



Reflexões sobre uma sequência didática de desenvolvimento de algoritmo com turma multisseriada utilizando Google Doodle e Scratch

Reflections about a didactic sequence of algorithm development with a multigrade class using Google Doodle and Scratch



Brunna Sordi Stock

Mestra em Educação

Universidade Federal do Rio Grande do Sul - UFRGS

Porto Alegre, RS - Brasil

brunna.stock@gmail.com



Marcus Vinicius de Azevedo Basso

Doutor em Informática na Educação

Universidade Federal do Rio Grande do Sul - UFRGS

Porto Alegre, RS – Brasil

mbasso@ufrgs.br

Resumo: O presente artigo versa sobre uma sequência didática realizada durante um projeto sobre programação com uma turma multisseriada do primeiro ciclo do Ensino Fundamental. Para trabalhar as ideias de algoritmo e de bloco de repetição, bem como a importância de passo a passo e de ordem, foi utilizado um Google Doodle e o software Scratch. A análise se dá com base nos estudos de Jean Piaget (1983) sobre os estágios de desenvolvimento, esquemas e tomada de consciência. Foi possível observar que as crianças estabeleceram uma relação entre os blocos de repetição dos dois recursos, adaptando o esquema utilizado para repetir um comando. Ainda, as atividades de pensamento computacional possibilitaram que o grupo refletisse sobre quais passos deveriam ser tomados para chegar ao objetivo desejado, um importante encadeamento lógico para a resolução de problemas. Desse modo, tanto o Doodle quanto o Scratch se mostraram importantes ferramentas para trabalhar o pensamento computacional.

Palavras chave: educação matemática; Scratch; Doodle; pensamento computacional; programação.

Abstract: This article presents a didactic sequence carried out during a project on programming developed with a multigrade class of the first cycle of elementary school. In it, a Google Doodle and Scratch software were used to work on the idea of algorithm and repetition block, as well as to discuss the importance of a step-by-step instruction and order. The analysis is based on Jean Piaget's (1983) studies on the stages of development, schemas and awareness. It was possible to observe that the children established a relationship between the repetition block of the two resources, both similarity and differentiation, adapting the scheme used. Also, the activities about computational thinking made it possible for the group to reflect on what steps should be taken to reach the goal, an important logical chain for problem solving. Both Doodle and Scratch proved to be important tools for working on computational thinking.

Keywords: mathematics education; Scratch; Doodle; computational thinking; programming.

Cite como

(ABNT NBR 6023:2018)

STOCK, Brunna Sordi; BASSO, Marcus Vinicius de Azevedo. Reflexões sobre uma sequência didática de desenvolvimento de algoritmo com turma multisseriada utilizando *Google Doodle e Scratch*. *Dialogia*, São Paulo, n. 40, p. 1-16, e21736, jan./abr. 2022. Disponível em: <https://doi.org/10.5585/40.2022.21736>.

American Psychological Association (APA)

Stock, B. S., & Basso, M. V. de. A. (2022, jan./abr.). Reflexões sobre uma sequência didática de desenvolvimento de algoritmo com turma multisseriada utilizando *Google Doodle e Scratch*. *Dialogia*, São Paulo, 40, p. 1-16, e21736. <https://doi.org/10.5585/40.2022.21736>.

1 Introdução

A escola, como tantos espaços, é um terreno de disputas de interesses. As questões do que ensinar, quando ensinar e como ensinar fazem parte (ou deveriam fazer) da constante reflexão de docentes que a constroem. Nesse sentido, quando falamos de pensamento computacional na escola, acreditamos que a primeira pergunta que deveria ser feita é: por quê? Há tantos saberes e habilidades interessantes de serem desenvolvidos, então, por que esse deveria fazer parte da sala de aula?

Não pretendemos, aqui, esgotar as respostas dessa pergunta, mas sim apresentar reflexões que podem ajudar a pensar sobre ela. Discutiremos uma sequência didática realizada com uma turma multisseriada do primeiro ciclo do Ensino Fundamental (1º, 2º e 3º anos) para elaborar um algoritmo utilizando *Google Doodle* e o *software Scratch* durante um projeto sobre programação. O planejamento dessa sequência e a análise se baseiam nos estudos de Jean Piaget (1983), os quais foram aprofundados pela primeira autora durante a graduação em Licenciatura em Matemática através do trabalho de Gérard Vergnaud (1996), orientando de Piaget, e na pós-graduação em Educação na linha de psicopedagogia. Descrevemos, a seguir, em que condições esse projeto foi desenvolvido para, após, aprofundarmos na análise da construção de algoritmos.

2 O projeto

Essa atividade foi realizada em uma escola privada de Porto Alegre/RS que utiliza a metodologia de projetos. A cada trimestre, os conteúdos e as habilidades, assim como a modalidade (oficina, projeto ou módulo), são definidos pela turma em conjunto com os e as docentes. Os projetos são elaborados a partir dos interesses da turma e das necessidades de aprendizagem estabelecidas em diretrizes nacionais, como a Base Nacional Comum Curricular - BNCC (BRASIL, 2018).

A sequência didática foi feita com uma turma multisseriada de estudantes de 1º, 2º e 3º anos dos Anos Iniciais do Ensino Fundamental, totalizando 11 crianças. A turma contava com a tutora responsável (pedagoga), a assistente (licenciada em História) e a professora de área (licenciada em Matemática e coautora desta publicação) para a mediação do processo. A partir de uma atividade do trimestre anterior com jogos on-line, a turma havia demonstrado curiosidade na elaboração de jogos, que foi aliada ao pensamento computacional e à resolução de problemas como necessidades de aprendizagem.

As Diretrizes para ensino de Computação na Educação Básica (SBC, 2019, p. 3) entendem o pensamento computacional como “a habilidade de sistematizar a atividade de resolução de

problemas, representar e analisar as soluções através de algoritmos”. Essa definição pode ser complexificada ao dialogarmos com a teoria de Jean Piaget a partir das discussões de Kátia Rocha e Marcus Basso (2021), que utilizam a ideia de objetos-de-pensar-com como instrumentos de pensamento que auxiliam o sujeito a refletir sobre conceitos. O autor e a autora entendem o pensamento computacional como “os processos de pensamento que utilizam elementos computacionais como objetos-de-pensar-com para contribuir nos processos de abstração reflexionante do sujeito” (ROCHA; BASSO, 2021, p. 427).

É nesse sentido que o pensamento computacional constitui um interesse: na sua potência para o processo de desenvolver habilidades cognitivas, em especial aquelas relacionadas à Matemática. Sem essas habilidades, a Matemática torna-se uma aplicação de fórmulas decoradas que servem apenas para problemas específicos do contexto da sala de aula. Por exemplo, não há sentido em falarmos sobre fórmulas de progressões geométricas e aritméticas no Ensino Médio se não realizamos atividades de observação e reconhecimento de padrões antes de formalizar o conceito de progressões. A Matemática tem muito mais de observação, inferência e verificação do que de fórmulas em si.

É importante ressaltar nossa ciência da limitação quanto ao tema, visto que esse projeto parte do olhar de uma professora de Matemática curiosa, mas cuja formação não contemplou formalmente o pensamento computacional. Porém, nossa observação aliada à teoria direciona para a questão central deste artigo, que é a sequência didática para trabalhar com pensamento computacional. Não temos a pretensão de romantizar essa prática e olhá-la como um exemplo fixo a ser seguido; ela é um exemplo a ser debatido, que trouxe surpresas e reflexões que consideramos relevantes de compartilhar.

Ainda, é imprescindível observar que esse projeto foi desenvolvido durante o período do isolamento social em função da Covid-19 no ano de 2020. Nesse momento, frente a tantas adaptações necessárias de currículo, espaços de convivência, ensino e, enfim, de vida, os objetivos dessa prática englobavam o pensamento computacional através de uma construção coletiva. Os encontros ao longo do trimestre se deram de forma síncrona através do aplicativo *Zoom*, em que a professora compartilhava a tela com a turma e, assim, realizavam coletivamente as alterações no *software* escolhido, o *Scratch*. Essa dinâmica teve o ônus de impossibilitar a criação individual por parte das crianças, porém, é justamente nessa construção coletiva que estava a sua potência: o contexto de isolamento demandava criar estratégias de interação e cooperação entre a turma que não se encontrava mais na escola. Logo, a escolha por um projeto de programação em que todas e

todos deveriam participar para o desenvolvimento do produto final foi algo intencional e decidido em diálogo com a coordenação.

O *Scratch*¹ é um *software* que utiliza blocos lógicos para desenvolver histórias interativas, jogos e animações, e foi sugerido para a turma pela sua facilidade de programação, possibilidades de uso e interface acessível para os e as estudantes. No planejamento coletivo do trimestre com a turma, foi apresentado o *software* através de um jogo e de uma animação criada no momento da aula. A turma acolheu a proposta e demonstrou curiosidade pelas possibilidades de criação. Nesse momento do planejamento, foi determinado que seriam nove encontros no total: um de organização, seis encontros de utilização do *Scratch* e desenvolvimento do produto final, um de apresentação dos produtos de cada grupo para toda a turma e avaliação do projeto, e, por fim, um de conclusão, avaliação individual e preenchimento dos registros pedagógicos no portal da escola.

Em qualquer projeto é necessário identificar em qual etapa de desenvolvimento se encontram as crianças para pensarmos o que podemos delimitar como expectativas de aprendizagem quanto ao tópico escolhido. Programar com crianças de seis anos não é o mesmo que com adolescentes de 15 anos. No caso dessa turma, havia o desafio de incluir estudantes em processo de alfabetização que recém tinham ingressado no 1º ano no contexto da virtualidade, algo que demanda leitura e escrita. Após as primeiras semanas, foi constatada a necessidade de dividir a turma em dois grupos: um com quatro estudantes em processo de alfabetização e que necessitavam de muita ajuda para a realização das atividades; outro com sete estudantes que já estavam mais além nesse processo, sendo em sua maioria do final do ciclo (3º ano). O primeiro grupo era acompanhado pela tutora e chamaremos de grupo 1; já o grupo 2 era acompanhado pela assistente.

A avaliação nessa escola se dá de forma processual através de atividades realizadas ao longo do trimestre e da elaboração de um produto final que formaliza os conhecimentos mobilizados no projeto. Na experiência didática que tratamos aqui, cada grupo elaborou uma atividade utilizando o *Scratch*: o grupo 1 decidiu fazer uma animação que consistia em colocar diferentes roupas em um centauro e em um cachorro; e o grupo 2 optou por criar um jogo ambientado no restaurante de um esquilo em que um dinossauro deveria ser alimentado com uma das comidas que voavam.

Na elaboração do projeto, os objetivos foram determinados tendo em mente que a expectativa de cada grupo seria diferente. Para tal, tomamos como parâmetro a BNCC (BRASIL, 2018) no componente da Matemática quanto aos objetos de conhecimento relacionados ao pensamento computacional e as Diretrizes (SBC, 2019). Assim, os objetivos da prática foram:

¹ O *Scratch* foi desenvolvido e é mantido pela Equipe do *Scratch* no grupo *Lifelong Kindergarten* no MIT (Instituto de Tecnologia de Massachusetts, sigla em inglês). Disponível em: <https://Scratch.mit.edu/>. Acesso em: 20 mar. 2022.

- 1) Desenvolver pensamento computacional;
- 2) Compreender a importância da ordem na resolução de problemas relacionados à programação em blocos;
- 3) Analisar problemas e avaliar soluções propostas;
- 4) Planejar a execução de uma tarefa e definir etapas para chegar ao objetivo;
- 5) Trabalhar cooperativamente, contribuindo com ideias e respeitando os e as colegas.

A habilidade geral sobre pensamento computacional foi pensada para dar conta das diferentes expectativas de aprendizagem de cada grupo e de acordo com o que seria desenvolvido no *Scratch*. Cada um desses objetivos, bem como as atividades, os encontros e as discussões no grupo, consistem em infinita fonte de reflexões e inquietações. Tendo em vista a limitação de um artigo, discutiremos a sequência didática que contempla os objetivos 1, 2, 3 e 4 quanto à construção de algoritmos.

3 Criando um algoritmo

As Diretrizes (SBC, 2019) definem algoritmos como modelos de processos que podem estar descritos em vários níveis de abstração. Mais precisamente, para resolver um problema, estipulamos um número finito de regras e procedimentos lógicos que, ao aplicar a um número finito de dados, leva à solução.

Falar de algoritmos na Matemática escolar é algo delicado, pois eles são muitas vezes vistos de forma negativa no processo de ensino. Evidentemente, uma educação voltada para decorar algoritmos perde o seu sentido em termos de desenvolvimento do raciocínio matemático e do pensamento computacional. Contudo, o uso de algoritmos não caracteriza por si só algo nefasto para a aprendizagem, pois é possível problematizar o seu uso ao pensar que há diversas formas de se resolver um problema e que ter que escolher qual delas é a mais conveniente é analisar as variáveis do contexto e mobilizar diferentes saberes. Ainda, entender o algoritmo como um passo a passo em que a ordem é relevante para chegar ao resultado permite, inclusive, construir algoritmos com as crianças, empenhando, assim, a reflexão sobre o processo – e é nesse sentido que entendemos o algoritmo como um objeto do conhecimento relevante na Matemática escolar.

As seguintes habilidades estão relacionadas a esse objeto do conhecimento para cada ano do ciclo em questão (SBC, 2019, p. 10):

Quadro 1 – Habilidades quanto ao objeto de conhecimento “algoritmos”

1º ano	Compreender a necessidade de algoritmos para resolver problemas; Entender a definição de algoritmos resolvendo problemas passo a passo.
2º ano	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 ano	Definir e executar algoritmos que incluam sequências, repetições simples (iteração definida) e seleções (descritos em linguagem natural e/ou pictográfica) para realizar uma tarefa, de forma independente e em colaboração.

Fonte: Elaborado pela autora (2022).

Tendo em vista essas habilidades, trabalhar na construção de um algoritmo nessa turma era trazer a necessidade de um passo a passo, compreender a importância da ordem e incluir elementos de repetição (em especial com estudantes de final de ciclo). Sabendo que os estádios de desenvolvimento (PIAGET, 1983) são variáveis, mas entendendo que a sua compreensão colabora para pensar nas atividades a serem realizadas, imaginava-se que os e as estudantes estavam na transição do estágio pré-operatório para o operatório concreto. No estágio pré-operatório se realiza a transição entre a inteligência sensório-motora e a inteligência representativa. Nessa fase, a criança ainda percebe o mundo de acordo com suas experiências, e não a partir de generalizações. Já no operatório concreto, a atividade cognitiva da criança torna-se operatória com o desenvolvimento da reversibilidade. Nesse estágio, a criança começa a resolver problemas concretos a partir da lógica e a lidar com o pensamento e questões conceituais.

Essa análise foi importante para pensar que utilizar somente o *Scratch* talvez fosse um passo muito grande para as crianças por implicar em pensar em algo geral como um passo a passo e pelo *software* utilizar blocos identificados por palavras. Assim, para propiciar um momento de discussão antes da utilização do *software* num formato de tutorial, optou-se por começar com um *Google Doodle*² de programação (Figura 1). Em cada etapa desse jogo é ensinada uma instrução para darmos ao coelho. Logo após, aparece uma atividade de aplicação dessa instrução e o objetivo é fazer o personagem comer todas as cenouras. A instrução é dada através do arraste dos blocos disponíveis para a caixa na parte inferior da tela e, para fazer o coelho se mover, é necessário apertar o botão *play* (seta laranja) depois que todos os comandos forem dados. Quando a instrução representa o caminho mais curto, isto é, que utilize menos blocos, recebe-se uma medalha. Esse jogo foi

² *Google Doodle* são as intervenções que a empresa *Google* faz no seu logotipo, seja no formato de imagem ou de animação, geralmente em datas comemorativas. Esse *Doodle* foi criado e disponibilizado em comemoração aos 50 anos de programação para crianças em 4 de dezembro de 2017. Disponível em: <https://www.Google.com/logos/2017/logo17/logo17.html?hl=pt-BR>. Acesso em: 20 fev. 2022.

escolhido por ser autoexplicativo quanto à janela de programação e por ter um design de blocos semelhante ao do *Scratch*.

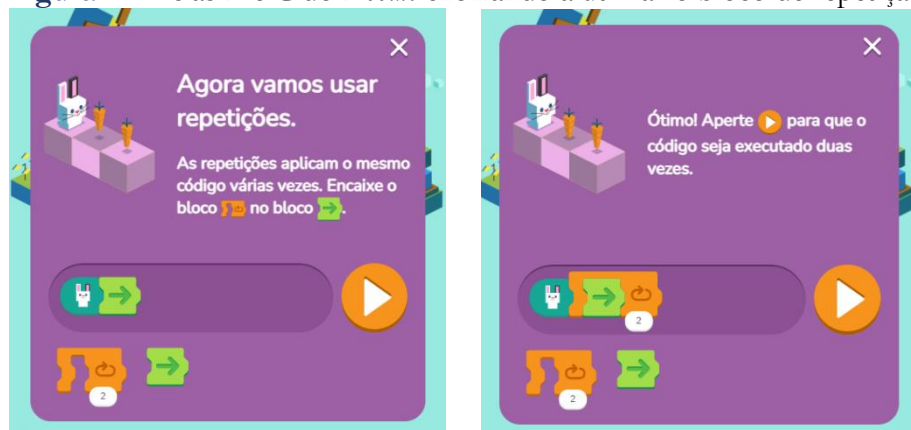
Figura 1 – Telas A e B do *Doodle* ensinando a programar a trajetória do coelho



Fonte: Elaborado pela autora (2022).

As crianças adoraram o jogo e, com ele, também foi possível trabalhar a contagem das casas que o coelho devia andar. Ainda, na próxima fase, é apresentado um bloco de repetição do código (Figura 2) que se assemelha com o bloco que seria usado no produto do grupo 2 para fazer as comidas voarem.

Figura 2 – Telas B e C do *Doodle* ensinando a utilizar o bloco de repetição



Fonte: Elaborado pela autora (2022).

Importante notar que o bloco de repetição do *Doodle* indica no canto inferior direito quantas repetições ele faz (no caso da Figura 2, duas vezes), ou seja, quem está jogando deve se adaptar a essa condição para determinar qual programação será feita. Por exemplo, na atividade proposta,

após as instruções, há um bloco de repetição de quatro vezes (Figura 3). Ao aparecer o bloco de repetição, foi perguntado para turma o que achavam que ele representava, ao que um aluno de 2º ano (que afirmava já ter aulas de programação fora da escola) respondeu ser um *loop*. Quando questionado sobre o que era um *loop*, esse aluno respondeu que era uma ordem de passos que se repetia.

Essa noção intuitiva foi levada para o grupo 1 (crianças em processo de alfabetização) e foi questionado se conseguiam pensar em algo que se repetisse indefinidamente, ao que uma aluna de 1º ano respondeu: “uma roda gigante”. Por mais que a roda gigante não siga indefinidamente uma sequência de comandos (afinal, ela para, gira, abre as portas etc. em função dos comandos dados por quem a opera), essa aluna conseguiu relacionar a algo que lhe era conhecido (a roda gigante) com uma nova linguagem (representação em blocos de comandos). Essa evidência de estabelecimento de relações mostra o potencial do pensamento computacional e de outros tipos de linguagem para o desenvolvimento de raciocínios complexos que saem da representação concreta.

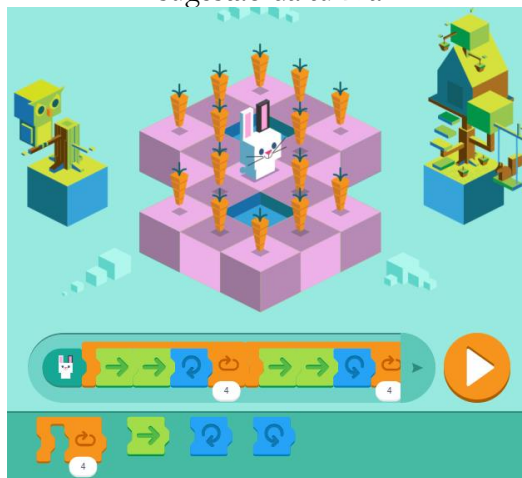
Figura 3 – Tela E do *Doodle* apresentando o caminho que deve ser feito e, na caixa embaixo da figura, a programação proposta pelas crianças



Fonte: Elaborado pela autora (2022).

A instrução sugerida pelas crianças na Figura 3 é excelente, pois faz com que o coelho realize o comando (frente, frente, virar à direita) quatro vezes, pegando, assim, todas as cenouras. Ambos os grupos chegaram nessa construção. Na sequência, foi apresentada uma nova situação para o coelho que envolvia utilizar o bloco de repetição mais de uma vez (Figura 4).

Figura 4 – Tela F do *Doodle* apresentando nova situação e, na caixa, a sugestão da turma



Fonte: Elaborado pela autora (2022).

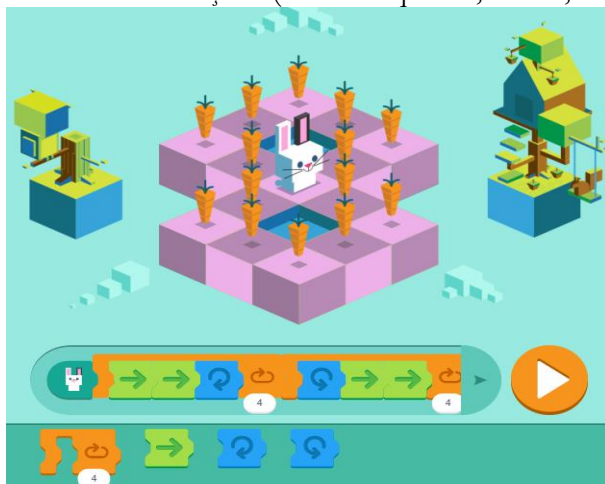
Ambos os grupos apresentaram essa solução. É possível observar que ela é uma repetição do que foi aprendido no contexto anterior (frente, frente, virar à direita) com uma pequena mudança, que é virar para a esquerda a fim de dar conta da parte superior do caminho. Quando questionadas sobre o motivo de utilizar o giro no outro sentido no segundo bloco de repetição, as crianças afirmaram que era porque o coelho tinha que ir para o outro lado agora. Aqui temos evidências da compreensão de sentido, visto que, antes, o movimento era para direita e agora há também uma sugestão para a esquerda em função da posição do coelho. Podemos interpretar essa utilização da mesma estratégia a partir da noção de esquema. Gérard Vergnaud (1996, p. 157) define um esquema como a “organização invariante da conduta para uma dada classe de situações”, ou seja, é o que evocamos para agir diante de uma situação por nós categorizada. Na situação em questão, as crianças já haviam elaborado um esquema que foi testado e verificado, que consistia no movimento (frente, frente, virar à direita). Fazer esse movimento na parte superior do caminho consistia em uma situação da mesma classe da anterior, implicando na utilização do mesmo esquema.

Porém, essa sugestão de programação não é viável, pois, ao finalizar o primeiro bloco de repetição, o coelho termina na mesma posição e sentido do início e, por isso, o segundo bloco de repetição não vai implicar na colheita das cenouras da parte superior. Isto é, o esquema não deu conta da situação. Logo ele precisa ser modificado na medida em que assimila (PIAGET, 1978) o objeto, ou seja, ao incorporar os dados da experiência para dar sentido a esse objeto. Ao fazer esse processo, dizemos que houve acomodação, que é o resultado das pressões exercidas pelo meio na elaboração desse esquema. Podemos afirmar, então, que o esquema designa a forma do

conhecimento que assimila os dados da realidade e que é suscetível à modificação por acomodação a essa realidade (MONTANGERO; MAURICE-NAVILLE, 1998).

Aqui temos um ponto importante do jogo: ele possibilitou verificar se a inferência estava correta e, caso contrário, analisar onde estava o erro, visto que o coelho faz o que for mandando nas instruções e cada bloco pisca na tela à medida que está sendo utilizado no caminhar do coelho. Ao visualizar que havia um problema de direção do coelho e que era necessário que ele mudasse de direção antes do novo bloco de repetição, o grupo 2 modificou o código (Figura 5). A modificação do esquema frente a um fracasso é considerada por Piaget (1978) um importante (porém, não único) passo para a tomada de consciência, ou seja, para a compreensão do êxito obtido na ação.

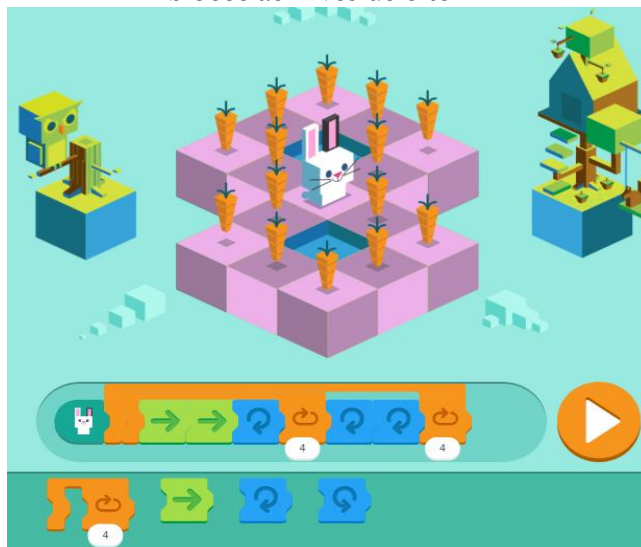
Figura 5 – Tela G do *Doodle* com alteração no código trocando a ordem das ações (virar à esquerda, frente, frente)



Fonte: Elaborado pela autora (2022).

A solução proposta dá conta do caminho e demonstra que o grupo 2 compreendeu a importância da ordem na elaboração de um passo a passo. Interessante perceber que as crianças, de forma geral, compreenderam que há uma ordem (andar e depois virar) para a primeira situação, contudo, somente o grupo das crianças mais velhas conseguiu fazer a correção na ordem de comandos para chegar em uma solução viável, talvez interpretando a ordem (frente, frente, virar) como única possível. Todavia, esse não é o caminho mais curto. Discutiu-se com o grupo 2 algumas possibilidades e não se chegou a uma proposta utilizando uma quantidade menor de blocos. A seguir, apresentamos uma solução mais curta, que não foi encontrada pelas crianças (Figura 6).

Figura 6 – Tela H do *Doodle* com uma solução mais curta utilizando sete blocos ao invés de oito

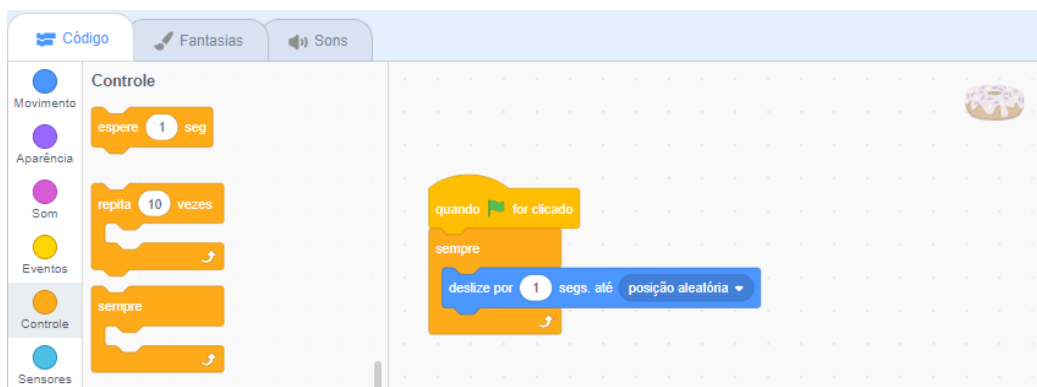


Fonte: Elaborado pela autora (2022).

Consideramos essa solução mais sofisticada por realizar um bloco de repetição dentro de outro bloco de repetição, o que acreditamos ter sido um empecilho para as crianças chegarem nessa proposta. Nela, é necessário entender que a mesma construção (na mesma ordem e com os mesmos blocos) pode ser utilizada para cada parte do trajeto, apenas dependendo de onde o coelho está. Ou seja, exige uma conservação do primeiro comando e um novo comando que apenas altere o sentido do coelho para efetuar a repetição.

Após essa atividade, discutimos com o grupo 2 sobre como faríamos, no *Scratch*, as comidas voarem no restaurante do esquilo sabendo que o objetivo do jogo era clicar com a seta na comida que o dinossauro pedisse. As crianças sugeriram utilizar algum comando no *Scratch* que fosse de repetição e fizesse o objeto se mover sempre e de forma inesperada. Buscamos em “Comandos” e encontramos o bloco “Sempre” (Figura 7), que se assemelha ao bloco de repetição do coelho no *Doodle*, mas sem um número delimitado de vezes. Para fazer um deslocamento inesperado, buscamos em “Movimentos” algo que fizesse o objeto trocar de posição na tela. Testamos o bloco “Vá para”, mas ele fazia o objeto desaparecer e reaparecer em outro lugar. Como o grupo queria dar a ideia de que as comidas estavam voando, optamos pelo bloco “Deslize”.

Figura 7 – Tela do objeto *donut* no *Scratch* com o comando realizado



Fonte: Elaborado pela autora (2022).

A utilização dos comandos do *Scratch* vai, em certa medida, na tentativa e erro pensando no que se quer chegar. Porém, é notável como as crianças conseguiram estabelecer uma relação entre o bloco de repetição do *Doodle* e o bloco “Sempre” do *Scratch*. Não obstante, o grupo identificou uma nova necessidade para o bloco “Sempre” que não havia no bloco de repetição do *Doodle*: interrupção. Uma aluna de final de ciclo afirmou que a comida escolhida (um *taco*) não poderia seguir voando quando selecionada, mas, sim, deveria parar quando fosse clicada, se deslocasse para a boca do dinossauro e, então, desaparecesse, dando a ideia de que havia sido ingerida. Aproveitando a frase dessa aluna, foi apresentado o bloco do tipo evento “Quando este ator for clicado” para criar uma condição para o clique da seta. Após, outro aluno sugeriu o bloco “Deslizar” para o *taco* ir até o dinossauro.

O grupo queria, ainda, que o dinossauro enviasse uma mensagem quando a comida fosse selecionada. Fomos à janela do dinossauro e as crianças não sabiam como sinalizar para ele que o *taco* já havia chegado e que o jogo precisava terminar, momento em que foi aproveitada essa ideia de sinalização para apresentar o evento “Quando receber comando”. A criação de um comando (aqui chamado de “oi” pelas crianças) como sinal do fim do jogo foi desafiadora por ser um evento aberto e não algo literal, como, por exemplo, um bloco do tipo “terminar o jogo”. Ainda, o comando apenas fazia o dinossauro dizer que o *taco* havia sido comido, mas o *taco* seguia na tela. A essa altura o grupo havia entendido que mais uma alteração deveria ser feita na janela de programação do *taco* e foi sugerido buscarmos algo que apagasse o *taco*, a saber o bloco “Esconda” (Figura 8).

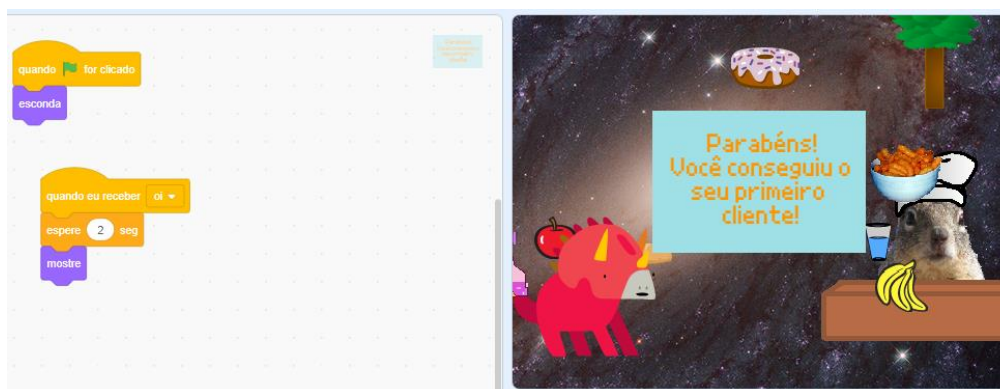
Figura 8 – Tela do objeto *taco* e tela do objeto dinossauro, respectivamente



Fonte: Elaborado pela autora (2022).

Finalizada a alteração no objeto dinossauro, o grupo sugeriu colocar uma mensagem para caracterizar o final do jogo. Rapidamente foi sugerida uma construção análoga à anterior utilizando o bloco “Quando eu receber comando” para sinalizar que a mensagem deveria aparecer naquele momento (Figura 9).

Figura 9 – Tela do objeto mensagem final



Fonte: Elaborado pela autora (2022).

No processo de construção do jogo, o grupo 2 demonstrou ser capaz de utilizar os blocos condicionais e de repetição, de aplicar o uso de um bloco em situações semelhantes, de formular um passo a passo para o movimento dos objetos, diferenciar as situações e adaptar estratégias considerando a ordem. É difícil dizer até que ponto a utilização do *Doodle* facilitou o processo de construção com o *Scratch*, mas é possível afirmar que as crianças conseguiram estabelecer relações entre os dois contextos e diferenciar os tipos de bloco de repetição.

Não obstante, foi possível desenvolver com ambos os grupos um importante caminho para a resolução de problemas, que é partir de onde se quer chegar para pensar o início do trajeto. As

ideias de alteração do jogo (por exemplo, fazer o *taco* desaparecer quando clicado com o grupo 2) promoviam a reflexão sobre o que precisávamos fazer para que aquilo ocorresse, ou seja, mesmo que muitas vezes o bloco escolhido fosse uma tentativa, ela não era completamente aleatória. Podemos observar esse mesmo encadeamento do pensamento quando analisamos um problema e selecionamos qual estratégia (seja ela um algoritmo ou não) para tentar resolvê-lo, saindo de uma lógica de tentativa e erro para uma reflexão sobre a ação, ou seja, nos encaminhando para uma tomada de consciência da ação.

Ainda, o grupo 1 também construiu um passo a passo com o *Doodle* utilizando corretamente o bloco de repetição, mas não adaptou o esquema relativo ao bloco de repetição para uma situação mais complexa. Cabe salientar que isso não era esperado, pois eram crianças em início de ciclo que precisavam de muito acompanhamento para a leitura e organização e sem um material manipulativo em mãos para realizar a tarefa. Ou seja, a sugestão de deslocamento do coelho se dava através de uma previsão com base na observação de uma tela plana e do estabelecimento de relações mentais entre os sinais dos blocos (setas, meia lua etc.) com o deslocamento. Por fim, não podemos esquecer a analogia estabelecida por uma aluna entre a ideia de repetição e a roda gigante, um indicativo da importância de fazermos perguntas que convoquem as crianças a relacionar as suas vivências com as situações de sala de aula para complexificar conceitos e lhes atribuir sentidos.

4 Considerações finais

Ao final dessa sequência didática utilizando o *Google Doodle* é possível afirmar que ele foi uma boa estratégia de ensino e de aprendizagem para discutir a relevância da ordem e de um passo a passo para cumprir um objetivo, tanto com estudantes de início quanto de final de ciclo. Por ser um jogo fechado em si mesmo, ou seja, sem possibilidades de criação, ele foi importante para garantir que ambos os grupos trabalhassem as habilidades que queríamos desenvolver no projeto. O *Scratch* também se mostrou uma ferramenta intuitiva para utilizar com as crianças, mas foi explorado de diferentes formas com cada grupo em função dos produtos finais.

Em relação aos objetivos do trimestre, a construção do grupo no *Scratch* foi excelente para compartilhar estratégias, discutir as ideias com colegas e mobilizar saberes coletivamente. Como tudo era dialogado, o tempo dessa construção possivelmente seria diferente do tempo de um produto individual, e pensamos que, em outro momento, desenvolver produtos individuais pode ser interessante para explorar a criatividade e interesses de cada estudante. Não obstante, é importante atentar para o fato de que, no grupo, algumas crianças podem ficar mais reservadas, o que dificulta o acompanhamento do processo. No caso dessa atividade, ter mais uma docente na

chamada síncrona foi imprescindível para ajudar a convocar essas crianças a participarem. Também foram úteis as atividades não plugadas enviadas para casa e os registros como diário de bordo escrito ou por áudio ao final de cada encontro.

Ademais, os estudos piagetianos contribuíram para elaborar essa proposta tendo em vista os estádios de desenvolvimento e as relações que esperávamos que as crianças fizessem. O olhar sobre as ações dos e das estudantes foi complexificado a partir da concepção de esquema, assimilação e acomodação. Esses conceitos também permitiram analisar essa sequência didática de forma crítica, interpretando o pensamento computacional como potente estratégia para compreensão de conceitos.

A partir desse projeto, identificamos a construção de algoritmos como algo relevante a ser desenvolvido com as crianças pela sua relação com a resolução de problemas. Entendemos que habilidades como construir um passo a passo, compreender a importância da ordem e identificar padrões, que são associadas a objetos do conhecimento do pensamento computacional, são necessárias em diferentes contextos e podem ser desenvolvidas em diversas atividades, inclusive de forma transversal no currículo, envolvendo todas as disciplinas. Para trazer essa perspectiva para a escola, é imperativo discutirmos mais entre docentes de diferentes áreas sobre as possibilidades do pensamento computacional.

Não podemos deixar de finalizar falando sobre os desafios da docência e de desenvolver projetos como esse. O tempo da aula é ínfimo comparado ao tempo de planejamento do projeto, de pesquisa para encontrar o *Doodle*, de investigação do *Scratch* para mediar a interação das crianças com o *software*, de estudo das teorias para direcionar a ação, de preparação das atividades e do plano de aula, de leitura dos diários de bordo etc. Não temos dúvidas sobre a importância de projetos como esse e tantos outros que colegas desenvolvem em contextos ainda mais adversos e com menos condições que as descritas aqui. Entretanto, sabendo da dedicação que nos é exigida e do valor de nosso trabalho, é que defendemos que falar de projetos sem falar de valorização docente é, na verdade, investir na precarização da educação.

Referências

BRASIL. Ministério da Educação. *Base Nacional Comum Curricular*. Brasília, 2018. Disponível em: http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518-versaofinal_site.pdf. Acesso em: 20 fev. 2022.

MONTANGERO, Jaques; MAURICE-NAVILLE, Danielle. *Piaget ou a inteligência em evolução*. Porto Alegre: ArtMed, 1998.

PIAGET, Jean. *O nascimento da inteligência na criança*. 3. ed. Rio de Janeiro: Zahar Editores, 1978.

PIAGET, Jean. *Problemas de psicologia genética*. São Paulo: Abril Cultural, 1983.

ROCHA, Kátia; BASSO, Marcus Vinicius de Azevedo. Pensamento Computacional na Formação de Professores de Matemática. *RENOTE - Revista Novas Tecnologias na Educação*, Porto Alegre, v. 19, n.2, p. 426-436, 2021. Disponível em: <https://doi.org/10.22456/1679-1916.121366>. Acesso em 28 fev. 2022.

SBC - SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. *Diretrizes para ensino de Computação na Educação Básica*. 2019. Disponível em: <https://www.sbc.org.br/educacao/diretrizes-para-ensino-de-computacao-na-educacao-basica>. Acesso em: 20 fev. 2022.

VERGNAUD, Gérard. A teoria dos campos conceituais. In: BRUN, Jean. *Didática das matemáticas*. Lisboa: Instituto Piaget, 1996, p. 155-191.