

Capítulo

1

Desenvolvendo o Pensamento Computacional por meio de Histórias: Uma Estratégia para Docentes do Ensino Básico

Juliana B.S. França, Beatriz Saburido, Angélica F.S. Dias

Abstract

Over the years, there has been a need to promote strategies for the dissemination of computing at basic levels of education in Brazil. There is, therefore, a genuine interest on the part of educators in promoting playful digital strategies in their pedagogical activities with their students to bring the school and academic universe closer to the reality of students. In this sense, this mini-course promotes the dissemination of the StoryCoding method for Educators to develop logical-creative skills in teachers, enabling them to promote computing in their teaching areas. In this mini-course, a historical overview will be drawn on the potential of computational thinking and how this area of knowledge can be applied to strengthen other disciplines through stories.

Resumo

Ao longo dos anos tem-se observado a necessidade de promover estratégias de disseminação da computação nos níveis básicos de educação no Brasil. Percebe-se, portanto, um real interesse dos educadores em promoverem estratégias lúdico-digitais em suas atividades pedagógicas com seus alunos, a fim de aproximar o universo escolar e acadêmico da realidade dos alunos. Neste sentido, este minicurso promove a disseminação do método StoryCoding para Educadores, com a finalidade de desenvolver competências lógica-criativas nos docentes, capacitando-os a promoverem a computação em suas diferentes áreas de ensino. Neste minicurso será traçado um panorama histórico sobre o pensamento computacional, e como esta área de conhecimento pode ser aplicada para fortalecer outras disciplinas através de histórias.

1.1. Introdução

Embora muitas pesquisas tragam os conceitos do pensamento computacional, poucas investigações destacam a importância da narrativa para apoiar a construção do conhecimento através de histórias. Este conhecimento pode contemplar de forma

gradativa a compreensão de que a máquina segue uma sequência de instruções que implementa um algoritmo. No entanto, para a construção de um algoritmo é necessário identificar e particionar o problema e sua complexidade até que se possa expressar suas soluções através de experiências e conhecimento prévio.

A literatura mostra que a linguagem de programação em blocos tem sido amplamente utilizada com o público infantil com o objetivo de desenvolver o pensamento computacional em crianças [França e Tedesco, 2019]. Aono et al. (2017), Rodriguez et al. (2015), discutem projetos cujo objetivo é o desenvolvimento do pensamento computacional em crianças por meio da programação de jogos com Scratch. Nessas pesquisas existe uma discussão efetiva sobre a inserção de competências computacionais nos Ensinos Fundamental e Médio, visando diminuir as dificuldades apresentadas pelos aprendizes na área de exatas, uma vez que tais dificuldades podem ser responsáveis pelo abandono de alunos dos cursos de computação na universidade. Esses autores defendem que quanto mais cedo os indivíduos desenvolvem competências lógicas, mais natural se torna o processo de aprendizagem da área de exatas e conseqüentemente das áreas de tecnologia. Outras pesquisas com foco no pensamento computacional também têm se destacado quanto ao desenvolvimento de competências acadêmicas que não necessariamente estão relacionadas ao desenvolvimento de software e jogos, por crianças [Mattos et al., 2017; Schneider et al. 2019; Junior et al. 2019].

A literatura traça o mapeamento simplificado de estratégias que conduzem o ensino da programação de computadores em Scratch através de narrativas (uso da técnica *Storytelling*), pode-se destacar Farias et al. (2019) e Gomes et al. (2017). A construção de histórias pode apoiar o fortalecimento e a construção de novos conhecimentos, do ponto de vista do aluno e do professor. Através destas narrativas, juntamente com conceitos - noções de estruturas de repetição, condicionais, paralelismo, análise de dados entre outros -, pode-se disseminar o ensino da programação combinando conceitos de pensamento computacional e programação em blocos.

Este capítulo tem por objetivo discutir o desenvolvimento do pensamento computacional através de histórias, na perspectiva do professor do ensino básico, através do método *StoryCoding*, que visa promover junto aos professores do ensino básico, a programação de narrativas em sala de aula. Na Grécia antiga, Platão dizia que todo aprendizado tem algum tipo de base emocional [Paviani 2013], seja ela positiva ou negativa. Faz parte desta pesquisa promover a interação de ensino-aprendizagem do pensamento computacional de forma lúdica e lógica, na medida que as práticas tecnológicas também influenciam as crianças e jovens para a área de STEM. Através da manipulação do Scratch elas são capazes de criar suas próprias histórias animadas e reproduzir para outros indivíduos suas experiências de forma lúdica, informal e em linguagem acessível ao universo infanto-juvenil.

Este capítulo está organizado em quatro seções. Na primeira são apresentados os elementos motivadores e objetivos deste capítulo. Na seção seguinte são apresentados os fundamentos do pensamento computacional e suas aplicações, com destaque para o ensino de programação através de uma linguagem em blocos, e através de narrativas. A

terceira seção apresenta e discute o método StoryCoding e por fim o capítulo é finalizado com as perspectivas futuras da área de ensino de programação para crianças e jovens a partir das práticas docentes no ensino básico.

1.2. Fundamentos do Pensamento Computacional e suas Aplicações

Com as constantes revoluções tecnológicas que cercam o mundo moderno, juntamente com o advento das linguagens de programação, os computadores estão cada vez mais presentes no dia a dia da humanidade. Segundo um estudo do Núcleo de Informação e Coordenação do Ponto BR – NIC.br (2021), estima-se que, em 2020, aproximadamente 81% da população brasileira com dez anos ou mais era usuária da rede de Internet, um aumento de sete pontos percentuais em relação a 2019 (74%). Nesse contexto, é importante que as pessoas sejam preparadas desde a infância para esta nova realidade, naturalizando esses novos conceitos.

Segundo Ignácio (2018), o Pensamento Computacional se apresenta como uma técnica de raciocínio e de comportamento que pode fazer com que o humano, ao atuar nas diversas áreas do conhecimento, com ou sem o uso de computador, tenha maior facilidade para organizar o pensamento, para trabalhar de forma colaborativa e para resolver problemas. No entanto, é importante destacar, que esta expressão pode sugerir um entendimento contrário, uma vez que essa modalidade de pensamento não está diretamente ligada ao uso de computadores e linguagens de programação.

Um dos precursores do conceito Pensamento Computacional foi Seymour Papert, em 1980, com seu trabalho “Mindstorms: Children, Computers and Powerful Ideas”, ele é considerado um dos principais pensadores que estudava sobre como a tecnologia pode influenciar a aprendizagem. Neste trabalho, temos:

"(...) eu tenho me descrito como um utópico educacional ~ não porque eu tenha projetado um futuro de educação em que as crianças são cercadas por alta tecnologia, mas porque acredito que certos usos da tecnologia computacional e das ideias computacionais podem fornecer novas possibilidades às crianças." pag 17.

Porém, somente a partir de 2006, o termo “Pensamento Computacional” começou a repercutir, através do trabalho de Jeannette Wing (2006). [Wing, 2006] afirma que o Pensamento Computacional é uma habilidade fundamental para todos, envolvendo a resolução de problemas, projeção de sistemas, e compreensão do comportamento humano, através de conceitos da ciência da computação.

O Pensamento Computacional ajuda a resolver problemas através da construção de pensamentos lógicos e reconhecimento de padrões. Brackmann (2017) diz que o Pensamento Computacional possui quatro pilares para apoiar esse objetivo (Figura 1), são eles:

1. decomposição: deve-se identificar um problema complexo e quebrá-lo em partes menores, com seus subproblemas, facilitando o gerenciamento;

2. reconhecimento de padrões: os subproblemas serão analisados individualmente com maior profundidade, investigando se problemas parecidos já foram solucionados anteriormente, dessa forma, é possível aplicar a resolução do problema ao contexto desejado;
3. abstração de um problema: deve-se focar apenas nos detalhes que são mais importantes, ignorando informações irrelevantes;
4. algoritmos: deve-se criar passos simples para resolver cada um dos subproblemas encontrados

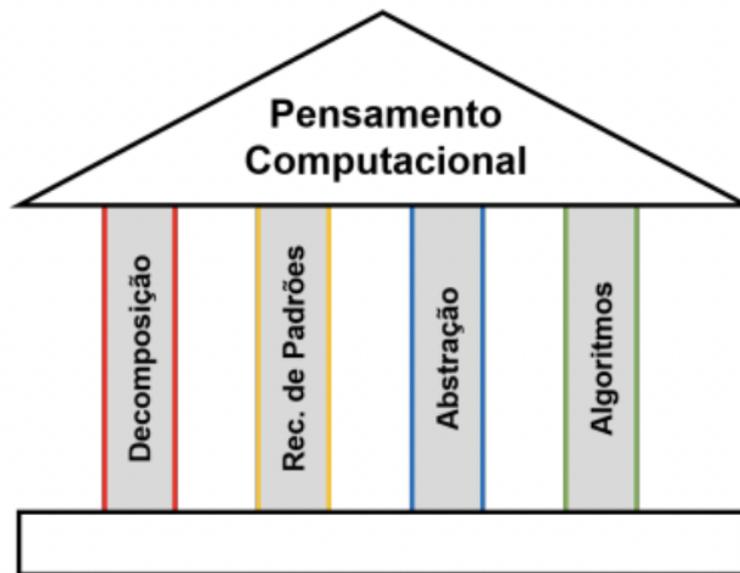


Figura 1: Pilares do Pensamento Computacional [Brackmann, 2017].

Nesse contexto, para atingir o objetivo principal de fazer um bolo, por exemplo, a aplicação dos pilares, resumidamente, seria: dividir a receita em categorias, conhecer as suas resoluções e os passos a serem seguidos.

1. ingredientes necessários: existem os ingredientes disponíveis? A pessoa deve checar se existem os ingredientes, caso contrário, ela deverá comprar;
2. infraestrutura necessária: existem um forno, formas e utensílios de cozinha disponíveis e adequados para uso? A pessoa deve checar se existem os acessórios necessários, caso contrário, ela deverá comprar;
3. modo de preparo adequado: a pessoa que executará a receita possui as habilidades necessárias para seguir o modo de preparo? A pessoa deverá checar se consegue fazer as receitas, caso contrário, ela deverá aprender as técnicas necessárias exigidas pelo preparo;

[Wing, 2006] adiciona que:

“Pensamento computacional é usar raciocínio heurístico na descoberta de uma solução. É planejar, aprender e agendar na presença da incerteza. É pesquisar, pesquisar e pesquisar

mais, resultando em uma lista de páginas da web, uma estratégia para vencer um jogo ou um contraexemplo.

(...)

Quando sua filha vai para a escola pela manhã, ela coloca em sua mala as coisas que precisará para o dia; isso é *prefetching* e *caching*. Quando seu filho perde suas luvas, você sugere que ele refaça seus passos; isso é *backtracking*.

(...)

O pensamento computacional vai ter se tornado impregnado na vida de todo mundo quando palavras como algoritmos e precondition tornarem-se parte do vocabulário.”

Num contexto também não muito distante da nossa realidade atual, o Pensamento Computacional também é útil para auxiliar na mitigação de riscos relacionados à segurança digital de crianças, adolescentes e adultos. No trabalho de [Kaefer, 2021], o Pensamento Computacional é aplicado dentro de um ambiente computacional para estimular comportamentos seguros na Internet.

Desse modo, o Pensamento Computacional é útil em vários aspectos da vida, sejam eles pessoais, empresariais ou escolares, por exemplo, no contexto escolar, quando alunos resolvem problemas complexos de matemática, fragmentando-os em partes menores e, através da lógica, resolvem primeiro as pequenas partes e depois avançam, diminuindo a sua complexidade.

1.2.1. Pensamento Computacional Desenvolvido Por Meio de Histórias

Histórias fazem parte da cultura dos seres humanos, segundo [Mateus et al., 2013], elas são uma maneira mais significativa que a humanidade encontrou para expressar experiências que nas narrativas realistas não acontecem. As histórias geralmente descrevem uma série de ações e eventos, que possuem uma ordem cronológica, organizada de acordo com as relações de causalidade entre as ações e os eventos. De acordo com Chatman (1978), as narrativas são divididas em duas categorias: a parte da história em si, que é composta justamente pela soma dos acontecimentos com os elementos existentes da cena (personagens e itens do cenário); a parte do discurso, que é composta pela forma que o conteúdo é contado ao receptor (Figura 2). O autor ainda adiciona que as histórias são compostas de eventos, compostos por ações e acontecimentos, e elementos existentes, compostos por personagens e cenários.

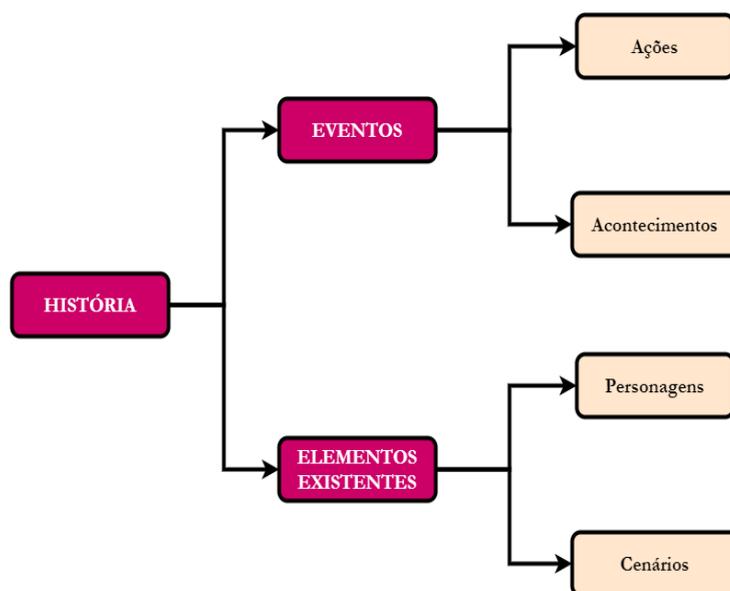


Figure 2: Caracterização das narrativas [Chatman, 1978], adaptado pelos autores.

Em [Graesser et al., 1980] temos que as histórias possuem variáveis de diversos tipos, uma trama desenvolvida a partir das relações estabelecidas pelo tema, noções de estruturas de repetição, condicionais, paralelismo, análise de dados, entre uma série de outros conceitos que foram estendidos para a área da Ciência da Computação, que fazem parte de como se manipula os objetos do mundo real, abstratos ou não.

Nesse contexto, Pires et al. (2018) correlacionaram elementos trabalhados em crianças, durante o processo de criação de histórias, com os pilares do Pensamento Computacional:

Tabela 1: Atividades desenvolvidas na criação de histórias e os pilares do Pensamento Computacional [Pires et al., 2018]

Pilar do Pensamento Computacional	Atividades Desenvolvidas
Decomposição	Para desenhar o <i>Storyboard</i> as crianças faziam “quadrinhos” no papel, para desenhar os acontecimentos em ordem cronológica. Ao decompor a história em diversos quadros, as crianças tornaram mais fácil o trabalho de transpor a história (escrita) para o desenho
Reconhecimento de padrões	Após o processo de separação por quadros, a criança pôde reaproveitar o desenho do personagem para diversos quadros. Ou seja, ao reconhecer os padrões entre frases da história, foi possível desenhar objetos baseados em outros já existentes, facilitando o processo
Abstração	No processo de construção dos desenhos, as crianças selecionam os elementos mais importantes da história para desenhar, como por exemplo, personagens e cenários
Algoritmo	A história e o <i>Storyboard</i> são algoritmos, pois tem início, meio e fim. Além das estruturas lógicas encontradas dentro da história.

Por isso, o Pensamento Computacional pode ser desenvolvido através de histórias, porque o processo de criação de um programa de computador e de uma narrativa

possuem elementos coincidentes, que devem ser levados em consideração para ter um produto final coeso, dentre as coincidências pode-se destacar:

- Programas de computadores e histórias possuem início, meio e fim;
- Programas de computadores e histórias precisam de eventos que são gatilhos de ações, trabalhando o conceito de causalidade;
- Programas de computadores e histórias possuem atores e cenários, que são aqueles que vão desempenhar uma ação em um determinado contexto.

O trabalho de [Farias et al., 2019] apresenta o relato de experiência de uma oficina que visa estimular o pensamento computacional, que também utilizou histórias para envolver as crianças participantes e seus familiares. Os autores afirmam que esse tipo de experiência lúdica e motivadora é capaz de estimular a capacidade lógica, cognitiva e criativa das crianças e destacam como outros benefícios de suas oficinas alguns pontos, como:

- Estímulo do raciocínio lógico, criatividade, autonomia, capacidade de resolução de problemas e trabalho em equipe, nas crianças beneficiadas pelo projeto;
- Aproximação da escola com as famílias.

Já o trabalho de [Silva, 2021] investiga a contribuição da programação em blocos e do pensamento computacional no ensino e aprendizagem da língua portuguesa através de histórias. Usando como ferramenta a plataforma Scratch, a autora visa articular a área do Português com as Tecnologias de Informação e Comunicação, com o objetivo de incentivar o domínio da Educação Literária de uma forma lúdica e divertida.

1.2.2. Pensamento Computacional Desenvolvidos através da Programação em Blocos

Para desenvolver o pensamento computacional em alunos do ensino básico é necessário estimular a formação de professores a fim de que eles sejam capacitados não apenas em ensinar programação, mas também estejam aptos em prover uma experiência de aprendizagem aos alunos de forma interdisciplinar [Fagerlund et al., 2022].

Por causa da imersão tecnológica que o mundo se encontra, estimular o pensamento computacional deveria ser uma tarefa dos educadores, daqueles que de fato acompanham os estudantes no seu dia-a-dia. Isso fica mais claro através do estudo de [Brezolin e Silveira, 2021], em que apontam que a maior parte das ações que envolvem a programação em blocos como disseminadora do pensamento computacional são realizadas no meio acadêmico, por estudantes de graduação e pós-graduação, sem o acompanhamento de seus professores. Os autores acrescentam que não fica claro o retorno dessas pesquisas para o ambiente escolar, criando uma necessidade de se investigar, também, o que realmente estes professores conhecem deste conjunto de ferramentas e como podem usufruir desse arsenal.

Esta transformação da programação textual surgiu com a intenção de mitigar as dificuldades encontradas no ensino de programação e, com isso facilitar a aprendizagem através da construção de ambientes de Linguagem de Programação Visual (Visual Programming Language - VPL), comumente conhecido como ambientes de

programação em blocos [Perni, 2021]. A aplicação da combinação dos conceitos de Pensamento Computacional e Programação em Blocos vem se tornando cada vez mais frequente. De acordo com a revisão sistemática de [Sousa et al., 2020], um total de 19 países já apresentam publicações sobre o uso de Programação em Blocos no ensino do Pensamento Computacional.

O uso da programação em blocos para o ensino do pensamento computacional é utilizado de várias maneiras, no trabalho de [Bobsin et al., 2020], por exemplo, os autores utilizaram esses conceitos para auxiliar a resolução de problemas investigativos de matemática na escola básica, através da apresentação de problemas graduais da plataforma code.org, para introduzir o desafio da solução de um problema específico no final.

1.3. Metodologias Ativas e o Desenvolvimento do Pensamento Computacional Por Meio de Histórias

A metodologia ativa é qualquer abordagem onde todos os alunos participam ativamente do processo de aprendizagem. O aprendizado ativo pode assumir muitas formas e ser trabalhado em qualquer disciplina. Comumente, os alunos se engajam em atividades pequenas ou grandes centradas em escrever, falar, resolver problemas ou refletir. A metodologia ativa está em contraste com os métodos tradicionais de ensino, pela mudança de papel do aluno. Na tradicional, os alunos são receptores passivos de conhecimento de um especialista. Na ativa, os alunos participam ativamente da construção do conhecimento com o apoio de um especialista. A mudança fundamental está no papel que o professor assume, passando de protagonista para coadjuvante no processo de aprendizado de seus alunos. De acordo com [Costa et al., 2020], as principais vantagens das metodologias ativas (centrada no aluno) são:

- Estimular as diferentes formas dos alunos aprenderem
- Estimular as competências da era atual
- Deixar as aulas mais dinâmicas

Enquanto que as principais desvantagens das metodologias ativas são destacadas por:

- Maior dificuldade na garantia de um aprendizado comum a todos
- Difícil aceitação dos alunos no começo (já que exige mais esforço)

Algumas metodologias ativas ganharam muita notoriedade nos últimos anos, tanto no meio científico, quanto em sala de aula. Vários pesquisadores buscaram comprovar a efetividade dessas metodologias, enquanto elas são praticadas por professores brasileiros em sala de aula. Dentre as principais metodologias ativas podemos citar: Ensino Híbrido, Aprendizagem baseada em problemas, Aprendizagem baseada em projetos, Rotinas de pensamento (*Thinking Routines*), Aprendizagem baseada em equipes. Na próxima subseção serão discutidos em detalhes as metodologias destacadas.

1.3.1. Métodos de Ensino

Nesta seção apresentaremos alguns métodos ativos de ensino difundidos na literatura. Para cada método será apresentada sua teoria correspondente, e o método baseado na

solução de problemas será especificado a fim de discutir a sua aplicabilidade com o método proposto para o desenvolvimento do pensamento computacional através de história, em sala de aula por educadores.

Ensino Híbrido - Sala de aula invertida (*Flipped Classroom*)

O *Flipped Classroom* é um modelo de educação formal que se caracteriza por mesclar dois modos de ensino: Online e Offline. Nesse sentido, a ideia é que a parte on-line e off-line se conectem e se complementam, proporcionando diferentes formas de ensinar e aprender. Existem vários modelos de ensino híbrido: os sustentados e os disruptivos. Os sustentados são conhecidos como: “rotação por estações”, “laboratório rotacional”, “rotação individual” e “sala de aula invertida”. Já os disruptivos são: “modelo flex”, “modelo à la carte”, “modelo virtual enriquecido” [Reidsema et al., 2017; Keengwe, 2014; Stohr et al., 2020; Schneiders, 2018; Trevelin et al., 2013; Awidi e Paynter, 2019].

Aprendizagem baseada em equipes (*Team-Based Learning*)

O *Team-Based Learning* [Michaelsen et al., 2008], [Reimschisel et al., 2017] é uma estratégia de ensino projetada para apoiar o desenvolvimento de equipes de aprendizado de alto desempenho e oferecer oportunidades para essas equipes se engajar em tarefas significativas de aprendizado [Fink, 2004]. A aprendizagem baseada em equipes é baseada em quatro princípios: (i) Os grupos devem ser adequadamente formados e gerenciados; (ii) os alunos devem ser responsabilizados por seus trabalhos em grupo; (iii) as tarefas de grupo devem promover tanto o aprendizado quanto o desenvolvimento de equipes; e (iv) os alunos devem ter feedback de desempenho frequente e oportuno [Johnson, 2009].

Aprendizagem baseada em problemas

A aprendizagem baseada em problemas (*Problem Based Learning*), é a construção do conhecimento a partir da discussão em grupo de um problema. Nela, o aluno estuda individualmente sobre determinado assunto antes da aula, e registra todas as suas dúvidas ou dificuldades. Durante a aula, acontecem discussões em grupo sobre os problemas apresentados e diferentes pontos de vista são considerados sobre a questão. Esta dinâmica tem por papel enriquecer os envolvidos, desenvolvendo neles novas competências e permitindo que novos horizontes sobre o problema sejam atingidos. Desta forma, a participação de cada aluno no grupo de discussão é essencial, incentivando o trabalho em equipe e a comunicação.

1.4. Método *StoryCoding*

O método *StoryCoding* nasce com o objetivo de estimular o ensino e as práticas da lógica de programação em sala de aula. Seu papel é trazer suporte aos docentes do ensino básico, a fim de que estes, nas diferentes áreas de conhecimento, possam fazer uso deste método para consolidar os conhecimentos trabalhados nas diversas disciplinas, e estimular as práticas lúdicas, lógicas e criativas. O método *StoryCoding* atua sobre dois pilares básicos: as narrativas (tendo por fundamento a teoria do *Storytelling*) e o raciocínio lógico.

1.4.1. Narrativas através das histórias: *Storytelling*

Segundo Mcsill et al. (2013) o *Storytelling* é definido como o velho hábito de contar histórias. Desde nossos ancestrais, o ato de se reunir em volta de uma fogueira para, ao fim de cada dia, contar sobre o que foi vivenciado ao decorrer dele é a forma mais eficaz de perpetuar a cultura e as práticas e conhecimentos culturais tão importantes para a sobrevivência de grupos. Ou seja, *Storytelling* é a arte de contar uma história por meio da palavra escrita, da música, da mímica, das imagens, do som ou dos meios digitais.

A história vem para a sala de aula como forma de estimular a aprendizagem do aluno. Nos últimos anos, as escolas têm enfrentado dificuldade em atualizar suas atividades frente aos recursos influenciadores do aluno. Segundo Salim et al. (2016) as dificuldades no processo de aprendizagem estariam relacionadas a necessidade dos alunos de colocarem em prática as rotinas, planejamentos e controle de processos cognitivos envolvidos na realização de uma dada atividade. E essas dificuldades, segundo ele, são decorrentes do uso inapropriado dos mecanismos do processo de informação.

O *Storytelling*, no contexto escolar, tem sido aplicado para investigar métodos de ensino [Salim et al., 2016; Lage et al., 2017], e também, para estimular a criatividade e curiosidade do aluno nas aulas para que, deste modo, se tornem mais participativos e interessados. Segundo Robin (2019) o uso do *Storytelling* aplicado à educação enriquece o processo estimulando a criatividade dos alunos durante o desenvolvimento de suas histórias, complementando os métodos tradicionais de ensino e personalizando a aprendizagem dos estudantes, além de melhorar a inter-relação entre eles em todo o processo através de sua participação ativa na construção do conhecimento.

O *Group Storytelling* é uma técnica de construção de histórias colaborativa, onde 2 ou mais pessoas participam na reconstrução de um acontecimento ou fato, seja de modo síncrono ou assíncrono, localmente ou de lugares diferentes, em pontos de processos diferentes ou diversos meios e mídias diferentes [Borges, 2011]. Esse modo de reconstrução de conhecimento é interessante, pois se os contadores se inter-relacionam dessa forma o conhecimento gerado terá o rico acréscimo do modo de ver de cada membro do grupo, gerando assim uma visão completa da história com a experiência individual de cada um.

A introdução da técnica *Group Storytelling* na rotina escolar é uma atividade de aprendizagem valiosa, pois a construção da história permite a concatenação dos fragmentos da mesma história produzidos pelos participantes da técnica. Desta forma, os fragmentos construídos por um participante são complementados, refutados ou comparados aos fragmentos de outra pessoa. Assim, é possível ter uma visão mais real e detalhista sobre a história.

O uso da técnica de construção de história para a disseminação do pensamento computacional em sala de aula permite o compartilhamento de conhecimento e troca, fazendo uso através de cenários de aprendizagem. Estes cenários podem ser construídos de forma colaborativa tradicional para apoiar o desenvolvimento do pensamento computacional utilizando uma plataforma de ensino.

1.4.2. Estrutura *StoryCoding*

O Método *StoryCoding* (Figura 3) aborda o ensino da computação aplicada e visa criar cenários para estimular o desenvolvimento do pensamento computacional em aulas práticas. Este método considera o ensino da lógica de programação, aplicado às narrativas reais ou lúdicas experienciadas pelos alunos ou estimuladas em aula. As primeiras iniciativas que originaram o método proposto ocorreram em oficinas aplicadas com crianças do ensino básico [França et al., 2021; Racca et al., 2021; Saburido et al., 2021].



Figura 3: Método *StoryCoding*.

O princípio do *StoryCoding* é consolidar o raciocínio lógico na programação de histórias. Para isso, é associado, portanto, a programação em blocos e o *Storytelling* para a prática de ensino-aprendizagem na Educação Básica. O método *StoryCoding* é um método de ensino baseado em Boud e Feletti (1998), Hmelo-Silver (2004), Savin-Baden e Major (2004). Este é um método que foca no ciclo da aprendizagem baseada em problema, bem como nas estratégias de colaboração que devem ser planejadas a fim de potencializar a sua aplicação. O método *StoryCoding* pode ser usado, portanto, para: (1) definir ações; (2) sistematizar as atividades (que envolvem as ações); (3) definir cenários de operação; (4) definir atores participantes do cenário e das atividades; e (5) Caracterizar atores. A sistematização deste método em sala de aula desenvolve nos alunos competências lógicas e de simplificação de problemas, além de fomentar discussões colaborativas a fim de que o objetivo seja alcançado em grupo.

1.4.1.1 Prática do *StoryCoding*

A prática do *StoryCoding* é conduzida em quatro fases: (1) Planejamento da prática pelo professor; (2) Definição dos grupos de trabalho; (3) Construção das histórias; (4) Automação das histórias.

A Fase 1 deve ocorrer junto à construção do plano de aula do professor. Este docente deve definir quais conceitos básicos serão trabalhados com os alunos, desde as áreas de conhecimento referentes à disciplina lecionada, até os conceitos lógicos e criativos necessários para a construção das histórias pelos alunos. Na Fase 2 o professor deverá propor que a turma se organize em grupos de até 3 alunos. O critério para formação desses grupos deverá ser por afinidade dos alunos, ou seja, eles mesmos escolhem com quem gostariam de trabalhar. Para a Fase 3, cada grupo deverá construir a

sua narrativa de trabalho, tendo por tema central o proposto pelo professor. Esta narrativa pode ser real ou lúdica.

Para a prática de automação das histórias, foram previstas ações como: (1) definição da história, (2) abstração da história, (3) decomposição da história em cenários, (4) construção do algoritmo através de estruturas lógicas na linguagem de programação em blocos.

Após a definição e a abstração das histórias, as ações da narrativa são decompostas em cenários de menor complexidade e, cada grupo de alunos, deverá planejar as ações de cada personagem, introduzindo-os na história, desenvolvendo o enredo e conectando as ações relacionadas aos acontecimentos, na ordem cronológica da narrativa criada. Em seguida, os cenários decompostos são automatizados na linguagem de programação em blocos Scratch.

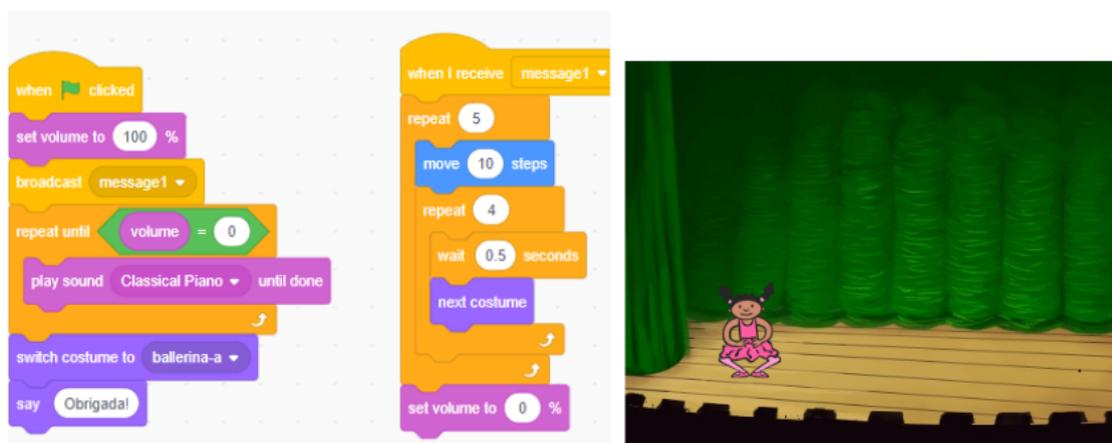


Figura 4: História da bailarina programada em Scratch por alunos do ensino básico.

Após a construção lógica da narrativa, os alunos apresentam seus projetos destacando a história programada e o seu código em Scratch. A Figura 4 mostra o resultado de uma história criada em Scratch por alunos do ensino básico. Esta história foi contada a partir da seguinte narrativa:

“A bailarina Cleo estava muito feliz. Esse era o seu grande dia! Após muita preparação, ela estava pronta para demonstrar 4 novas posições e repeti-las 5 vezes para o seu público. Sua apresentação estava para acontecer em um belo teatro. Cleo sonhava com as cortinas verdes abertas e o início de seu espetáculo.”

Considerando a narrativa acima, o grupo de alunos trabalhou elementos de repetição, atores, diálogos e caracterização de cenários, além de representar conceitos referentes às práticas de dança.

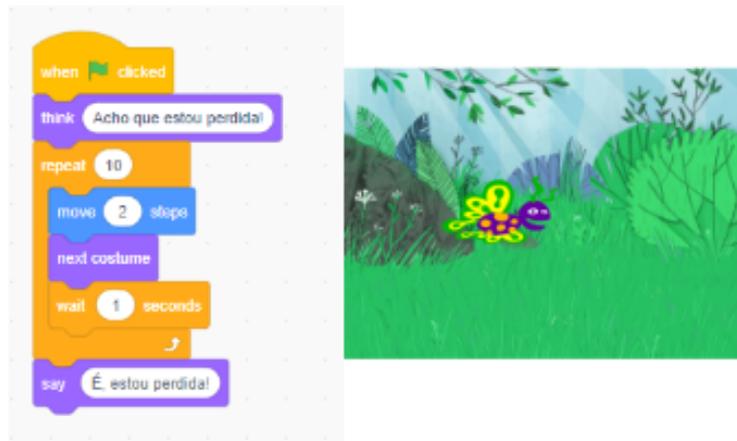


Figura 5: História da borboleta Sissi programada em Scratch por alunos do ensino básico.

A Figura 5 apresenta o resultado de uma história programada em Scratch por um grupo de alunos do ensino básico. Essa história narra a aventura da borboleta Sissi e trabalha aspectos como orientação espacial, caracterização de cenários, atores, e diálogos:

“A borboleta Sissi ama passear na floresta. Ela é muito sapeca! Em uma tarde de sol, borboleta Sissi saiu para passear em uma região da floresta desconhecida, e se perdeu. Em sua caminhada ela percebeu que estava perdida.”

1.5. Conclusão

Neste capítulo foi apresentado o Método *StoryCoding*, que tem por objetivo discutir o desenvolvimento do pensamento computacional através de histórias, na perspectiva do professor do ensino básico. Faz parte deste método apoiar os docentes do ensino básico no estímulo aos seus alunos sobre a prática da programação de computadores na linguagem Scratch, a fim de consolidar conceitos de diferentes áreas de conhecimento.

Este método tem por fundamento a solução de problemas, uma vez que seus princípios estão fundamentados na estratégia de simplificar grandes problemas em situações de menor complexidade a fim de que a automação seja conduzida. Este capítulo apresentou também a dinâmica de implementação do Método *StoryCoding* em sala de aula e discutiu dois exemplos práticos criados por grupos de alunos do ensino básico.

Como próximos passos do método *StoryCoding*, espera-se disseminá-lo entre docentes do ensino básico e acompanhar de forma sistemática a aceitação das práticas inerentes ao método pelos professores e alunos. Espera-se também manter um repositório das histórias programadas a fim de acompanhar o desenvolvimento dos alunos e seus grupos.

Agradecimentos

Esta pesquisa recebeu apoio parcial da Faperj através do projeto #E-26/211.367/2019 (248406), referente à pesquisadora Juliana França.

Referências

- Aono, A. H., Rody, H. V. S., Musa, D. L., Pereira, V. A., & Almeida, J. (2017, July). A utilização do scratch como ferramenta no ensino de pensamento computacional para crianças. In *Anais do XXV Workshop sobre Educação em Computação*. SBC.
- Awidi, I. T., & Paynter, M. (2019). The impact of a flipped classroom approach on student learning experience. *Computers & Education*, 128, 269-283.
- Bobsin, R., Nunes, N., Kologeski, A., & Bona, A. (2020). O Pensamento Computacional presente na Resolução de Problemas Investigativos de Matemática na Escola Básica. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação*, (pp. 1473-1482). Porto Alegre: SBC. doi:10.5753/cbie.sbie.2020.1473
- Borges, M. (2011). Conhecimento coletivo. *Sistemas Colaborativos*. Pimentel, Mariano; Fuks, Hugo.(Orgs). Rio de Janeiro: Elsevier, 198.
- Boud, D., & Feletti, G. (Eds.). (1998). The challenge of problem-based learning. Psychology Press.
- Brackmann, C. (2017) Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica.
- Brezolin, C., & Silveira, M. (2021). Panorama Brasileiro de Uso de Ferramentas para Desenvolvimento do Pensamento Computacional e Ensino de Programação . In *Anais do XXIX Workshop sobre Educação em Computação*, (pp. 398-407). Porto Alegre: SBC. doi:10.5753/wei.2021.15931
- Chatman, S. B.(1978). Story and discourse: Narrative structure in fiction and film. Londres: Cornell University Press.
- Costa, J. A. C., de Oliveira, J. D., & Dantas, D. R. (2020) Metodologias ativas e suas contribuições no processo de ensino-aprendizagem. *Série Educar-Volume 40 Prática Docente*, 8.
- Farias, C., Cruz, V., Farias, J., Braz, D., Brito, B., & Carvalho, A. (2019). Estimulando o Pensamento Computacional: uma experiência com ScratchJr. In *Anais do XXV Workshop de Informática na Escola*, (pp. 197-206). Porto Alegre: SBC. doi:10.5753/cbie.wie.2019.197
- Fink LD. 2004. Beyond small groups: harnessing the extraordinary power of learning teams. In: Michaelsen LK , Knight AB, Fink DL, editors. *Team-based learning: a transformative use of small groups in college teaching*. Sterling, VA: Stylus.
- França, Juliana B. S., Saburido, Beatriz & Dias, Angélica F. S.. (2021). Desenvolvendo o Pensamento Computacional de Meninas através de Histórias. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 32. , 2021, Online. *Anais [...]*. Porto Alegre: Sociedade Brasileira de Computação, 2021. p. 931-942. DOI: <https://doi.org/10.5753/sbie.2021.216837>.
- França, R., & Tedesco, P. (2019, November). Pensamento computacional: Panorama dos grupos de pesquisa no brasil. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)* (Vol. 30, No. 1, p. 409).
- Gomes, V., Pontes, R., Camelo, C., Cavalcanti, G., & Perkusich, M. (2017, October). Ensino de programação para crianças e adolescentes: um estudo exploratório. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação* (Vol. 6, No. 1, p. 490).

- Graesser, A. C., et al. (1980). "Advanced Outlines, Familiarity, and Text Genre on Retention of Prose". *The Journal of Experimental Education*, vol. 48, no 4, p. 281-90. Taylor and Francis+NEJM, <https://doi.org/10.1080/00220973.1980.11011745>.
- Hmelo-Silver, C. E. (2004). Problem-based learning: What and how do students learn?. *Educational psychology review*, 16(3), 235-266.
- Ignácio, W. (2018). O Pensamento Computacional na educação brasileira e o papel das instituições de ensino tecnológico.
- Janne Fagerlund, Mikko Vesisenaho a, Päivi Häkkinen. (2022). Fourth grade students' computational thinking in pair programming with Scratch: A holistic case analysis. *International Journal of Child-Computer Interaction*: 33, 100511 .
- Johnson, C. (2009). Team-Based Learning for Health Professions Education: A Guide to Using Small Groups for Improving Learning. *The Journal of Chiropractic Education*, 23(1), 47.
- Junior, B., Cavalheiro, S., & Foss, L. (2019, Novembro). Revisitando um Jogo Educacional para desenvolver o Pensamento Computacional com Gramática de Grafos. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)* (Vol. 30, No. 1, p. 863).
- Kaefer, J. C. (2021). Um Ambiente Computacional Para Desenvolvimento De Recursos Digitais De Estímulo ao Comportamento Seguro Na Internet Por Crianças E Adolescentes.
- Keengwe, J. (Ed.). (2014). Promoting active learning through the flipped classroom model. IGI Global.
- Lage, B., França, J., Dias, A., Vivacqua, A., & Borges, M. (2017, October). Aplicando o group storytelling no compartilhamento de experiências e na avaliação em sala de aula. In *Anais do Workshop de Informática na Escola* (Vol. 23, No. 1, pp. 165-174).
- Mateus, Ana do Nascimento Biluca, et al. (2013). "A importância da contação de história como prática educativa na educação infantil". *Pedagogia em Ação*, vol. 5, no 1, periodicos.pucminas.br, <http://periodicos.pucminas.br/index.php/pedagogiacao/article/view/8477>.
- Mattos, M. M., de Araújo, L. P., Zucco, F. D., Hein, N., da Cunha, K. Z., de Oliveira, G. C., ... & Bursoni, P. L. (2017, July). Aplicação da prática colaborativa no desenvolvimento de um jogo para o ensino da programação. In *Anais do XIV Simpósio Brasileiro de Sistemas Colaborativos* (pp. 1536-1545). SBC.
- Mcsill, James. (2013). 5 Lições de storytelling: fatos, ficção e fantasia. 1. Ed. São Paulo: Dvs Editora. In Santos, Leonardo Schwertner Dos. *Storytelling: O Poder da Narrativa Estratégica Dentro do Branding e Marketing*. P. 5.
- Michaelsen, L. K., & Sweet, M. (2008). The essential elements of team-based learning. *New Directions for Teaching and Learning*, 2008(116), 7-27. doi:10.1002/tl.330
- Núcleo da Informação e Coordenação do Ponto BR -NIC.br. (2021). TIC DOMICÍLIOS - Pesquisa Sobre o Uso das Tecnologias de Informação e Comunicação nos Domicílios Brasileiros - 2020. Disponível em: https://cetic.br/media/docs/publicacoes/2/20211124201233/tic_domicilios_2020_livro_eletronico.pdf.
- Papert, S. (1985). LOGO: Computadores e educação. Tradução de José Armando Valente; Beatriz Bitelman; Afira Vianna Ripper. São Paulo: Brasiliense.

- Paviani, J. (2013). Platão & a educação. Autêntica.
- Perni, A.P. J.; Silva, D.E.; Valentim, N.M. C. Um benchmark de ferramentas de programação em blocos que podem ser utilizadas nas salas de aula do Ensino Médio. X Congresso Brasileiro de Informática na Educação (CBIE 2021). Anais do XXXII Simpósio Brasileiro de Informática na Educação (SBIE 2021). DOI: 10.5753/sbie.2021.217765
- Pires, F. G. de S., et al.(2018). Uma análise cognitiva entre a emergência de padrões em narrativas infantis e elementos do pensamento computacional. p. 1193. DOI.org (Crossref), <https://doi.org/10.5753/cbie.sbie.2018.1193>.
- Racca, Beatriz Saburido & França, Juliana Baptista dos Santos. (2021) StoryGirl: Uma rede colaborativa de apoio à criança através de histórias reais programadas em Scratch. In: PESQUISAS EM ANDAMENTO - SIMPÓSIO BRASILEIRO DE SISTEMAS COLABORATIVOS (SBSC), 16. , 2021, Evento Online. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação. p. 54-61. DOI: https://doi.org/10.5753/sbsc_estendido.2021.16036.
- Reidsema, C., Kavanagh, L., Hadgraft, R., & Smith, N. (2017). The Flipped Classroom. Practice and Practices in Higher Education. Ed. Springer.
- Reimschisel, T., Herring, A. L., Huang, J., & Minor, T. J. (2017). A systematic review of the published literature on team-based learning in health professions education. *Medical Teacher*, 39(12), 1227–1237. doi:10.1080/0142159x.2017.1340636
- Robin, B. R., & McNeil, S. G. (2019). Digital storytelling. *The International Encyclopedia of Media Literacy*, 1-8.
- Rodriguez, C., Zem-Lopes, A. M., Marques, L., & Isotani, S. (2015, October). Pensamento Computacional: transformando ideias em jogos digitais usando o Scratch. In *Anais do Workshop de Informática na Escola* (Vol. 21, No. 1, pp. 62-71).
- Saburido, B., Dias, A. F. S. & França, J. B. S.. (2021) StoryGirl: programando através de narrativas. In: WOMEN IN INFORMATION TECHNOLOGY (WIT), 15. , 2021, Evento Online. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, p. 355-359. ISSN 2763-8626. DOI: <https://doi.org/10.5753/wit.2021.15881>.
- Salim, Roberta; Dias, Angélica F. S.; França, Juliana B. S.; Lage, Bruno B.; Borges, Marcos R. S.. O Uso de Storytelling para Apoiar o Compartilhamento de Conhecimento. In: SIMPÓSIO BRASILEIRO DE SISTEMAS COLABORATIVOS (SBSC), 13. , 2016, Belém. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2016 . p. 1593-1606. ISSN 2326-2842. DOI: <https://doi.org/10.5753/sbsc.2016.9512>.
- Savin-Baden, M., & Major, C. H. (2004). Foundations of problem-based learning. McGraw-hill education (UK).
- Schneiders, L. A. (2018). O método da sala de aula invertida (flipped classroom). Lajeado: ed. da UNIVATES.
- Schneider, G., Bernardini, F., & Boscaroli, C. (2019, Novembro). Ensino do pensamento computacional por meio da internet das coisas: Possibilidades e desafios. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)* (Vol. 30, No. 1, p. 169).
- Silva, V. I. C. (2021) Contar histórias com o Scratch: o contributo do ambiente gráfico de programação para a aprendizagem da língua.

- Sousa, L. de L., Farias, E. J. & Carvalho, W. V. de. (2020). Programação em Blocos Aplicada no Ensino do Pensamento Computacional: Um Mapeamento Sistemático. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 31. , 2020, Online. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação. p. 1513-1522. DOI: <https://doi.org/10.5753/cbie.sbie.2020.1513>.
- Stöhr, C., Demazière, C., & Adawi, T. (2020). The polarizing effect of the online flipped classroom. *Computers & Education*, 147, 103789.
- Trevelin, A. T. C., Pereira, M. A. A., & de Oliveira Neto, J. D. (2013). A utilização da “sala de aula invertida” em cursos superiores de tecnologia: comparação entre o modelo tradicional e o modelo invertido “flipped classroom” adaptado aos estilos de aprendizagem. *Revista de estilos de aprendizagem*, 6(12).
- Wing, J. M. (2006). Pensamento Computacional: um conjunto de atitudes e habilidades que todos, não só cientistas da computação, ficaram ansiosos para aprender e usar. Tradução: Cleverson Sebastião dos Anjos. *Revista Brasileira de Ensino de Ciência e Tecnologia*, Ponta Grossa, v. 9, n.2, p. 1-10, maio/ago. 2016.