

Universidade Estadual do Norte do Paraná

Repositório Institucional UENP

<https://repositorio.uenp.edu.br>

Programa de Pós-Graduação em Ensino

Produtos educacionais

2021

Abstração na educação básica: o que ensinar? e como ensinar?

FANTINATI, Regiane Ezequiel

Universidade Estadual do Norte do Paraná

<https://repositorio.uenp.edu.br/handle/123456789/676>

Baixado de Repositório Institucional UENP



**UNIVERSIDADE ESTADUAL DO NORTE DO
PARANÁ**

Campus Cornélio Procópio

**PROGRAMA DE PÓS-GRADUAÇÃO EM ENSINO
MESTRADO PROFISSIONAL EM ENSINO**

REGIANE EZEQUIEL FANTINATI

PRODUÇÃO TÉCNICA EDUCACIONAL

**ABSTRAÇÃO NA EDUCAÇÃO BÁSICA: O QUE ENSINAR?
E COMO ENSINAR?**

REGIANE EZEQUIEL FANTINATI

PRODUÇÃO TÉCNICA EDUCACIONAL

ABSTRAÇÃO NA EDUCAÇÃO BÁSICA: O QUE ENSINAR? E COMO ENSINAR?

Produção Técnica Educacional apresentada ao Programa de Pós-Graduação em Ensino da Universidade Estadual do Norte do Paraná – *Campus* Cornélio Procópio, como requisito parcial à obtenção do título de Mestre em Ensino.

Orientadora: Profa Dra Selma dos Santos Rosa

Ficha catalográfica elaborada pelo autor, através do
Programa de Geração Automática do Sistema de Bibliotecas da UENP

Fa Fantinati, Regiane Ezequiel
Abstração na Educação Básica: O que ensinar? E como ensinar? / Regiane Ezequiel Fantinati; orientadora Selma dos Santos Rosa - Cornélio Procópio, 2021.
62 p. :il.

Produção Técnica Educacional (Mestrado Profissional em Ensino) - Universidade Estadual do Norte do Paraná, Centro de Ciências Humanas e da Educação, Programa de Pós-Graduação em Ensino, 2021.

1. Pensamento Computacional. 2. Abstração. 3. Computação na Educação Básica. 4. Ensino de Computação. 5. Educação Básica. I. Santos Rosa, Selma dos, orient. II. Título.

LISTA DE ILUSTRAÇÕES

Figura 1: Esquema do Ensino em Espiral adaptado para o ensino da Abstração	13
Figura 2: Organização da proposta para o ensino de Abstração nos anos iniciais do EF	15
Figura 3: Visão geral da apresentação das atividades, agrupadas por curso na plataforma Code.org	32
Figura 4: Ambiente de programação Code.org.....	33
Figura 5: Ambiente da atividade Mapa Divertido	36
Figura 6: Ambiente da atividade Criar um Monstro - Catálogo	38
Figura 7: Ambiente da atividade Labirinto Sequência (com setas de	40
Figura 8: Ambiente da atividade Labirinto Sequência (com as direções)	41
Figura 9: Ambiente da atividade Labirinto Laços	43
Figura 10: Ambiente da atividade Laços de Repetição.....	45
Figura 11: Ambiente da Atividade Abelha Condicionais	47
Figura 12: Ambiente da atividade Labirinto Condicionais	49
Figura 13: Ambiente da atividade Labirinto Condicionais 2	50
Figura 14: Ambiente da atividade Fazendeira Laços Enquanto.....	52
Figura 15: Ambiente da atividade Hora do Código Minecraft.....	53
Figura 16: Esquema do Ensino em Espiral adaptado para o ensino da Abstração para os anos iniciais do EF	57

LISTA DE QUADROS

Quadro 1: Eixos de Conhecimentos da área de Computação organizados pela SBC e pelo CIEB.....	10
Quadro 2: Estrutura curricular proposta	18
Quadro 3: Rubricas de avaliação da unidade de conhecimento Abstração.....	25
Quadro 4: Síntese com link das atividades propostas para o ensino da Abstração ..	54

SUMÁRIO

1	<i>INTRODUÇÃO</i>	7
2	<i>FUNDAMENTAÇÃO TEÓRICO-METODOLÓGICA</i>	10
2.1	PRÁTICAS PEDAGÓGICAS: COMO DESENVOLVER AS HABILIDADES PROPOSTAS POR ESTA PTE	16
2.2	AVALIAÇÃO DA UNIDADE DE CONHECIMENTO ABSTRAÇÃO.....	26
2.3	ATIVIDADES DA PLATAFORMA CODE.ORG	31
3	<i>PRODUÇÃO TÉCNICA EDUCACIONAL</i>	34
3.1	ATIVIDADES PARA O DESENVOLVIMENTO DA ABSTRAÇÃO	35
3.1.1	Atividades para o 1º ano	35
3.1.2	Atividades para o 2º ano	39
3.1.3	Atividades para o 3º ano	44
3.1.4	Atividades para o 4º ano	48
3.1.5	Atividades para o 5º ano	51
4	<i>PROPOSTA DE ENSINO EM ESPIRAL</i>	57
5	<i>CONSIDERAÇÕES FINAIS</i>	59
	<i>REFERÊNCIAS</i>	60

1 INTRODUÇÃO

Em todo o mundo, os formuladores de políticas educacionais argumentam que às crianças deve ser ensinado como a tecnologia funciona, bem como quais habilidades de pensamento desenvolvidas por meio da programação são úteis em um contexto mais amplo (ROSE; HABGOOD; JAY, 2018). Essa argumentação está causando aumento na inserção, como disciplina ou tema transversal, da programação, do Pensamento Computacional (PC) ou da CC(CC) no currículo da Educação Básica (EB). Como exemplo dessa inserção, citamos as ferramentas de programação baseadas em blocos que se tornam cada vez mais presentes nos anos iniciais do Ensino Fundamental (LIEBE, 2019; ROSE; HABGOOD; JAY, 2018). Entretanto, observamos que, ao programar, alunos se deparam com dificuldades, resultando em *scripts* longos, blocos duplicados e problemas para identificar e corrigir erros, causados pela falta da habilidade de abstrair, tão fundamental na CC e no desenvolvimento do PC (ROSE; HABGOOD; JAY, 2019).

Para Fayer *et al.*, (2017) o ensino da Abstração contribuirá para capacidade de tornar os alunos criadores de tecnologias o que corrobora com os estudos de Brennan e Resnick (2012), que em sua pesquisa, relatam uma experiência que aponta que a Abstração pode ser trabalhada de diferentes maneiras: inicialmente, com a concepção do problema e, depois, traduzida na programação com o uso de variáveis e blocos de programação para chegar à solução do problema proposto. Para esses autores, o processo de Abstração pode ser conceituado como a construção de algo grande, juntando peças menores, como uma prática importante para a resolução de problemas. Essa prática, de acordo com esses mesmos autores, desenvolve nos alunos habilidades como Abstração e modularização.

De acordo com Kramer (2007), a capacidade de Abstração é uma das maneiras de se produzir programas mais claros e elegantes. Segundo o autor, seus melhores alunos nas disciplinas relacionadas à programação eram capazes de lidar com a complexidade dos problemas, produzindo modelos eficientes. Uma das

preocupações do autor está relacionada à incapacidade de Abstração de problemas, como não conseguir identificar o que é realmente importante em um problema.

Para Wing (2011), o mais importante processo cognitivo necessário para o PC é a Abstração a qual está presente na escrita de algoritmos, na compreensão e organização de um sistema e em diversas situações que envolvem a resolução de um problema (PASQUAL 2018). A Abstração é considerada por muitos pesquisadores uma habilidade essencial em computação (KRAMER, 2007), havendo, no entanto, muitos desafios associados ao seu ensino.

De acordo com Liebe (2019), existe uma falta significativa de pesquisas educacionais que orientem o ensino de CC na EB e as escassas pesquisas sobre o tema não abordam o ensino da Abstração em CC.

Statter e Armoni (2020) consideram a Abstração, em CC, muito difícil de ensinar e de aprender. Ainda de acordo com os autores, a literatura descreve a Abstração, em CC, como um conceito difícil de ensinar em todas as faixas etárias, sendo mais desafiador ensinar Abstração para os alunos mais novos segundo as teorias cognitivas, especialmente conforme as teorias clássicas do desenvolvimento cognitivo de Piaget

Statter e Armoni (2020) também afirmam que a Abstração deve ser ensinada em toda a sequência dos currículos de CC, devendo ser revista continuamente de maneira apropriada à idade, seguindo a abordagem em espiral proposta por Bruner (1960).

Frente ao exposto, em que explanamos sobre as possibilidades e relevância de ensinar Abstração enquanto unidade do PC aos alunos mais novos, apresentamos, como Produto Educacional¹, uma proposta de ensino com base na abordagem de ensino em espiral de Bruner (1960), para o ensino da Abstração para alunos dos anos iniciais da Educação Básica (faixa etária de 6 a 11 anos de idade).

¹ Produto Educacional é um instrumento que se configura como produção desenvolvida pelo orientador e orientando, totalmente vinculado ao trabalho de dissertação, com a finalidade de resolução de um problema específico de sala de aula, sendo aplicável e utilizável, e que, a partir de sua proposta didática, possa ajudar, modificar e transformar maneiras de ensinar e aprender. O produto não assume o papel de apenas uma exigência burocrática, mas é o resultado do trabalho científico, fruto da dissertação, tendo como objetivo o retorno à comunidade escolar no sentido de levar a pesquisa para dentro da escola (BATALHA, 2019).

Compreendemos que a abordagem em espiral, em que os tópicos são revisitados em todos os anos de ensino, de forma que possam, gradativamente ser aprofundadas em conteúdo, conceito e nível de Abstração contribuirá para a construção do raciocínio abstrato.

Para tanto, utilizamos a plataforma digital Code.org que aborda conteúdos de programação em blocos a qual, por sua vez, dispõe de recursos de práticas simuladas e experimentadas que propiciam o estímulo e a capacidade de Abstração da criança desde a Educação Infantil.

Também nos alicerçamos em diretrizes e referenciais curriculares para o ensino de Computação na Educação Básica elaborados pela Sociedade Brasileira da Computação (SBC) e pelo Centro de Inovação para a Educação Brasileira (CIEB), ambos com propostas alinhadas a competências gerais da Base Nacional Comum Curricular (BNCC).

2 FUNDAMENTAÇÃO TEÓRICO-METODOLÓGICA

A fim de elaborar a proposta para o ensino de Abstração enquanto unidade do PC por meio de uma abordagem em espiral, nos embasamos nas diretrizes e referenciais curriculares para o Ensino de Computação na EB elaborados pela SBC e pelo CIEB e alinhadas a competências gerais da BNCC.

Em suas diretrizes e referências curriculares, tanto a SBC e quanto o CIEB organizaram os conhecimentos da área de Computação a serem trabalhados na EB em três eixos, conforme o Quadro 1.

Quadro 1: Eixos de Conhecimentos da área de Computação organizados pela SBC e pelo CIEB

Conhecimentos da área de Computação		
	EIXOS	UNIDADES
SBC	Pensamento Computacional	Abstração
		Análise
		Automação
	Cultura Digital	Tecnologia e Sociedade
		Cidadania Digital
		Letramento Digital
	Mundo Digital	Codificação
		Processamento
		Distribuição
Currículo de referência		
	EIXOS	UNIDADES
CIEB	Pensamento Computacional	Reconhecimento de Padrões
		Decomposição
		Algoritmos
		Abstração
	Cultura Digital	Tecnologia e Sociedade
		Cidadania Digital
		Letramento Digital
	Tecnologia Digital	Representação de Dados
		Hardware e Software
Comunicação e Redes		

Fonte: a autora.

Conforme apresentado no Quadro 1, o ensino de Computação está organizado, segundo a SBC e o CIEB, em três eixos, subdivididos em conceitos aqui denominados, de unidades² de conhecimento. Cada unidade corresponde ao desenvolvimento de uma ou mais habilidades.

A SBC (2018) considera necessário trabalhar, já nos anos iniciais do Ensino Fundamental (EF), conceitos relacionados às estruturas abstratas do eixo PC, com uma abordagem crítica e estratégica para a utilização das bases computacionais nas diferentes áreas de conhecimento, com vistas, principalmente, ao desenvolvimento de competências para a resolução de problemas.

Já o CIEB, consoante o Quadro 1, apresenta, como unidades de conhecimento do PC o Reconhecimento de Padrões, Algoritmos, Decomposição e a Abstração. Importa ressaltar que essas unidades se inter-relacionam.

Para a elaboração desta proposta, trabalharemos com a unidade Abstração, do eixo PC, apontada no Quadro 1, habilidade que tem sido considerada o pilar fundamental da resolução de problemas no enfoque computacional, uma vez que ajuda a entender complexidades proporcionando a clareza necessária para solucioná-lo (SBC, 2017) estando o desenvolvimento dessas habilidades que convergem para a resolução de problemas complexos – consideradas essenciais para os alunos do século XXI – em consonância com a BNCC.

De acordo com Wing (2011), o mais importante processo cognitivo para o PC é a Abstração, imprescindível para identificar e ressaltar propriedades essenciais comuns a um conjunto de objetos ou dados. Ainda de acordo com Wing (2006), a Abstração é o conceito mais importante do PC, uma vez que o processo de abstrair é utilizado em diversos contextos e envolve “a escolha de um objeto, seu isolamento dos demais a ele relacionado, assumindo-se esse objeto isolado como o mais importante a ser tratado em um dado momento de um processo de resolução de um problema” (FANTINATI ; SANTOS ROSA, p. 14 não publicado), diminuindo a complexidade e proporcionando a clareza necessária diante de uma situação-problema.

² Adotamos o termo unidade por ser uma denominação utilizada na BNCC para organizar as habilidades por unidades de conhecimento e unidades temáticas

De acordo com Fuller *et al.* (2007) citado por Liebe (2019), a capacidade de usar a Abstração efetivamente é uma habilidade ensinável, o que vai ao encontro de Bruner (1960) citado por Statter e Armoni (2020, p. 2):

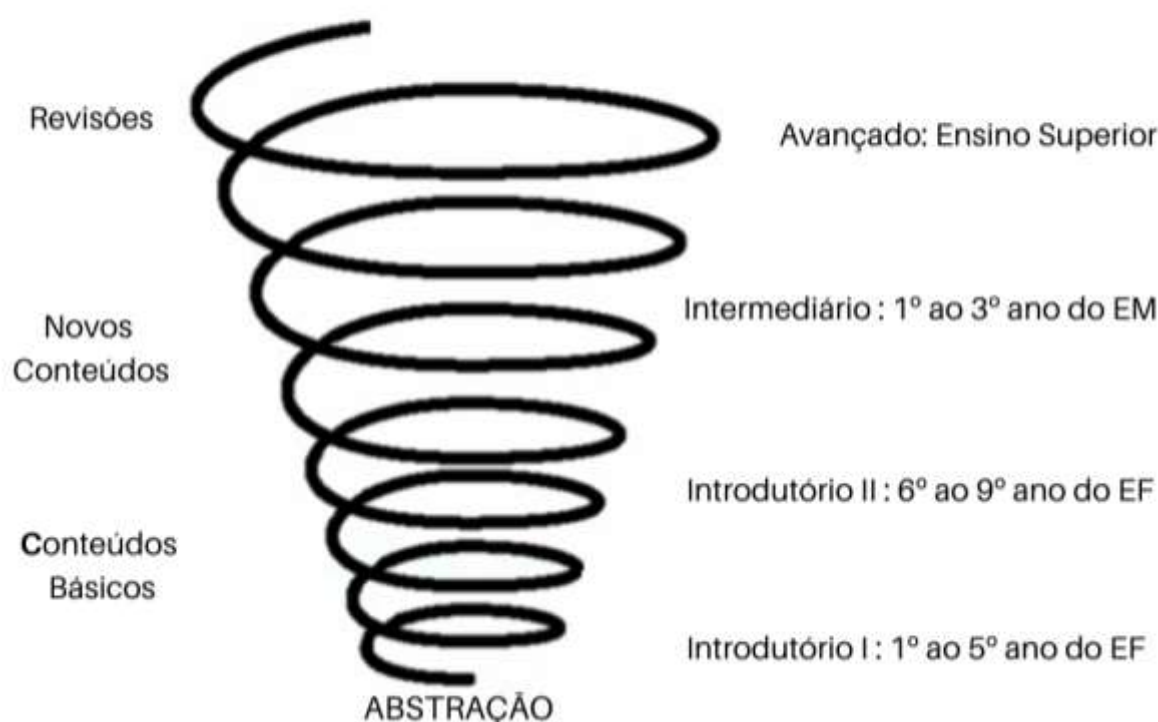
A Abstração deve refletir-se em toda a sequência dos currículos de ciências da computação (do Ensino Fundamental ao ensino médio até a graduação) começando o mais cedo possível, seguindo a abordagem em espiral de Bruner. De acordo com a abordagem em espiral, ela também deve ser revisitada continuamente de maneira apropriada à idade nesta sequência da educação em ciência da computação (STATTER e ARMONI, 2020, p. 2)

Para Bruner (1960), o conceito da aprendizagem em espiral pode anunciar-se da seguinte forma: qualquer ciência pode ser ensinada, pelo menos nas suas formas mais simples, uma vez que os mesmos tópicos serão, posteriormente, retomados e aprofundados. Bruner (1960) igualmente argumenta que o currículo deve ser organizado de forma que possam ser trabalhados, periodicamente, os mesmos conteúdos, mas cada vez com maior profundidade. Além disso, deve ser ensinado o mais cedo possível, como também ser revisado em todos os níveis de ensino até os níveis de graduação e, em cada caso, de maneira apropriada à idade (STATTER; ARMONI, 2020).

Considerando a sua importância, o ensino de Abstração em CC, pode ser organizado de acordo com a estrutura de Bruner(1960), ou seja, em espiral. Statter e Armoni (2020) o utilizaram no Ensino Médio, enquanto nós, na presente PTE abordaremos esse ensino nos anos iniciais da EB (do 1º ao 5º ano do EF). Assim, o ensino da Abstração, enquanto unidade do PC constituirá a base da espiral, considerando que este é o primeiro conhecimento dos alunos com essa unidade em CC. Tal conteúdo poderá continuar a ser revisitado cada vez com maior profundidade em outros níveis de ensino, conforme Figura 1.

Figura 1: Esquema do Ensino em Espiral adaptado para o ensino da Abstração

Ensino da Abstração enquanto unidade do Pensamento Computacional



Fonte: Adaptado da abordagem em espiral de Bruner(1960)

Para trabalhar a proposta para o ensino de Abstração com abordagem em espiral, utilizamos, como material de referência, as atividades da plataforma Code.org, as quais foram selecionadas de forma que, gradualmente, pudessem ser revisitadas e aprofundadas para aumentar o nível de Abstração. A escolha se deu por essa plataforma se apresentar como uma proposta de ensino que tem, como um dos objetivos, desenvolver o PC desde os alunos pré-alfabetizados, por ser considerada uma boa forma de os alunos começarem a resolver problemas reais por meio do estímulo à capacidade de Abstração com recurso à prática simulada e experimentada (BARRADAS *et al.*, 2019).

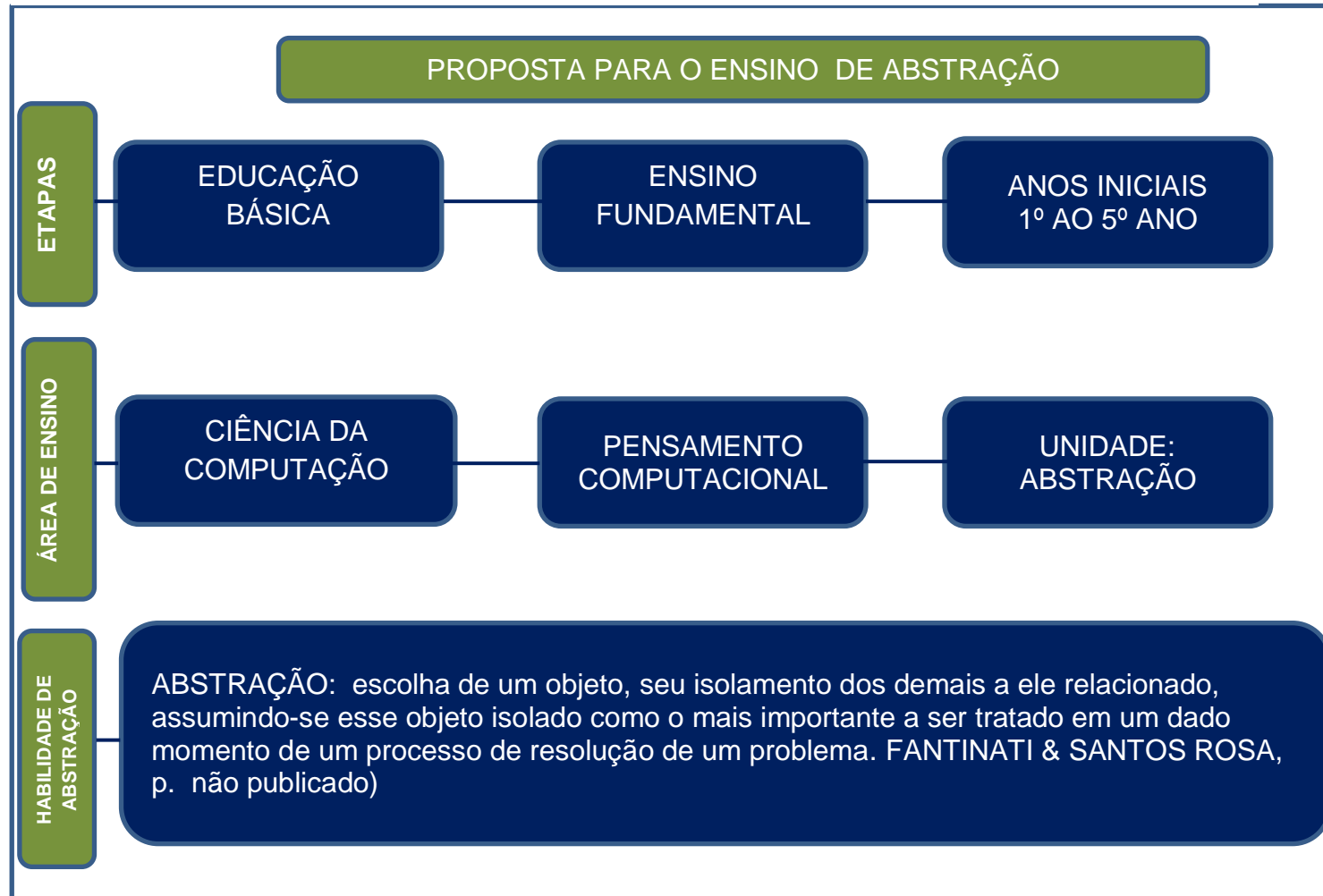
A plataforma Code.org trabalha com atividades de programação em blocos, interface visualmente amigável, e com um objetivo prático para o ensino dos conceitos iniciais da programação. Abrange conceitos de sequências, estrutura de condicionais, estrutura de repetição (*loops*), funções, utilização de operações

matemáticas, operadores lógicos – VERDADEIRO, FALSO, MAIOR ou MENOR QUE, entre outros.

Para além do exposto, a escolha dessa plataforma deu-se também pelo fato de a autora dessa PTE utilizá-la frequentemente em suas atividades docentes e por considerá-la adequada para desenvolver nos alunos habilidades de Abstração que é seu foco de pesquisa. O acompanhamento de tais atividades, ao longo das etapas dos anos iniciais do EF – conforme os alunos avançam para os próximos anos, avança também o grau de dificuldade das atividades – possibilitou à autora observar os avanços que os alunos apresentam ano após ano, independente do seu grau de dificuldade inicial. Da mesma forma, possibilitou que os alunos observassem e comentassem esse avanço.

Entre os conteúdos que estimulam o desenvolvimento do raciocínio abstrato, optamos por trabalhar com atividades que abrangem a programação, pois, de acordo com Alves (2016), os conceitos básicos de programação podem ser aplicados na resolução de problemas de diferentes áreas de conhecimento, mesmo que ainda não estejam diretamente ligados à programação, evidenciando que esses conceitos são valiosos em todos os contextos e fundamentais no ensino da Abstração. A Figura 2 apresenta a organização da nossa proposta de ensino.

Figura 2: Organização da proposta para o ensino de Abstração nos anos iniciais do EF



Fonte: a autora.

A proposta de ensino, conforme apresentado na Figura 2, tem, como público-alvo, alunos dos anos iniciais da EB (1º ao 5º ano do EF). A área de ensino é a CC, especificamente a unidade de conhecimento Abstração, uma das unidades que compõem o PC, conforme referido anteriormente, podendo ser implementada de forma transversal ao currículo.

2.1 PRÁTICAS PEDAGÓGICAS: COMO DESENVOLVER AS HABILIDADES PROPOSTAS POR ESTA PTE

De acordo com CIEB (2018), as habilidades são o elemento essencial do currículo, uma vez que apontam o que precisa ser desenvolvido pelo aluno e orientam a prática docente. A habilidade “Abstração” é compreendida, conforme já mencionado na Introdução e na Figura 2, como “a escolha de um objeto, seu isolamento dos demais a ele relacionado, assumindo-se esse objeto isolado como o mais importante a ser tratado em um dado momento de um processo de resolução de um problema”, (FANTINATI & SANTOS ROSA, p.14 não publicado)

Considerando que a proposta para o ensino de Abstração, enquanto unidade do PC, ora apresentada tem, como base, a abordagem em espiral, organizamos as habilidades inerentes à Abstração de forma a progredir e aprofundar gradativamente de acordo com o ano de ensino, do 1º ao 5º ano do EF.

Para o desenvolvimento das habilidades de Abstração já descritas, esta PTE apresenta sugestões de práticas pedagógicas com uso do computador e sem o uso do computador (atividades desplugadas). Para elaborar tais sugestões, seguimos orientações de Bruner (1960, p.32):

[...] toda ideia pode ser representada de maneira honesta e útil nas formas de pensamento da criança em idade escolar, em que as primeiras representações podem, posteriormente, tornar-se mais poderosas e precisas, com maior facilidade, graças a essa aprendizagem anterior (BRUNER, 1960 p.32).

Assim, selecionamos atividades que, de acordo com o ano de ensino, pudessem, gradativamente, ser aprofundadas em conteúdo, conceito e

nível de Abstração de forma que, igualmente, pudessem ser revisitados proporcionando reflexão, compreensão e desenvolvimento do raciocínio abstrato.

Tendo em vista que a área desta proposta de ensino é a CC, as atividades – abreviadas no quadro que as mostra (Quadro 2) como Atv1, Atv2 e assim sucessivamente – incluem conceitos de algoritmos, sequenciação e ciclos com conteúdos básicos de lógica de programação, como variáveis, comandos, laços de repetição e condicionais.

A estrutura curricular proposta nesta PTE – Quadro 2 –também traz definições do ano escolar em que a atividade será aplicada, do nome da atividade, da situação-problema, da habilidade de Abstração a ser desenvolvida pelo aluno, da rubrica de avaliação (de acordo com o Quadro 3, a ser apresentado nas páginas subsequentes), das habilidades e das competências gerais de acordo com a BNCC e as habilidades relacionadas ao Ensino da Computação, de acordo com a SBC, para as etapas do EF, com ênfase no eixo PC.

Cada habilidade preconizada pela BNCC (2018, p. 51, 95, 279, 287, 291, 411) tem um código identificador, dividido em quatro partes: o primeiro par de letras indica a etapa da EB, o primeiro par de números indica o ano a que se refere a habilidade, o segundo par de letras indica o componente curricular, o último par de números indica a posição da habilidade na numeração sequencial do ano ou do bloco de anos. Exemplo: o código [EF03MA12] indica EF – Ensino Fundamental; 03 - o ano; MA – Matemática; 12 – o número da habilidade. A coluna Rubrica de Avaliação contém as rubricas (Quadro 3), associada a cada atividade apresentadas no Quadro 2. ,

Apresentamos, conforme Quadro 2, a estrutura curricular para o ensino de Abstração enquanto unidade do PC.

Quadro 2: Estrutura curricular proposta

Ano	Atividades	Situação-problema	Habilidade de Abstração	Rubrica de Avaliação	Habilidades BNCC	Competências Gerais BNCC (2019)	Habilidades Computação com ênfase no PC- SBC
1º	Atv1: Mapa Divertido	Localizar qual caminho (criando um algoritmo ao juntar as setas) leva Flurb ao tesouro.	Decidir quais detalhes de um problema são importantes. Decidir quais detalhes de um problema podem ser ignorados.	1, 2	[EF15LP09] - Expressar-se em situações de intercâmbio oral com clareza, preocupando-se em ser compreendido pelo interlocutor e usando a palavra com tom de voz audível, boa articulação e ritmo adequado.	[CG02] Exercitar a curiosidade intelectual e recorrer à abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.	[PC] Criar passos para solução de problemas relacionados ao movimento do corpo e trajetórias espaciais.
	Atv 2 Criar um Monstro	Identificar o que cada monstro (ex.: Zumbi, Frankstein, Duende) tem de diferente e	Identificar os detalhes que são responsáveis pelas diferenças e usar a estrutura geral para	1, 2, 9, 11, 16	[EI03ET05] Classificar objetos e figuras de acordo com suas	[CG04] – Utilizar diferentes linguagens – verbal (oral ou visual-motora,	[PC] Representar os passos de uma tarefa por meio de uma

		abstrair as diferenças para criar uma lista de instruções (algoritmos) que possam ser usadas por todos para recriar um dos monstros, orientados por essas instruções.	encontrar uma solução que seja válida para mais de um problema.		semelhanças e diferenças [EI03ET01] Estabelecer relações de comparação entre objetos, observando suas propriedades.	como Libras, e escrita), corporal, visual, sonora e digital, bem como conhecimentos das linguagens artística, matemática e científica, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao entendimento mútuo.	notação pictórica, de forma organizada e relacional [PC] Compreender uma situação-problema criando e identificando sequências de passos de uma tarefa para sua solução.
2º	Atv 1 Labirinto Sequência	Programar o personagem <i>Angry Bird</i> para que cumpra o desafio de chegar até o porco desviando de objetos encontrados pelo caminho e usando somente a quantidade necessária de blocos (quantidade limitada).	Decidir quais detalhes de um problema são importantes e quais detalhes de um problema podem ser ignorados (em um contexto de maior complexidade).	1, 2, 6	[EF03MA12] Descrever e representar, por meio de esboços de trajetos ou utilizando croquis e maquetes, a movimentação de pessoas ou de objetos no espaço, incluindo mudanças de direção e sentido,	[CG02] Exercitar a curiosidade intelectual e recorrer à abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e	[PC] Utilizar linguagem lúdico-visual para representar algoritmos.

					com base em diferentes pontos de referência.	criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.	
	Atv 2 Labirinto Laços	Programar, como na atividade anterior, o personagem <i>Angry Bird</i> para que chegue até o porco desviando de objetos encontrados pelo caminho. Mas, agora, nessa atividade, é aprofundado o grau de dificuldade, uma vez que diminuiu a quantidade máxima de blocos a ser utilizada, e foi acrescentado o bloco com o comando "Repita", necessário para a resolução do desafio proposto.	Reforçar a habilidade da atividade anterior e identificar e eliminar os elementos que se repetem em um determinado problema encapsulando os elementos repetidos. (Decidir quais detalhes de um problema podem ser ignorados)	1, 2, 4, 6, 10, 12	[EF03MA12] - Descrever e representar, por meio de esboços de trajetos ou utilizando croquis e maquetes, a movimentação de pessoas ou de objetos no espaço, incluindo mudanças de direção e sentido, com base em diferentes pontos de referência.	[CG02] Exercitar a curiosidade intelectual e recorrer à abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.	[PC] Definir e simular algoritmos (descritos em linguagem natural ou pictográfica) construídos como sequências e repetições simples de um conjunto de instruções básicas (avance, vire à direita, vire à esquerda, etc.).
3º	Atv 1: Abelha: laços de	Programar a personagem para que percorra o labirinto identificando e	Compreender e utilizar a modularização ou reuso do bloco de programação.	1, 2, 4, ,10, 12, 13, 14	[EF04MA03] - Resolver e elaborar problemas com números naturais	[CG02] Exercitar a curiosidade intelectual e recorrer à	[PC] Identificar as principais abstrações para

	repetição	coletando a quantidade de mel e o néctar a produzir, utilizando a estrutura de repetição.			envolvendo adição e subtração, utilizando estratégias diversas, como cálculo, cálculo mental e algoritmos, além de fazer estimativas do resultado.	abordagem própria das ciências, incluindo a investigação, a reflexão, a análise crítica, a imaginação e a criatividade, para investigar causas, elaborar e testar hipóteses, formular e resolver problemas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das diferentes áreas.	construir processos – escolha, composição e repetição – simulando e definindo algoritmos simples que representem situações do cotidiano infantil.
	Atv 2: Abelha: condicionais	Programar a personagem Abelha para que percorra o labirinto e colete néctar ou faça mel sem saber a quantidade a se obter, uma vez que a quantidade não está explícita na atividade. Esses procedimentos devem ser executados com a estrutura de condicionais.	Extrair, dentro de um problema matemático, conceitos fundamentais para a resolução de problemas encapsulando-o em estruturas de condicionais.	1, 2, 3, 4, 6, 10, 12, 13, 16	[EF01MA01] Utilizar números naturais como indicador de quantidade ou de ordem em diferentes situações cotidianas e reconhecer situações em que os números não indicam contagem nem ordem, mas sim código de	[CG05] Compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos,	[PC] Identificar subproblemas comuns em problemas maiores e a possibilidade do reuso de soluções.

					identificação.	resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.	
4º	Atv 1 Labirinto: condicionais	<p>Programar o personagem para que escolha qual a melhor decisão a ser tomada para chegar ao destino, utilizando a estrutura de condicionais. Exemplo: quando se deparar com uma curva, utilizar o comando SE para verificar a condicional virar à esquerda ou à direita.</p> <p>Retomar e aprofundar a atividade 1 em que o personagem Zumbi verifica uma condição: se há caminho à frente, à direita ou à esquerda. Se sim, ele escolhe uma instrução; se não, ele escolhe outra instrução, ou seja, faz</p>	Usar procedimentos para encapsular um conjunto de comandos frequentemente repetidos e relacionados à tomada de decisão.	1, 2, 3, 4, 6, 9, 10, 12, 13, 16	[EF03MA12] - Descrever e representar, por meio de esboços de trajetos ou utilizando croquis e maquetes, a movimentação de pessoas ou de objetos no espaço, incluindo mudanças de direção e sentido, com base em diferentes pontos de referência.	[CG05] Compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.	[PC] Colaborar e cooperar na proposta e execução de soluções algorítmicas utilizando decomposição e reuso no processo de solução.

	Atv 2: Labirinto: condicionais 2	uma coisa OU outra.					
5º	Atv: 1 Fazendeira: Laços (comando Enquanto)	Programar a personagem Fazendeira para utilizar comandos já aprendidos nas atividades anteriores e utilizar o comando Enquanto: “Enquanto uma instrução na parte superior for verdadeira, faça alguma coisa”. Nessa atividade, a personagem deve ser programada para que ENQUANTO houver uma pilha de terra, essa terra deve ser movida.	Usar procedimentos para encapsular um conjunto de comandos frequentemente repetidos e relacionados à tomada de decisão: estrutura de repetição e estrutura de condicionais.	1, 2, 3, 4, 5, 6, 9, 10, 12, 13, 16	[EF03HI11] Identificar diferenças entre formas de trabalho realizadas na cidade e no campo, considerando também o uso da tecnologia nesses diferentes contextos. [EF03MA06] - Resolver e elaborar problemas de adição e subtração com os significados de juntar, acrescentar, separar, retirar, comparar e completar quantidades, utilizando diferentes estratégias de cálculo exato ou aproximado, incluindo cálculo mental.	[CG05] Compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.	[PC] Compreender a técnica de decompor um problema para solucioná-lo. ([PC]) Colaborar e cooperar na proposta e execução de soluções algorítmicas utilizando decomposição e reuso no processo de solução.

	<p>Atv 2:</p> <p>Hora do código: Minecraft</p>	<p>Programar o personagem do Minecraft de forma que não caia na lava. O desafio é que a lava está sob as pedras, impossibilitando a visualização. É necessário, então, que o aluno programe o personagem para verificar se existe lava naquele lugar antes de avançar.</p>	<p>Rever, aprofundar e elevar o nível de Abstração de encapsulamento.</p> <p>Usar procedimentos de encapsulamento organizando as informações em estruturas que possam auxiliar na resolução de problemas.</p>	<p>1, 2, 3, 4, 5, 6, 9, 10, 12, 13, 15, 16</p>	<p>[EF03MA12]</p> <p>Descrever e representar, por meio de esboços de trajetos ou utilizando croquis e maquetes, a movimentação de pessoas ou de objetos no espaço, incluindo mudanças de direção e sentido, com base em diferentes pontos de referência.</p>		<p>[PC]</p> <p>Identificar subproblemas comuns em problemas maiores e a possibilidade do reuso de soluções.</p>
--	--	--	---	--	--	--	---

Fonte: a autora.

A estrutura curricular apresentada no Quadro 2 tem, como propósito, contribuir para o ensino da Abstração de forma a ajudar o aluno a entender problemas complexos, proporcionando clareza necessária para a sua resolução. Essa habilidade pode ser útil em quaisquer área e contextos.

Pré-requisitos para o professor

Como já mencionado, a proposta utiliza as atividades da plataforma Code.org, uma plataforma de ambiente amigável que permite que programas sejam construídos com blocos encaixados como quebra-cabeça, não exigindo necessidade de conhecimento prévio de outras linguagens de programação, uma vez que ele é ideal para pessoas que estão começando a programar, incluindo crianças. Ressaltamos que a atividade disponibiliza aos professores planos de ensino e vídeos explicativos. Esta proposta também contribui com a organização de atividades da plataforma para orientar professores no ensino da Abstração.

Nesse caminho, as atividades aqui propostas podem ser trabalhadas de forma transversal ao currículo, de modo interdisciplinar e por professores de qualquer área de conhecimento. Nesta proposta, as atividades abrangem áreas como Matemática, História e Língua Portuguesa e áreas correlatas à CC.

Para a condução das atividades, sugerimos ao professor:

- Iniciar os conceitos básicos de lógica de programação com atividades desplugadas sugeridas por esta PTE ou outra que o professor possa contextualizar com situações rotineiras dos alunos.
- Apresentar a plataforma Code.org e instruir os alunos sobre como utilizá-la.
- Propor a utilização da plataforma Code.org pelos alunos individualmente ou em dupla.
- Antes de iniciar um conceito ou conteúdo novo na plataforma, o professor pode introduzir, conforme sugerido, atividades desplugadas de forma que contextualize o conteúdo com situações rotineiras dos alunos.

Importante ressaltar que, embora tenhamos selecionado duas atividades por ano de ensino, totalizando dez atividades, o professor, de acordo

com o seu planejamento, pode decidir sobre qual, como e com que grau de profundidade abordará os conteúdos sugeridos, podendo também trabalhar com outras atividades disponibilizadas na plataforma Code.org e aprofundar mais ou menos cada conteúdo, detendo-se mais em alguns deles, se necessário, para que os alunos desenvolvam a habilidade de Abstração proposta.

2.2 AVALIAÇÃO DA UNIDADE DE CONHECIMENTO ABSTRAÇÃO

A avaliação tem, como propósito, auxiliar os docentes a acompanhar os alunos no desenvolvimento das habilidades propostas, buscando identificar se foram ou não desenvolvidas.

Para isso, sugerimos dois tipos de avaliação que podem ser aplicadas de forma complementar. A primeira é a utilização pelo professor do recurso disponível na plataforma Code.org por meio do qual é possível acompanhar a turma e também os alunos individualmente. Na plataforma, há relatórios de quantos desafios os alunos concluíram e quantas linhas de código usaram, se cumpriram parcialmente ou integralmente uma atividade ou ainda se a deixaram de realizar. Caso o aluno tenha chegado ao resultado, mas não tenha atingido todos os objetivos propostos (por exemplo, não utilizou a estrutura de repetição), tem a possibilidade de realizar a atividade novamente até obter o resultado esperado.

A segunda sugestão é o uso de rubricas de avaliação que, segundo Stevens e Levi (2005), consiste em uma ferramenta que indica, em uma escala, as expectativas específicas para uma determinada tarefa. Ao elaborar uma rubrica de avaliação o professor pode criar vários critérios, bem como inserir uma escala de pontuação para cada um deles.

O Quadro 3 apresenta uma proposta de rubricas de avaliação para Abstração, enquanto unidade do PC.

Quadro 3: Rubricas de avaliação da unidade de conhecimento Abstração

Identificação	Autores	Habilidade de Abstração	Não atende 0 a 2	Atende minimamente Insuficiente 3 a 4	Atende parcialmente Regular 5 a 6	Atende Bom 7 a 8	Atende completa- mente Muito Bom 9 a 10
1	Selby (2015); Cetin; Dubinsky (2017); Rijke, <i>et al.</i> (2017); Wing (2008)	Decidir quais detalhes de um problema são importantes.					
2	Selby (2015); Cetin; Dubinsky (2017); Rijke, <i>et al.</i> (2017); Wing (2008)	Decidir quais detalhes de um problema podem ser ignorados.					
3	Costa (2016)	Usar procedimentos para encapsular um conjunto de comandos frequentemente repetidos de uma função: usar as estruturas de condicionais.					
4	Costa (2016)	Usar procedimentos para encapsular um conjunto de comandos frequentemente repetidos de uma função: usar estrutura de repetição.					

5	Costa (2016)	Usar da Abstração para decompor um problema em subproblemas.					
6	Brennan e Resnick (2012)	Aplicar a modularização: Construir algo grande unindo conjuntos de partes (peças) menores.					
7	Costa (2016)	Deduzir conclusões a partir de fatos.					
8	Costa (2016)	Usar metáforas.					
9	Selby (2015); Cetin; Dubinsky (2017); Rijke <i>et al.</i> (2017); Melham (2012); Curzon <i>et al.</i> (2014)	Sintetizar					

10	Brennan e Resnick (2012)	Compreender e utilizar a modularização.					
11	CODE (2020)	Identificar as diferenças entre objetos e descrevê-las dentro dos padrões definidos.					
12	Rose; Habgood; Jay (2019)	Identificar a reutilizar o código (blocos de programação).					
13	Cetin e Dubinsky (2017)	Extrair, dentro de um problema matemático, conceitos fundamentais para a resolução;					
14	CODE (2020) Brennan e Resnick (2012)	Identificar semelhanças entre objetos e aplicá-los em diferentes contextos;					

15	Curzon et al (2014)	Ocultar a complexidade dos dados ou objetos;					
16	Curzon et al (2014)	Filtrar informações no desenvolvimento de soluções;					

Fonte: a autora.

As rubricas apresentadas no Quadro 3 estão associadas a cada atividade exposta no Quadro 2, havendo mais de uma rubrica de avaliação que servirá de parâmetro para os professores elaborarem seus *feedbacks* sobre as atividades propostas, reunindo elementos para indicar as dificuldades e os avanços dos alunos, as lacunas identificadas e o que precisa ser melhorado para que a proposta atinja o objetivo que é a construção do raciocínio abstrato nos alunos dos anos iniciais da EB.

2.3 ATIVIDADES DA PLATAFORMA CODE.ORG


A Code.org® é uma organização sem fins lucrativos dedicada a expandir o acesso à CC nas escolas de Ensino Fundamental e Médio. A plataforma – também denominada de Code.org –oferece uma variedade de cursos para estudantes explorarem os conceitos de programação, pensamento computacional, cidadania digital e desenvolvimento de jogos interativos e histórias. Conta com uma quantidade significativa de recursos que possibilitam aprender e ensinar (CODE, 2020). Porém, em virtude do objetivo deste estudo, detalharemos com mais precisão somente algumas das atividades que vão ao encontro da proposta apresentada.

A plataforma mantém uma estrutura comum na qual as atividades são agrupadas por cursos de acordo com a faixa etária. Conforme exposto na Figura 3, cada curso compõe, em média, 20 aulas (lições) com vários níveis, com quantidade que variam até o máximo de 16 níveis em formato de tutorial. Essa estrutura conta também com planos de aula e atividades *off-line* disponíveis para o professor, tendo a função de introduzir questões concretas antes das atividades direcionadas pelos tutoriais.

Figura 3: Visão geral da apresentação das atividades, agrupadas por curso na plataforma Code.org

Curso 3

O Curso 3 foi desenvolvido para os alunos que já fizeram o Curso 2. Os alunos se aprofundarão nos tópicos de programação introduzidos nos cursos anteriores, buscando soluções flexíveis para problemas mais complexos. No final deste curso, eles vão criar histórias e jogos interativos que podem compartilhar com qualquer pessoa. Este curso é recomendado para alunos de 4º-5º ano.



Tente agora Obter ajuda

Nome da Lição	Progresso
1. Pensamento computacional	Atividade off-line 1 2
2. Labirinto	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
3. Artista	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
4. Caça-sóis funcionais	Atividade off-line 1 2
5. Artista: Funções	1 2 3 4 5 6 7 8 9 10
6. Abelha: Funções	1 2 3 4 5 6 7 8 9 10 11
7. Abelha: Condicionais	1 2 3 4 5 6 7 8 9 10
8. Labirinto: Condicionais	1 2 3 4 5 6 7 8 9 10 11 12
9. Composição	Atividade off-line 1
10. Jogo de dados	Atividade off-line 1 2
11. Artista: Laços Aninhados	1 2 3 4 5 6 7 8 9 10 11 12
12. Fazendeira: Laços Enquanto	1 2 3 4 5 6 7 8 9
13. Abelha: Laços Aninhados	1 2 3 4 5 6 7 8 9 10 11 12 13
14. Abelha: Depuração	1 2 3 4 5 6 7 8 9 10 11 12 13
15. Pingue-pongue	1 2 3 4 5 6 7 8 9 10
16. Laboratório: Crie uma história	1 2 3 4 5 6
17. Laboratório: Crie um jogo	1 2 3 4 5 6 7
18. Internet	Atividade off-line 1 2
19. Crowdsourcing	Atividade off-line 1 2
20. Cidadania digital	Atividade off-line 1
21. Artista: padrões	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Fonte: <https://studio.code.org/s/course>

Na plataforma Code.org, o *layout* dos ambientes segue um padrão comum: toda atividade tem a área de ilustração do problema com personagens conhecido das crianças, como *Angry Birds*®, *Elza*®, *Minecraft*®, zumbis, entre outros. Conta, também, com a explanação do objetivo e as instruções de resolução, os blocos e a área para a programação, conforme apresentado na Figura 4.

Figura 4: Ambiente de programação Code.org



Fonte: <https://studio.code.org/s/course2/stage/8/puzzle/4>

A plataforma é composta, principalmente, por atividades *on-line*, com temas e níveis de dificuldades de acordo com o público-alvo.

Cada atividade apresenta, logo no início, as instruções necessárias e o desafio proposto, com a opção de áudio para alunos que ainda não estão alfabetizados. Na parte esquerda da tela, é exibido o ambiente do desafio. A parte direita da tela é o espaço destinado para a programação. Para realizar o desafio, o aluno deve selecionar os blocos de comandos necessários e encaixá-los, de forma sequencial, como se fossem um jogo de quebra-cabeça.

Para verificar se o código desenvolvido está correto, se pode ser executado de outra forma ou se precisa de correções para atingir o resultado esperado, o aluno clica no botão “Executar” e, se o objetivo não foi atingido, ele pode refazer a programação quantas vezes forem necessárias, além de receber orientação por mensagens com dicas da atividade. Na parte superior da tela, é possível acompanhar o desenvolvimento do aluno.

Cada círculo apresentado no topo da página, conforme Figura 4, representa uma etapa da atividade. Se, no desafio, o aluno atingiu os objetivos propostos, o círculo é colorido de verde escuro. No caso de o aluno ter chegado ao resultado, mas não ter atingido todos os objetivos propostos – por exemplo, não utilizou a quantidade mínima de blocos proposta para tal atividade – o círculo é colorido em verde claro tendo o aluno, então, a possibilidade de realizar a atividade novamente até obter o resultado esperado.

3 PRODUÇÃO TÉCNICA EDUCACIONAL

A Produção Técnica Educacional apresentada neste documento é parte integrante da Dissertação de Mestrado intitulada: “**Ciência Da Computação nos anos iniciais da Educação Básica**:uma proposta para o ensino de Abstração” disponível em: <http://www.uenp.edu.br/mestrado-ensino>. Para maiores informações, entrar em contato com a autora pelo email: refantinati@gmail.com.

A seguir, apresentamos a Produção Técnica Educacional.

3.1 ATIVIDADES PARA O DESENVOLVIMENTO DA ABSTRAÇÃO

Esta seção apresenta atividades que, disponibilizadas pela plataforma Code.org selecionamos para fazer parte de nossa proposta para desenvolver a Abstração, enquanto unidade do PC, em alunos dos anos iniciais do EF (1º ao 5º ano), usando a abordagem de ensino em espiral proposta por Bruner (1960).

Atividades do Code.org

Iniciaremos com conteúdos básicos - a base da espiral - introduzindo conceitos de Abstração como unidade de conhecimento do PC para crianças que serão introduzidas em CC, ou seja, para alunos do 1º ano (6 e 7 anos) evoluindo, gradativamente, conforme avançam no conteúdo e no nível de ensino até chegar no 5º ano (10 e 11 anos). Selecionamos, na plataforma Code.org, entre as atividades que compõem, duas para cada ano de ensino ora enfocado.

Da mesma forma, de cada atividade, discorreremos sobre a situação-problema apresentada ao aluno e o que esperamos que seja desenvolvido. Importante ressaltar que os alunos devem realizar todas as atividades sugeridas em cada fase e que descrevemos apenas 2 atividades por ano para exemplificar a habilidade desenvolvida nas respectivas fases da atividade descrita.

Cabe lembrar, conforme apresentado na Figura 2, que as atividades são agrupadas na plataforma com a denominação de curso, que cada curso é composto por diversas aulas (lições) e que, a partir da progressão dessas aulas, novos conceitos são introduzidos enquanto outros já apresentados são reforçados.

3.1.1 Atividades para o 1º ano

No Brasil, não há a obrigatoriedade de se alfabetizar as crianças durante a Educação Infantil. De acordo com a BNCC (BRASIL, 2018), a alfabetização das crianças deverá ocorrer até o segundo ano do Ensino Fundamental. Consideraremos então, para o desenvolvimento das atividades do 1º ano, que os alunos não estejam alfabetizados.

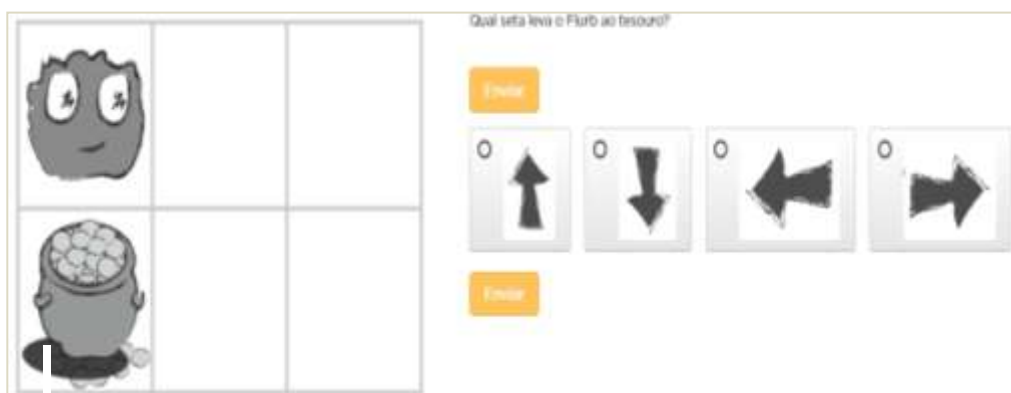
As atividades escolhidas para estimular o desenvolvimento do pensamento abstrato nessa fase de ensino, são atividades desplugadas³ disponíveis na plataforma Code.org. As duas atividades selecionadas são Mapas Divertidos e Criar um Monstro, indicadas para alunos na faixa etária de 4 a 6 anos.

Atividade 1 - Primeira atividade para o 1º ano: Mapas Divertidos

Essa primeira atividade (Figura 5) busca introduzir a criança no conceito e na construção de algoritmos. A plataforma disponibiliza o material a ser impresso: um mapa, o *Flurb* (personagem) e as flechas. Com o material em mãos, os alunos serão orientados a listar etapas para mover caracteres ao redor de um mapa, organizando as direções para atingir uma meta predeterminada que é, no caso, chegar ao tesouro, possibilitando uma previsão de aonde o personagem *Flurb* chegará seguindo uma lista de etapas.

Sugerimos também, antes de iniciar a atividade, que o professor, desenhe um labirinto no chão e convide os alunos a percorrer este labirinto, de forma que o "*Flurb*" seja substituído por um aluno e o tesouro poderia ser substituído por um objeto, corroborando com a implementação da computação desplugada, fortalecendo a ideia de que é possível aprender técnicas de programação sem uso de computadores.

Figura 5: Ambiente da atividade Mapa Divertido



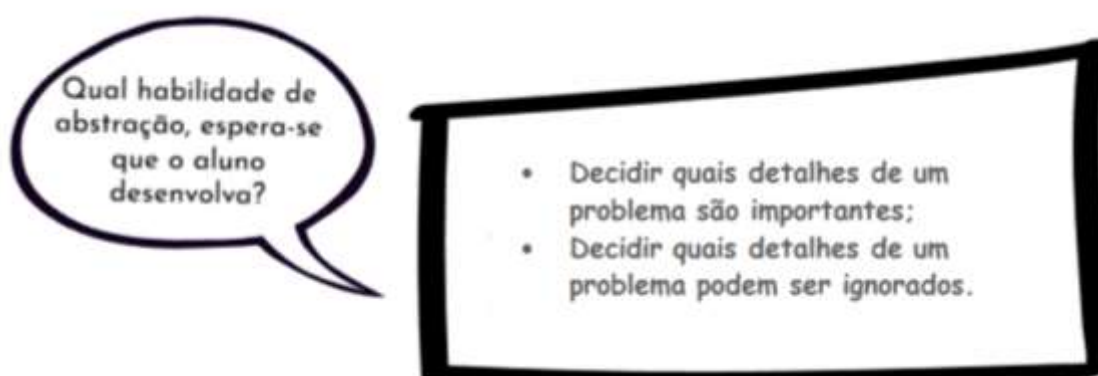
Fonte: <https://curriculum.code.org/pt-br/csf-1718/coursea/5/>

³ A atividade desplugada consiste em atividades desenvolvidas com o objetivo de ensinar os fundamentos da CC sem a necessidade de usar computadores.

A atividade Mapa Divertido tem o objetivo de introduzir o aluno nos conceitos de CC, possibilitando o conhecimento dos primeiros comandos sequenciais de programação (algoritmo), sem que, necessariamente, seja necessário usar um computador

Situação-problema da atividade Mapas divertidos:

O aluno deverá localizar o caminho (criando um algoritmo ao juntar as setas) que leva *Flurb* ao tesouro.



Para Wing (2006), a Abstração está presente na escrita de algoritmos, na compreensão e na organização de um sistema e em diversas situações que envolvem a resolução de um problema. Nessa primeira atividade, para o 1º ano, cuja proposta é iniciar o ensino de Abstração, com a construção de algoritmos, recorreremos à estratégia de ensino desplugada, conforme já relatado. Para isso, utilizamos um material concreto, ou seja, manuseável, possibilitando à criança a visualização do todo. Essa proposta vai ao encontro da afirmação de Santos (2018) de que o docente deve prover diferentes contextos para que os estudantes compreendam tanto os conceitos abstratos quanto as manipulações simbólicas desses conceitos.

Espera-se que o aluno desenvolva a habilidade de centrar-se unicamente nas propriedades relevantes da atividade que é chegar, por meio de setas direcionais, até o objeto solicitado, ignorando o restante do caminho considerado irrelevante para atingir o objetivo proposto, ou seja, reduzindo a

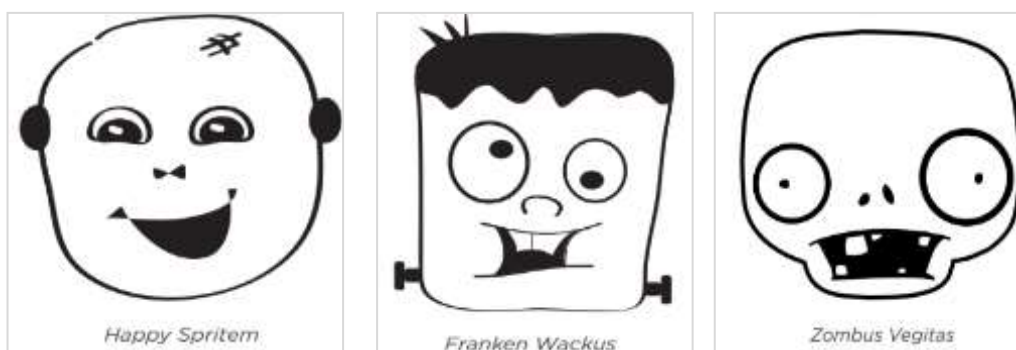
complexidade para focar na ideia principal. Com isso, o aluno será conduzido ao entendimento da situação-problema, primeiramente interagindo com um objeto concreto, para que após, por meio da construção de um algoritmo, a atividade seja compreendida e solucionada.

Atividade 2 - Segunda atividade para o 1º ano: Criar um Monstro

Essa segunda atividade pode ser realizada com a classe toda ao mesmo tempo. É apresentado aos alunos um catálogo de monstros (disponível na plataforma). Cada monstro precisa ser classificado de acordo com as informações atribuídas a ele no catálogo (Figura 6). Primeiro, os alunos analisarão os monstros para ver o que todos eles têm em comum e, então, abstrairão as diferenças para criar uma lista de instruções que possam ser usadas por todos para recriar um dos monstros do catálogo. Os alunos desenharão esses monstros somente com as instruções, ou seja, sem vê-los.

Para auxiliar o professor na execução dessa atividade, orientamos que acessem o plano de aula completo, disponível em <https://cutt.ly/KzzzFyS> e assistam ao vídeo com a explicação da atividade em <https://youtu.be/injJWiSA0pw>.

Figura 6: Ambiente da atividade Criar um Monstro - Catálogo

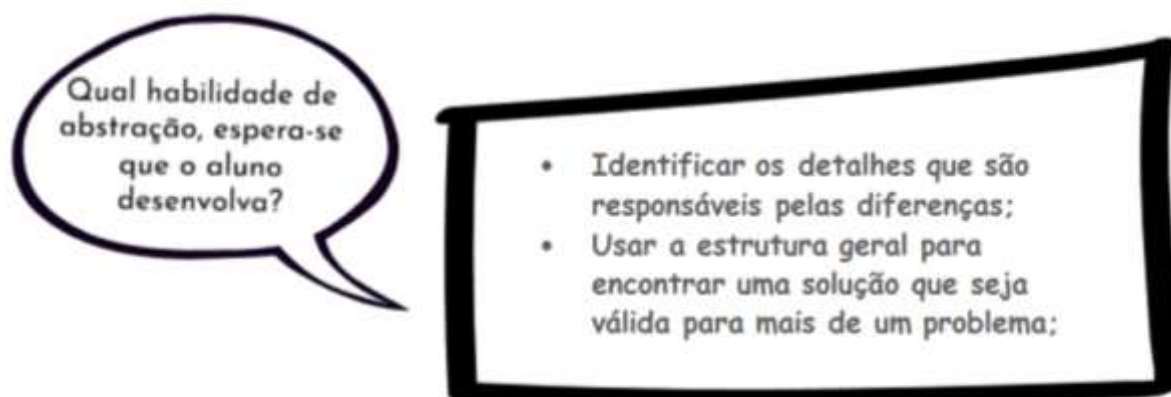


Fonte: <https://studio.code.org/unplugged/unplug2.pdf>

Situação-problema da atividade Criar um Monstro

Os alunos terão que identificar o que cada monstro tem de diferente, dentro de padrões já definidos, ou seja, abstrairão as diferenças para criar uma lista de instruções (algoritmos) que podem ser usadas por todos para recriar um dos

monstros do catálogo, orientados por essas instruções. Com isso, desenvolverão a habilidade de Abstração exposta abaixo.



Espera-se que o aluno consiga identificar o que é diferente (características) em cada monstro e que, embora todos tenham olhos, um monstro pode ter olhos de “doidão”, enquanto outro, olhos de “duende”. O aluno será capaz, por exemplo, de olhar um monstro e reconhecer que o monstro tem cabeça de zumbi, olhos de duende, boca de ranzinza ou não tem nenhuma das características em questão. Desenvolverá a habilidade de abstrair os detalhes que caracterizam as diferenças e usar essas informações para criar um Algoritmo (orientações) para que outra equipe desenhe um determinado monstro.

3.1.2 Atividades para o 2º ano

Após ter introduzido os alunos do 1º ano no conteúdo de Abstração com atividades desplugadas, as próximas atividades, para os próximos anos, serão com recursos *on-line* também disponíveis na plataforma Code.org para que os alunos possam experimentar a programação com blocos a partir dos conceitos já apresentados com a atividade desplugada.

As atividades selecionadas para o 2º ano foram: Labirinto Sequência, que se apresenta subdividido – com setas direcionais e com as direções (DIREITA, ESQUERDA, AVANCE) – e Labirinto Laços.

Atividade 3 - Primeira atividade para o 2º ano: Labirinto Sequência

A atividade Labirinto Sequência, aborda o ensino de algoritmo (sequência de passos) e a escolhemos porque possibilita aos alunos revisitar o conceito da atividade Mapas Divertidos, aplicada de forma desplugada. Agora, a atividade será realizada no computador. A atividade Labirinto Sequência (com setas direcionais) encontra-se na Figura 7.

Figura 7: Ambiente da atividade Labirinto Sequência (com setas de direcionamento)



Fonte: <https://studio.code.org/s/course1/stage/4/puzzle/10>

A atividade Labirinto Sequência, com as direções (VIRAR À DIREITA, VIRAR À ESQUERDA E AVANCE, pode ser visualizada na Figura 8.

Figura 8: Ambiente da atividade Labirinto Sequência (com as direções)



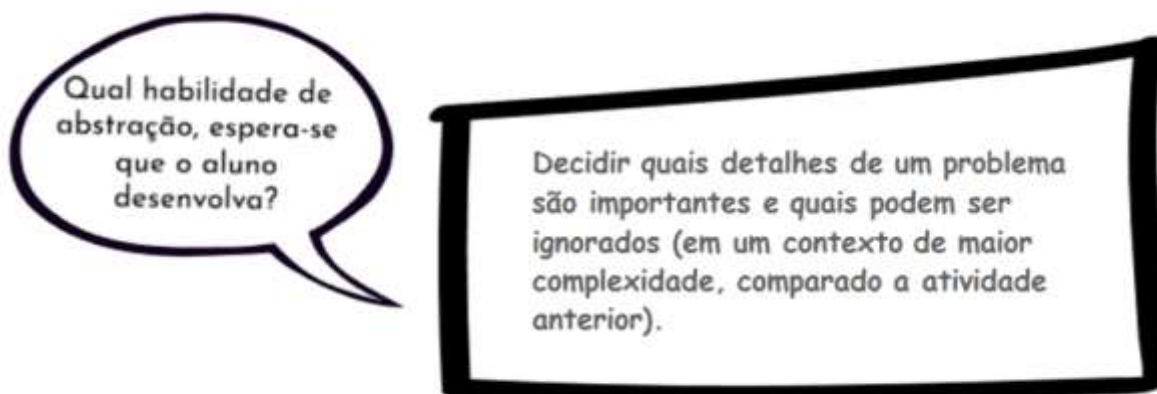
Fonte: <https://studio.code.org/s/course2/stage/3/puzzle/4>

Situação- problema da atividade Labirinto Sequência

O aluno deverá programar o personagem *Angry Birds* para que chegue até o porco desviando de objetos encontrados pelo caminho, limitando-se a quantidade máxima de blocos definida na atividade.

Nessa atividade, o aluno deve programar ações do pássaro, personagem do *Angry Birds*, escolhendo qual o caminho o levará até o Porco Verde. Ao longo de todos os níveis e seus desafios, o modo de resolução segue o mesmo princípio: conforme avança o nível, aumenta a distância percorrida e a quantidade de obstáculos para a resolução do desafio proposto.

A primeira atividade proposta, para o 1º ano, consistia em trabalhar o ensino de algoritmos, de modo que, inicialmente, recorrêssemos às atividades desplugadas, ou seja, utilizássemos materiais concretos, manuseáveis para iniciar os alunos em atividades que desenvolvam a Abstração, diferente dessa atividade para o 2º ano, que é *on-line* e cuja habilidade de Abstração a ser desenvolvida encontra-se num contexto de maior complexidade.



Nessa atividade Labirinto Sequência com direções, espera-se que o aluno raciocine logicamente sobre a atividade que experimentou, manipulando-as, agora, de forma simbólica (simulada), como um aprofundamento da habilidade de Abstração proposta para o 1º ano que era centrar-se unicamente nas propriedades relevantes da atividade, ignorando detalhes irrelevantes para atingir o objetivo proposto, mas aumentando o nível de complexidade, com o acréscimo do desafio de o aluno precisar desviar dos objetos do labirinto, como as dinamites, por exemplo, e com uma quantidade limitada de passos. A pretensão é, gradualmente, aumentar o nível da habilidade de Abstração do aluno.

Atividade 4 - Segunda atividade para o 2º ano: Labirinto Laços

Com a segunda atividade selecionada, o objetivo é reforçar o aprendizado da atividade anterior e introduzir a estrutura de repetição.

As estruturas de repetição também conhecidas como laços (*loops*) são utilizados para executar, repetidamente, uma instrução ou bloco de instruções enquanto determinada condição estiver sendo satisfeita.

A atividade é iniciada com a retomada da estrutura de programação em sequência, a exemplo da Figura 8, e, no segundo nível, inicia-se a mesma tarefa, mas com a inserção do laço de repetição, conforme Figura 9. O professor deve enfatizar a comparação entre essas duas estruturas.

Figura 9: Ambiente da atividade Labirinto Laços

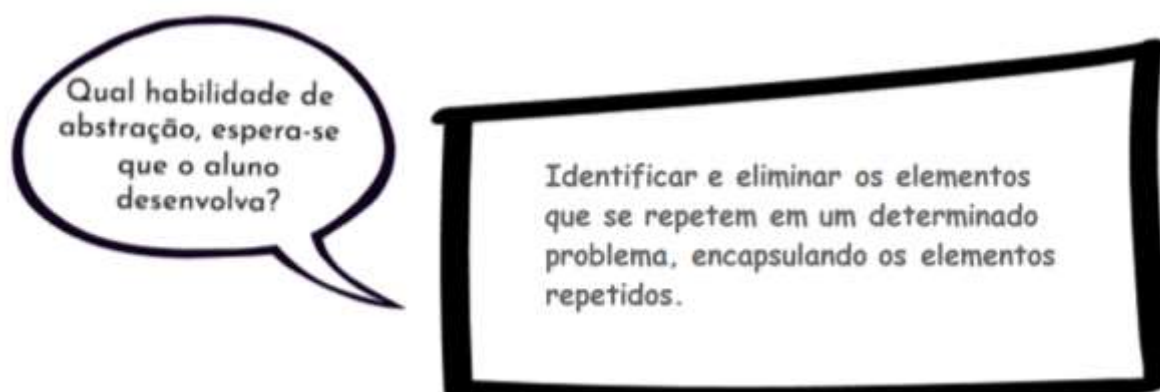


Fonte: <https://studio.code.org/s/course2/stage/3/puzzle/4>

Sugerimos que, antes de iniciar essa atividade, o professor utilize atividades desplugadas para trabalhar com o conceito “laço de repetição”. A plataforma disponibiliza vídeos com sugestões de atividade para a introdução do conceito de estruturas de repetição, citamos como exemplo, um vídeo de dança. Os alunos aprenderão uma coreografia simples e, então, serão instruídos a repeti-la.

Situação-problema da atividade Labirinto Laços

O aluno deverá programar, como na atividade anterior, o personagem *Angry Birds* para que chegue até o porco desviando de objetos encontrados pelo caminho, mas, agora, nessa atividade, objetiva-se aprofundar o grau de dificuldade, exigindo um maior nível de Abstração com a inserção do bloco de comando *REPITA*, o aluno terá que identificar um conjunto de comandos (blocos) repetidos e substituí-los pelo bloco *REPITA*. Ilustramos a abaixo a habilidade de Abstração a ser desenvolvida nesta atividade.



Espera-se que o aluno, ao analisar o problema, descarte as redundâncias (irrelevantes) abstraindo (eliminando) os blocos duplicados, e que os substitua por blocos que podem ser reutilizados (repetição), promovendo uma programação mais enxuta, somente com os blocos necessários. Esta atividade tem como objetivo, além de reforçar a habilidade da atividade anterior, desenvolver a habilidade de Abstração de identificar e eliminar os elementos que se repetem em um determinado problema encapsulando os elementos repetidos. (Decidir quais detalhes de um problema podem ser ignorados).

3.1.3 Atividades para o 3º ano

As atividades selecionadas para o 3º ano foram Laços de Repetição e Abelha Condicionais.

Atividade 5 - Primeira atividade para o 3º Ano: Laços de Repetição

Esta atividade retoma o conteúdo dos laços de repetição, vistos no 2º ano, mas com um grau maior de dificuldade, pois, agora, o labirinto apresenta mais um desafio, conforme apresentado na Figura 10: para chegar ao destino, o aluno precisa programar a abelha para coletar néctar e produzir mel.

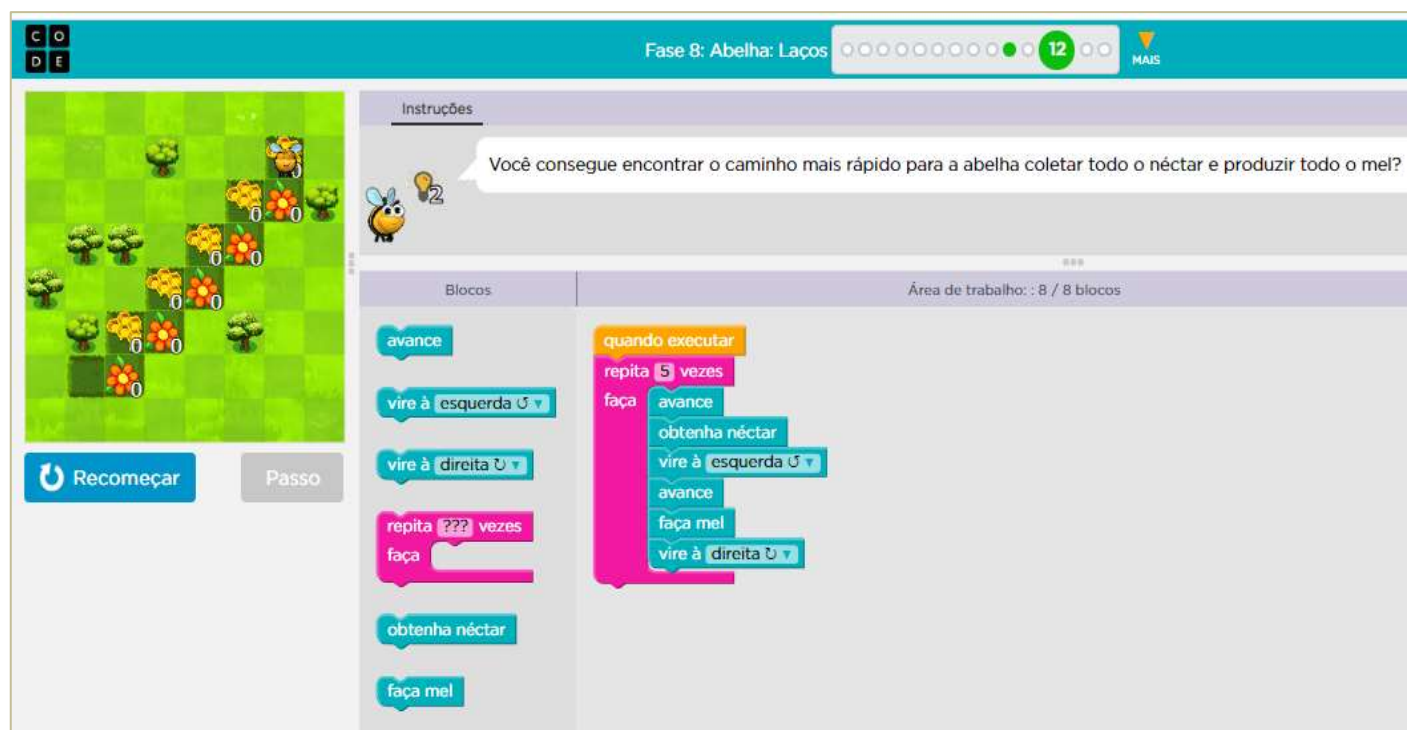
Conforme já mencionado, os Laços de repetição, também conhecidos como *loops*, são comandos que permitem iteração de código, ou seja, que comandos presentes no bloco sejam repetidos diversas vezes. É uma repetição

de uma atividade ou tarefa. A plataforma Code.org, traz uma explicação da atividade:

Assim que abro uma colmeia para a inspecionar, retiro uma moldura, inspeciono-a, e depois coloco-a de volta no seu local. Seguidamente, eu repito essa tarefa várias vezes. Isso é um loop, ou seja, fazer a mesma atividade várias vezes. Tudo o que você colocar dentro do bloco repita será repetido o número de vezes que você quiser. (CODE, 2017).

Para auxiliar no conceito de laços de repetição e no desenvolvimento da atividade sugerimos que o professor assista ao vídeo disponibilizado pela plataforma: <https://youtu.be/4W9TAXGJyyU>

Figura 10: Ambiente da atividade Laços de Repetição



Fonte: <https://studio.code.org/s/course2/stage/8/puzzle/12>

Situação-problema da atividade Laços de Repetição

O aluno deverá programar a personagem para que percorra o labirinto identificando a quantidade de mel a coletar e o néctar a produzir, colocando no bloco *REPITA*, o número de vezes necessário para que a ação (quantidade de

mel a coletar e o néctar a produzir) seja concluída (ao invés de, por exemplo, arrastar os blocos avance e obtenha néctar três vezes, o aluno deve arrastá-los uma vez cada e repetir a ação três vezes). A atividade tem por objetivo, desenvolver a habilidade de Abstração exposta abaixo.



Esta atividade busca contribuir para o reforço da habilidade da atividade anterior (descartar as redundâncias, eliminando os blocos duplicados, substituindo-os por blocos que podem ser reutilizados), mas com um grau maior de dificuldade. Isso porque a inserção dos blocos de “coletar néctar” e “produzir mel”, junto aos laços de repetição (já estudados pelo aluno) aumentará a quantidade de variáveis no programa, também aumentando sua complexidade e, conseqüentemente, exigindo do aluno um maior nível de Abstração.

Atividade 6 - Segunda atividade para o 3º Ano: Abelha Condicionais

Nesta atividade, os alunos revisitarão os conteúdos dos laços de repetição e iniciarão o conceito de condicionais. A sugestão é que, antes de introduzir as aulas com condicionais, o professor inicie o conceito com uma atividade desplugada. Cabe informar que a plataforma disponibiliza um plano de ensino denominado “Condicionais: codificação com cartas” e apresenta também um vídeo, com o mesmo tema. Essa aula traz as estruturas de condicionais, especialmente no que diz respeito a sua relação com os *laços de repetição* e comandos *SE*.

A “estrutura condicional” possibilita a escolha de um grupo de ações e estruturas a serem executadas quando determinadas condições são ou não satisfeitas. O comando que define essa estrutura é representado pela palavra SE que tem um sentido condicional. Assim, o aluno terá que analisar o bloco condicional no qual o cenário apresentado, conforme a Figura 10, diz que, SE houver néctar ou mel MENOR que, IGUAL a, ou MAIOR que um número, ENTÃO algo deve ser feito (obter néctar ou fazer mel). Nesta atividade, o bloco não informa a quantidade de néctar ou de mel a obter. Apresenta, apenas, o ponto de interrogação. Nesse exemplo, o aluno programará com a condição “SE a flor roxa tem néctar MAIOR QUE 0, obtenha néctar”. Isso fará com que a abelha continue coletando néctar enquanto for necessário. O ambiente dessa atividade encontra-se na Figura 11.

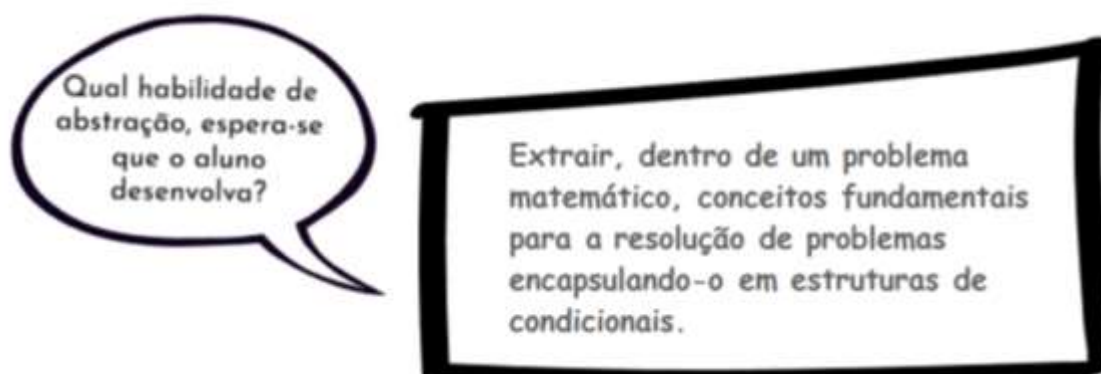
Figura 11: Ambiente da Atividade Abelha Condicionais

The screenshot displays the 'Abelha Condicionais' activity interface. At the top right, it says 'Fase 13: Abelha: Condicionais'. The main area is divided into a grid on the left and a programming panel on the right. The grid shows a bee on a green field with several flowers, including a purple one. The programming panel has an 'Instruções' section with a bee icon and text: 'Este campo tem flores vermelhas e roxas. As flores vermelhas têm uma quantidade conhecida de néctar, mas o néctar das flores é desconhecido. Colete todo o néctar.' Below this is a 'Blocos' section with various code blocks: 'avance', 'vire à esquerda', 'vire à direita', 'obtenha néctar', 'faça mel', 'repita ??? vezes faça', and 'se néctar = 1 faça'. At the bottom left of the grid area are 'Executar' and 'Passo' buttons.

Para melhor compreensão do conceito de *Condicionais*, sugerimos que o professor assista ao vídeo: <https://youtu.be/g8BQLKnrj2U>.

Situação-problema da Atividade Abelha Condicionais

O aluno deverá programar a personagem para que ela percorra o labirinto e colete néctar ou faça mel sem saber a quantidade a se obter, uma vez que a quantidade não está explícita na atividade (SE houver néctar ou mel MENOR que, IGUAL a, ou MAIOR que um número, ENTÃO algo deve ser feito). A atividade tem por objetivo, desenvolver a habilidade de Abstração exposta abaixo.



Espera-se que o aluno compreenda a informação apresentada pela atividade extraíndo o conceito matemático fundamental, dos símbolos: MAIOR QUE, MENOR QUE e IGUAL A, usados para comparar números e expressões. Esse conceito matemático deverá ser implementado na estrutura condicional ATÉ QUE a condição seja satisfeita.

3.1.4 Atividades para o 4º ano

As atividades selecionadas para o 4º ano foram Labirinto Condicionais e Labirinto Condicionais 2.

Atividade 7 - Primeira atividade para o 4º ano: Labirinto Condicionais

A atividade Labirinto Condicionais, possibilita ao aluno revisitar os conteúdos de Laços de Repetições e as estruturas de condicionais, agora com grau maior de dificuldade. Na presente atividade o bloco *SE* ajuda o personagem Zumbi a tomar uma decisão. O aluno o programará para verificar o cenário ao seu redor e decidir o melhor caminho que o levará ao seu destino. O personagem terá que ser programado usando os blocos de laços de repetições e os blocos condicionais para cumprir o desafio proposto nesta atividade, conforme apresentado na Figura 12.

Figura 12: Ambiente da atividade Labirinto Condicionais



Fonte: <https://studio.code.org/s/course3/stage/8/puzzle/3>

Nessa atividade, o aluno programará o personagem utilizando o bloco *SE* para tomar uma decisão. Exemplo, ao usar o bloco que diz “SE houver um caminho à esquerda, faça”, o aluno deve usar o comando vire à esquerda dentro dele. O personagem está sendo programado para ver as coisas ao seu redor, ver se há um caminho à esquerda e, caso o caminho exista, ele deve virar à esquerda.

Atividade 8 - Segunda atividade para o 4º ano: Labirinto Condicionais 2

A segunda atividade indicada para o 4º ano visa reforçar a atividade anterior, mas com o acréscimo de mais um comando condicional: o bloco *SE/SENÃO*, conforme Figura 13.

Figura 13: Ambiente da atividade Labirinto Condicionais 2

The screenshot displays the Scratch environment for the 'Labirinto Condicionais 2' activity. At the top, the title bar reads 'Fase 8: Labirinto: Condicionais' with a progress indicator showing 8 out of 10 steps. The main workspace shows a maze with a zombie character and several eggs. The 'Instruções' panel contains the following text: 'O bloco se/senão verifica uma condição e, em seguida, faz uma coisa OU outra. Use este novo bloco para me levar até o girassol.' The 'Blocos' panel shows a script starting with 'quando executar' followed by a 'repetir até' loop containing a 'se houver caminho à frente' block. The 'Área de trabalho' shows 3/5 blocks used.

Fonte: <https://studio.code.org/s/course3>

O funcionamento da estrutura **SE/SENÃO** baseia-se no resultado de uma expressão booleana (**VERDADEIRO** ou **FALSO**) em que o fluxo do algoritmo segue para um bloco de instruções ou **NÃO**.

Nessa atividade, o personagem Zumbi verifica uma condição: se há caminho à frente, à direita ou à esquerda. Se **SIM**, ele escolhe uma instrução; SE **NÃO**, escolhe outra instrução, ou seja, faz uma coisa **OU** outra.

Situação-problema das atividades Labirinto Condicionais e Labirinto Condicionais 2

O aluno deverá programar o personagem para, quando se deparar com o problema apresentado, optar pela melhor decisão (**SE/SENÃO**) a ser tomada para chegar ao destino utilizando a estrutura de condicionais e laços de repetição e a quantidade máxima de blocos estabelecidos. A habilidade de Abstração a ser desenvolvida com essa atividade pode ser visualizada na ilustração abaixo.



Espera-se que o aluno compreenda a estrutura (condicional e laços de repetição) elevando o nível de Abstração de encapsular um conjunto maior de comandos frequentemente repetidos e relacionados à tomada de decisão.

3.1.5 Atividades para o 5º ano

As atividades selecionadas para o 5º ano foram Fazendeira - Laços Repetição e Hora do Código Minecraft.

Atividade 9 - Primeira atividade para o 5º ano: Fazendeira Laços Enquanto

Nessa atividade, os alunos revisitarão conceitos já aprendidos, como os laços de repetição, e serão introduzidos a um novo comando: o bloco *ENQUANTO*.

O comando *ENQUANTO* é recomendado quando o número de repetições for desconhecido. Essa estrutura repete uma sequência de comandos enquanto uma determinada condição (especificada por meio de uma expressão lógica) for satisfeita.

A estrutura desse comando é apresentada de forma simples nessa atividade (Figura 14): “enquanto uma instrução for verdadeira, faça alguma coisa”. Nessa atividade, a personagem deve ser programada para que *ENQUANTO* houver uma pilha de terra, essa terra deve ser movida até que o solo fique plano.

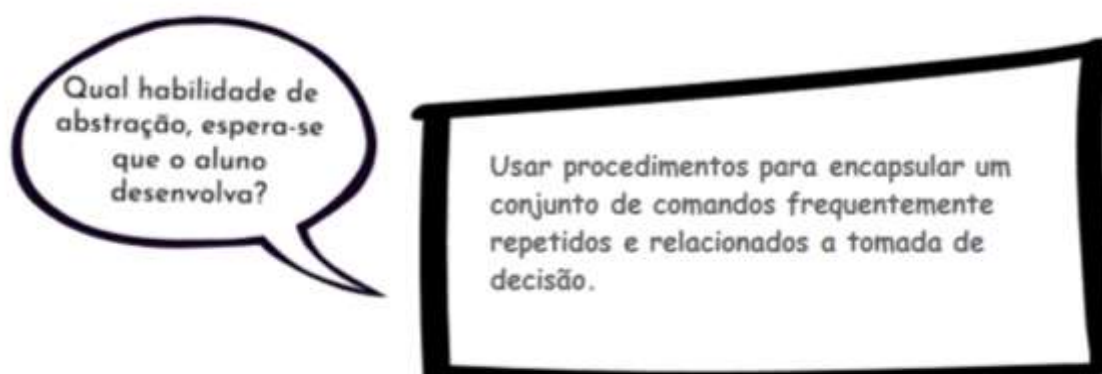
Figura 14: Ambiente da atividade Fazendeira Laços Enquanto



Fonte: <https://studio.code.org/s/course3/stage/12/puzzle/8>

Situação-problema da atividade Fazendeira - Laços Enquanto

O aluno deverá programar o personagem para que ENQUANTO uma instrução não for verdadeira, faça alguma coisa, compreendendo que existem duas situações distintas nessa mesma atividade. Na ilustração abaixo encontra-se a habilidade de Abstração a ser desenvolvida com essa atividade.

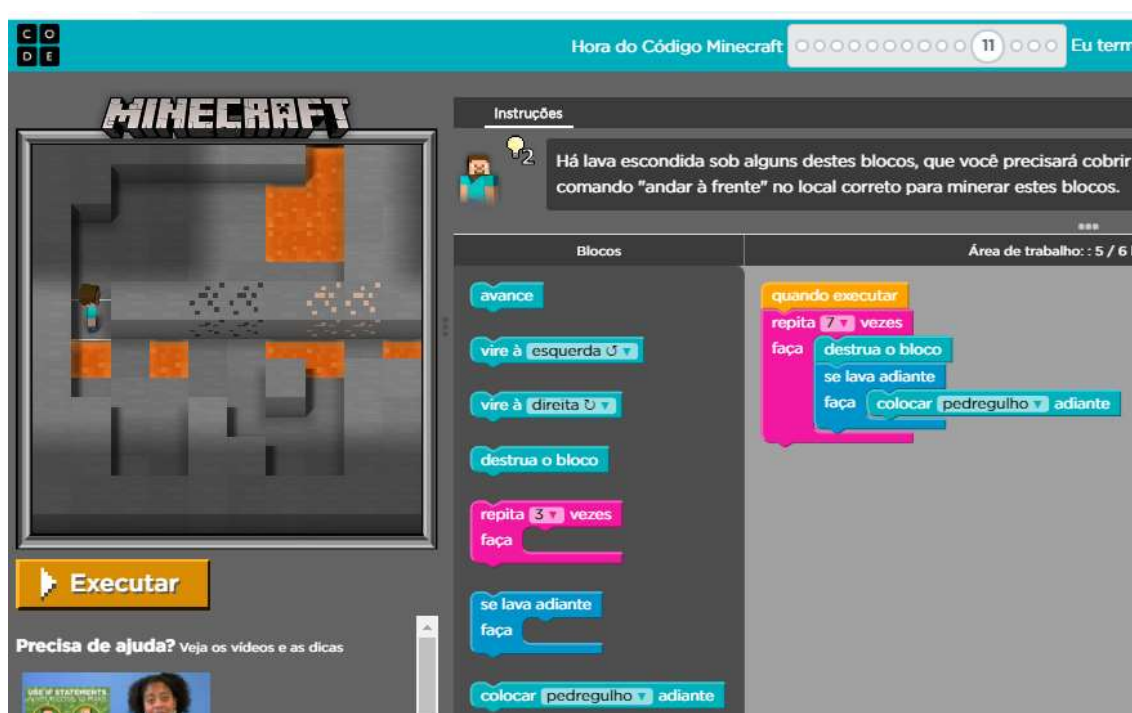


Espera-se que o aluno compreenda a estrutura condicional, elevando seu nível de Abstração para encapsular um conjunto de comandos frequentemente repetidos e que pode ser utilizado em diferentes situações de resolução de problemas..

Atividade 10 - Segunda atividade para o 5º ano: Hora do Código Minecraft

A segunda atividade faz parte das atividades da Hora do Código e conta com o personagem do *Minecraft*®, conforme Figura 15.

Figura 15: Ambiente da atividade Hora do Código Minecraft



Fonte: <https://studio.code.org/s/mc/stage/1/puzzle/11>

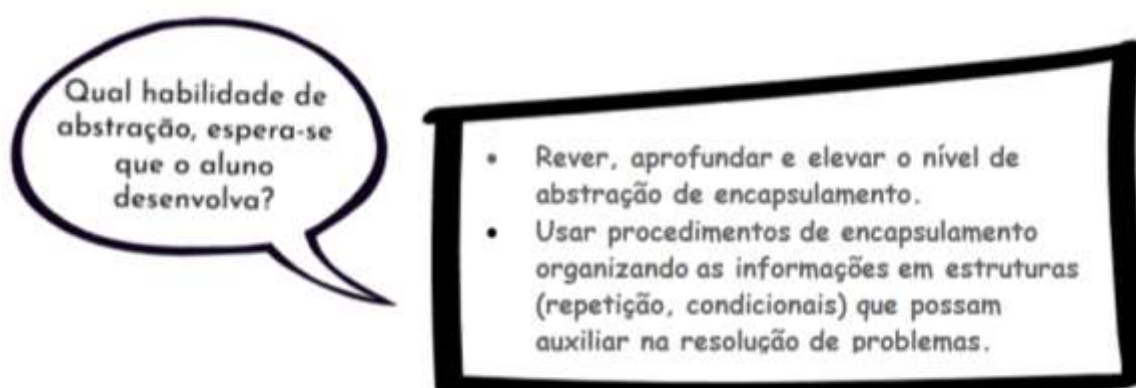
A atividade tem, como objetivo, reforçar o aprendizado das estruturas de programação já aprendidas, como as estruturas sequenciais, os laços de repetição (*loops*) e as condicionais.

Nesta atividade, o aluno não deve deixar o personagem do Minecraft cair na lava. O desafio é que a lava está sob as pedras, impossibilitando a visualização. É necessário, então, que o aluno programe o personagem para verificar se existe lava naquele lugar antes de avançar. *SE* sim, *ENTÃO* ele precisa

colocar uma pedra na frente antes de avançar, até chegar ao local correto para minerar esse bloco

Situação-problema da atividade Hora do Código Minecraft

O aluno deverá programar o personagem para que chegue até o destino sem deixar que caia na lava. Com isso, desenvolverá a habilidade de Abstração exposta abaixo.



Espera-se que, frente a um cenário com diversas informações, o aluno compreenda um problema complexo e consiga extrair, em meio a tantas informações, a que é relevante para atingir o objetivo proposto.

No Quadro 4, sintetizamos a presente proposta com definições do ano no qual a atividade será aplicada, o nome com link para acessar a atividade, a situação-problema e a habilidade de Abstração a ser desenvolvida pelo aluno.

Lembramos que os alunos devem realizar todas as atividades sugeridas em cada fase e que descrevemos apenas 2 atividades por ano para exemplificar a habilidade desenvolvida nas respectivas fases da atividade descrita.

Quadro 3: Síntese com link das atividades propostas para o ensino da Abstração

Ano	Atividades	Situação-problema	Habilidade de Abstração
1º	Mapa divertido	Localizar qual caminho (criando um algoritmo ao juntar as setas) leva Flurb ao tesouro.	Decidir quais detalhes de um problema são importantes. Decidir quais detalhes de um problema podem ser ignorados.

	<u>Criar um Monstro</u>	Identificar o que cada monstro (ex.: Zumbi, Frankstein, Duende) tem de diferente e abstrair as diferenças para criar uma lista de instruções (algoritmos) que possam ser usadas por todos para recriar um dos monstros, orientados por essas instruções.	Abstrair os detalhes que são responsáveis pelas diferenças para criar uma lista de instruções (algoritmos).
2º	<u>Labirinto Sequência</u>	Programar o personagem <i>Angry Birds</i> para que cumpra o desafio de chegar até o porco desviando de objetos encontrados pelo caminho e usando somente a quantidade necessária de blocos (quantidade limitada).	Decidir quais detalhes de um problema são importantes e quais detalhes de um problema podem ser ignorados (em um contexto de maior complexidade).
	<u>Labirinto Laços</u>	Programar, como na atividade anterior, o personagem <i>Angry Bird</i> para que chegue até o porco desviando de objetos encontrados pelo caminho, mas, agora, nessa atividade, foi aprofundado o grau de dificuldade, uma vez que diminuiu a quantidade máxima de blocos a ser utilizada, e acrescentado o bloco com o comando, “Repita” necessário para resolução do desafio proposto.	Reforçar habilidade da atividade anterior. Identificar e eliminar os elementos que se repetem em um determinado problema, encapsulando os elementos repetidos. Decidir quais detalhes de um problema podem ser ignorados.
3º	<u>Abelha: Laços de Repetição</u>	Programar a personagem para que percorra o labirinto identificando e coletando a quantidade de mel e o néctar a produzir, utilizando a estrutura de repetição.	Compreender e utilizar a modularização. Compreender o reuso dos blocos de programação.
	<u>Atividade Abelha Condicionais</u>	Programar, a personagem Abelha para que percorra o labirinto e colete néctar ou faça mel sem saber a quantidade a se obter, uma vez que a quantidade não está explícita na atividade, procedimentos que devem ser executados com a estrutura de condicionais.	Capacidade de extrair, dentro de um problema matemático, conceitos fundamentais para a resolução de problemas encapsulando-o em estruturas de condicionais.
4º	<u>Labirinto condicionais - Abelhas</u>	Programar o personagem para que escolha qual a melhor decisão a ser tomada para chegar ao destino, utilizando a estrutura de condicionais. Exemplo: quando se deparar com uma curva, utilizar o comando SE para verificar a condição “virar à esquerda ou à direita”.	Usar procedimentos de encapsulamento organizando as informações em estruturas (repetição, condicionais) que possam auxiliar na resolução de problemas.
	<u>Labirintos Condicionais - Zumbi</u>	Retomar e aprofundar a atividade anterior (Labirinto condicionais) em que o personagem Zumbi verifica uma condição, se há caminho à frente, à direita ou à esquerda. Se sim, ele escolhe uma instrução; se não, escolhe outra instrução, ou seja, faz uma coisa OU outra.	

5º	<u>Fazendeira</u> <u>Laços Enquanto</u>	Programar a personagem Fazendeira para utilizar comandos já aprendidos nas atividades anteriores e utilizar o comando <i>ENQUANTO</i> . “Enquanto uma instrução na parte superior for verdadeira, faça alguma coisa”. Nessa atividade, a personagem deve ser programada para que, <i>ENQUANTO</i> houver uma pilha de terra, essa terra deve ser removida.	Usar procedimentos para encapsular um conjunto de comandos frequentemente repetidos e relacionados à tomada de decisão: estrutura de repetição e de condicionais.
	<u>Hora do Código</u> <u>Minecraft -</u> <u>Condicionais</u>	Programar o personagem do Minecraft de forma que não caia na lava. O desafio é que a lava está sob as pedras, impossibilitando a visualização. É necessário, então, que o aluno programe o personagem, utilizando as estruturas de repetição e condicionais para verificar se existe lava naquele lugar antes de avançar	Rever, aprofundar e elevar o nível de Abstração de encapsulamento. Usar procedimentos de encapsulamento organizando as informações em estruturas (repetição, condicionais) que possam auxiliar na resolução de problemas.

Fonte: a autora.

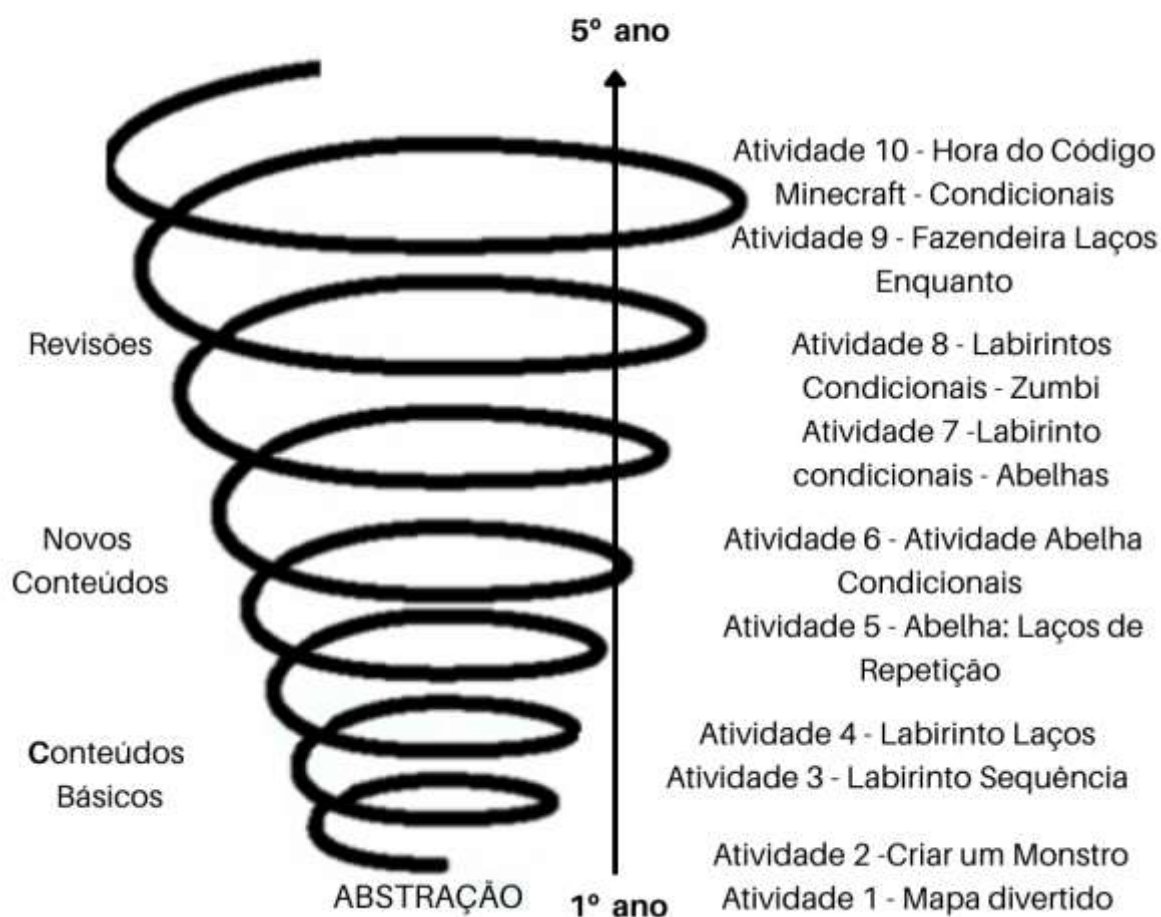
Conforme apresentado no Quadro 4, esta proposta de ensino apresenta sugestões de práticas pedagógicas com uso do computador e de práticas que não utilizam o computador (atividades desplugadas) para o ensino de Abstração, enquanto unidade do PC nos anos iniciais da EB.

4 PROPOSTA DE ENSINO EM ESPIRAL

Conforme já mencionado, esta PTE traz uma proposta para o ensino de Abstração para os anos iniciais do EF, com atividades que visam estimular o desenvolvimento do raciocínio abstrato, utilizando recursos da plataforma Code.org e uma abordagem de ensino em espiral proposta por Bruner(1960). Esta abordagem propõe que um mesmo tema comece a ser ensinado o mais cedo possível, aplicado em diferentes contextos e revisitado em todos os níveis de ensino, cada vez com maior grau de complexidade e de profundidade, conforme Figura 16.

Figura 16: Esquema do Ensino em Espiral adaptado para o ensino da Abstração para os anos iniciais do EF

Ensino de Abstração enquanto unidade do Pensamento Computacional - 1º ao 5º ano do Ensino Fundamental



Nesta proposta, focaremos na base da espiral com conteúdos básicos introdutórios do 1º ao 5º ano e com atividades selecionadas da Code.org, de acordo com o ano de ensino, com as proposições à saber:

1º ano

Compreender uma situação-problema criando e identificando sequências de passos (algoritmo) para a solução de um problema com base na capacidade de decidir quais detalhes de um problema são importantes ignorando detalhes irrelevantes.

2º ano

Resolver e elaborar problemas que envolvam a identificação e eliminação de informações redundantes (duplicadas) utilizando estratégias como a de usar procedimentos para encapsular um conjunto de comandos frequentemente repetidos, iniciando com a estrutura de repetição simples com o comando *REPITA*.

3º ano

Resolver e elaborar problemas que envolvam a identificação de subproblemas comuns em problemas maiores e a possibilidade do reuso de soluções, utilizando procedimentos de encapsulamento, como a estrutura de repetição, comando *REPITA*, e a estrutura de condicionais utilizando os comandos *SE* e *ENTÃO*.

4º ano

Resolver e elaborar problemas utilizando a modularização, ou seja, construindo algo grande unindo conjuntos de partes (peças) menores utilizando encapsulamento de estrutura de repetição e condicionais, como a estrutura *SE-ENTÃO-SENÃO* em que, com base no resultado de uma expressão booleana (*VERDADEIRO* ou *FALSO*), o fluxo do algoritmo segue para um bloco de instruções ou não.

5º ano

Resolver e elaborar problemas envolvendo a modularização e o reuso dos blocos de programação utilizando o encapsulamento com a estrutura de repetição *ENQUANTO*, em que os passos dentro desse bloco são repetidos enquanto a expressão booleana (*VERDADEIRO* ou *FALSO*) resultar *VERDADEIRO*.

5 CONSIDERAÇÕES FINAIS

Com base na análise dos estudos realizados para a elaboração desta pesquisa, compreendemos que a Abstração é uma das principais habilidades que os alunos devem adquirir no desenvolvimento do PC (WING, 2006). Identificamos, porém, alguns desafios, como pesquisas escassas do conceito e ensino de Abstração na área da CC, além de a Abstração ser considerada difícil de ser ensinada, principalmente para os alunos mais novos, menores de 12 anos de idade.

Frente a essa realidade, elaboramos uma proposta para o ensino de Abstração, enquanto unidade do PC, para alunos dos anos iniciais da EB, com base na abordagem de ensino em espiral de Bruner(1960), utilizando as atividades da plataforma Code.org e alicerçadas nas diretrizes curriculares da SBC e do CIEB, alinhadas as propostas da BNCC.

Esperamos que esta abordagem, junto com as atividades do Code.org, possam estimular a capacidade de Abstração com suas práticas simuladas e experimentadas, que contribuam para a elaboração de um Produto Educacional que proporcione às crianças, desde o início do EF, o desenvolvimento do raciocínio abstrato de acordo com a sua faixa etária, propiciando a habilidade de Abstração de forma progressiva e significativa para a criança.

A presente proposta foi desenvolvida visando contribuir com o ensino da Abstração enquanto uma das unidades que compõem o PC, com desenvolvimento de atividades e elaboração de rubricas de avaliação, na expectativa de contribuir com pesquisas e materiais para a aplicação prática em sala de aula de forma que possa ser trabalhada tanto como um tema transversal em quaisquer áreas quanto na disciplina de Informática.

Esperamos, também, trazer contribuições para que esta PTE possa ser utilizada na formação de professores e em orientações para o currículo de Ensino de Computação.

REFERÊNCIAS

- ALVES, N. D. C. *et al.* Ensino de computação de forma interdisciplinar em disciplinas de história no ensino fundamental: um estudo de caso. **Revista Brasileira de Informática na Educação**. Florianópolis. 2016
- BARRADAS, R., Lencastre, J. A., Soares, S., & Valente, A. (orgs.) (2019). **Desenvolvimento do pensamento computacional em idades precoces usando a plataforma code**. In E. C. da Silva & S. L. Javaroni (org.) *Pensamento Computacional: pesquisas, práticas e concepções*. São Paulo: UNESP.
- BRENNAN, K.; RESNICK, M. New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association*, Vancouver, Canada.
- BRUNER, J. **O Processo da Educação**. 5. ed. Harvard University Press, Cambridge, Ma.1966.
- CETIN, I.; DUBINSKY, E. Reflective abstraction in computational thinking. **Journal of Mathematical Behavior** 47 (2017) 70–80.
- CIEB: **Centro de Inovação para a Educação Brasileira**. Currículo de Referência em Tecnologia e Computação. Disponível em: <https://curriculo.cieb.net.br/impressao/>.
- CODE. **CODE, 2020**. Disponível em: <https://code.org/>
- COSTA, S. B.; **Desenho de interface para o desenvolvimento do Pensamento Computacional no Ensino Básico**: análise do Scratch. Dissertação (Mestrado em Novos Media e Práticas Web). Universidade Nova de Lisboa (FCSH/NOVA)
- CURZON, P. M. DORLING, T. Ng, C. SELBY, and J. WOOLLARD. Developing computational thinking in the classroom: a framework. **Computing at School, Available from community.computing at school.org.uk/resources/2324**, June 2014.
- FULLER, U., JOHNSON, C., AHONIEMI, T. et al (2007). Developing a computer science specific learning taxonomy. **ITiCSE working group report on innovation and technology in computer science education**. doi: 10.1145/1345443.1345438
- KRAMER, J. **Is abstraction the key to computing? Communications of the ACM** . 50, 4 (2007), 36-42
- LIEBE, C. **An Examination of Abstraction in K-12 Computer Science Education**. 2019. Tese (Doutorado em Filosofia)Walden University. Michigan State University
- MELHAM, T. Modelling, abstraction, and computation in systems biology: A view from computer science **Progress in Biophysics and Molecular Biology**. Volume 111, Issues 2–3, April 2013, Pages 129-136
- ROSE, S.P.; HABGOOD, M.P.J.; JAY, T.; Using Pirate Plunder to Develop Children's Abstraction Skills in Scratch. CHI EA '19: Extended Abstracts of the 2019 CHI

Conference on Human Factors in Computing Systems May 2019 Paper No.: LBW0172 Pages 1–6

ROSE, S.P.; HABGOOD, M.P.J.; JAY, T.; Pirate Plunder: Game-Based Computational Thinking Using Scratch Blocks. In Proceedings of the 12th **European Conference for Game Based Learning**. 556–564.

SELBY, C.C. Relationships: computational thinking, pedagogy of programming, and Bloom's Taxonomy. WiPSCE '15: **Proceedings of the Workshop in Primary and Secondary Computing Education**. November 2015 Pages 80–87

SBC - **Sociedade Brasileira de Computação. Diretrizes para ensino de Computação na Educação Básica.** Disponível em <https://www.sbc.org.br/educacao/diretrizes-para-ensino-de-computacao-na-educacao-basica>. Acessado em 01/05/2020.

SBC - **Sociedade Brasileira de Computação. Currículo de referência.2017.** Disponível em: <https://www.sbc.org.br/documentos-da-sbc/category/131-curriculos-de-referencia>. Acessado em 01/09/2020.

STATTER, D; ARMONI M. Teaching Abstraction in Computer Science to 7th Grade Students. **ACM Transactions on Computing Education** January 2020 Article No.: 8

WING, J. Pensamento Computacional – Um conjunto de atitudes e habilidades que todos, não só cientistas da computação, ficaram ansiosos para aprender e usar. **Revista Brasileira de Ensino de Ciência e Tecnologia**, v. 9, n. 2, 2016. Disponível em: <<https://periodicos.utfpr.edu.br/rbect/article/view/4711>>. Acesso em 03 jan de 2019