trabalhar em equipe, para que possam se comunicar e colaborar com outras pessoas de forma eficaz.

"A falta de acesso à internet é como estar em um mundo sem luz, onde as oportunidades são obscurecidas e a igualdade é negada." (Touré, 2007). A romantização da facilidade ao acesso a internet na sociedade ainda é algo recorrente, mesmo com a popularização da tecnologia e dos eletrônicos é necessário lembrar que nem todos têm acesso a computadores e à internet. A falta de acesso à internet nas escolas é um problema significativo em muitas partes do mundo. De acordo com a UNESCO, em 2020, cerca de 3,7 bilhões de pessoas no mundo ainda não tinham acesso à internet, e muitas escolas em áreas rurais ou de baixa renda não têm acesso à internet de alta velocidade ou mesmo conexões confiáveis.

Isso pode ter um impacto significativo na educação das crianças, especialmente quando se trata de ensino de programação e tecnologia. Sem acesso à internet, as crianças podem não ter acesso a recursos de programação online ou ferramentas de desenvolvimento, o que pode limitar sua capacidade de aprender e praticar programação. A falta de acesso à Internet nas escolas pode ser atribuída a vários fatores, incluindo falta de infraestrutura, falta de investimento em tecnologia educacional e disparidades socioeconômicas. Resolver esse problema exigirá que governos, escolas e organizações sem fins lucrativos trabalhem juntos para fornecer acesso à Internet e tecnologia educacional a todas as crianças, independentemente de onde vivam ou de sua condição socioeconômica.

A carência na acessibilidade da internet para toda a população encadeia-se na falta de acesso ao ensino de programação na infância, sendo um problema significativo em muitas partes do mundo, especialmente em áreas de baixa renda e

em países em desenvolvimento. A falta de recursos, como computadores e acesso à internet, bem como a falta de professores capacitados em programação, pode limitar a capacidade das crianças de aprender habilidades tecnológicas importantes.

Isso pode ter consequências a longo prazo para o futuro dessas crianças, já que habilidades em programação são cada vez mais importantes em muitas áreas profissionais. Além disso, a falta de diversidade na indústria de tecnologia pode ser exacerbada pela falta de acesso ao ensino de programação para crianças de minorias e grupos sub-representados. Para enfrentar esse problema, é necessário investir em recursos e programas de educação em programação para crianças em idade escolar, bem como garantir que esses recursos estejam disponíveis para todas as crianças, independentemente de sua localização geográfica ou situação socioeconômica. Também é importante trabalhar para aumentar a diversidade na indústria de tecnologia, fornecendo oportunidades iguais para todos aqueles que desejam aprender e trabalhar nessa área.

O ensino da programação computacional na infância é alvo de críticas devido ao incentivo do acesso a computadores e eletrônicos precocemente.

Embora os computadores e a tecnologia tenham trazido muitos benefícios para a sociedade moderna, há preocupações significativas sobre o impacto negativo que o uso excessivo desses dispositivos pode ter no desenvolvimento físico, cognitivo e social das crianças (Adaptado de Council on Communications and Media, 2013, p. 7).

Uma maneira de lidar com essa questão é com a implementação da programação desplugada, uma abordagem de ensino de programação que não envolve o uso de um computador ou dispositivo eletrônico. Utilizando atividades práticas e lúdicas para ensinar conceitos de programação de forma mais visual e interativa. Essas atividades podem incluir jogos, quebra-cabeças, desafios matemáticos, cartas, entre outros.

A programação desplugada é uma forma eficaz de introduzir conceitos de programação para crianças e jovens que ainda não têm acesso a computadores ou dispositivos eletrônicos, ou para aqueles que têm dificuldades em aprender através de métodos tradicionais. Além disso, a programação desplugada pode ser usada como um complemento ao ensino tradicional de programação, ajudando os alunos a entenderem os conceitos básicos antes de passarem para a programação em computadores. Atuando como um meio para diminuir o acesso das crianças a

computadores e eletrônicos enquanto aprende e se aprofunda no mundo da computação.

6 CONSIDERAÇÕES FINAIS

O rápido avanço da tecnologia tem despertado interesse no ensino de programação desde a infância, devido à automação generalizada e influência em diversos setores da sociedade. Essa prática demonstra benefícios significativos para o desenvolvimento das crianças em várias áreas de suas vidas. Esta pesquisa explora esses benefícios e seu impacto direto na infância. Além de abordar possíveis problemas que o estímulo do ensino da programação na infância sem o devido acompanhamento pode gerar.

Foi observado a importância de alguns estímulos nos primeiros anos de vida, como também os estágios do desenvolvimento da criança e como cada indivíduo passa por todas essas etapas com uma variação de tempo em cada estágio. Abordando o ensino da lógica de programação e do pensamento computacional, os meios de ensino até opiniões críticas sobre esse intensivo precoce. A revisão apresentada no texto abordou suas vantagens, exibindo seus benefícios e relatos de estudiosos do tema, também foi relatado possíveis problemas que a computação pode gerar sem um acompanhamento de profissionais e pessoas preparadas para abordar o seu ensino.

O trabalho relata a relevância do Pensamento Computacional na infância e como ele está presente em várias áreas do conhecimento. Foi citado e especificado diferentes softwares direcionados ao ensino de programação para crianças. Além de apresentar e detalhar o desenvolvimento da programação computacional para crianças no Brasil, tais como seus problemas na implementação e desafios.

Logo, esta pesquisa não traz um posicionamento definitivo para o estímulo da programação na infância e o uso dessa prática como algo para todos, mas sim dados e informações pertinentes sobre o tema, como também opiniões de profissionais que incentivam essa prática ou se opõem a esse estímulo logo nos primeiros anos de vida.

De acordo com esse trabalho, este estudo terá continuidade, no qual sugiro a realização das seguintes etapas:

- Aprofundar os estudos sobre a programação e como ela pode afetar na vida de cada indivíduo.
- Pesquisar sobre a possibilidade de inclusão que a programação pode causar na sociedade e no ensino.

- Analisar projetos de programação na infância, como relatório das atividades praticadas.

REFERÊNCIAS BIBLIOGRÁFICAS

ARAÚJO, M. A. S. **Ensino de programação para crianças: uma análise da situação atual**. Monografia (Licenciatura em computação) - Instituto Federal de Educação, Ciência e Tecnologia do Sertão Pernambucano, Campus Petrolina, Petrolina - PE, 2014.

ATLASSIAN. **Atlassian** - Software Development and Collaboration Tools. Disponível em: <https://www.atlassian.com/>. Acesso em: 25 maio 2023.

BEZERRA, Fábio; DIAS, Klissiomara. Programação de Computadores no Ensino Fundamental : Experiências com Logo e Scratch em Escola Pública. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 22. , 2014, Brasília. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2014. p. 229-238. ISSN 2595-6175.

BLIKSTEIN, P. (2008). **O pensamento computacional e a reinvenção do computador na educação**. Disponível em: <http://bit.ly/1IXIbNn>. Acesso em: 20 fev. 2023.

BRACKMANN, P. C. **Desenvolvimento do Pensamento Computacional Através de Atividades Desplugadas na Educação Básica**. Santa Maria, 2017. 226 p. Tese (Pós-Graduação em Informática na Educação) - Universidade Federal do Rio Grande do Sul.

CARLOS, L.; GODINHO, J.; GOMIDE, J. (2018). **Um relato de experiência da escola de verão de programação para crianças**. Workshop de Informática na Escola, volume 24, p 41.

CASTRO, Adriane de. **O uso da programação Scratch para o desenvolvimento de habilidades em crianças do ensino fundamental**. 2017. 124 f. Dissertação (Mestrado em Ensino de Ciência e Tecnologia) - Universidade Tecnológica Federal do Paraná, Ponta Grossa, 2017.

CARVALHO, Rafael. Por que aprender programação é tão crucial quanto saber ler. **Exame**, 21 ago. 2015. Disponível em: <https://exame.com/carreira/por-que-aprender-programacao-e-tao-crucial-quanto-saber-ler/>. Acesso em: 18 jan. 2023.

CODE.ORG. Disponível em: <https://code.org/>. Acesso em: 28 jan. 2023.

COSTA, Vera Rita. Computação para os pequenos. **Observatório da Imprensa**, São Paulo, 3 jun. 2014 Disponível em: https://www.observatoriodaimprensa.com.br/e-noticias/_ed801_computacao_para_os_pequenos/. Acesso em: 30 dez. 2022.

CORDEIRO, A. M.; OLIVEIRA, G.M.; RENTERÍA, J.M. Revisão sistemática: uma revisão narrativa. **Rev. Col. Bras. Cir**, Rio de Janeiro, v. 34, n. 6, p. 428-431, 2007.

CUNHA, Alberto. PROGRAMAÇÃO DESPLUGADA! - CONHEÇA ESSA PRÁTICA SEM O USO DE APARELHOS ELETRÔNICOS!. Publicado pelo canal Aprendiz 21. 2020. Disponível em: https://www.youtube.com/watch?v=Zd_ZOk_InYo. Acesso em: 30 jan. 2023.

Dijkstra, E. W.. **A Discipline of Programming**. Prentice-Hall, 1976.

Eduardo O.C. Chaves; V.W. Setzer, **O Uso de Computadores em Escolas: Fundamentos e Críticas**, São Paulo: Ed. Scipione, 1988, pp. 69-127.

ENGLE, Patrice; LUCAS, Jane E. **Cuidados para o desenvolvimento da criança(CDC): manual de orientação às famílias**.2012.Disponível em: https://www.mds.gov.br/webarquivos/publicacao/crianca_feliz/Cuidados_para_desenvolvimento_crianca.pdf. Acesso em: 03 maio 2023.

FERNANDES, K. R. **Ensino de Computação com SCRATCH no Ensino Fundamental – Um Estudo de Caso**. Revista Brasileira de Informática na Educação, Porto Alegre, v. 22, n. 2, p. 117-132, jul./dez. 2014.

FERRARI, Carlos Gilberto Melchior Rodrigues Sansalone. O surgimento da informática e sua chegada ao Brasil. **Portal e Educação**. 2013. Disponível em: <http://www.portaleducacao.com.br/iniciacao-profissional/artigos/47410/osurgimento-da-informatica-e-sua-chegada-ao-brasil>. Acesso em: 04 nov. 2022.

Fly Educação e Cultura. 29 nov 2021. Diversidade no mundo da tecnologia: Um mercado de trabalho mais inclusivo. **Fly e Educação**. Disponível em: <https://www.flyeducacao.org/post/diversidade-no-mundo-da-tecnologia-um-mercado-de-trabalho-mais-inclusivo>. Acesso em: 15 mar. 2023.

FUNDAÇÃO TELEFÔNICA VIVO. Programaê!. 2017. **Fundação Telefônica Vivo**. Disponível em: <https://www.fundacaotelefonicavivo.org.br/programae/>. Acesso em: 10 fev. 2023.

GALVÃO, T. F. PEREIRA, M. G. Revisões sistemáticas da literatura: passos para sua elaboração. **Revisão Sistemática**, n. 23, v.1, p. 183-184, 2014.

GARLET, D.; BIGOLIN, N., M.; SILVEIRA, S., R. Ensino de Programação de Computadores na Educação Básica: um estudo de caso. **Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica**. v. 9, n. 2. 2018. Disponível em: <http://periodicos.unifacef.com.br/index.php/resiget/article/view/1604>. Acesso em: 5 fev. 2023.

GERALDES, W. B. **Programar é bom para as crianças? Uma visão crítica sobre o ensino de programação nas escolas**. Texto Livre, Belo Horizonte-MG, v. 7, n. 2, p. 105–117, 2014. DOI: 10.17851/1983-3652.7.2.105-117. Disponível em: <https://periodicos.ufmg.br/index.php/textolivre/article/view/16677>. Acesso em: 29 jun. 2023.

GIL, A. C. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas, 2002.

GIL, A. C. **Métodos e técnicas de pesquisa social**. 6. ed. São Paulo: Atlas, 2008.

GODOY, Leopoldo. Sistema operacional para PCs vai se tornar irrelevante, diz criador do Linux. **G1**. São Paulo, 02 set 2010. Disponível em: <https://g1.globo.com/tecnologia/noticia/2010/09/sistema-operacional-para-pcs-vai-se-tornar-irrelevante-diz-criador-do-linux.html>. Acesso em: 20 jan. 2023.

GOMES, Patrícia. Codecademy ensina programação de graça. **Porvir**. Disponível em: <http://porvir.org/porcriar/codecademy-ensina-programacao-de-graca-pelo-mundo/20120828>. Acesso em: 31 dez. 2022.

HAPPY CODE. (2023). **Happy Code**. Disponível em <https://www.happycodeschool.com/>. Acesso em: 10 jan. 2023.

Instituto Brasileiro de Geografia e Estatística (IBGE). Disponível em: <https://www.ibge.gov.br>. Acesso em: 25 abr. 2023.

KAFAI, Y. B; BURKE, Q. (2013). **Computer Programming Goes Back to School**. Education Week.

MARCONI, Marina de Andrade; LAKATOS, Eva Maria. **Fundamentos de metodologia científica**. 5. ed. São Paulo: Atlas, 2003.

LARIVÉE, S. & MICHAUD, N. (1980). L'ordinateur au secours de l'inadaptation, **Revue des Sciences de l'Éducation**, 6(3): pp. 451-472..

LIMA, B. N. M.; Figueiredo, R. M. C. **INICIATIVAS PARA ENSINO DE PROGRAMAÇÃO PARA CRIANÇAS: UMA REVISÃO SISTEMÁTICA**. 2016. Universidade de Brasília, Faculdade Gama, Distrito Federal, Brasil. Disponível em: <https://www.tecsi.org/contecsi/index.php/contecsi/13CONTECSI/paper/viewFile/4123/2650>. Acesso em: 05 dez. 2022.

LOSSO, Renta. Crianças X computadores: benefícios e males da era tecnológica. 16 ago. 2010. **IG**. Disponível em: <http://delas.ig.com.br/filhos/criancas-x-computadoresbeneficios-e-males-da-era-tecnologica/n1237749844018.html>. Acesso em: 08 abr 2023. Especial para o iG São Paulo.

LU, J. G.; FLETCHER, R. J. G. **The ecology of birds**. Cambridge: Cambridge University Press, 2009.

MARCIANO, J. J. S; MENDES, L. A. M. **Migração de Sistemas Computacionais para o Paradigma SOA**. Tese (Graduação em Ciência da Computação) - Universidade Presidente Antônio Carlos (UNIPAC), Barbacena. Disponível em: <https://ri.unipac.br/repositorio/wp-content/uploads/2019/07/Jim-Jones-da-Silveira-Marciano.pdf>. Acesso em: 10 dez. 2022.

MARCONI, M. A.; LAKATOS, E. M. **Fundamentos de metodologia científica**. 5. ed. São Paulo: Atlas, 2003.

MATTOS, P. C. **Tipos de Revisão de Literatura**. 2015, Disponível em: <https://www.fca.unesp.br/Home/Biblioteca/tipos-de-evisao-de-literatura.pdf>. Acesso em: 24 fev. 2023.

MIRANDA, G L. (1990). **Linguagem LOGO**. Disponível em: <http://hdl.handle.net/10400.12/2860>. Acesso em: 15 mar. 2023.

NOGUEIRA, Makeliny Oliveira Gomes; LEAL, Daniela. **Teoria da aprendizagem: um encontro entre os pensamentos filosófico, pedagógico e psicológico**. 2. ed. Curitiba: Intersaberes, 2015.

OLIVEIRA, Milena; SOUZA, Anderson; FERREIRA, Aline; BARREIROS, Emanoel. **Ensino de lógica de programação no ensino fundamental utilizando o Scratch: um relato de experiência**. WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 2014, Brasília. Porto Alegre: Sociedade Brasileira de Computação, 2014 . p. 239-248.

OLIVEIRA, S.B.C. et al. **PROGRAMAÊ! UMA PROPOSTA DE ENSINO DE LÓGICA DE PROGRAMAÇÃO NAS SÉRIES INICIAIS DO ENSINO FUNDAMENTAL**. 2017. Instituto Federal de Tocantins. Disponível em: <http://propi.ifto.edu.br/ocs/index.php/jice/8jice/paper/view/8640/3837>. Acesso em: 17 dez. 2022.

PAPERT, Seymour. **A máquina das crianças: repensando a escola na era da informática**. 4. ed. Porto Alegre: Artes Médicas, 1985.

PAPERT, S. **A máquina das crianças: repensando a escola na era da informática**. Tradução: Sandra Costa. Porto Alegre: Artmed, 2008. p. 224.

PAPERT, S. (1980). **Mindstorms: children, computers and powerful ideas**. New York: Basic Books.

PENINA, M. Quem disse que precisa de computador para programar? Conheça os benefícios das atividades desplugadas. **Educação Faber Castell**, São Paulo. Disponível em: <http://www.educacao.faber-castell.com.br/quem-disse-que-precisa-de-computador-para-programar-conheca-os-beneficios-das-atividades-desplugadas/#:~:text=%E2%80%9CEscrever%20c%C3%B3digos%20n%C3%A3o%20%C3%A9%20s%C3%B3,%E2%80%94%20%E2%80%9C%C3%A9%20para%20todos%E2%80%9D>. Acesso em: 12 nov. 2022.

PEREIRA, A., P., A. Andrei Cardozo. **O Scratch para o ensino de programação com crianças do 2º CEB**. 2021. Dissertação (Tese de Mestrado em Sistemas e Tecnologias de Informação para as Organizações) – Escola Superior de Tecnologia e Gestão de Viseu, Universidade de Viseu, Portugal.

PEREIRA, J. C. R., RAPKIEWICZ, C. 2004. **O Processo de Ensino-Aprendizagem de Fundamentos de Programação: Uma Visão Crítica da Pesquisa no Brasil**, WEI RJES.

PEREIRA, L. 2013 Escolas Defendem Ensino de Programação a Crianças e Adolescentes. **Olhar Digital**, 06 Fev. 2013. Disponível em: <http://olhardigital.uol.com.br/noticia/escolas-defendem-ensino-de-programacao-acrianças-e-adolescentes/35075>. Acesso em: 10 jan. 2023.

PIAGET, J. **A formação do símbolo na criança: Imitação, jogo e sonho imagem e representação**. 3^o ed. Rio de Janeiro: LTC, 1964.

PIAGET, J. **O juízo moral na criança**. Tradução Elzon Lenardon. 2. ed. São Paulo: Summus, 1994.

PIAGET, J. **Seis estudos de psicologia**. 22. ed. Rio de Janeiro: Forense Universitária, 1994.

PretaLab. (s.d.). **PretaLab: Inovação e Tecnologia com protagonismo negro**. Disponível em: <https://www.pretalab.com/>. Acesso em: 30 abr. 2023.

RACKMANN, Christian Puhlmann. **Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica**. 2017. p. 226. Tese (Doutorado em Informática na Educação) -Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017.

RAU, Maria C. T. **Educação infantil: práticas pedagógicas de ensino e aprendizagem**. 1^o ed. Curitiba: InterSaberes, 2012.

RESNICK, M. (2014). Aprender a programar, programar para aprender. **Transformar** 2014. São Paulo, Brasil, 28 jul. 2014.

RESNICK, Mitchel. **Sowing the Seeds for a More Creative Society**. Learning and Leading with Technology, 18-22, 2007.

RIBEIRO, J. L. P. Revisão de investigação e evidência científica. **Psicologia, Saúde & Doenças**, v.15 n.3, p. 671–682, 2014.

SANTOS, G.C. Análise bibliométrica dos artigos publicados como estudos bibliométricos na História do Congresso Brasileiro de Custos. **Pensar contábil**. Rio de Janeiro, v.17. n. 62, p. 4 – 13, jan/abr. 2015.

SANTOS, I. E. **Manual dos métodos e técnicas de pesquisa científica**. 8. ed. Rio de Janeiro: Impetus, 2011.

SBC, Sociedade Brasileira de Computação (2016). **Computação na Base Nacional Comum Curricular**. Elaborada pela comissão de Educação da SBC em colaboração com a Comissão Especial de Informática na Educação e membros da SBC.

SCAICO, P. D., et al. (2013). **Ensino de programação no ensino médio: Uma abordagem orientada ao design com a linguagem scratch**. *Revista Brasileira de Informática na Educação*. Anais do XVIII WIE.

SETZER, V. W. **O computador no Ensino: Nova Vida ou Destruição?**. In: CHAVES, E. O. C.; SETZER, V. W. O uso de computadores em escolas: Fundamentos e Críticas, São Paulo: Ed. Scipione, 1988, p. 69-127. Disponível em: www.ime.usp.br/~vwsetzer/computador-no-ensino.html. Acesso em: 25 jan. 2023

SETZER, V.W. **Meios Eletrônicos e Educação: uma visão alternativa**, 2a. ed. São Paulo: Ed. Escrituras, 2002, pp. 85-134. Disponível em: <https://www.ime.usp.br/~vwsetzer/PqQdCo.html>. Acesso em: 20 nov. 2022.

SILVA, J., C. **Ensino de programação para alunos do ensino básico: um levantamento das pesquisas realizadas no Brasil**. Revisão Bibliográfica (Bacharel em Ciência da Computação) - Universidade Universidade Federal da Paraíba, 2017.

SUPERGEEKS. (s.d). **Nossa História**. Disponível em: <https://www.supergeeks.com.br/quem-somos>. Acesso em: 08 fev. 2023.

LOSSO, Renta. **Crianças X computadores: benefícios e males da era tecnológica**. 16, ago.2010. Disponível em: <http://delas.ig.com.br/filhos/criancas-x-computadoresbeneficios-e-males-da-era-tecnologica/n1237749844018.html>. Acesso em: 08 dez. 2022. Especial para o iG São Paulo.

TAVARES, M. (s.d). **Criança e jovem sem tevê: postura radical ou sensata?** [Entrevista em vídeo]. Disponível em: <https://www.ime.usp.br/~vwsetzer/entrev/entrevista-pontocom-1009.html>. Acesso em: 18 de janeiro de 2023.

TORVALDS, L. (2014). Entrevista sobre tecnologia e inovação [Entrevista pessoal]. **Business Insider**.

VALENTE, J. (2018) Estudo aponta a falta de diversidade no mundo da tecnologia e inovação. **Agência Brasil**, 07 ago. 2018. Disponível em: <https://agenciabrasil.ebc.com.br/direitos-humanos/noticia/2018-08/estudo-aponta-falta-de-diversidade-no-mundo-da-tecnologia-e>. Acesso em: 15 maio 2023.

VIEIRA, J. G. S. **Metodologia da pesquisa científica na prática**. Curitiba: Fael, 2010.

XAVIER, G. F. C. **Lógica de programação**. 2018. Edição 13. Editora Senac São Paulo, 2018. Disponível em: https://books.google.com.br/books?hl=pt-BR&lr=&id=lxW8EAAAQBAJ&oi=fnd&pg=PP5&dq=l%C3%B3gica+de+programa%C3%A7%C3%A3o&ots=47lqkfmCW9&sig=hpoj5xY3D6dkftYIM_4sF4l_OVc#v=onepage&q=l%C3%B3gica%20de%20programa%C3%A7%C3%A3o&f=false. Acesso em: 03 abr. 2023.

ZANATTA, Andrei Cardozo. **PROGRAMAÇÃO DE COMPUTADORES PARA CRIANÇAS Metodologia do CODE CLUB Brasil**. 2015. Dissertação (Graduação

em Tecnologias da Informação e Comunicação) – Universidade Federal de Santa Catarina Campus Araranguá, Universidade de Santa Catarina, Araranguá, 2015.

ZANETTI, H. A. P.; BORGES, M. A. F.; LEAL, V. C. G.; MATSUZAKI, I. Y. Proposta de ensino de programação para crianças com scratch e pensamento computacional. **Tecnologias, Sociedade e Conhecimento**, Campinas, SP, v. 4, n. 1, p. 43–58, 2017. DOI:10.20396/tsc.v4i1.14484. Disponível em: <https://econtents.bc.unicamp.br/inpec/index.php/tsc/article/view/14484>. Acesso em: 29 fev. 2023.

WING, J. M. (2006). **Computational thinking**. Communications of the ACM.