

Mestrado Profissional em Educação
Profissional e Tecnológica - Profept

PRODUTO EDUCACIONAL

UMA SEQUÊNCIA DIDÁTICA PARA
DESENVOLVER LÓGICA
COMPUTACIONAL UTILIZANDO A
PLATAFORMA CODE.ORG

JORDANA VILELA MARTINS
JÚLIO CÉSAR FERREIRA

INSTITUTO FEDERAL GOIANO

INSTITUTO FEDERAL GOIANO

Mestrado Profissional em Educação
Profissional e Tecnológica - Profept

SEQUÊNCIA DIDÁTICA
DESENVOLVIDA COMO PARTE DOS
REQUISITOS PARA A OBTENÇÃO DO
TÍTULO DE MESTRA EM EDUCAÇÃO
PROFISSIONAL E TECNOLÓGICA DO
PROGRAMA DE PÓS-GRADUAÇÃO EM
EDUCAÇÃO PROFISSIONAL E
TECNOLÓGICA, OFERTADO PELO
CAMPUS CERES DO INSTITUTO
FEDERAL GOIANO.

UMA SEQUÊNCIA DIDÁTICA PARA
DESENVOLVER LÓGICA
COMPUTACIONAL UTILIZANDO A
PLATAFORMA CODE.ORG

JORDANA VILELA MARTINS
JÚLIO CÉSAR FERREIRA

Este obra está licenciado
com uma Licença Creative
Commons Atribuição-Não
Comercial 4.0 Internacional.



RESUMO

A partir da investigação de como o pensamento computacional como metodologia ativa de ensino de introdução à lógica computacional contribui com a formação integral dos estudantes de uma turma de ensino técnico de informática integrado ao médio em um campus do Instituto Federal no interior de Goiás, esta sequência didática, baseada no pensamento computacional, propôs uma metodologia ativa de ensino de introdução à lógica computacional. O tema foi delimitado a resolução de problemas. A finalidade da proposta foi desenvolver lógica a partir de uma metodologia ativa utilizando o pensamento computacional.

A Sequência Didática (SD) foi planejada para ser aplicada junto à disciplina de Lógica de Programação, Introdução à Lógica de Programação ou afins e também em projetos de Introdução ao Pensamento Computacional. Os conhecimentos prévios que o professor necessita para a aplicação da SD englobam Lógica Computacional e linguagem de programação em blocos. Já para o aluno espera-se que possua conhecimentos relacionados à utilização de tecnologias digitais para realização das atividades na plataforma utilizada.

QUEM SOMOS?

QUEM SOMOS?



JORDANA VILELA MARTINS

GRADUADA EM LICENCIATURA EM COMPUTAÇÃO E SISTEMAS PARA INTERNET PELO INSTITUTO FEDERAL DO TRIÂNGULO MINEIRO

MESTRA EM EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA DO INSTITUTO FEDERAL GOIANO



JÚLIO CÉSAR FERREIRA

PHD EM MATHSTIC PELA UNIVERSIDADE DE RENNES 1 (FRANÇA)

PROFESSOR NO INSTITUTO FEDERAL GOIANO

SUMÁRIO

1	INTRODUÇÃO	5
2	DEFINIÇÃO DE PRODUTO EDUCACIONAL E SEQUÊNCIA DIDÁTICA	6
3	DESCRIÇÃO DO PRODUTO EDUCACIONAL	8
4	BASE TEÓRICA PARA A SEQUÊNCIA DIDÁTICA	10
5	SOBRE A PLATAFORMA CODE.ORG	12
6	DESENVOLVIMENTO DO PRODUTO EDUCACIONAL	14
7	SEQUÊNCIA DIDÁTICA PARA DESENVOLVER LÓGICA COMPUTACIONAL UTILIZANDO A PLATAFORMA CODE.ORG	15
7.1	SÍNTESE DA APLICAÇÃO DA SEQUÊNCIA DIDÁTICA	16
7.2	RESUMO DA SEQUÊNCIA DIDÁTICA	18
7.3	ENCONTRO 1	19
7.4	ENCONTRO 2	22
7.5	ENCONTRO 3	16
7.6	CONTEÚDOS	27
	REFERÊNCIAS	30
	APÊNDICES	31







1 INTRODUÇÃO

Buscamos, nesse produto, apresentar uma Sequência Didática (SD) que propõe o ensino de Lógica Computacional (comandos e funções dos parâmetros da programação em blocos) no contexto das aulas de Lógica de Programação. Esta proposta faz parte da dissertação de Mestrado Profissional em Educação Profissional e Tecnológica (Profept) intitulada: Metodologias Ativas no Ensino Integrado: Pensamento Computacional como Metodologia de Ensino de Lógica Computacional.

A SD aborda introdução à Lógica de Programação utilizando programação em blocos, por meio de situações didáticas com foco nos anos iniciais do ensino técnico integrado ao médio utilizando recursos da plataforma de programação Code.org, tendo em vista o documento oficial BNCC e as Diretrizes Curriculares Nacionais Gerais para a Educação Profissional e Tecnológica.

Vamos abordar aqui: definição de produto educacional (PE) situando as algumas possíveis alternativas de produtos e afinando para a descrição do PE desenvolvido; uma apresentação e descrição da plataforma Code.org; e, por fim, a proposta do PE que é uma Sequência Didática (SD) utilizando a plataforma Code.org.



2 DEFINIÇÃO DE PRODUTO EDUCACIONAL E SEQUÊNCIA DIDÁTICA

Como parte da dissertação para o Mestrado Profissional, necessita-se criar um produto educacional. Os produtos educacionais são ferramentas pedagógicas elaboradas para viabilizar a prática pedagógica, havendo a necessidade de constante avaliação e reestruturação para contribuições com os atos de aprender e ensinar. (FREIRE; GUERRINI; DUTRA, 2016).

O produto educacional como objeto para auxílio educacional deve ser desenvolvido com sustentação em pesquisa científica buscando contribuir para a prática profissional na educação, apresentando uma proposta de ensino ou de mudanças na realidade de uma instituição de ensino. Há uma vasta gama de produtos educacionais. Alguns deles são apresentados na imagem a seguir.

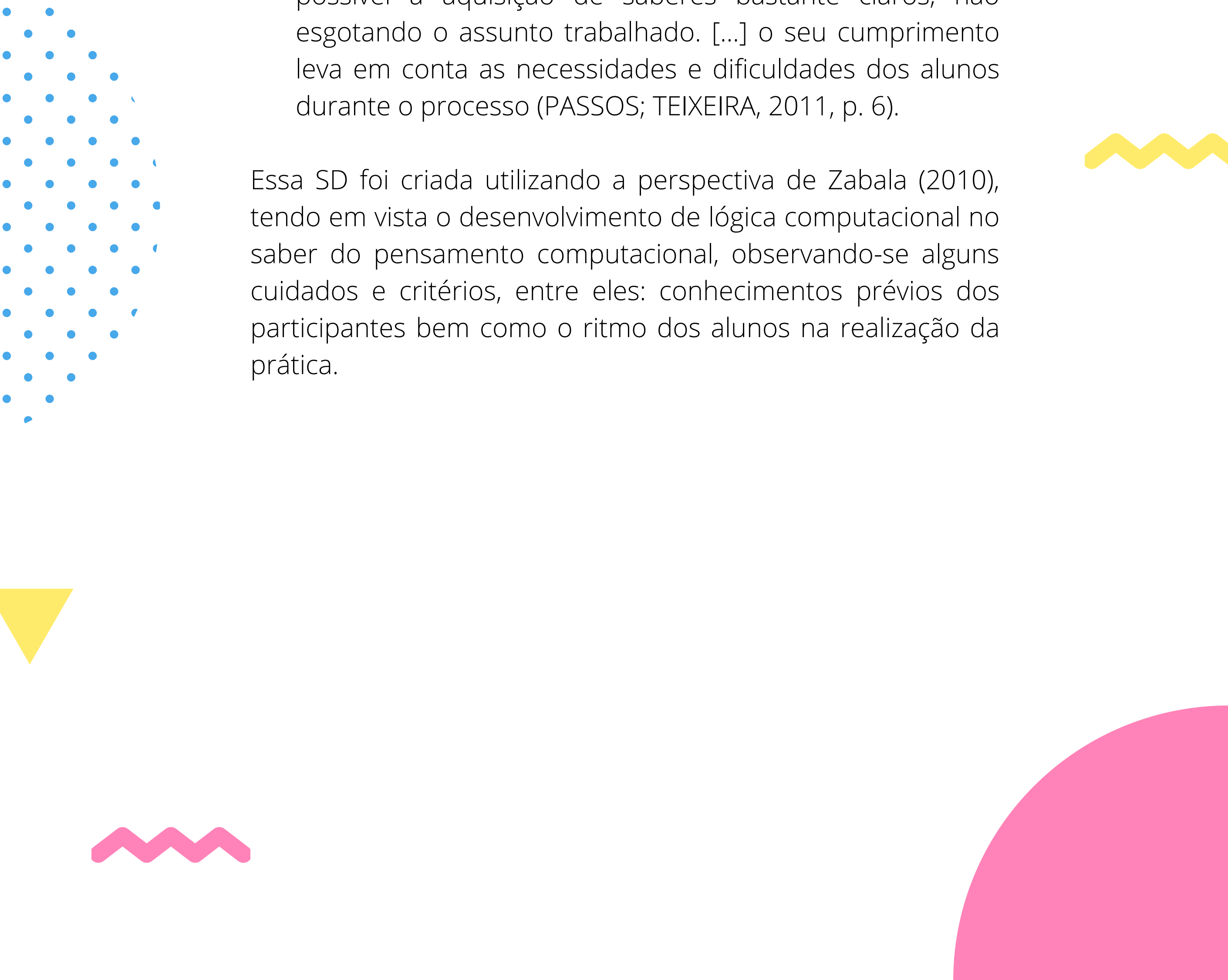




SEQUÊNCIA DIDÁTICA

Este produto educacional tem o formato de sequência didática (SD). Segundo Passos e Teixeira (2011), a sequência didática tem como característica tratar de:

Uma série de situações que se estruturam ao longo de uma quantidade pré-fixada de aulas, estas situações, devidamente estruturadas, têm como objetivo tornar possível a aquisição de saberes bastante claros, não esgotando o assunto trabalhado. [...] o seu cumprimento leva em conta as necessidades e dificuldades dos alunos durante o processo (PASSOS; TEIXEIRA, 2011, p. 6).



Essa SD foi criada utilizando a perspectiva de Zabala (2010), tendo em vista o desenvolvimento de lógica computacional no saber do pensamento computacional, observando-se alguns cuidados e critérios, entre eles: conhecimentos prévios dos participantes bem como o ritmo dos alunos na realização da prática.

3 DESCRIÇÃO DA SEQUÊNCIA DIDÁTICA

Utilizamos a plataforma Code.org nesta SD visando maior envolvimento dos participantes com relação às atividades propostas e possibilitando a realização de atividades síncronas e assíncronas.

Essa SD tem como objetivo disponibilizar aos professores que ensinam lógica de programação nos anos iniciais do ensino médio mais uma possibilidade de realizar o ensino de lógica computacional, de acordo com os pressupostos de Zabala (2010) ao pensar realização da atividade como prática educativa buscando abranger os eixos factual, conceitual, pedagógico e atitudinal.

Procurou-se trazer a essa proposta, preceitos de Kaplún (2003) durante a criação do material educativo com o intuito de gerar meios de desenvolver uma experiência de aprendizado abrangendo os eixos conceitual, pedagógico e comunicacional.

A dinâmica da SD baseada na plataforma Code.org tem atividades que são compostas por uma sucessão de desafios relacionados a um personagem fictício que envolve conceitos de lógica computacional para a tomada de decisões de maneira que esse personagem possa atingir seu objetivo dentro do cenário proposto.

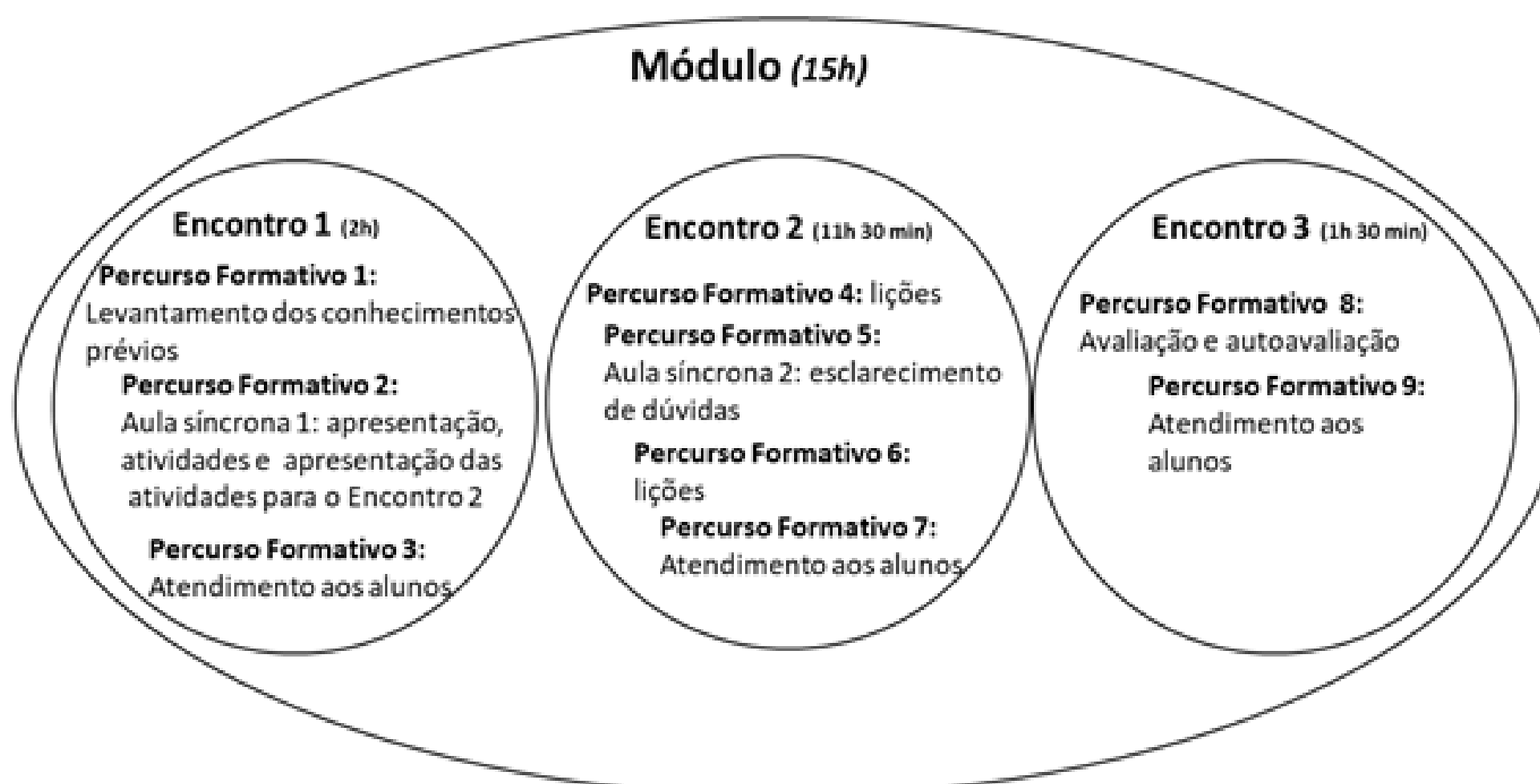
Planejada para ser desenvolvida em um módulo de 15 horas, esta SD está dividida em 3 encontros. O Encontro 1 é reservado para a investigação dos conhecimentos prévios (Apêndice A) que os alunos trazem com eles e para a introdução aos recursos da plataforma Code.org por meio do desenvolvimento de lições e atividades da plataforma junto com os alunos.

O Encontro 2 está organizado de modo a possibilitar que o aluno construa o conhecimento durante a prática das atividades constantes nas lições do Code.org previamente elaboradas.

No Encontro 3, além da avaliação formativa e processual desenvolvidas nos dois encontros anteriores, é realizada a avaliação (Apêndice D) e autoavaliação (Apêndice E) dos alunos.

Todos os encontros são constituídos por percursos formativos síncronos (aulas síncronas e atendimento aos alunos) e assíncronos (reservados para o trabalho ativo no aluno).

Uma visão mais geral da SD é apresentada na figura a seguir, onde, para cada encontro do módulo há um tempo estipulado para a realização da proposta do percurso formativo



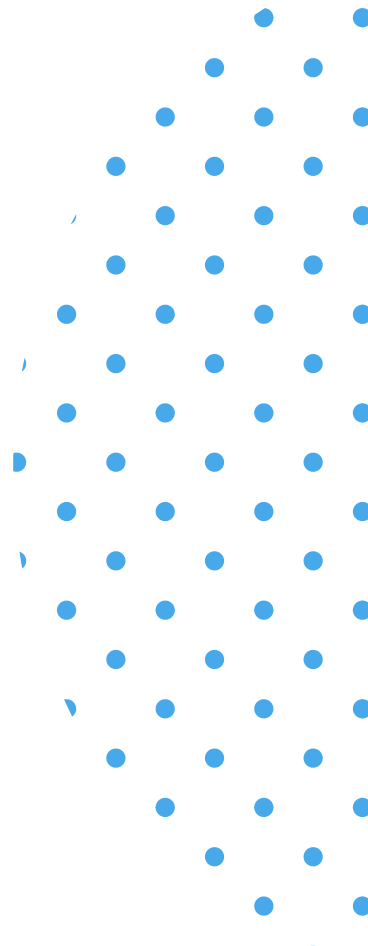
Buscamos nessas atividades abranger circunstâncias em que podem emergir diferentes noções de lógica computacional em articulação com algumas unidades temáticas propostas na BNCC (2018). Entre elas, a busca por “[...] consolidar os conhecimentos [...] e agregar novos, ampliando o leque de recursos para resolver problemas mais complexos, que exigem maior reflexão e abstração.” (BNCC, 2018, p. 471).

Destacamos também a busca por avançar na identificação de padrões a fim de estabelecer generalizações, propriedades e algoritmos. E, a resolução de problemas, de investigação e de desenvolvimento de projetos com potencial para desenvolver raciocínio, representação, comunicação e argumentação, favorecidos pelo desenvolvimento do pensamento computacional.

Somando-se a isso, vislumbramos possibilitar a articulação com as Diretrizes Curriculares Nacionais Gerais para a Educação Profissional e Tecnológica (2021) ao passo que a SD oportuniza contemplar métodos, técnicas, ferramentas e outros elementos das tecnologias.




4 BASE TEÓRICA PARA A SEQUÊNCIA DIDÁTICA




Para Zabala (2010), a prática educativa pode compreender quatro tipos de conteúdos: **factual**, que alcança conhecimentos de fatos e situações, dados e fenômenos; **conceitual**, que envolve conceitos que os estudantes devem saber a partir da prática educativa; **procedimental**, que são os caminhos que devem ser percorridos para se chegar aos objetivos da prática educativa; e **atitudinal**: que compreende valores, atitudes e normas. Para a SD criada, buscamos um enlace entre os tipos de conhecimento.

Zabala (2010) afirma que “a aprendizagem quase nunca pode ser considerada acabada, já que sempre existe a possibilidade de ampliar ou aprofundar seu conhecimento, de fazê-la mais significativa” (ZABALA, 2010, p. 43), ou seja, a aprendizagem está sempre em construção.


Nesse sentido, os conteúdos conceituais são importantes na medida em que representam a expansão de campos para a construção da aprendizagem. Nos conteúdos procedimentais, segundo Zabala (2010), incluímos **aprendizagem de técnicas**, métodos no desenvolvimento de habilidades que são realizadas com alguma finalidade. Já os conteúdos atitudinais correspondem à **construção de atitudes e valores** no que se refere à informação recebida, tendo em vista a intervenção do aluno em sua realidade.





A respeito dos conteúdos de caráter atitudinal, Zabala (2010) afirma que não podem ser aprendidos através da simples leitura, mas que “os textos escritos podem servir como complementos ou indutores de reflexões, diálogos e debates, ao mesmo tempo em que como **veiculadores de determinados valores**” (ZABALA, 2010, p. 183).



Ao elaborar um produto educacional devemos incluir como proposta trabalhar conteúdos atitudinais como valores, normas e atitudes que, segundo Zabala (2010), podem possibilitar a vivência com o mundo que cerca o aprendiz.





Além disso, ao elaborar o produto educacional como material educativo buscou-se em Kaplún (2003) atentar para alguns cuidados e critérios que devem ser observados principalmente durante a sua aplicação, compreendendo o **material educativo** como algo que **possibilita o desenvolvimento de uma experiência de aprendizado.**

Kaplún (2003) relata a necessidade de pensar sobre os procedimentos necessários para o planejamento e a elaboração de produtos educacionais, sobretudo materiais textuais. Para a elaboração do material educativo de qualidade, sugere basear o produto educativo em três eixos: **conceitual, pedagógico e comunicacional.**

Kaplún (2003) pontua que o **eixo conceitual** é definido como o **conhecimento da matéria em questão**, os conceitos que a articulam e os debates gerados por ela, a escolha das ideias centrais e o tema. Na SD aplicada, o eixo conceitual abarca conceitos de lógica computacional.

Quanto ao **eixo pedagógico**, Kaplún (2003, p. 49) define como “o **articulador principal de um material educativo**” estabelecendo um ponto de partida e um ponto de chegada a quem o material está direcionado. Segundo o autor, através desse eixo é proposto um caminho na busca por efetivas mudanças e enriquecimento de concepções, percepções e valores.

Ao pensar o eixo pedagógico, para Kaplún temos que, no mínimo “[...] conversar sobre o tema com os sujeitos que serão, potencialmente, usuários do material” (KAPLÚN, 2003, p. 50), buscando levar em conta o conhecimento prévio do envolvido no processo de aprendizagem bem como o meio no qual ele está inserido. Na SD aplicada, o eixo pedagógico abrange os conhecimentos prévios dos alunos como a utilização de meio digital para realização das atividades adentrando a realidade dos alunos nativos digitais.

Por fim, o **eixo comunicacional** para Kaplún (2003) trata-se do **veículo no qual se percorre o eixo pedagógico.** O autor sugere o uso da criatividade, da capacidade de brincar e de romper moldes. Na SD aplicada, o eixo comunicacional envolve a familiarização da plataforma com o contexto do dia a dia dos alunos.

Em síntese, caracteriza-se a SD aplicada, segundo os três conteúdos propostos por Kaplún (2003) da seguinte forma: eixo conceitual como a lógica computacional; eixo pedagógico como aprender a aprender utilizando a lógica do pensamento computacional dividida em decomposição, abstração, reconhecimento de padrões e algoritmos; e eixo comunicacional como estrutura de “jogos” baseada na plataforma Code.org.

5 SOBRE A PLATAFORMA CODE.ORG

“Code.org é uma organização sem fins lucrativos dedicada a expandir o acesso à ciência da computação nas escolas e aumentar a participação de mulheres jovens e estudantes de outros grupos sub-representados” (CODE, 2021). Para isso, a Code.org, apoiada por empresas como Microsoft, Facebook, Amazon, Infosys Foundation, Google, entre outros, tem o objetivo de levar a oportunidade de aprender Ciência da Computação aos alunos de todas as escolas, de maneira que possa ser incorporada à educação básica de ensino fundamental e médio por meio de uma plataforma de acesso.

A Code.org defende que a Ciência da Computação deve ser uma base para todos os alunos. Atualmente, com alcance mundial, seus cursos estão disponíveis em mais de 67 idiomas e acessados em mais de 180 países (CODE, 2021).

Entre as características oferecidas pela plataforma, destacam-se: interatividade entre os recursos; possibilidade para o aluno de aprender seguindo seu próprio ritmo de aprendizagem; e possibilidade para o educador de criar, gerenciar e acompanhar turmas com seus alunos a partir da plataforma.

Pela interface de entrada da plataforma Code.org (figura a seguir) é possível ter acesso a catálogo de cursos, projetos com acesso livre ou projetos com acesso por conta.



Dentre estes, o acesso por conta possibilita ao professor a criação e gerenciamento de turmas, e possibilita ao aluno participação e acompanhamento das atividades da turma.

Catálogo de Cursos Projetos Sobre

Já tens uma conta? Inicia sessão

Nome utilizador ou e-mail

Palavra-passo

[Esqueceste-te da palavra-passo?](#)

[Iniciar sessão](#)

[Ainda não criaste uma conta?](#)

[Criar uma conta](#)

Insere o teu código de verificação de 6 letras

Código (ABCDE) [Go](#)

[Continuar com Google](#)

[Continuar com Facebook](#)

[Continuar com Microsoft](#)

OU

Queres experimentar programar sem teres de criar uma conta?

Festa da Dança

Featuring Katy Perry, Shawn Mendes, Panic! At The Disco, Lil Nas X, Jonas Brothers, Nicki Minaj, and 34 more!

Minecraft

Use sua criatividade e habilidades para resolver problemas para explorar e construir mundos aquáticos com código!

Frozen

Use a programação para explorar a magia e a beleza do gelo com Anna e Elsa.

Labirinto clás...

Explora as noções básicas da Ciência da Computação. Milhões de pessoas já experimentaram.

[Política de privacidade](#) | [Voluntariado para traduzir o nosso conteúdo](#) | [Ajuda e suporte](#) | [Termos](#)

Português (Brasil)

© Code.org, 2021

Os parceiros da Amazon, Google e da Microsoft ajudaram a criar estes materiais.

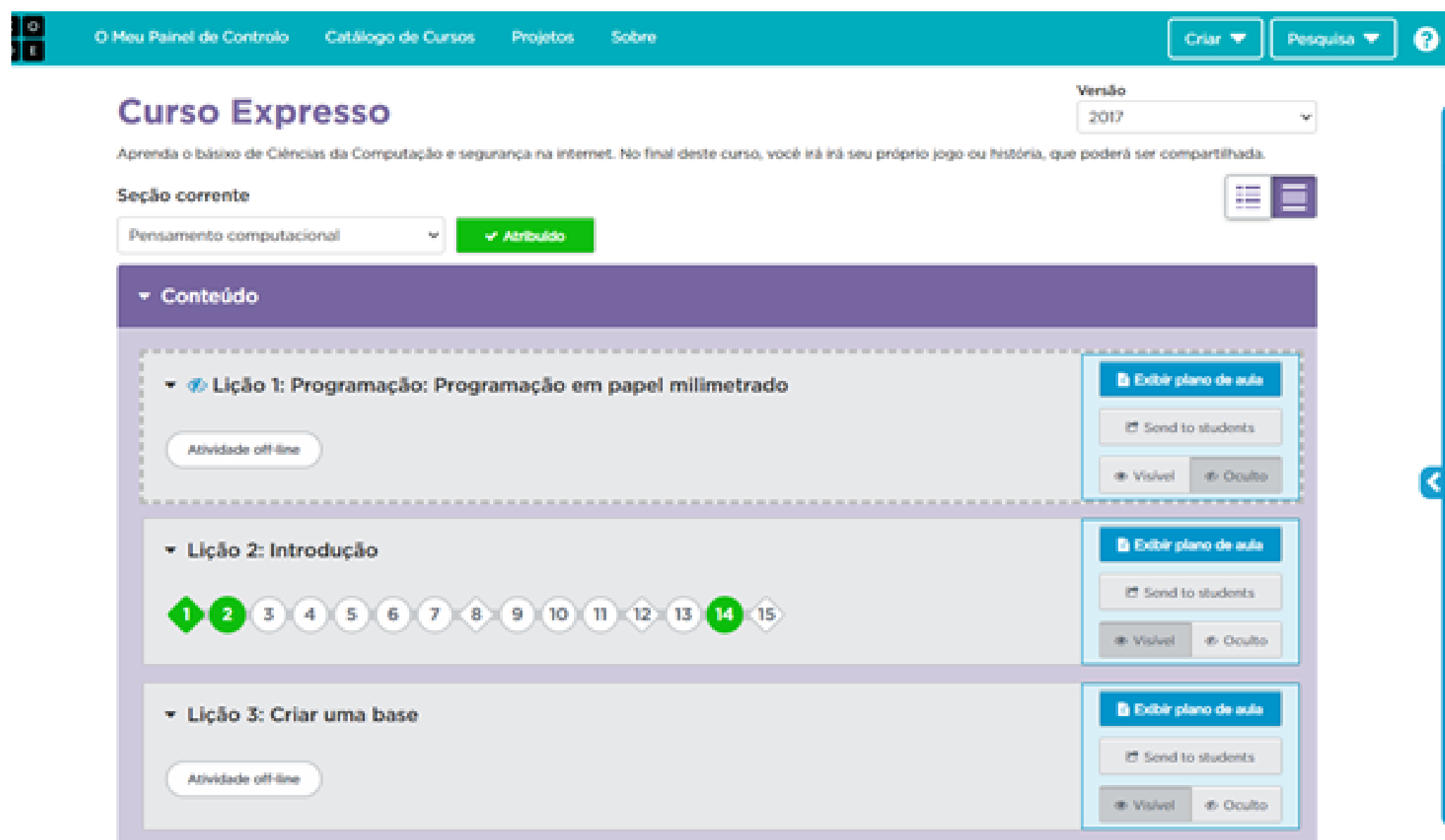
Minecraft™ de 2011 Microsoft. Todos os Direitos Reservados. Star Wars™ © 2019 Disney. Todos os direitos reservados.

Para a SD proposta, utilizamos uma conta de professor e contas para alunos de maneira que se seja possível acessar às atividades de cada aluno dentro da plataforma.

O *Apêndice B* apresenta as etapas para criação de conta no Code.org para professor, bem como a criação de turma e escolha do curso, enquanto o *Apêndice C* apresenta as etapas para criação de conta no Code.org para professor e para os alunos

6 DESENVOLVIMENTO DO PRODUTO EDUCACIONAL

O produto educacional “Uma sequência didática para desenvolver lógica computacional utilizando a plataforma Code.org” foi desenvolvido a partir de lições do “Curso Expresso” versão 2017 da plataforma Code.org. Esta versão foi escolhida devido à estabilidade, suporte para o idioma português e o nível das lições.



Para realização desta sequência didática é necessário que o professor e que os alunos tenham uma conta na plataforma Code.org.

Está em dúvida sobre como criar estas contas?

Lá no final você encontra o passo a passo para criar estas contas você encontra no APÊNDICE B (página 34) e no APÊNDICE C (página 39).

The background features a vibrant, abstract design with various geometric elements. In the top left, there's a pink triangle. The top right has a large yellow circle with a pink triangle and a cyan wavy line. A cyan circle with blue dots is on the left. The bottom right has a large pink circle with a cyan wavy line. The bottom left has a yellow triangle and a pink wavy line. A cyan triangle is on the right side. The number '7' is positioned at the top left of the main text area.

7

UMA SEQUÊNCIA DIDÁTICA PARA DESENVOLVER LÓGICA COMPUTACIONAL UTILIZANDO A PLATAFORMA CODE.ORG

Esta sequência didática (SD) foi pensada para ser realizada remotamente, por isso, ferramentas didáticas, de avaliação e de comunicação foram propostas para esta finalidade.

Ao término das atividades foi planejada uma avaliação e uma autoavaliação.

Todos os encontros podem ser aplicados individualmente e remotamente pelo computador, celular ou tablet com acesso à internet.

Para iniciar, leia atentamente as próximas páginas, nelas você encontrará uma introdução sobre o pensamento computacional, a abordagem de aprendizagem e sua relação com o formato escolhido para a SD proposta aqui.

7.1 SÍNTESE DA APLICAÇÃO DA SEQUÊNCIA DIDÁTICA

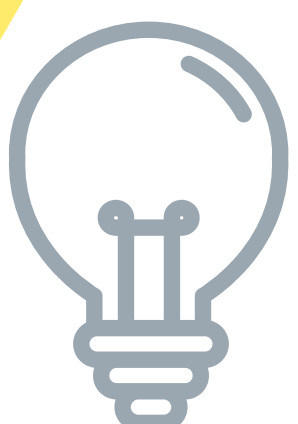
O PENSAMENTO COMPUTACIONAL é “uma distinta forma de pensamentos com conceitos básicos da Ciência da Computação para resolver problemas, desenvolver sistemas e para entender o comportamento humano, habilidade fundamental para todos.” WING (2006).

O pensamento computacional traz como características: a formulação de problemas usando o computador como ferramentas para ajudar a resolvê-los; a organização lógica e análise de dados; a representação de dados por meio de abstrações e simulações; a automação de soluções considerando o pensamento algorítmico; a identificação, análise e realização de soluções possíveis buscando alcançar efetiva combinação de etapas e recursos; e a generalização e transferência desse processo de resolução de problemas para uma ampla variedade de problemas. (VALENTE, 2019).

Seguindo a corrente construtivista de aprendizagem, acreditamos que para que o aluno aprenda ele deve se apropriar do conhecimento. Durante esse processo, ele elabora estratégias para a solução de um problema que envolve investigação e reflexão sobre a prática realizada (PAPERT, 1988, 2008). Dessa forma, possibilitando o ato de “pensar sobre a forma de pensar”, o aprendiz pode adotar o melhor estilo cognitivo para a solução de um problema e descobrir seu “estilo de pensamento”.

Nesse contexto, a lógica desenvolvida pelo Pensamento Computacional oferece a possibilidade de desenvolvimento tanto cognitivo quanto emocional dos educandos com a construção de significados mediante o “pensar o pensar”.

Segundo Zabala (2010), para que o aluno **aprenda significativamente** a aula deve ser pensada para abranger diferentes aspectos de aprendizagem que podem ser atingidos de acordo com a abordagem que se dá ao conteúdo. O autor classifica os conteúdos de acordo com a predominância de cada aula em conteúdo: factual, conceitual (Atividade: 1), procedimental (Atividade 2) e atitudinal (Atividade 3).



Factuais: fatos, acontecimentos, relativo a repetições - **Decorar**

Conceituais: situações, dados e fenômenos concretos e singulares - **Saber**

Procedimentais: um conjunto de ações dirigidas para um fim - **Saber fazer**

Atitudinais: valores, atitudes e normas - **Admitir ser**

As atividades propostas nessa SD buscaram abranger esses quatro pontos de aprendizagem e tem o objetivo conduzir o aluno a desenvolver conhecimentos a partir dos quatro níveis propostos por Zabala (1998) de maneira a não somente encontrar a melhor resolução das atividades propostas, mas, desenvolver processos mentais por meio do Pensamento Computacional.

Assim, espera-se que o aluno avance em competências e habilidades para resolver problemas do seu dia a dia, além da capacidade de pensar de forma estruturada na resolução de problemas, desenvolvendo sua criatividade e **raciocínio lógico computacional**.

A estratégia utilizada com as atividades envolve o desenvolvimento de ações exemplificadas pela existência de um personagem que o aluno conduzirá por meio de comandos, movendo-o na direção correta, com movimentos estratégicos para que atinja o seu objetivo.



Para isso, o aluno deve sistematizar a sequência de ações mais eficaz para **atingir seu objetivo**. Durante a construção da resolução, o aluno deve refletir sobre a melhor estratégia visto que uma vez executada, o personagem seguirá os passos propostos, atingindo ou não o objetivo, dessa forma, levando o aluno a refletir e participar da construção do próprio conhecimento.

O Pensamento Computacional relaciona-se a essa proposta na medida em que o aluno apropria-se do conhecimento relacionado ao conceito e no processo de construção do próprio conceito podendo utilizá-lo em outras situações.

A seguir, você verá o resumo de como poderá ser aplicada a Sequência Didática.



7.2 RESUMO DA SEQUÊNCIA DIDÁTICA

Tempo estimado para aplicação da sequência didática: 15 horas (três encontros).

Público alvo: alunos da educação básica.

Requisito mínimo: ter acesso a um computador, celular ou tablet com acesso à internet.

Tema: Lógica Computacional.

Conteúdos trabalhados: Resolução de problemas e Lógica Computacional.

Objetivo: desenvolver a Lógica Computacional a partir de estruturas de pensamento computacional utilizando a plataforma Code.org

Habilidades desenvolvidas na BNCC: 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.

Materiais utilizados: computador, celular ou tablet com acesso à internet.

Recursos utilizados: Code.org, Google Classroom, Google Meet, Google Forms.

As atividades realizadas durante a SD podem ser sintetizadas em três encontros que são compostos por **Percursos Formativos** que podem ser observados com mais detalhes no quadro abaixo:

Encontro	Percorso formativo	Tempo estimado
Encontro 1	1- Levantamento dos conhecimentos prévios	20 min.
	2- Aula síncrona 1: apresentação do Code, realização de atividades e apresentação das atividades para o aluno	50 min.
	3- Atendimento aos alunos	50 min.
Encontro 2	4- Alunos fazendo um conjunto de lições do Code.org	5h
	5- Aula síncrona 2: esclarecimento de dúvidas	40 min.
	6- Continuidade alunos fazendo lições do Code.org	5h
	7- Atendimento aos alunos	50 min.
Encontro 3	8- Aula síncrona 3: avaliação e auto avaliação	40 min.
	9- Atendimento aos alunos	50 min.
Total:		15h



A seguir, a descrição de cada um dos três Encontros.



7.3 ENCONTRO 1

O **Encontro 1**, com duas horas de duração, traz três Percursos Formativos:

- 1) levantamento dos conhecimentos prévios;
- 2) apresentação da plataforma Code.org;
- 3) atendimento ao aluno.

7.3.1 Percurso formativo 1: Levantamento dos conhecimentos prévios (duração 20 min.).

Objetivo geral: Estabelecer os conhecimentos prévios dos alunos e possibilitar definir a aplicação das próximas atividades.

Resumo: Disponibilização do Formulário com questões sobre lógica computacional.

1. O professor envia questionário sobre lógica computacional (Apêndice B);
2. Baseados no que já conhecem, os alunos respondem o formulário;
3. As respostas são analisadas pelo professor e os conteúdos das próximas aulas são alterados de acordo com as necessidades de aprendizado dos alunos.

Organização: sala do Google Classroom.



7.3 ENCONTRO 1

7.3.1 Percurso formativo 2: apresentando a plataforma Code.org. (duração 50 min.)

Objetivo: apresentar a plataforma de realização das atividades. Dialogar com os alunos sobre a resolução de problemas propostos. Desenvolver o conceito de programação por blocos.

Resumo: os alunos terão o primeiro contato com a plataforma Code.org

1. O professor disponibiliza os links de acesso para a realização das atividades;
2. O professor disponibiliza manual de acesso às plataformas que serão utilizadas (Apêndice C);
3. Os alunos, juntamente com o professor, acessam a plataforma Code.org;
4. Alunos e professor interagem sobre os recursos que serão utilizados;
5. O professor disponibiliza as atividades que serão realizadas pelos alunos (Apêndice H);
6. Os alunos realizam a primeira atividade juntamente com o professor e colegas e trocam conhecimentos sobre a resolução das atividades propostas.

Organização da turma: em encontro virtual pelo Google Meet.

Busca por soluções: realizada por meio das conversas entre os alunos e professor durante o encontro.

Exposição do conceito: o professor demonstra recursos de utilização da plataforma e aproveita as propostas dos alunos para elaborar um dos caminhos para solução do problema, gerando um conceito construído pelo grupo para criar meios mais eficazes para a resolução dos problemas propostos.

Generalização: os alunos acompanham o professor e realizam a sua atividade observando se a lógica proposta, experimentando em outras situações das novas condições da atividade que segue.

Realização da atividade e compartilhamento de dúvidas: os alunos aplicam o modelo a outras situações. Os alunos participam do compartilhamento de dúvidas entre si e com o professor.

Realização do exame: participação dos alunos e realização da atividade proposta.

Avaliação: o professor comunica aos alunos os resultados obtidos ao final do encontro.

Reflexão: os alunos respondem um formulário sobre as atividades com questões auto reflexivas disponibilizado no Classroom.

Atividades de casa: o professor disponibiliza o itinerário da atividade seguinte para que seja realizada em casa.

7.3 ENCONTRO 1

7.3.3 Percurso formativo 3: atendimento aos alunos (duração 50 min.).

Objetivo: desenvolver conceitos de programação em blocos e Lógica Computacional.

Resumo: após os alunos terem o primeiro contato com a plataforma e realizado a tentativa de solucionar os problemas das lições:

1. O professor disponibiliza aos alunos horários de atendimento individual ou em grupo para desenvolverem soluções sobre as Lições disponíveis as quais os alunos estejam com maior dificuldade;
2. Os alunos que manifestam interesse em relação a participar desse tipo de dinâmica marcam um encontro com encontro remoto individual com o professor;
3. O professor instiga o aluno a resolver o problema.

Organização: em encontro virtual pelo Google Meet ou Whatsapp. Busca por soluções: realizada por meio das conversas entre os alunos e professor durante o encontro.

Generalização: os alunos juntamente com o professor realizam as lições observando a lógica proposta experimentando as novas atividades.

Compartilhamento de dúvidas: os alunos participam do compartilhamento de dúvidas entre si e com o professor.

7.4 ENCONTRO 2

O encontro 2 tem duração de 11h 30 min.

7.4.1 Percurso formativo 4: realização das atividades da plataforma Code.org (duração: 5 horas).

Objetivo: desenvolver a lógica computacional a partir de estruturas de Pensamento Computacional utilizando a plataforma Code.org., além de utilizar sintaxes lógicas usadas nas linguagens de programação e incluir conceitos dos quatro pilares do Pensamento Computacional: abstração, decomposição, reconhecimento de padrões e algoritmos.

Resumo: por meio do contexto da plataforma (semelhante a um jogo) os alunos são envolvidos no tema em torno dos aspectos problemáticos da resolução de problemas. As atividades são realizadas pelo aluno e por ser totalmente on-line, o aluno determina o dia e a hora em que realiza a atividade (dentro de um espaço de tempo limite estipulado pelo professor).

Proposição de problemas ou questões: os alunos, dirigidos pela proposta do curso na plataforma Code.org respondem sobre cada um dos problemas e situações propostos construindo a sequência lógica.

Organização da turma: cada aluno realiza o estudo individual utilizando a plataforma Code.org e experimentando diferentes formas de resolver o problema por meio da técnica de construção do código por blocos.

Propostas das fontes de informação: os alunos propõem fontes de informação mais apropriadas em caso de dúvidas em alguma das questões (troca de informação de fontes por alunos e professor pelo grupo no Google Classroom).

Elaboração das conclusões: os alunos coletiva ou individualmente elaboram conclusões referentes às questões e aos problemas propostos.

Generalização das conclusões e síntese: a partir da realização das atividades, os alunos criam seus modelos e princípios que deduzem o trabalho realizado.

Prova ou exame: a plataforma Code.org possibilita ao professor visualizar as tentativas e acertos dos alunos, bem como o seu processo de desenvolvimento. Dessa forma, esse recurso oferece informações valiosas no processo de avaliação sendo utilizado como ferramenta para tal.

Avaliação: a plataforma Code.org informa ao aluno, instantaneamente e individualmente, seu progresso após a realização de cada atividade realizada.

7.4 ENCONTRO 2

7.4.2 Percurso formativo 5: esclarecimento de dúvidas oriundas das atividades desenvolvidas no percurso formativo 4 (duração 40 min).

Objetivo: desenvolver conceitos de programação por blocos e Lógica Computacional.

Resumo: após os alunos terem o primeiro contato com a plataforma e realizado a tentativa de solucionar os problemas das lições:

1. O professor disponibiliza aos alunos, no Classroom da turma, um encontro síncrono para que grupo possa desenvolver soluções sobre as lições que o grupo de alunos estejam com maior dificuldade;
2. Os alunos que manifestam interesse em participar desse percurso comparecem ao encontro remoto dialogam sobre as suas soluções e investigam possibilidades de otimização de suas respostas;
3. O professor atua como mediador do encontro.

Organização: em encontro virtual pelo Google Meet com link enviado disponibilizado no Classroom da turma.

Busca por soluções: realizada por meio das conversas entre os alunos e professor durante o encontro.

Generalização: os alunos juntamente com o professor analisam as suas soluções, comparam-nas desenvolvem habilidades para experimentar as atividades seguintes.

Compartilhamento de dúvidas: os alunos participam do compartilhamento de dúvidas entre si e com o professor.

7.4 ENCONTRO 2

7.4.3 Percurso formativo 6: continuidade da realização das atividades da plataforma Code.org(duração: 5 horas).

Objetivo: desenvolver a Lógica Computacional a partir de estruturas de Pensamento Computacional utilizando a plataforma Code.org. além de utilizar sintaxes lógicas usadas nas linguagens de programação e incluir conceitos dos quatro pilares do Pensamento Computacional: abstração, decomposição, reconhecimento de padrões e algoritmos.

Resumo: os alunos realizam as lições da plataforma Code.org.

Proposição de problemas ou questões: Os alunos, dirigidos pela proposta do curso na plataforma Code.org respondem sobre cada um dos problemas e situações propostos construindo a sequência lógica.

Organização da turma: cada aluno realiza o estudo individual utilizando a plataforma Code.org e experimentando diferentes formas de resolver o problema por meio da técnica de construção do código por blocos.

Propostas das fontes de informação: os alunos propõem fontes de informação mais apropriadas em caso de dúvidas em alguma das questões (troca de informação de fontes por alunos e professor pelo Google Classroom).

Elaboração das conclusões: os alunos, coletiva ou individualmente, elaboram conclusões referentes às questões e aos problemas propostos.

Generalização das conclusões e síntese: a partir da realização das atividades, os alunos criam seus modelos e princípios que deduzem o trabalho realizado.

Prova ou exame: a plataforma Code.org possibilita ao professor visualizar as tentativas e acertos dos alunos, bem como o seu processo de desenvolvimento. Dessa forma, esse recurso oferece informações no processo de avaliação sendo utilizado como ferramenta para tal.

Avaliação: a plataforma Code.org informa ao aluno, instantaneamente e individualmente, seu progresso após a realização de cada atividade realizada.

7.4 ENCONTRO 2

7.4.4 Percurso formativo 7: atendimento aos alunos. (Duração 50 min.)

Objetivo: desenvolver conceitos de programação em blocos e Lógica Computacional.

Resumo: após os alunos adquirirem mais familiaridade com os recursos e interface da plataforma Code.org e terem realizado a tentativa de solucionar os problemas das lições restantes:

1. O professor disponibiliza aos alunos, no grupo da turma, horários de atendimento individual ou em grupo para desenvolverem soluções sobre as Lições disponíveis as quais os alunos estejam com maior dificuldade;
2. Os alunos que manifestam interesse em relação a participar desse tipo de dinâmica agendam um encontro com encontro remoto individual com o professor;
3. O professor instiga o aluno a resolver o problema.

Organização: em encontro virtual pelo Google Meet ou Whatsapp.

Busca por soluções: realizada por meio das conversas entre os alunos e professor durante o encontro.

Generalização: os alunos, juntamente com o professor, realizam as lições observando a lógica proposta experimentando as novas atividades.

Compartilhamento de dúvidas: os alunos participam do compartilhamento de dúvidas entre si e com o professor.



7.5 ENCONTRO 3

O Encontro 3 tem duração de 1h 30 min.



7.5.1 Percurso Formativo 8: avaliação e autoavaliação (duração: 40 min.)

Objetivo: identificar a habilidade de formação e solução de problemas, baseando-se nos conceitos fundamentais da Computação.

Resumo: aplicação de instrumento de avaliação constituído por questões de múltipla escolha e aplicação de instrumento de autoavaliação.

Organização da turma: em encontro virtual pelo Google Meet com link disponibilizado no Google Classroom.

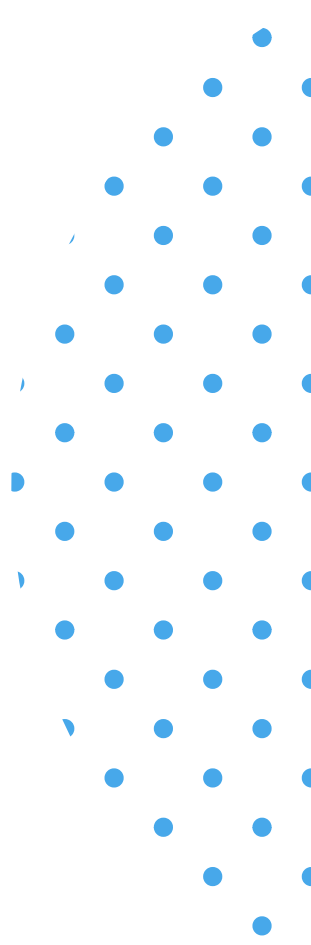



Prova ou exame: os alunos realizam remotamente a avaliação e autoavaliação (Apêndice E).

7.5.2 Percurso Formativo 9: atendimento aos alunos com devolutiva das avaliações (duração: 50 min.)

Objetivo: estabelecer um feedback ao aluno com relação ao seu desempenho durante os percursos formativos.

Resumo: os alunos e o professor dialogam sobre a experiência de aprendizagem, sobre os recursos utilizados e sobre a forma de avaliação e desempenho próprio.

Organização da turma: em encontro virtual pelo Google Meet com link disponibilizado no Google Classroom.



7.6 CONTEÚDOS

Os conteúdos e descrição das lições planejadas para os percursos formativos 2, 4 e 6 são aqueles que incluem a realização de Atividades dentro da plataforma Code.org.

Encontro	Percurso formativo	Tempo estimado
Encontro 1	2- Aula síncrona 1: apresentação do Code, realização de atividades e apresentação das atividades para o aluno	50 min.
Encontro 2	4- Alunos realizando um conjunto de lições do Code.org	5h
	6- Continuidade alunos fazendo lições do Code.org	5h
Total:		10h 50 min.

As atividades propostas no formato da plataforma Code.org dentro desses três percursos formativos são compostas por Lições que, por sua vez, são constituídas por Exercícios.

A seguir, um quadro contendo todas as lições e exercícios de cada Percurso Formativo que utilizamos na plataforma Code.org para esta SD.

7.6 CONTEÚDOS

Conteúdos	Descrição das Lições
<p>Percurso Formativo 2 (50 min.) Explorar a plataforma Code.org. Analisar padrões de combinação e solução de problemas.</p> <p>Lições: Curso Expresso [1] Lição 2 Introdução [2] Exercícios: 1 a 13.</p>	<p>Os alunos irão explorar a plataforma Code.org. Essa pode ser a primeira experiência deles com codificação visual com blocos de maneira a explorar padrões. Baseado em ações experimentais com a plataforma, a tarefa será apresentada aos alunos de forma que eles explorem a sequência de passos e ações para obter determinado resultado. Eles serão incentivados a realizarem novas atividades sem o auxílio do professor.</p> <p>[1]: https://studio.code.org/s/express-2017 [2]: https://studio.code.org/s/express-2017/stage/2/puzzle/2</p>
<p>Percurso Formativo 4 (5h) Atividades da plataforma Code.org.</p> <p>Curso Expresso: Lição 3 [3] Curso Expresso: Lição 4 Depuração [4] Exercícios: 1 a 10.</p> <p>Lição 7 Laços aninhados [5] Exercícios: 1 a 13.</p>	<p>Essa será a primeira experiência dos alunos com a repetição computacional de padrões na plataforma. Eles irão desenvolver familiarização inicial com comandos e funções dos parâmetros da programação em blocos. Serão motivados a conjecturar conexões entre os blocos de instruções construídos e a respectiva execução do personagem criando conexões entre representações. Os alunos deverão identificar erros e propor melhorias no código</p> <p>[3]: https://studio.code.org/s/express-2017/lessons/3/levels/1 [4]: https://studio.code.org/s/express-2017/lessons/4/levels/1 [5]: https://studio.code.org/s/express-2017/lessons/7/levels/1</p>
<p>Percurso Formativo 6 (5h) Criar eventos para a execução do personagem.</p> <p>Lição 14: Eventos com Flappy [6] Exercícios: 1, 2, 3, 4, 5, 6, 7, 9, 10.</p> <p>Lição 33: Construir um projeto [7] Exercício 2</p>	<p>Essa aula aborda conceitos relativos à criação de um jogo inserindo comandos em um contexto. Por exemplo, “Quando clicar”, “Reproduza”, “Quando passar um obstáculo”, “Quando atingir um obstáculo”, “Quando bater no chão...” são blocos de comandos que podem ser utilizados para dar ações aos personagens assimelhando-se a um jogo. Dessa forma, os alunos serão incentivados a criar.</p> <p>[6]: https://studio.code.org/s/express-2017/lessons/14/levels/1 [7]: https://studio.code.org/s/express-2017/lessons/33/levels/2</p>
<p>Percurso Formativo 8 (40 min.) Analisar padrões de combinação e solução de problemas.</p> <p>Avaliação [8]</p>	<p>No PF 8, o aluno deverá realizar uma avaliação com questões de múltipla escolha relacionadas ao conteúdo trabalhado durante as aulas anteriores, explorando desenvolvimento de padrões, algoritmos e lógica.</p> <p>[8]: Apêndice E.</p>

Os exercícios que compõem essas lições podem ser em formato de:

vídeos...



exercício de
múltipla escolha...



... ou programação
em blocos



(este último, representando a maioria dos exercícios da plataforma)

Cada um dos exercícios das lições da plataforma estão melhores descritas no **Apêndice F** como "Sequência de atividades Code detalhada"

REFERÊNCIAS

BRACKMANN, Christian Puhlmann. Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica. 2017. 226 f. Tese (Doutorado em Informática na Educação) - Universidade Federal do Rio Grande do Sul, Centro de Estudos Interdisciplinares em Novas Tecnologias na Educação, Programa de Pós-graduação em Informática na Educação, Porto Alegre, BRRS, 2017.

BRASIL. Ministério da Educação. Base Nacional Comum Curricular. Brasília, 2018.

CIEB. Currículo de tecnologia e computação. 2020. Disponível em: <<https://curriculo.cieb.net.br/>>. Acesso em 12 abr. 2020.

CODE. 2021. Disponível em: <https://studio.code.org>. Acesso em: 01 fev. 2021.

FREIRE, Gabriel Gonçalves; GUERRINI, Daniel; DUTRA, Alessandra. 2016. O Mestrado Profissional em Ensino e os Produtos Educacionais: A Pesquisa na Formação Docente. Porto das Letras. 2, 1 (dez. 2016), p. 100 - 114.

KAPLÚN, Gabriel. Material educativo: a experiência de aprendizado. Comunicação & educação. n. 27, p. 40 a 60, mai. a ago. 2003. Disponível em: <https://doi.org/10.11606/issn.2316-9125.v0i27p46-60>. Acesso em: 22 set. 2019.

LIBÂNEO, José Carlos. Pedagogia e pedagogos, para quê? São Paulo: Cortez, 1998.

MORAN, José Manuel , MASETTO, Marcos. T; BEHRENS, Marilda Aparecida. . Novas Tecnologias e Mediação Pedagógica. Ed. Papirus, 2000.

VALENTE. Pensamento Computacional, Letramento Computacional ou Competência Digital? Novos desafios da educação. Revista Educação e Cultura Contemporânea, v. 16, n. 43, p. 147-168, 2019.

WING, Jeannette Marie. Computational Thinking Benefits Society. Social Issues in Computing, 2014. Disponível em: <<http://socialissues.cs.toronto.edu/2014/01/computational-thinking/>>. Acesso em: 01 out. 2020.

ZABALA, Antoni. A prática educativa: como ensinar; tradução Ernani F. da F. Rosa. Porto Alegre: Artmed. 224 p. 2010.

APÊNDICES

APÊNDICE A – CONHECIMENTOS PRÉVIOS

As perguntas a seguir são para melhor entendermos os seus conhecimentos sobre Pensamento Computacional e Lógica.

*Os campos com * são de preenchimento obrigatório

1. Por favor, digite seu nome completo: * _____
2. Turma: _____
3. Você já tinha ouvido falar no termo “pensamento computacional”? *
Sim
Não
4. O que você sabe sobre lógica de programação? *
Não sei nada.
Sei teoria.
Sei teoria e já pratiquei.
5. Você acha que o pensamento computacional tem alguma relação com a resolução de problemas? *
Sim. Tudo a ver.
Não. Nada a ver.
Não sei.
6. Você acha que o pensamento computacional tem alguma relação com o desenvolvimento da lógica? *
Sim
Não
7. Você já utilizou algum programa, aplicativo ou site para aprender lógica? *
Sim
Não
8. Já conhece a plataforma Code.org? *
Sim. Já ouvi falar.
Sim. Já utilizei.
Não conheço.
9. Conhece a linguagem Scratch? *
Sim. Já ouvi falar.
Sim. Já utilizei.
Não conheço.
10. O que você sabe sobre programação? *
Não sei nada.
Sei pouco.
Sei muito.

11. Baseando-se na imagem a seguir, marque o que está ERRADO:

- Passo A
- Passo B
- Passo C
- Passo D
- Não sei responder

Para levar o "Pac-Man" até o fantasma pelo caminho indicado, qual passo da sequência está incorreto?

avance → Passo A
 vire à esquerda U ▾ → Passo B
 avance → Passo C
 vire à esquerda U ▾ → Passo D
 avance

12. Baseando-se na imagem a seguir, marque a alternativa CORRETA:

- Alternativa A
- Alternativa B
- Alternativa C
- Alternativa D
- Não sei responder

Qual sequência leva o "Pac-Man" até o fantasma pelo caminho indicado?

Alternativa A: × 5 (Right arrow)

Alternativa B: × 3 (Right arrow)

Alternativa C: × 4 (Right arrow)

Alternativa D: × 2 (Right arrow)

13. Baseando-se na imagem a seguir, marque a alternativa do passo ERRADO:

- Passo A
- Passo B
- Passo C
- Passo D
- Não sei responder.

Para que o "Pac-Man" chegue até o fantasma pelo caminho indicado, qual passo da sequência está incorreto?

repetir até [ghost icon]

faça

- se houver caminho à frente ▾
 - faça avance → Passo A
- se não Se houver caminho à direita U ▾
 - faça vire à esquerda U ▾ → Passo C
- se não vire à direita U ▾ → Passo D

APÊNDICE B - CONTA DO PROFESSOR NA PLATAFORMA CODE.ORG

Tutorial Code.org para o professor

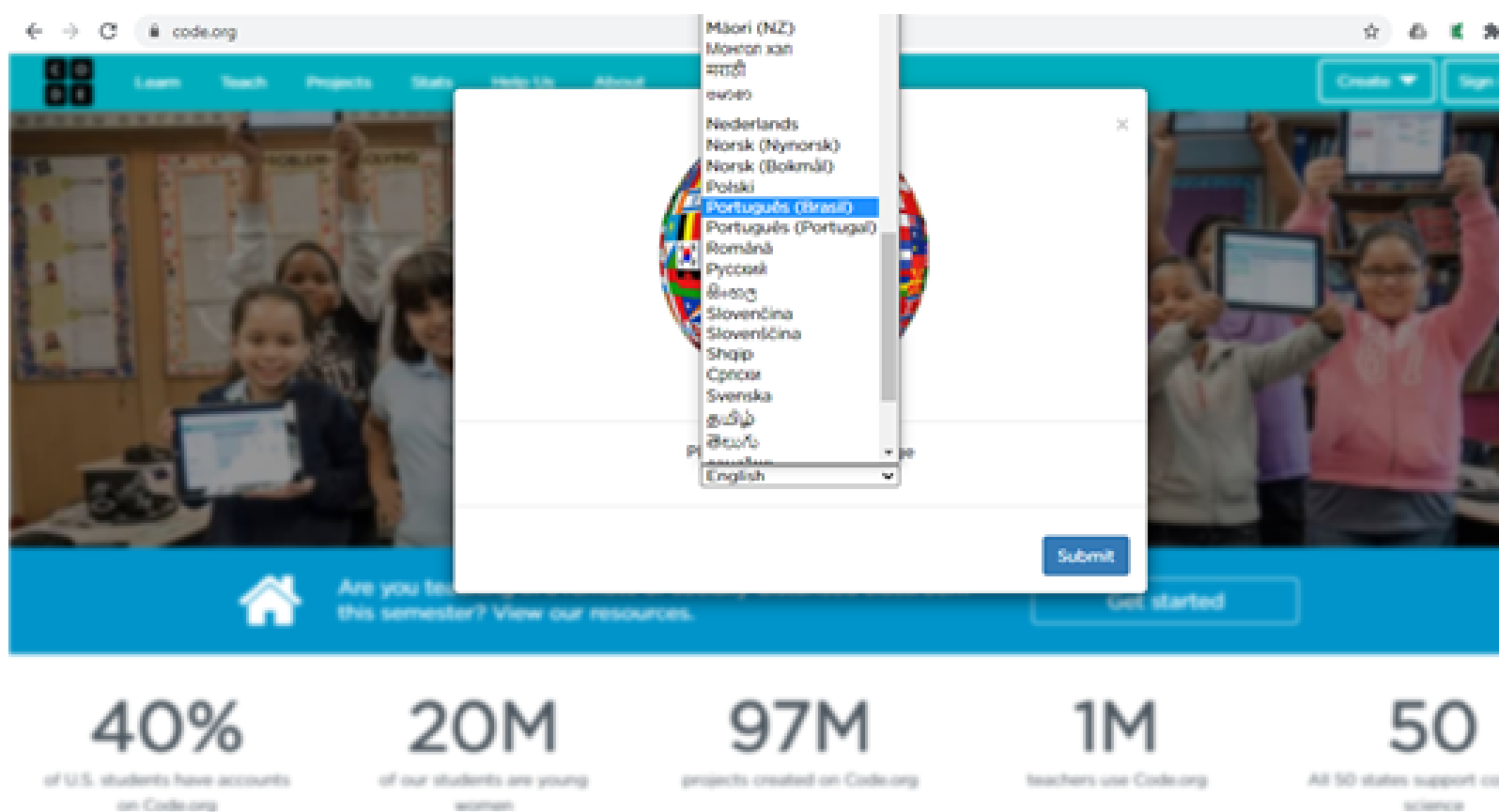
Olá professor! Este tutorial traz o passo a passo para a utilização da plataforma Code.org com os seus alunos. Esperamos que goste!

O code.org é uma plataforma de cursos de Lógica Computacional baseada na **resolução de problemas**. Traz as seguintes vantagens para os usuários:

- É gratuito. Você só precisa de um e-mail válido para utilizar.
- Você pode criar uma turma e disponibilizar o curso da plataforma em sua turma.
- Seus alunos não precisam ter um e-mail porque você poderá disponibilizar para eles um código para que entrem na sua turma.
- Você poderá acompanhar o progresso de seus alunos.
- Você poderá visualizar as tentativas dos alunos, bem como seus erros e acertos durante o desenvolvimento das respostas das atividades propostas no seu curso.
- A estrutura das atividades funciona com programação em blocos e pode ser exemplificado com programação em código que é gerada automaticamente.
- Você pode realizar atividades diversas que são disponibilizadas na plataforma.

Nesse tutorial, vamos criar um usuário e uma turma na plataforma Code.org.

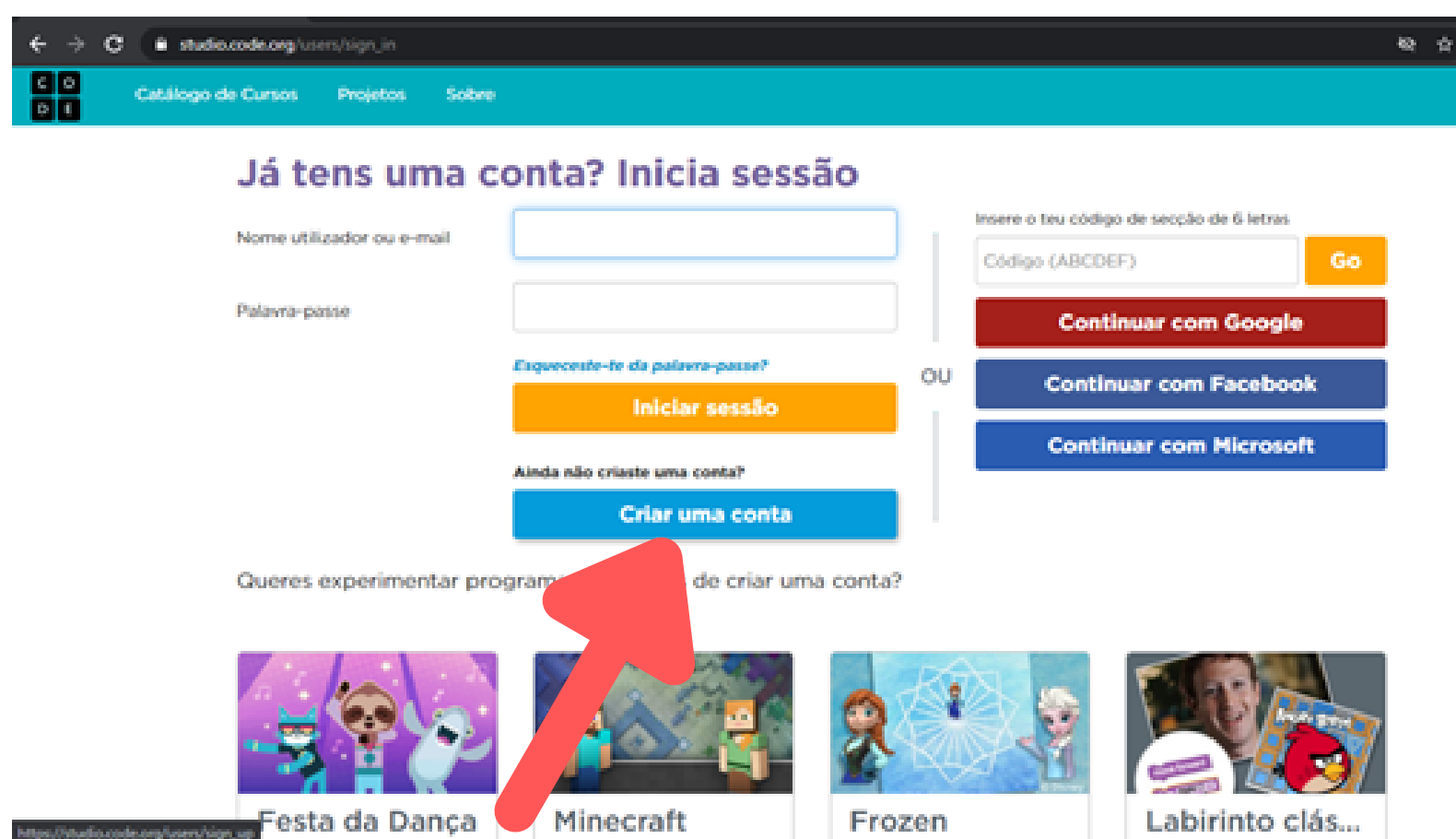
1 O primeiro passo é acessar o link: <https://code.org/> em seguida escolher o idioma Português.



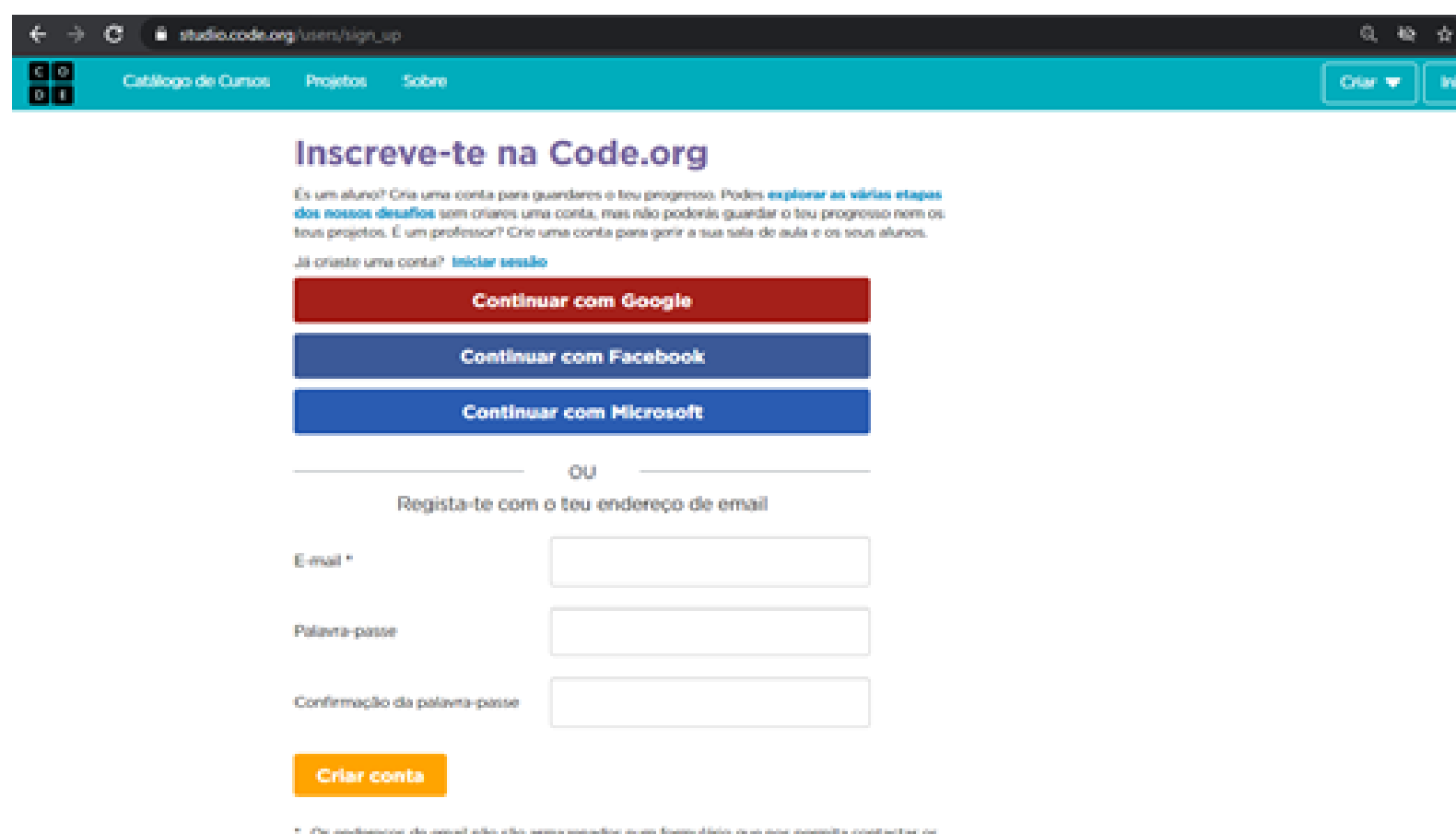
2 Em seguida, seleccione o botão “Entrar” (canto superior direito).



3 E crie um login digitando um e-mail válido e uma senha através da opção “Criar uma conta”.



4 Você poderá entrar com alguma conta de e-mail já existente, ou registrar seu endereço de e-mail criando uma senha e confirmando-a.



CRIANDO UMA TURMA

Após criar a conta, e entrar na plataforma, você poderá criar uma “Sala de aula” utilizando o menu “Painel de controle”.



Em “Seções de Sala de aula”, clique na opção “Criar seção”.



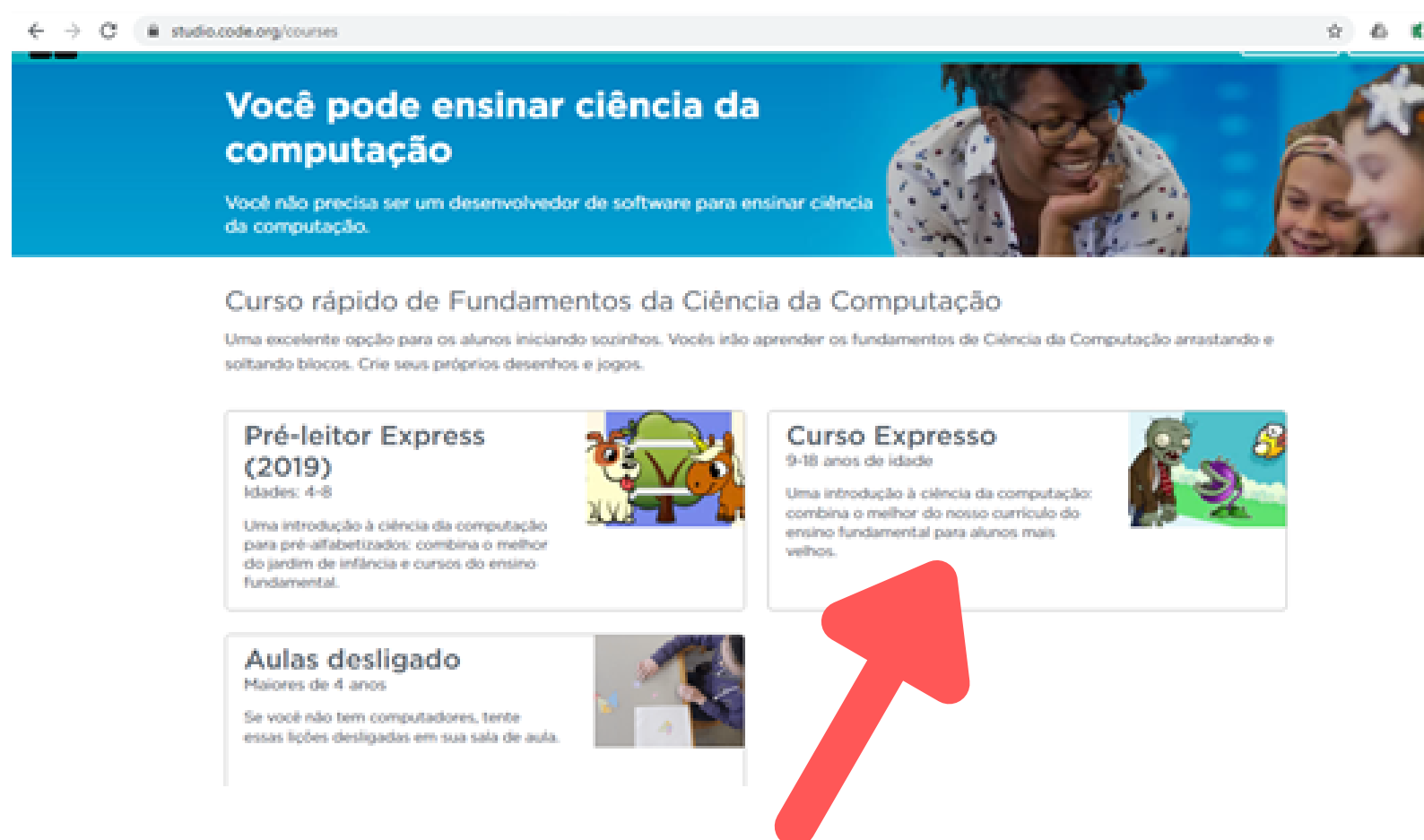
Você poderá escolher como serão feitos os logins dos alunos de acordo com a idade da turma. Assim, os seus logins poderão ser realizados pelo Google Classroom, e-mails, nicknames ou imagens.



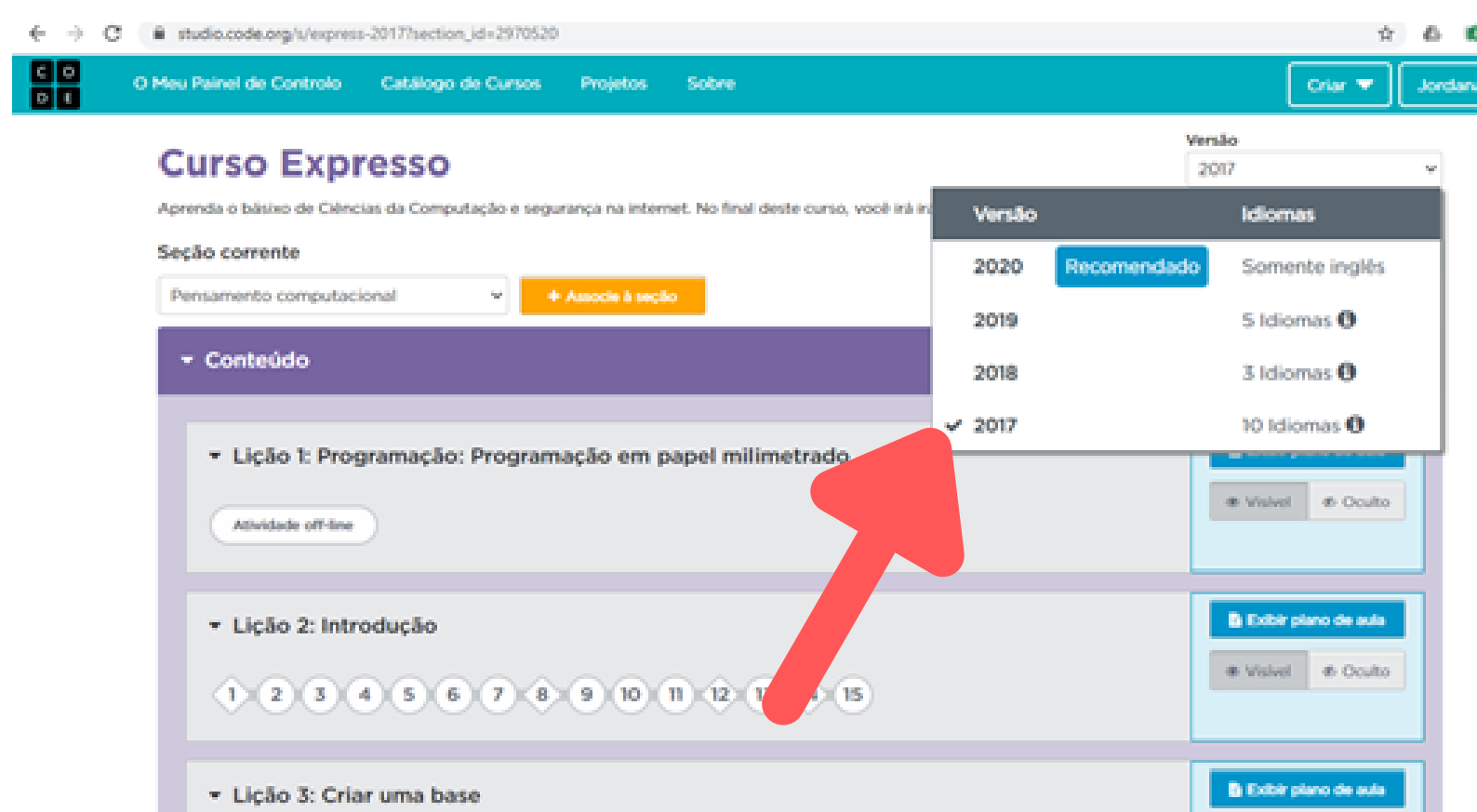
ENCONTRE UM CURSO

Você pode encontrar entre os vários cursos oferecidos pela plataforma aquele que melhor se adapta ao perfil dos seus alunos.

No nosso caso, optamos por fazer o curso expresso para alunos de 9 a 18 anos de idade, disponível no link: https://studio.code.org/s/express-2017?section_id=2970520



A versão do Code.org utilizada na Sequência Didática é a Versão 2017.

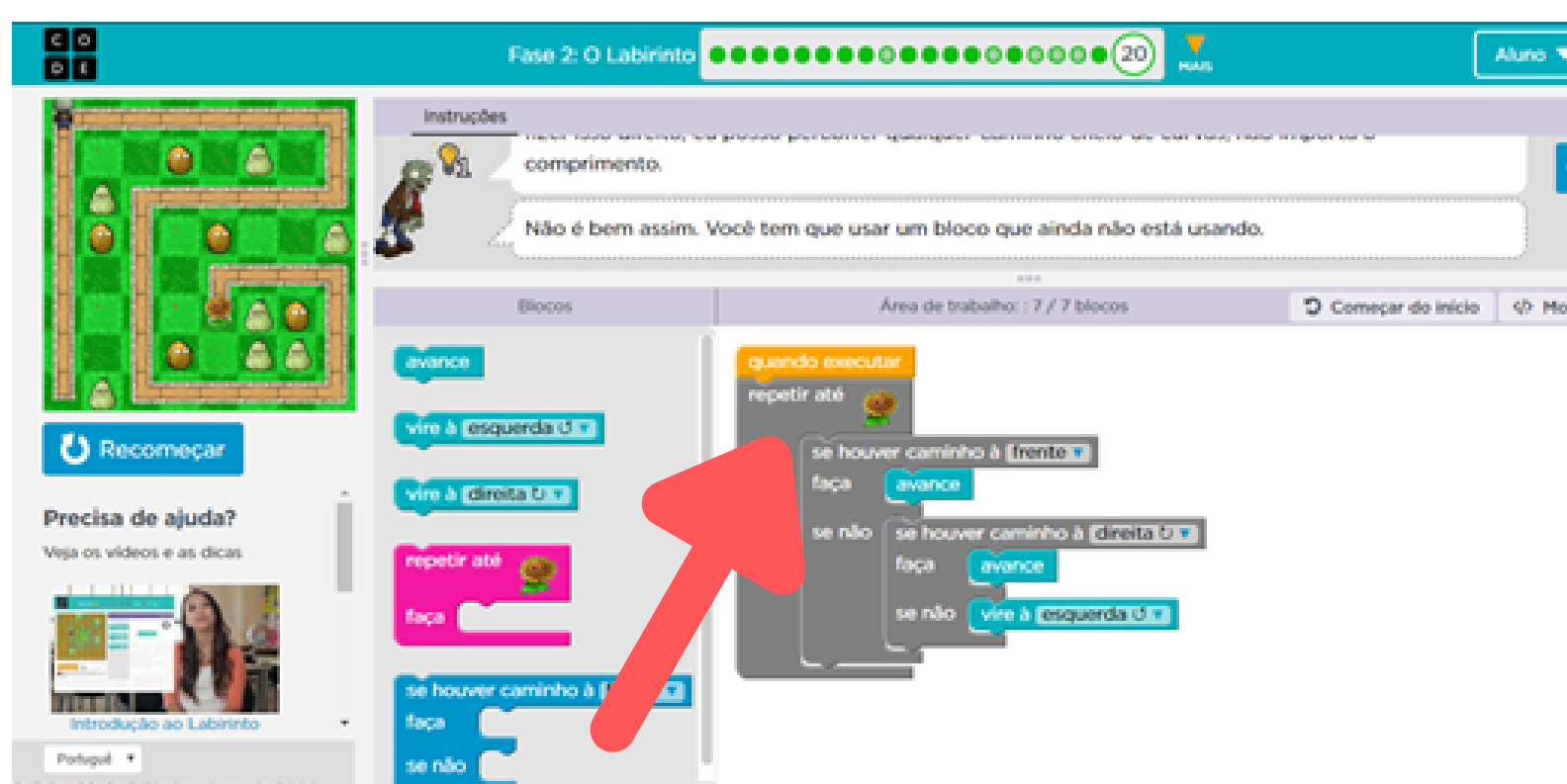


Você poderá escolher quais atividades do curso estarão disponíveis para os alunos, uma vez que algumas atividades poderão ser ocultadas.

Mas como funciona?

As atividades são realizadas por meio de estruturas de blocos (são bem parecidas com Scratch). Dessa forma, o aluno deverá "arrastar os blocos de comandos" de forma a montar uma estrutura que resolva o problema proposto da melhor forma possível.

No exemplo a seguir, o zumbi deve percorrer o caminho até o final do trajeto.



ENCONTRE UM CURSO

Durante a realização das atividades o aluno terá os seguintes *feedbacks* da plataforma: conseguirá saber se realizou a tarefa com precisão ou se poderia ter feito o bloco de forma mais otimizada.

Para você, professor, os benefícios são maiores ainda uma vez que traz alguns filtros de resultados por turma (apresentando uma visão geral), por aluno (visão geral do que ele realizou) ou por atividade (possibilitando analisar as suas respostas).

Assim, com o filtro por aluno, é possível acompanhar o nível de progresso do aluno incluindo as atividades que cada aluno realizou ou se ainda não o fez.

O PENSAMENTO COMPUTACIONAL
Atribuído a: Introdução acelerada para o curso de CC

Progresso Respostas de texto Avaliações/Pesquisas Projetos Estatísticas Gerenciar alunos

Selecionar um curso ou unidade: Introdução acelerada para o curso de CC
Ver Por: Lições Níveis

Lições tentadas em Introdução acelerada para o curso de CC

Lição	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Aluno 1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Status da Lição: Não iniciado, Em andamento, Concluído (perfeito), Concluído (muitos blocos)

Ao selecionar o filtro por atividade, você poderá observar como cada aluno se saiu em cada questão.

O PENSAMENTO COMPUTACIONAL
Atribuído a: Introdução acelerada para o curso de CC

Progresso Respostas de texto Avaliações/Pesquisas Projetos Estatísticas Gerenciar alunos

Selecionar um curso ou unidade: Introdução acelerada para o curso de CC
Ver Por: Lições Níveis
Pular para a lição: 2: O Labirinto

Níveis tentados em Introdução acelerada para o curso de CC

Lição	2: O Labirinto																				3
Tipo de nível	[Ícone]																				[Ícone]
Aluno 1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	A

Tipo de nível Detalhes do nível Status do nível

Além disso, ao selecionar determinada questão de um aluno específico, você terá acesso ao código construído pelo aluno, tendo em mãos um material rico para análise do desenvolvimento e da estrutura lógica utilizada.

Além disso, é fornecido a data e o horário de realização para cada atividade realizada pelo aluno. Podendo ser aliada no controle de tarefas e avaliação

Fase 2: O Labirinto

Instruções: Quando um bloco estiver cinza, significa que você não pode excluí-lo. Resolva esse desafio usando que faz 3 repetições. Tente colocar esses 3 blocos dentro do bloco "repita" em cinza: avance, avan...

Apenas visualizar: 6 / 5 blocos

quando executar
repita 3 vezes
faça
repita 3 vezes
faça
avance
avance
vire à direita 90°

Painel do professor
Ver página como: Estudante Professor
Aluno 1
Última Atualização: 13/05/2020 13:20:15
Solução do Exemplo 1
Ver seção: O PENSAMENTO CO
Visualizar o Painel de Controle do Professor
Eu
Aluno 1

Tudo pronto para começar? Então vamos lá!

Tutorial Code.org para o aluno

Olá, você está preparado para começar a programar utilizando blocos? Este é o tutorial para o primeiro acesso ao nosso curso. Vamos lá?

A plataforma: Code.org é uma plataforma de cursos de Lógica Computacional que iremos utilizar.

Funciona assim: você pode entrar no site, criar seu usuário e seguir com atividades diversas que são disponibilizadas.

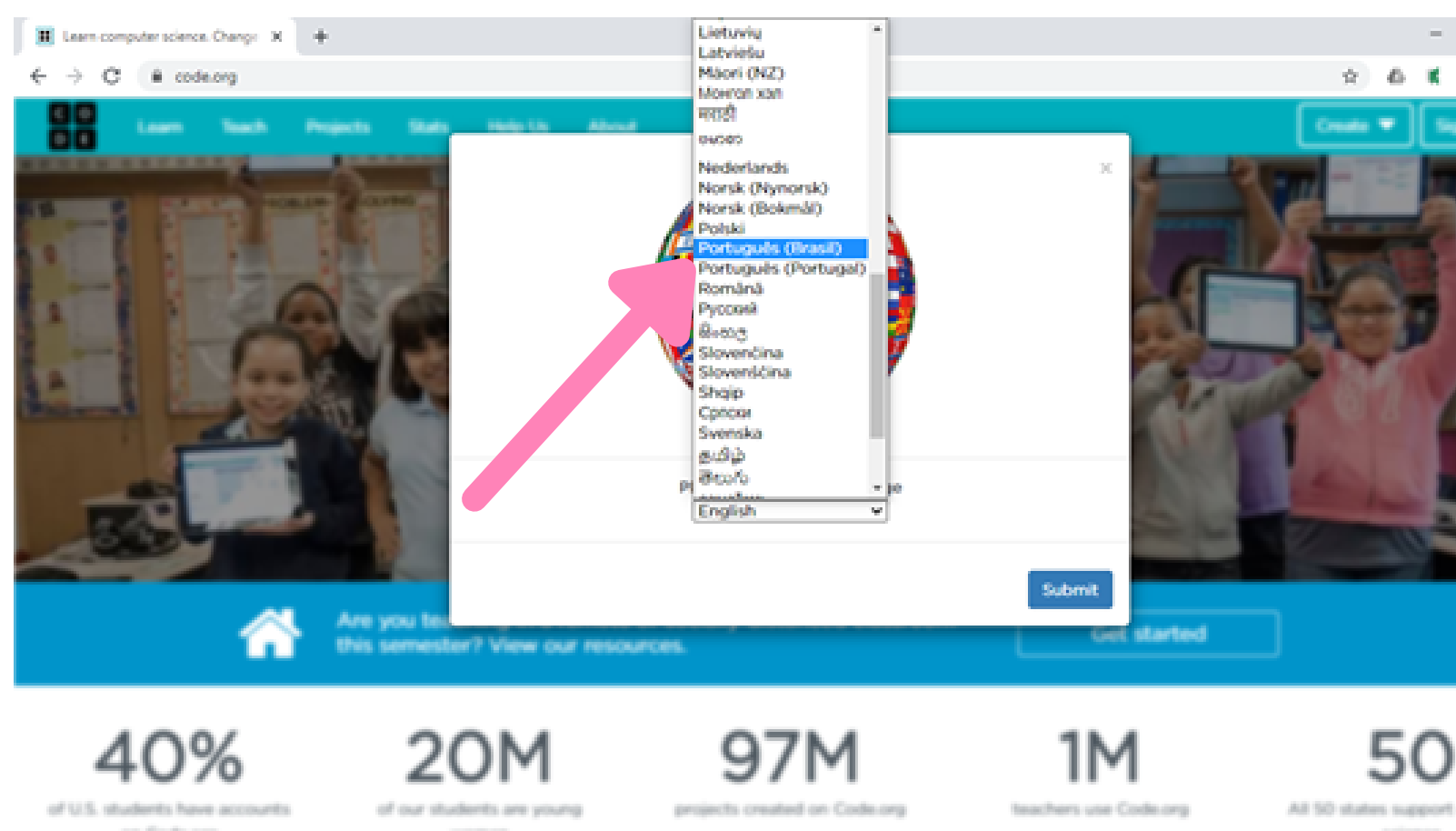
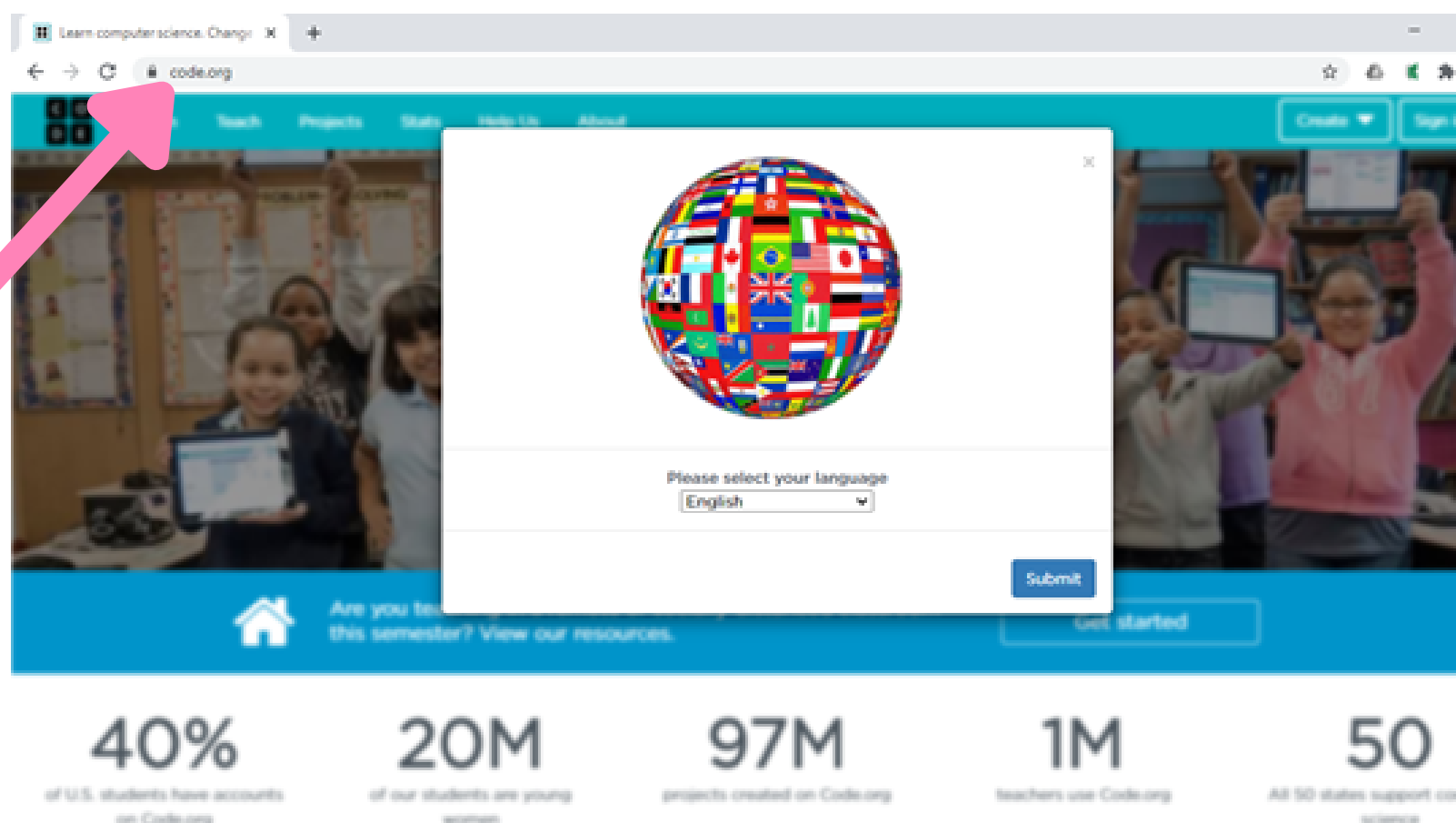
Nós vamos te mostrar o passo a passo de como você deve fazer, tá ok? Você deve entrar por meio de um código da “turma” criada pelo seu professor. Nessa turma, que você vai participar, as atividades serão as mesmas para você e para seus colegas.

Nas atividades que faremos utilizando essa plataforma você deverá encontrar soluções e atingir um objetivo final utilizando programação por blocos.

Vai ser divertido e muito produtivo. **Vamos começar?**

Entrando na plataforma do curso

1- O primeiro passo é acessar o link: <https://code.org/>. e escolha o idioma Português .

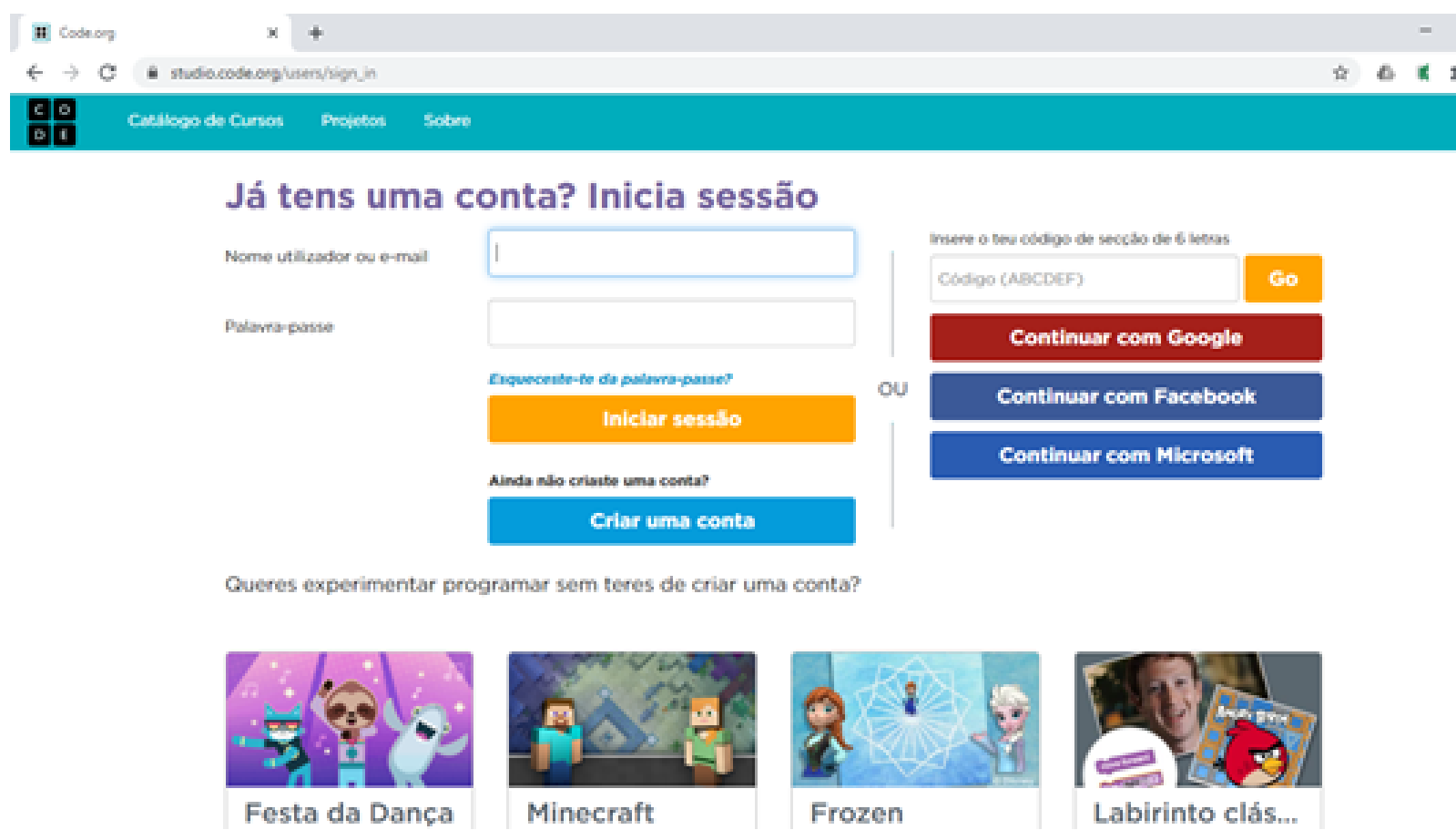


CODE PARA O ALUNO

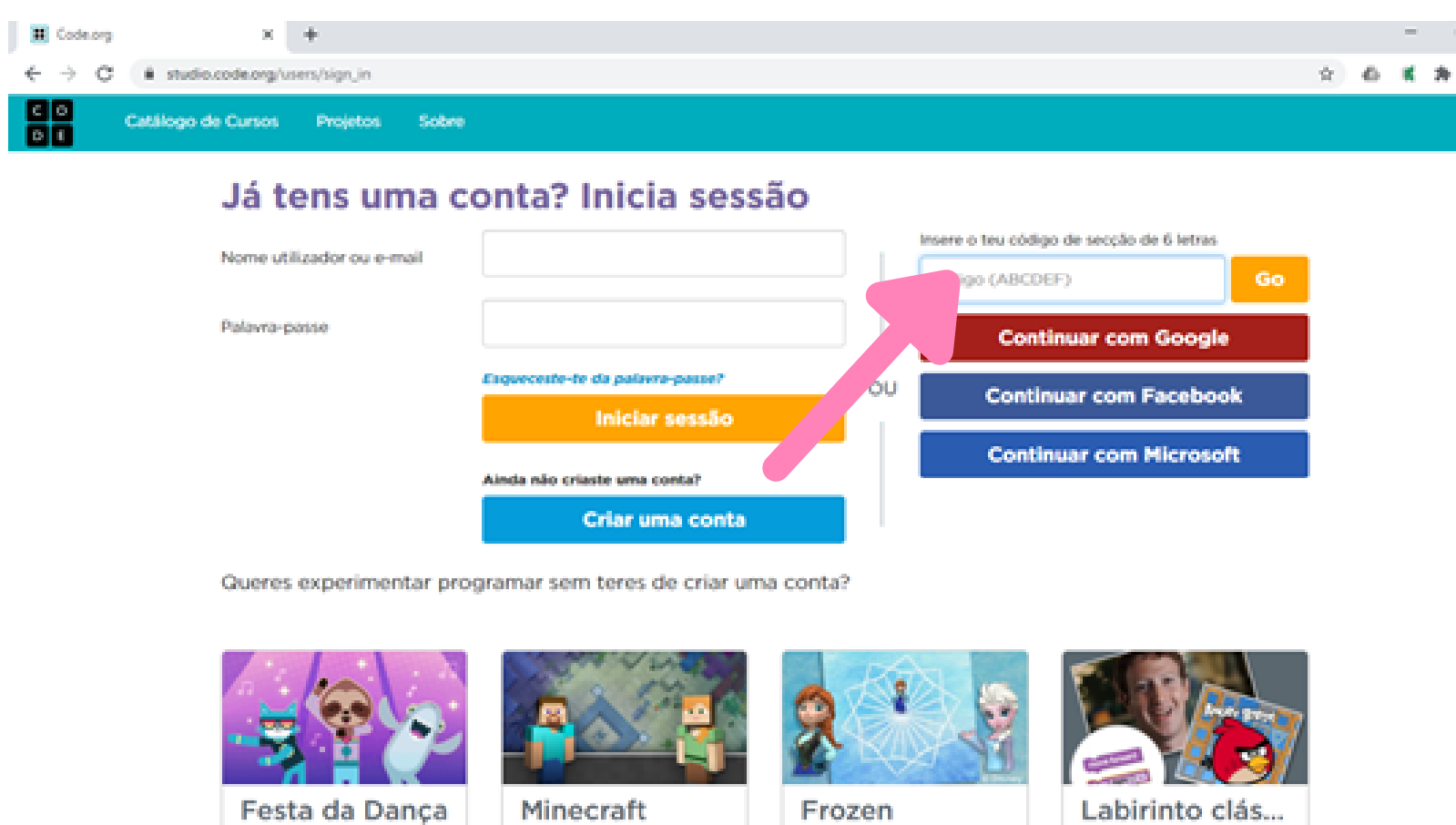
Clique na opção "Entrar" (no canto superior direito da tela)



-Digite seu nome de utilizador e a sua palavra-passe (senha). E clique em "Iniciar Sessão".



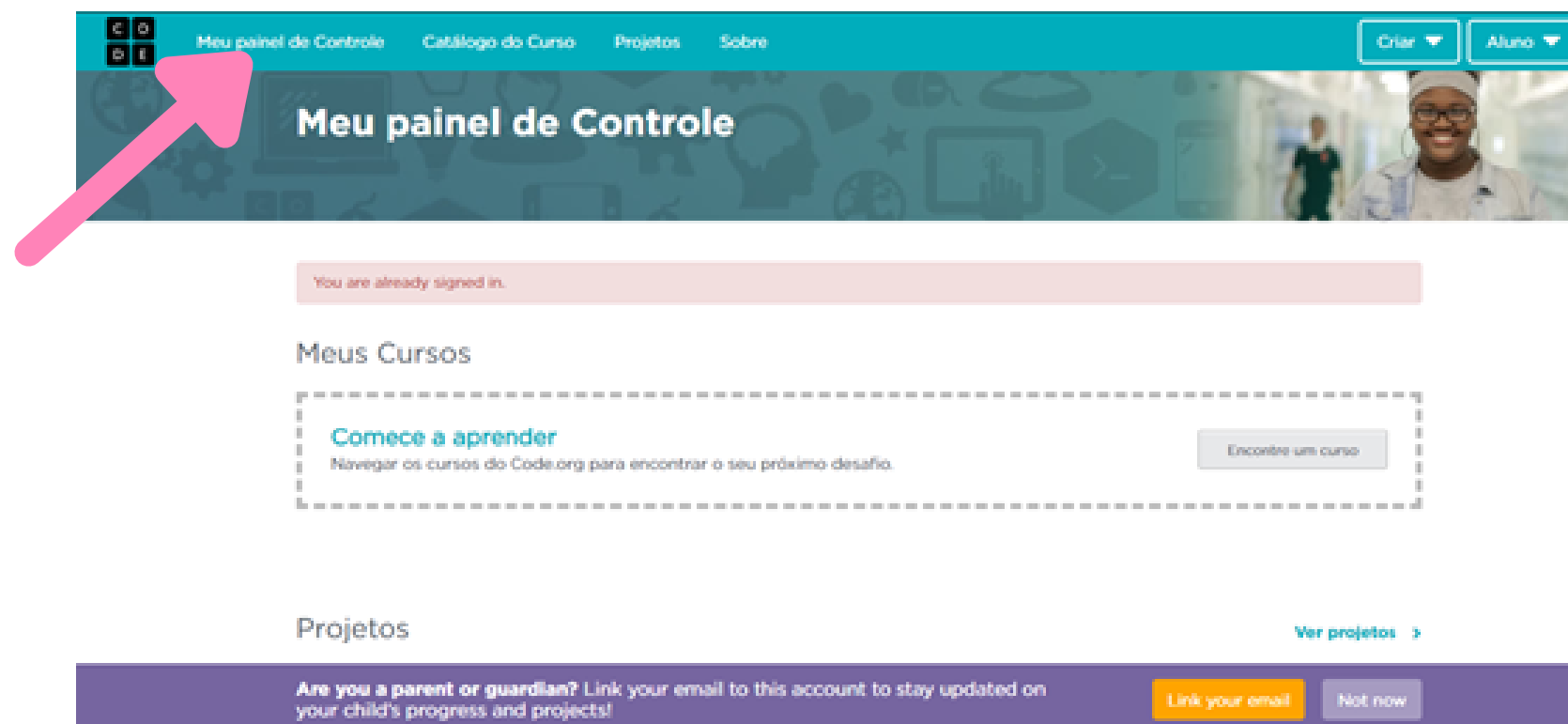
Ou... insira o código da seção (esse código foi informado pelo seu professor no início do curso)



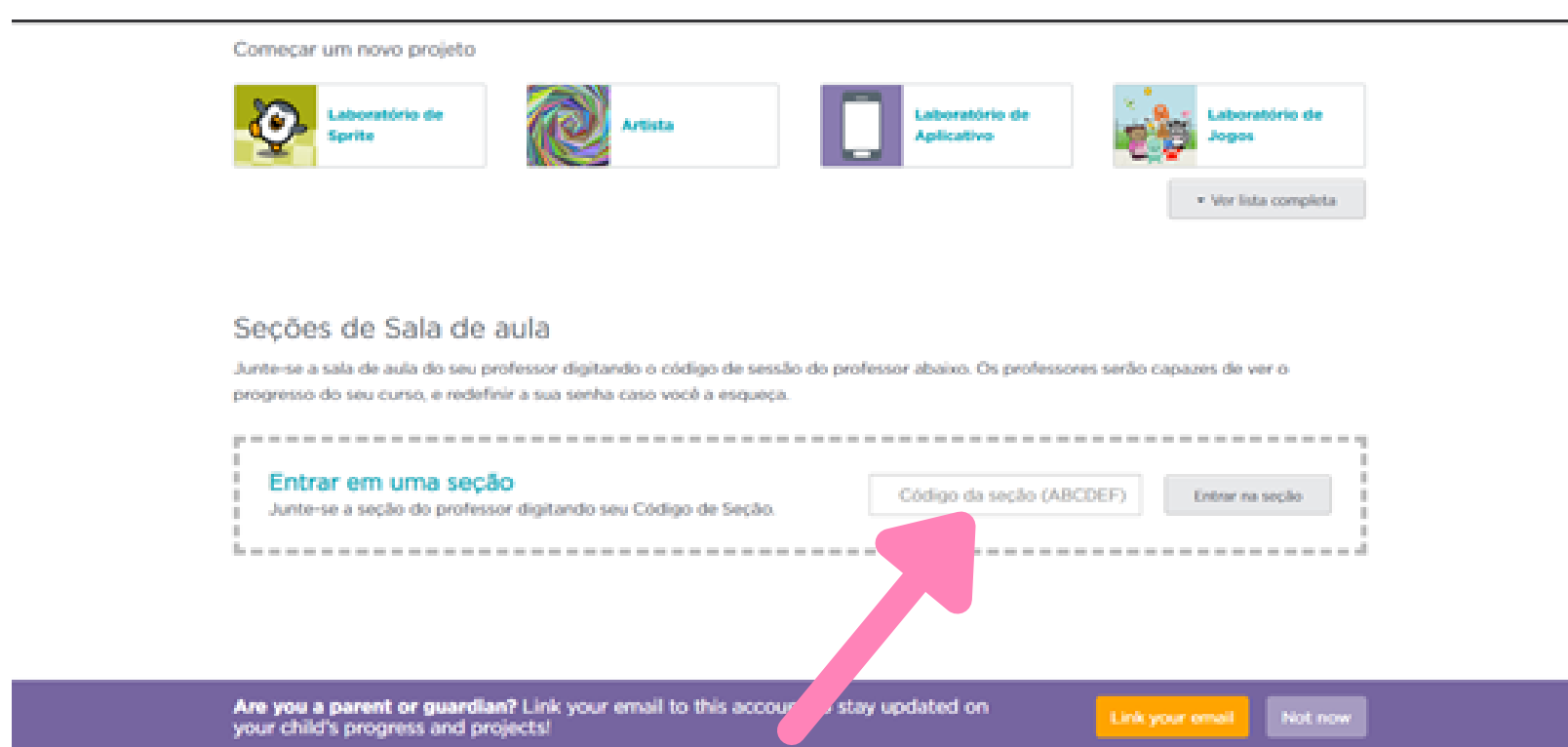
Muito bem! Agora que você digitou o código da turma você já pode começar um curso!

COMEÇANDO O CURSO

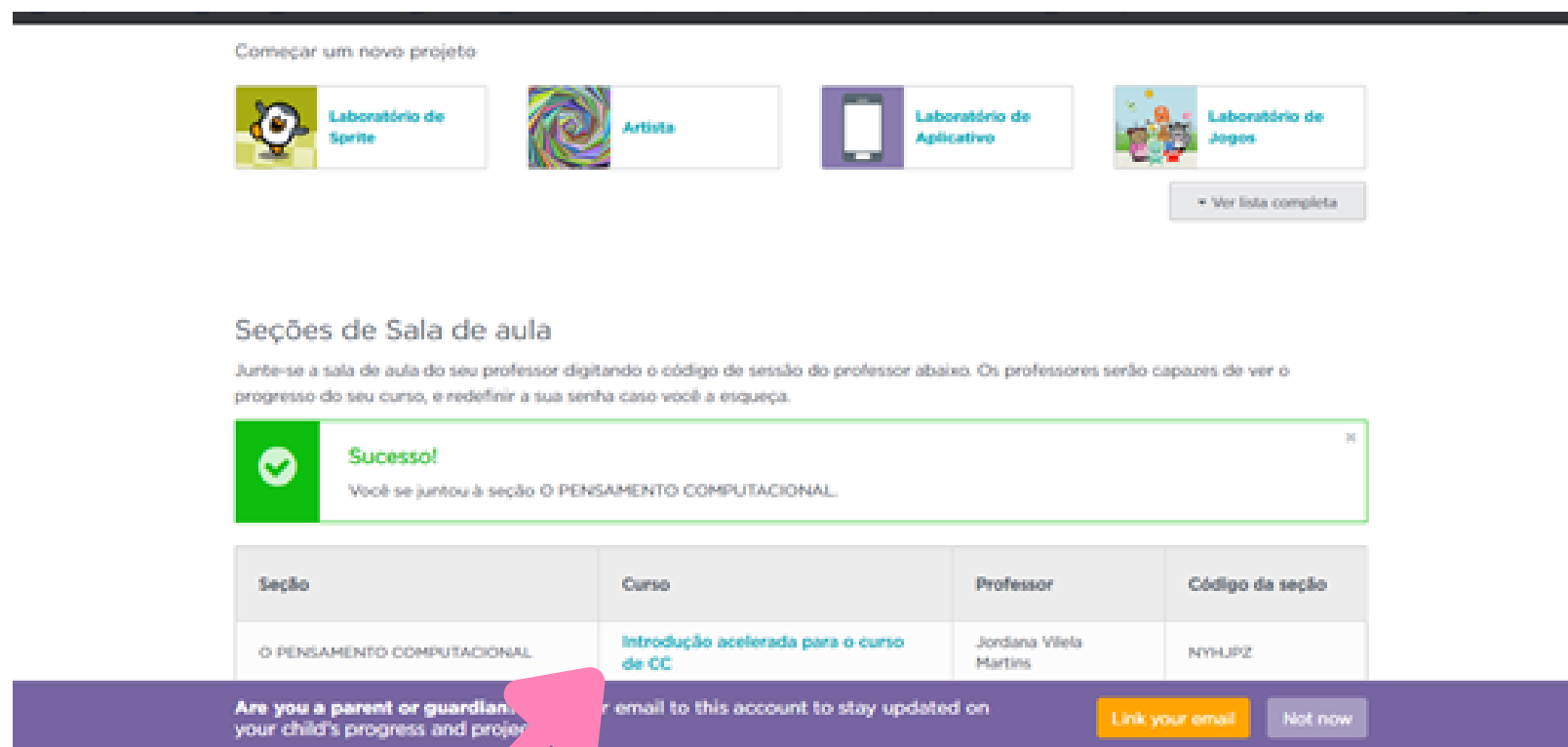
Para iniciar o seu curso basta clicar na opção “Meu painel de controle”.



No final da página tem um campo chamado “Seções de Sala de aula”, localize a opção “Entrar em uma seção” e no campo “Código da seção (ABCDEF)” digite o código da sua turma que o seu professor comunicou a você.



Depois disso, irão aparecer os cursos que você está fazendo (incluindo a turma que ele acabou de entrar).



E agora? Como funciona isso?

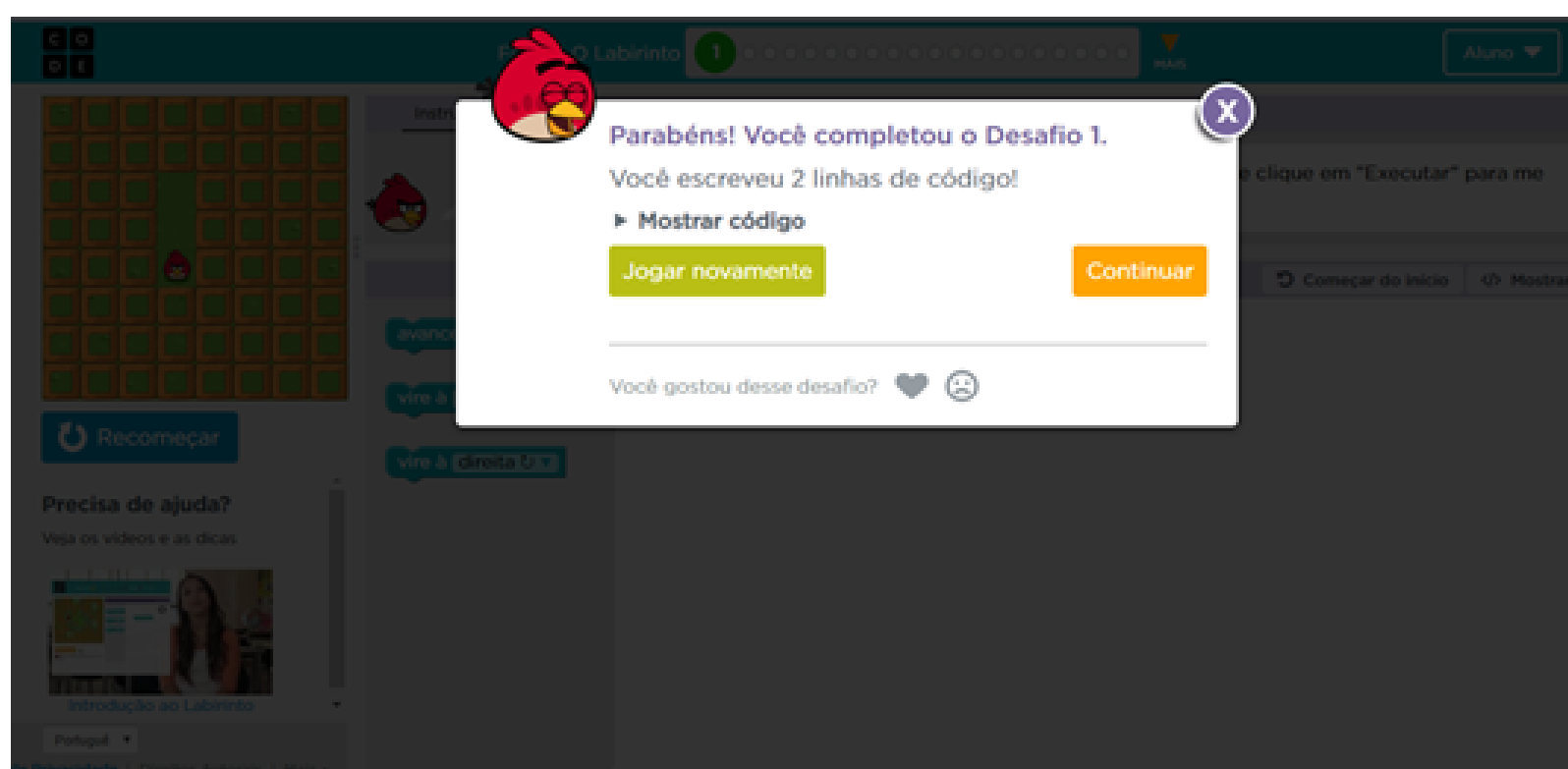
COMO FUNCIONA

Ao selecionar a turma que você acabou de entrar, são listadas as atividades a serem feitas.



Em cada atividade há dicas de como devem ser realizadas, além disso, na mesma hora você já pode ver se a solução que você propôs é aceitável.

Assim, se você fizer a resolução dentro dos padrões esperados, você receberá uma mensagem de parabéns.



Não se esqueça que a aprendizagem é construída passo a passo e que você também aprende com os seus erros.

Está pronto para começar?

Então, bons estudos!!!

APÊNDICE D - QUESTIONÁRIO COM QUESTÕES SOBRE LÓGICA COMPUTACIONAL

Olá! Essa avaliação é composta por 10 questões de múltipla escolha. Todas as questões são de preenchimento obrigatório (com 4 alternativas de resposta). As questões marcadas com * são de preenchimento obrigatório, entretanto, se você não souber a resposta poderá selecionar a última opção ("Não sei a resposta!").

Em todas as questões, o objetivo é fazer o personagem Pac Man (amarelo) chegar ao Blinky (fantasma vermelho). Vamos começar?

1. Endereço de e-mail: *
2. Seu nome completo: *

Pergunta 1 - Marque a alternativa correta: *

- Alternativa A
- Alternativa B
- Alternativa C
- Alternativa D
- Não sei responder.

<p>Qual sequência leva o "Pac-Man" até o fantasma pelo caminho indicado?</p>	<p>Alternativa A</p> <p>→ → ↓</p> <hr/> <p>Alternativa B</p> <p>→ ↓ ↓</p> <hr/> <p>Alternativa C</p> <p>→ → ↓ ↓</p> <hr/> <p>Alternativa D</p> <p>↓ ↓ →</p>
--	---

Pergunta 2 - Marque a alternativa correta: *

- Alternativa A
- Alternativa B
- Alternativa C
- Alternativa D
- Não sei responder.

<p>Qual comando está faltando na sequência para levar o "Pac-Man" até o fantasma pelo caminho indicado?</p> <p>← ← ↑ ? → → →</p>	<p>Alternativa A</p> <p>→</p> <hr/> <p>Alternativa B</p> <p>←</p> <hr/> <p>Alternativa C</p> <p>↑</p> <hr/> <p>Alternativa D</p> <p>↓</p>
--	---

Pergunta 3 - Marque o passo que está ERRADO: *

- Passo A
- Passo B
- Passo C
- Passo D
- Não sei responder.

Para levar o "Pac-Man" até o fantasma pelo caminho indicado, qual passo da sequência está incorreto?

avance → Passo A
 vire à esquerda U ▾ → Passo B
 avance → Passo C
 vire à esquerda U ▾ → Passo D
 avance

Pergunta 4 - Marque a alternativa correta: *

- Alternativa A
- Alternativa B
- Alternativa C
- Alternativa D
- Não sei responder.

Qual sequência leva o "Pac-Man" até o fantasma pelo caminho indicado?

Alternativa A: x 5 (Right arrow)
 Alternativa B: x 3 (Right arrow)
 Alternativa C: x 4 (Right arrow)
 Alternativa D: x 2 (Right arrow)

Pergunta 5 - Marque a alternativa correta: *

- Alternativa A
- Alternativa B
- Alternativa C
- Alternativa D
- Não sei responder.

Qual sequência leva o "Pac-Man" até o fantasma pelo caminho indicado?

Alternativa A: repita 4 vezes, faça repita 3 vezes, faça avance, vire à direita U ▾, avance
 Alternativa B: repita 3 vezes, faça repita 4 vezes, faça avance, vire à direita U ▾, avance
 Alternativa C: repita 3 vezes, faça repita 3 vezes, faça avance, vire à direita U ▾, avance
 Alternativa D: repita 4 vezes, faça avance, repita 3 vezes, faça vire à direita U ▾, avance

Pergunta 6 - Marque a alternativa correta: *

- Alternativa A
- Alternativa B
- Alternativa C
- Alternativa D
- Não sei responder.

Qual sequência leva o "Pac-Man" até o fantasma pelo caminho indicado?

Alternativa A: Repete até chegar ao (Right, Down arrows)
 Alternativa B: Repete até chegar ao (Right, Up arrows)
 Alternativa C: Repete até chegar ao (Down, Left arrows)
 Alternativa D: Repete até chegar ao (Down, Right arrows)

Pergunta 7 - Marque a alternativa correta: *

- Alternativa A
- Alternativa B
- Alternativa C
- Alternativa D
- Não sei responder.

<p>Qual sequência leva o "Pac-Man" até o fantasma pelo caminho indicado?</p>	<p>Alternativa A</p>	<p>Alternativa B</p>
	<p>Alternativa C</p>	<p>Alternativa D</p>

Pergunta 8 - Marque a alternativa correta: *

- Alternativa A
- Alternativa B
- Alternativa C
- Alternativa D
- Não sei responder.

<p>O que falta na seguinte sequência para levar o "Pac-Man" até o fantasma pelo caminho indicado?</p>	<p>Alternativa A</p>
	<p>Alternativa B</p>
	<p>Alternativa C</p>
	<p>Alternativa D</p> <p>Tanto a alternativa A como a alternativa C estão corretas</p>

Pergunta 9 - Marque o passo que está ERRADO: *

- Passo A
- Passo B
- Passo C
- Passo D
- Não sei responder.

<p>Para que o "Pac-Man" chegue até o fantasma pelo caminho indicado, qual passo da sequência está incorreto?</p>	
--	--

Pergunta 10 - Marque a alternativa correta: *

- Alternativa A
- Alternativa B
- Alternativa C
- Alternativa D
- Não sei responder.

<p>Qual sequência leva o "Pac-Man" pelo caminho indicado até os morangos e faz o "Pac-Man" comer o número de morangos indicado?</p>	<p>Alternativa A</p>	<p>Alternativa B</p>
	<p>Alternativa C</p>	<p>Alternativa D</p>



APÊNDICE E – AUTO AVALIAÇÃO E A AVALIAÇÃO DA ATIVIDADE

As perguntas a seguir são relativas à sua experiência com relação ao uso da plataforma Code.org e sobre a sua percepção de aprendizagem. Queremos saber a sua opinião sobre as atividades desse módulo. São cinco perguntinhas!

Pergunta 1- De 0 a 10 (sendo 0 péssimo e 10 excelente), como você acha que tenha se saído no teste? *

1 2 3 4 5 6 7 8 9 10

Pergunta 2- Você considera que aprendeu com as atividades deste módulo (sendo 1 aprendi pouco e 10 aprendi muito):

1 2 3 4 5 6 7 8 9 10

Pergunta 3- Você consegue ver pontos positivos com relação à realização das atividades deste módulo? Poderia dizer quais?

Pergunta 4- Você consegue ver pontos negativos com a realização das atividades deste módulo? Poderia dizer quais?

Pergunta 5- Qual o seu nível de satisfação em relação à: *

Plataforma Code.org Não fiquei satisfeito Fiquei satisfeito! Fiquei muito satisfeito

Programação em blocos

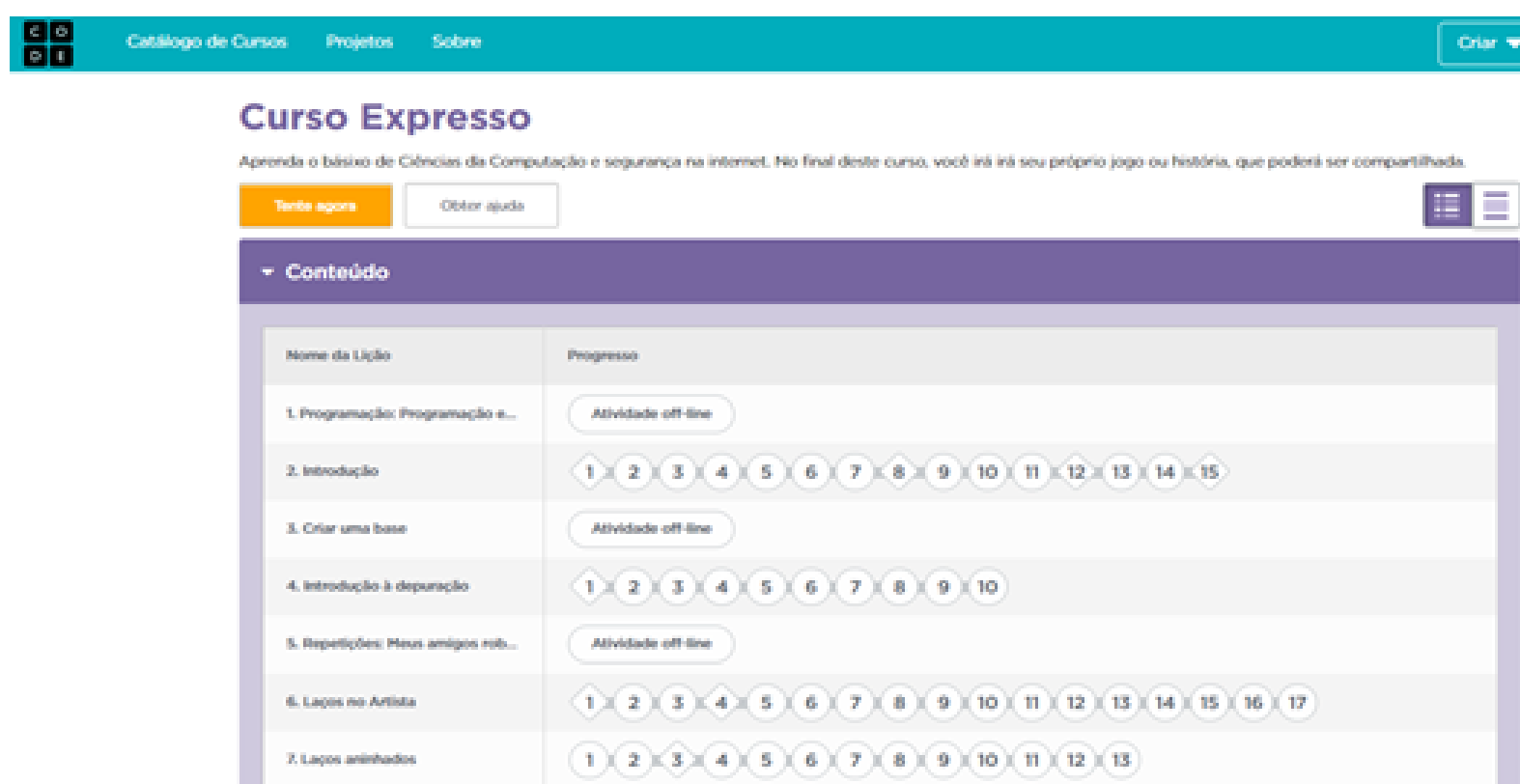
Atividades de lógica



ATIVIDADES - CODE.ORG

Descritivo das atividades a serem realizadas na Sequência Didática para Desenvolver Lógica Computacional Utilizando a Plataforma Code.org

Na plataforma Code.org, realizaremos algumas lições do Curso Expresso, versão 2017 que está disponível no seguinte endereço: <https://studio.code.org/s/express-2017>



As lições realizadas são as seguintes:

- **Lição 2 Introdução Exercícios 1 ao 10**
- **Lição 4 Depuração - Exercícios: 1 a 10.**
- **Lição 7 Laços aninhados - Exercícios: 1 a 13.**
- **Lição 14: Eventos com Flappy - Exercícios: 1, 2, 3, 4, 5, 6, 7, 9, 10.**
- **Lição 33: Construir um projeto**

E serão melhor descritas a seguir.

PERCURSO FORMATIVO 2

As lições realizadas no Percurso Formativo 2 serão:

- Lição 2: Introdução ao quebra-cabeça on-line. Exercícios: 1 a 11.

Visão geral: (segundo a descrição da própria plataforma Code.org, [tradução nossa]). Os alunos começarão com uma introdução (ou revisão, dependendo da experiência de sua classe) do espaço de trabalho online da Code.org. Os alunos aprenderão a funcionalidade básica da interface, incluindo o botão Executar, a funcionalidade de arrastar para excluir e conectar os blocos.

Os objetivos (segundo a plataforma Code.org [tradução nossa]): ordenar os comandos de movimento como etapas sequenciais em um programa. Modificar um programa existente para solucionar erros. Dividir uma longa sequência de instruções na maior sequência repetível.

PERCURSO FORMATIVO 2

Nos exercícios a serem realizados, o aluno tem como objetivo levar o personagem Scratch até a sua noz executando corretamente os “passos” que o conduzem até lá. Utilizando programação por blocos, para o aluno, a execução das atividades assemelha-se a um quebra-cabeça em que as peças necessitam ser encaixadas. Entretanto, no decorrer das atividades o aluno perceberá que há mais de uma forma de se obter o mesmo resultado.

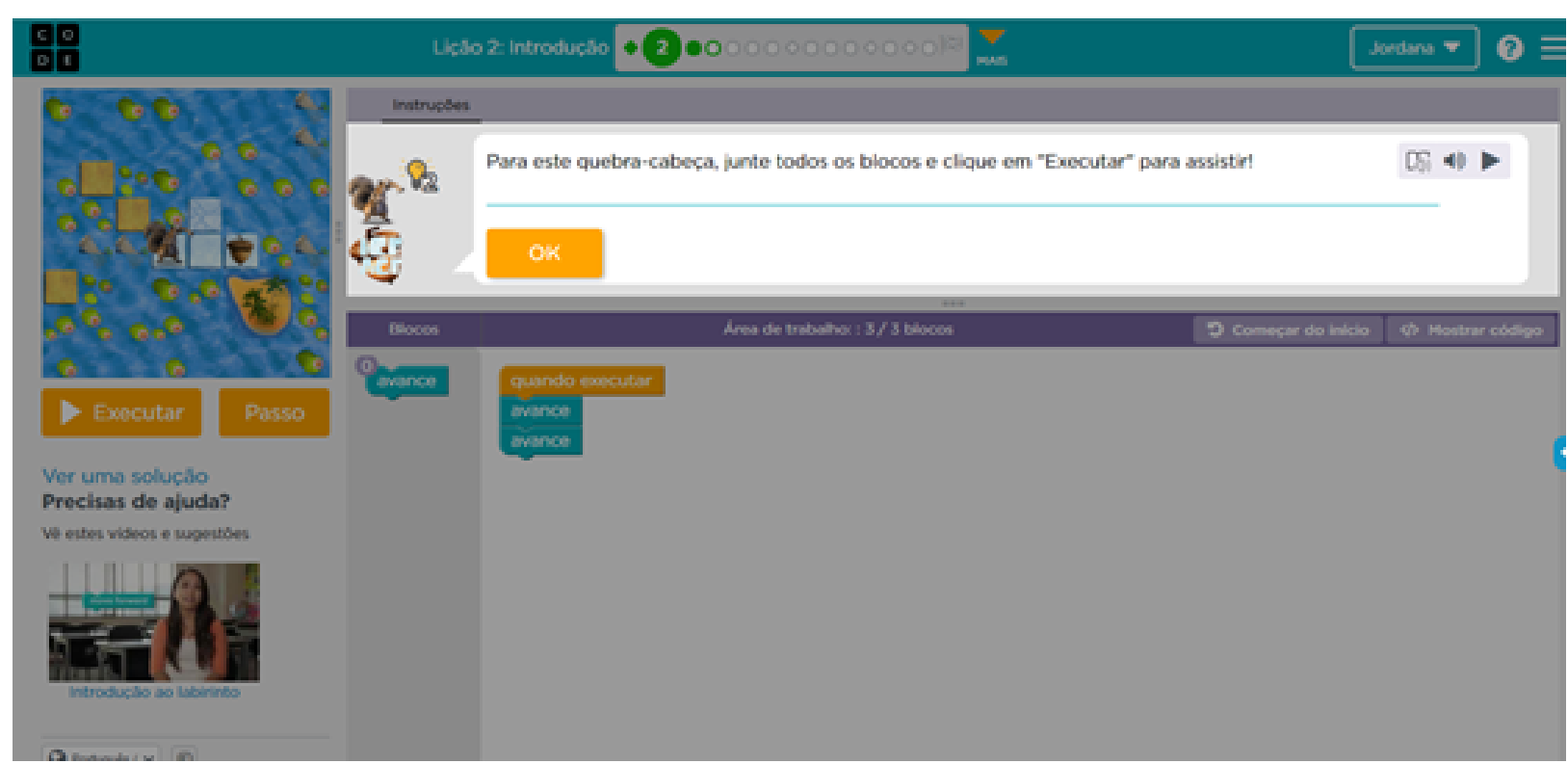
LIÇÃO 2: INTRODUÇÃO AO QUEBRA-CABEÇA ON-LINE. EXERCÍCIOS: I A II.

o Exercício 1: Vídeo: Introdução para o Code Studio

Explicação sobre as funcionalidades e a interface da plataforma relativas à programação em blocos.

o Exercício 2: o aluno irá entender o funcionamento da interface da plataforma Code.org que utiliza a estrutura de programação por blocos (basicamente criar um bloco único de código utilizando (arrastando) trechos de código e executar.

Ações por meio do personagem: andar em linha reta.



o Exercício 3: o aluno poderá observar que haverá obstáculos que dificultam seu objetivo, entretanto, isso ainda não é um problema para ele.

Ações por meio do personagem: andar em linha reta.



o Exercício 4: nesse exercício, o aluno deve retirar blocos do código que já está posicionado em sua área de trabalho para que o personagem realize seu objetivo corretamente. Além disso, poderá observar que há novas ações disponíveis para o seu personagem (virar para a direita e para a esquerda).

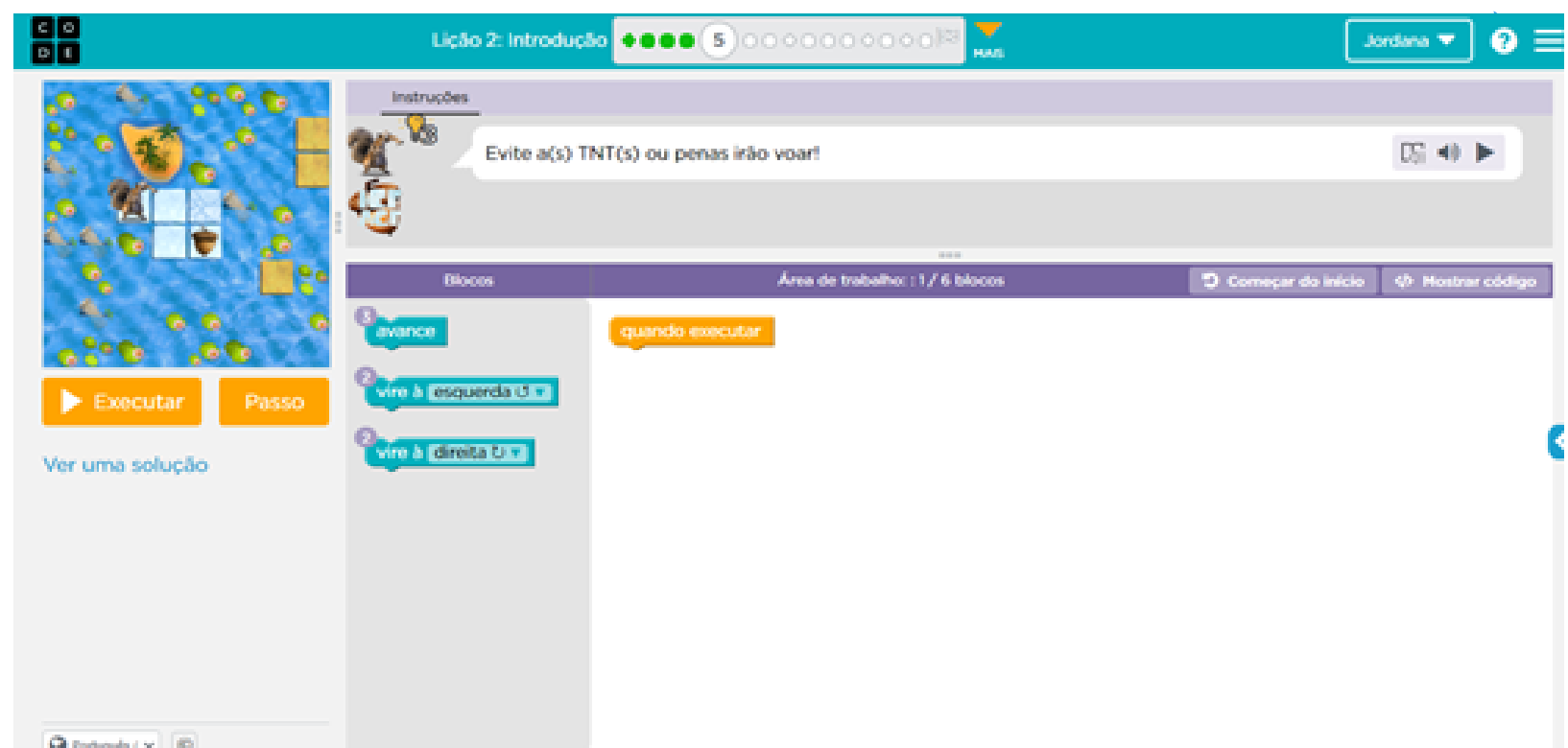
Ações por meio do personagem: seguir em frente, virar para a direita, virar para a esquerda.



PERCURSO FORMATIVO 2 - LIÇÃO 2

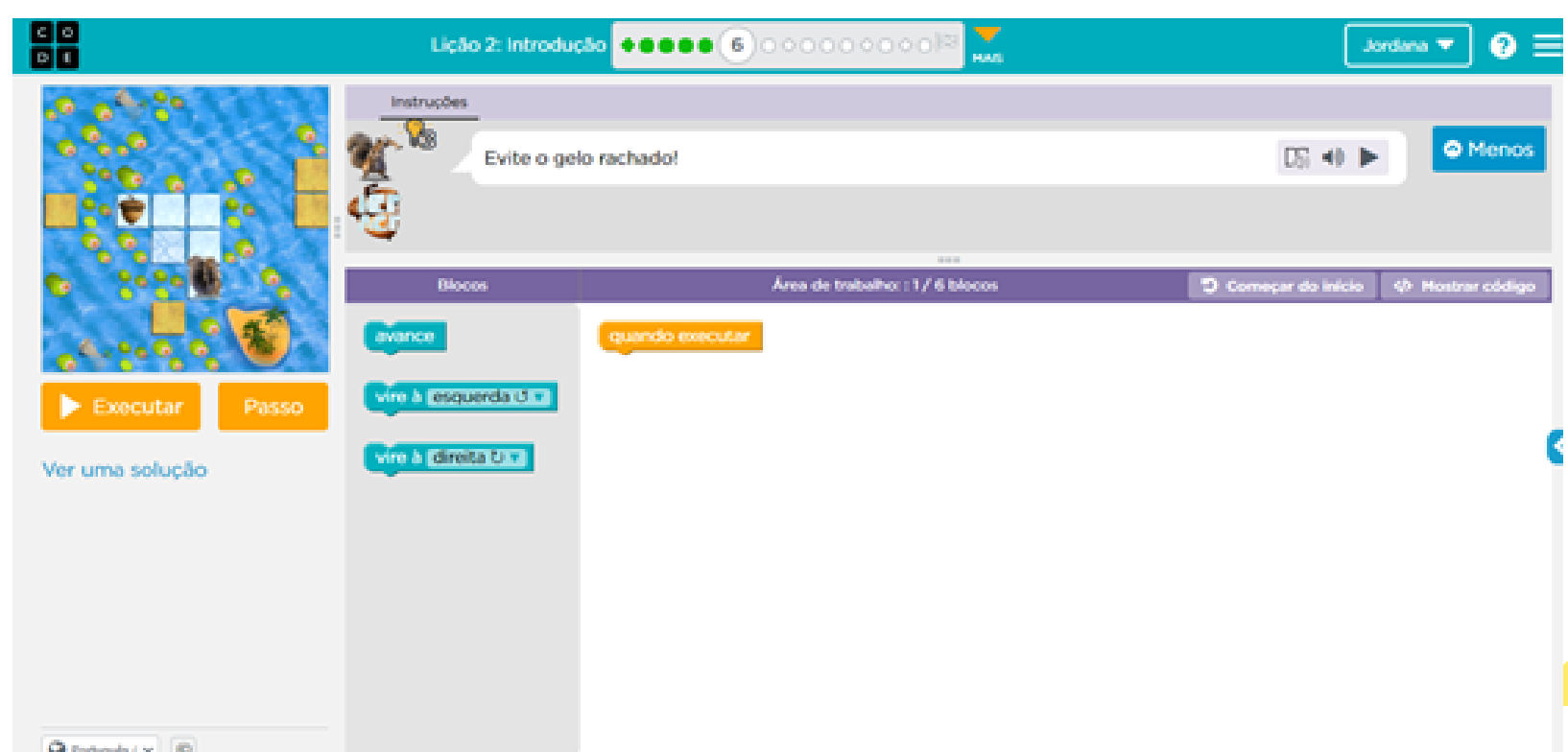
o Exercício 5: Neste exercício, o aluno deverá atingir seu objetivo, entretanto deverá evitar que o personagem caia.

Ações por meio do personagem: seguir em frente, virar para a direita, virar para a esquerda.



o Exercício 6: neste exercício, assim como no anterior, o aluno deverá atingir seu objetivo evitando obstáculos em seu caminho.

Ações por meio do personagem: seguir em frente, virar para a direita, virar para a esquerda.



o Exercício 7: esse exercício traz um questionamento ao aluno sobre a ação que será executada, levando-o a refletir sobre o posicionamento correto dos blocos no código e exibe para o aluno um novo item em sua lista de blocos que serão utilizadas nos próximos exercícios: as estruturas de repetição. Após responder o questionamento e executar, uma resposta é exibida ao aluno informando se está correta ou incorreta a sua resposta

Ações por meio do personagem: seguir em frente, virar para a direita, virar para a esquerda, utilizar estruturas de repetições.

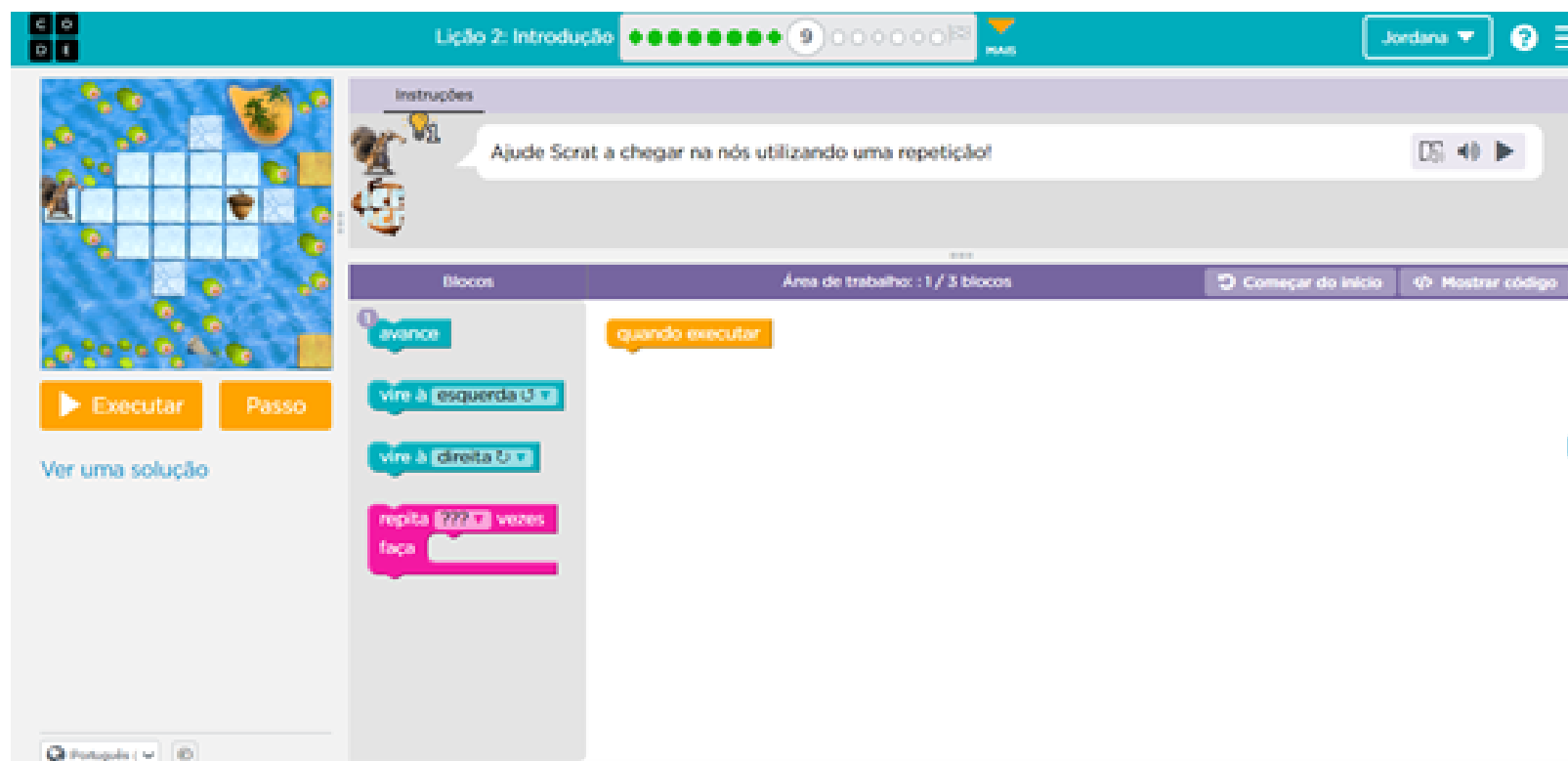
PERCURSO FORMATIVO 2 - LIÇÃO 2

o **Exercício 8:** Vídeo Laços Repita.

Ações por meio do personagem: Discorre sobre os benefícios de se utilizar estruturas para repetição na programação.

o **Exercício 9:** O primeiro exercício da sequência em que o aluno deverá utilizar estrutura de repetição em seus blocos. Nesse exercício ele terá a chance de realizar a ação semelhante aos outros realizados (apenas avançando) ou reduzir código (utilizando a estrutura de repetição).

Ações por meio do personagem: seguir em frente, virar para a direita, virar para a esquerda, utilizar estruturas de repetições.



o **Exercício 10:** Semelhante ao exercício anterior, o aluno deverá utilizar estrutura de repetição em seu código.

Ações por meio do personagem: seguir em frente, virar para a direita, virar para a esquerda, utilizar estruturas de repetições.



o **Exercício 11:** Apresenta um desafio para o aluno trazendo um código já pronto na área de trabalho, entretanto, com um percurso maior a ser percorrido pelo personagem.

Ações por meio do personagem: seguir em frente, virar para a direita, virar para a esquerda, utilizar estruturas de repetições.



PERCURSO FORMATIVO 4

As lições realizadas no Percurso Formativo 4 serão Lição 3, Lição 4 e Lição 7.

LIÇÃO 3: CRIAR UMA BASE

Visão geral: Os alunos deverão realizar a depuração[2] observando os blocos e fazendo alterações necessárias para que o personagem atinja seu objetivo.

Objetivos: analisar blocos e ordenar os comandos de movimento. Modificar um programa existente para solucionar erros. Dividir uma longa sequência de instruções na maior sequência repetível.

Exercício 1: Vídeo: Criar uma Base. O vídeo aborda sobre planejamento e persistência na realização de atividades (inclusive programação) e sobre aprender com os próprios erros.

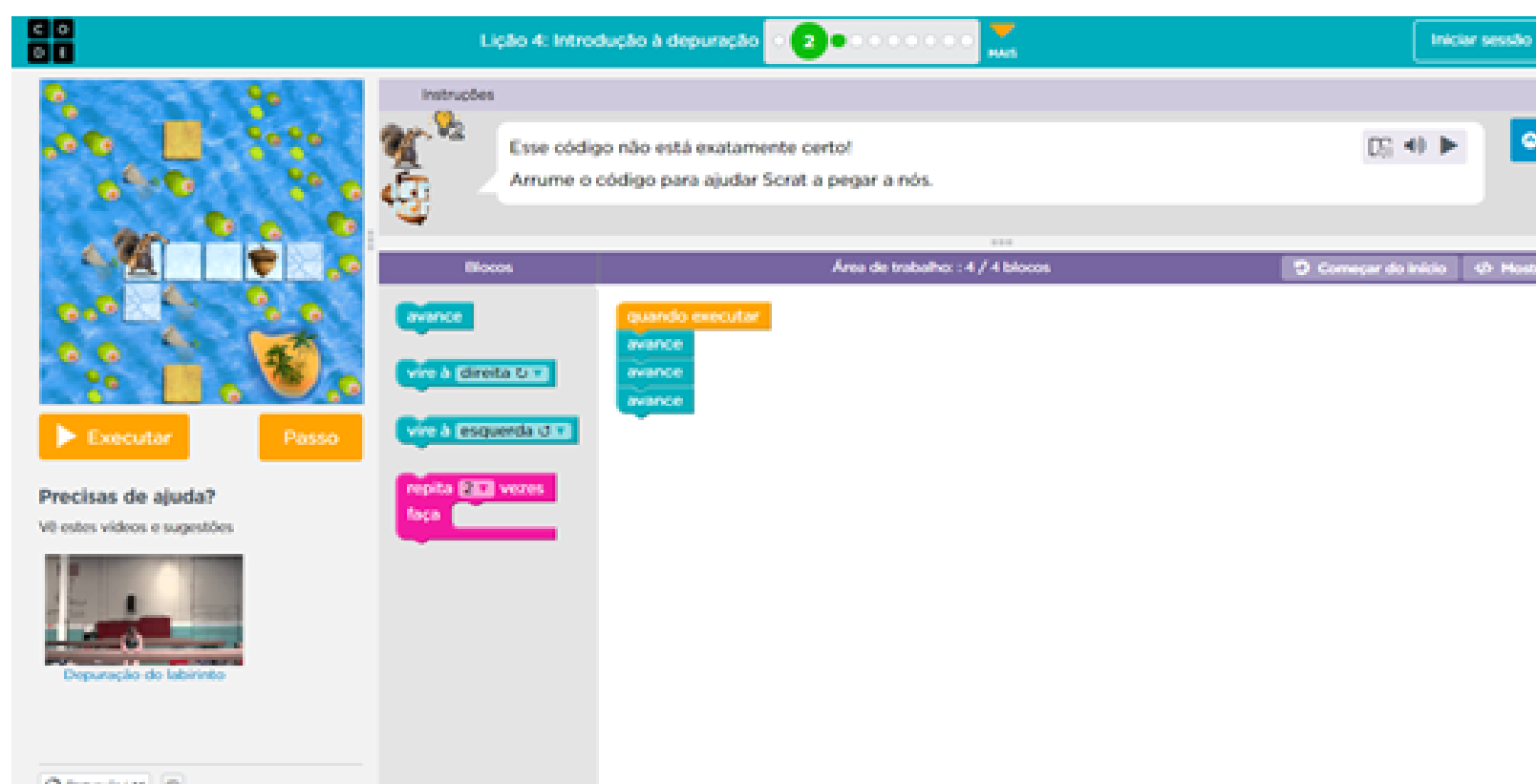
LIÇÃO 4: INTRODUÇÃO À DEPURAÇÃO. EXERCÍCIOS: 1 A 10.

Visão geral: Os alunos deverão realizar a depuração[1] observando os blocos e fazendo alterações necessárias para que o personagem atinja seu objetivo.

Objetivos: analisar blocos e ordenar os comandos de movimento. Modificar um programa existente para solucionar erros. Dividir uma longa sequência de instruções na maior sequência repetível

o Exercício 1: Vídeo: Analisando. Explica sobre depuração e sobre como encontrar e corrigir erros na plataforma utilizando o botão “Step” (ou “Passo”) que executa o código a cada bloco de maneira que o aluno poderá identificar os erros no código e corrigi-los.

o Exercício 2: o aluno deverá analisar a estrutura do código apresentado (estrutura dos blocos) e fazer alterações para que o personagem atinja os seus objetivos. Ações por meio do personagem: seguir em frente.



PERCURSO FORMATIVO 4 - LIÇÃO 4

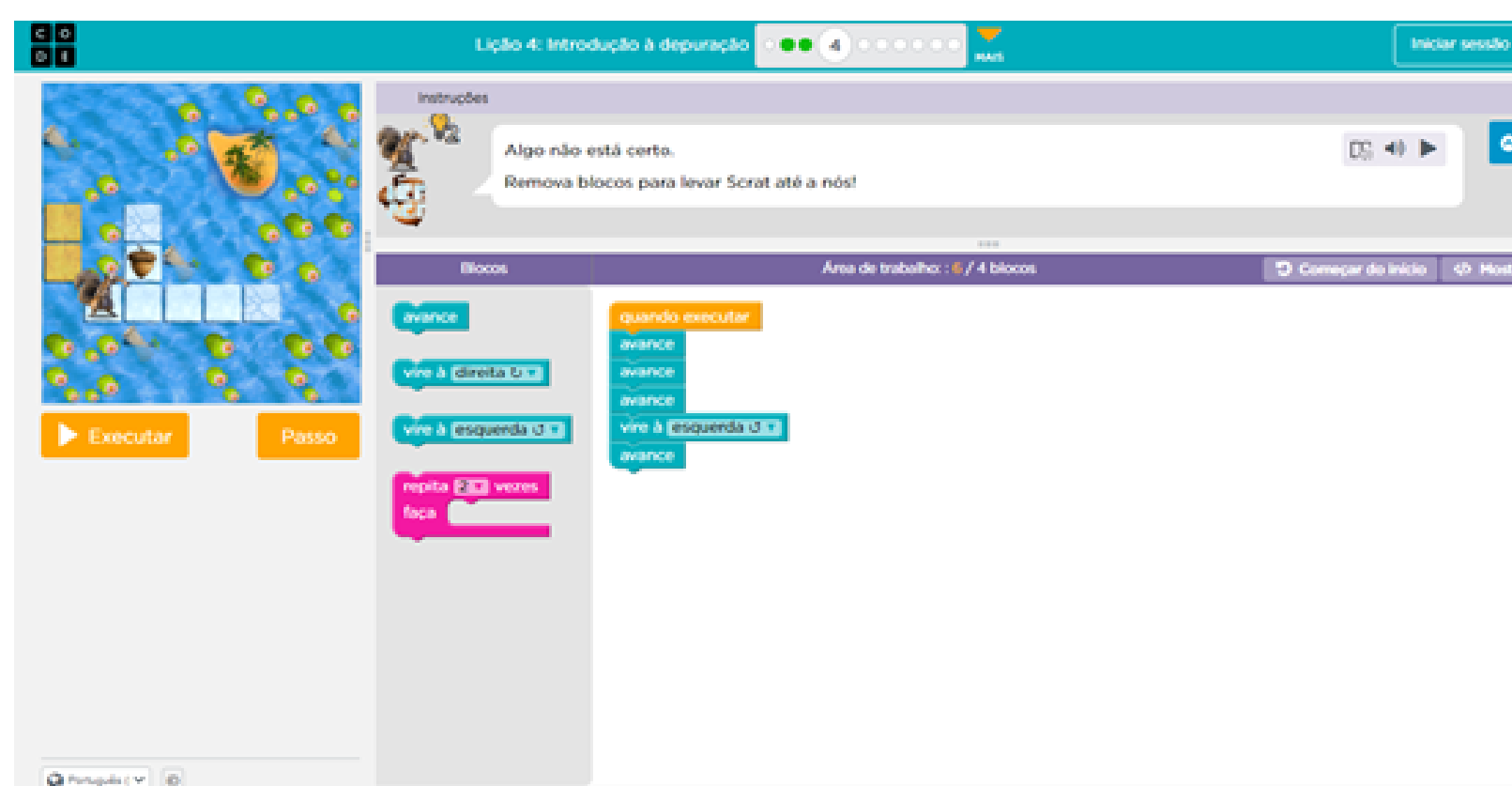
o Exercício 3: o aluno deverá analisar a estrutura do código apresentado (estrutura dos blocos) e fazer alterações para que o personagem atinja os seus objetivos. Poderá utilizar estruturas de repetição trabalhadas na aula anterior.

Ações por meio do personagem: seguir em frente, virar.



o Exercício 4: o aluno deverá analisar a estrutura do código apresentado (estrutura dos blocos) e fazer alterações para que o personagem atinja os seus objetivos.

Ações por meio do personagem: seguir em frente e virar.



o Exercício 5: o aluno deverá analisar a estrutura do código apresentado (estrutura dos blocos) e fazer alterações para que o personagem atinja os seus objetivos.

Ações por meio do personagem: seguir em frente e virar.



PERCURSO FORMATIVO 4 - LIÇÃO 4

o Exercício 6: o aluno deverá analisar a estrutura do código apresentado (estrutura dos blocos) e fazer alterações para que o personagem atinja os seus objetivos. Blocos que não podem ser apagados. Um desafio para o aluno e demanda um pouco mais de tempo para ser realizado.

Ações por meio do personagem: seguir em frente e virar.



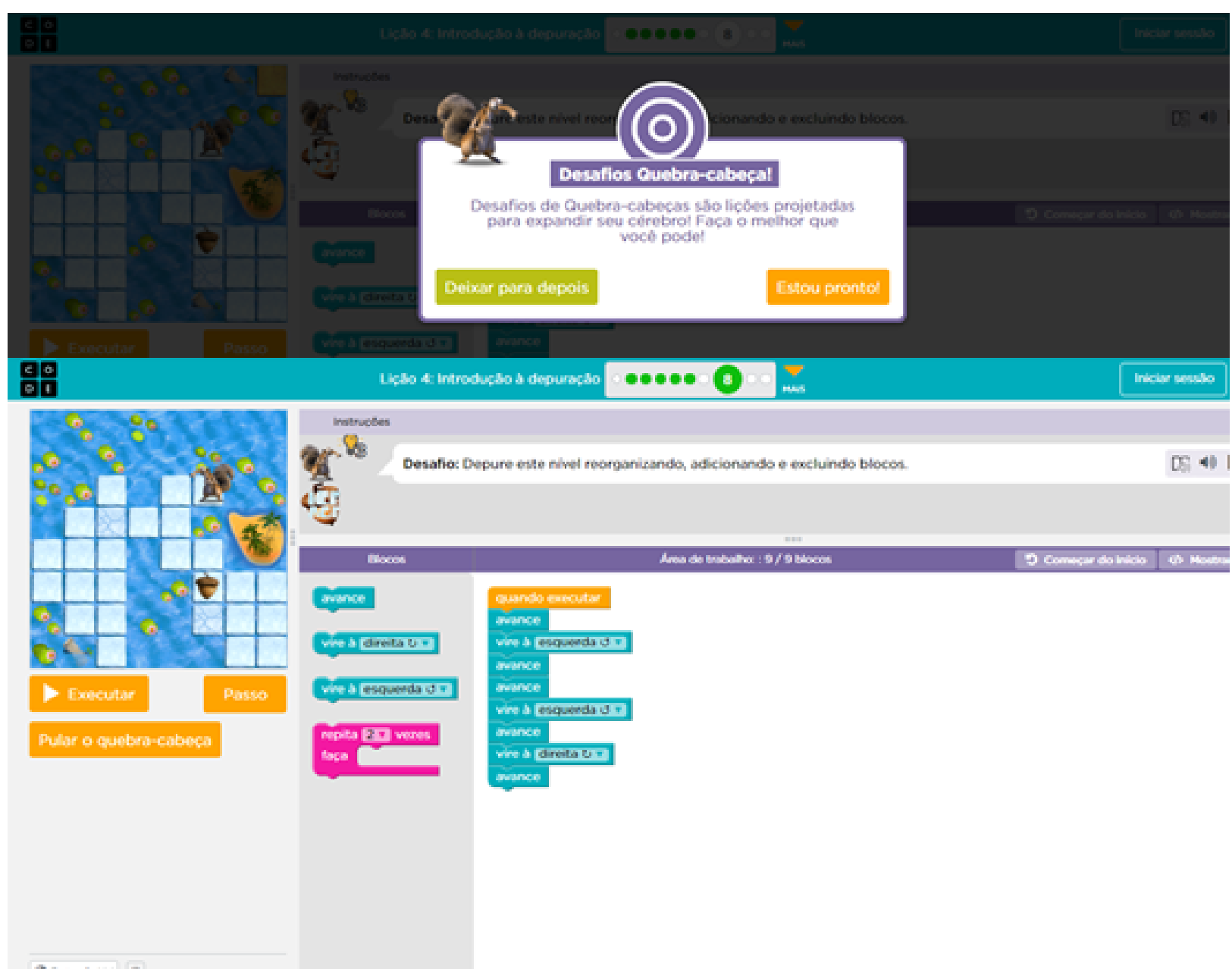
o Exercício 7: o aluno deverá analisar a estrutura do código apresentado e fazer alterações para que o personagem atinja os seus objetivos.

Ações por meio do personagem: seguir em frente e virar.



o Exercício 8: o aluno deverá analisar a estrutura do código apresentado (estrutura dos blocos) e fazer alterações para que o personagem atinja os seus objetivos. Blocos que não podem ser apagados. Um desafio para o aluno e demanda um pouco mais de tempo para ser realizado.

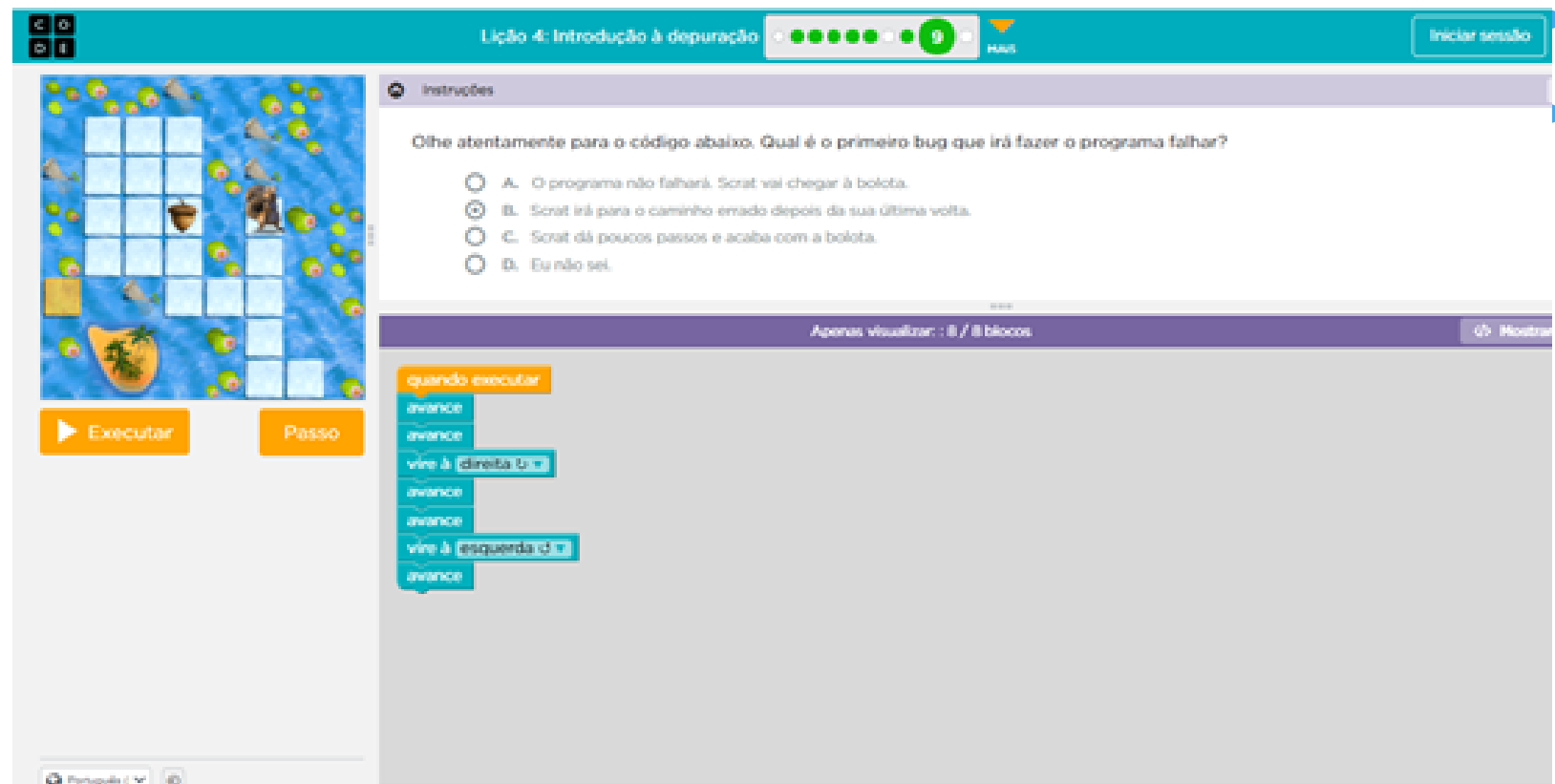
Ações por meio do personagem: seguir em frente e virar.



PERCURSO FORMATIVO - LIÇÃO 4

o Exercício 9: o aluno deverá analisar a estrutura do código apresentado (estrutura dos blocos) e fazer alterações para que o personagem atinja os seus objetivos. Blocos que não podem ser apagados. Um desafio para o aluno e demanda um pouco mais de tempo para ser realizado. – Questão para o aluno – forma de avaliar

Ações por meio do personagem: seguir em frente e virar.



o Exercício 10: o aluno deverá analisar a estrutura do código apresentado e fazer alterações para que o personagem atinja os seus objetivos e possivelmente usar estruturas de repetição para otimização de código.

Ações por meio do personagem: seguir em frente e virar.



PERCURSO FORMATIVO 4

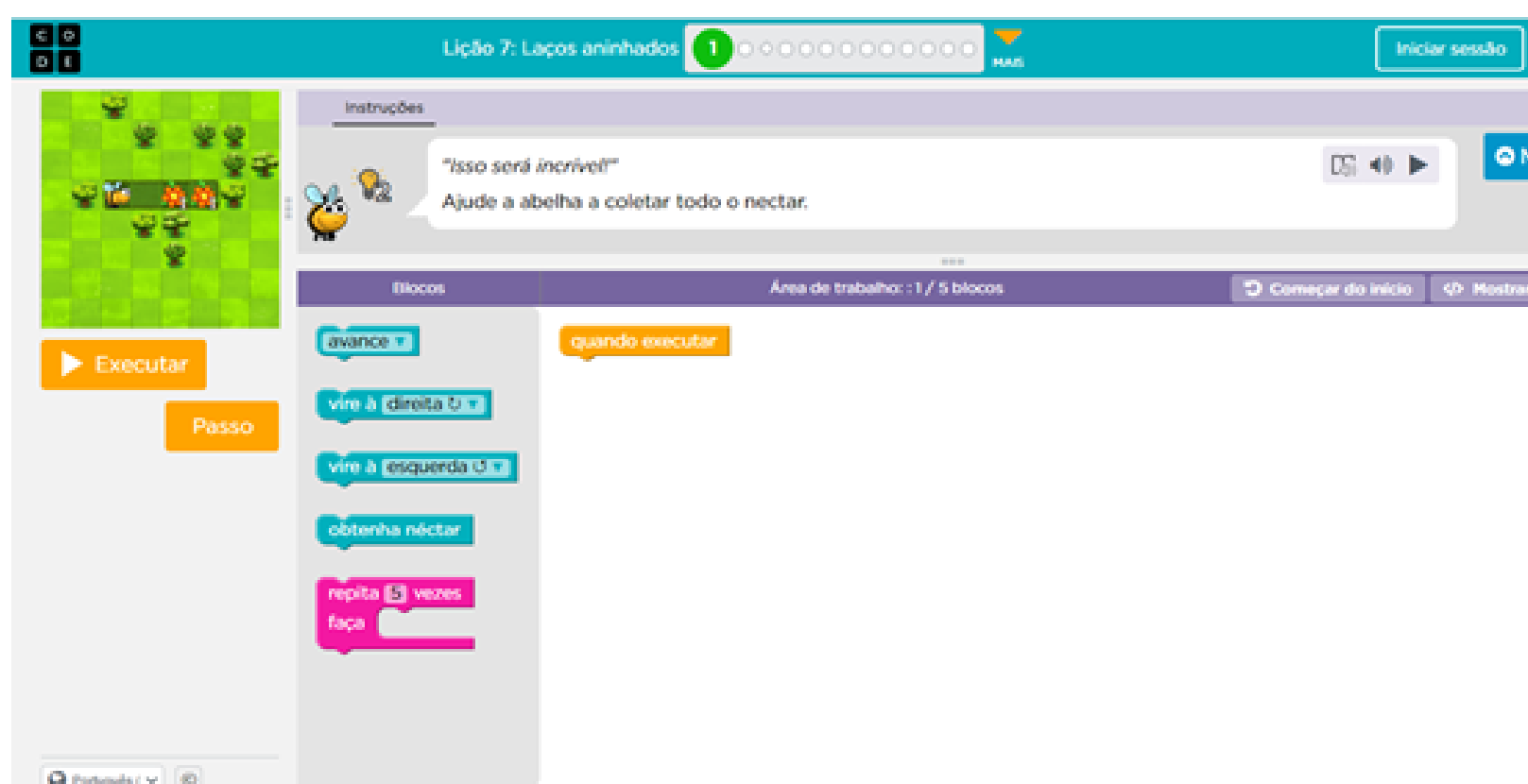
LIÇÃO 7: LAÇOS ANINHADOS. EXERCÍCIOS: 1 AO 13.

Visão geral: Os alunos poderão utilizar laços aninhados (estruturas de repetição) para otimizar ações realizadas pelo personagem.

Objetivos: analisar blocos e ordenar os comandos de movimento. Dividir uma longa sequência de instruções na maior sequência repetível e utilizar estruturas de repetição para realizar ações otimizando o código.

o Exercício 1: o aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo. Além de chegar a determinado local, o objetivo do personagem inclui realizar uma ação (coletar néctar). Para isso, o aluno deverá utilizar estruturas de repetição trabalhadas na aula anterior.

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda, obter néctar.



o Exercício 2: o aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo. Além de chegar a determinado local, o objetivo do personagem inclui realizar uma ação (coletar néctar). O aluno poderá utilizar estruturas de repetição aninhadas.

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda, obter néctar.



PERCURSO FORMATIVO 4 - LIÇÃO 7

o **Exercício 3:** Vídeo: Laços aninhados. Junções de loop aninhado.

o **Exercício 4:** o aluno deverá analisar a estrutura do código e responder à questão informando qual ação terá o personagem ao clicar em “Executar”.

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda, obter néctar.

Lição 7: Laços aninhados 4

Instruções

Olhe atentamente para os laços aninhados abaixo. O que acontecerá quando você clicar em "Executar"?

A. A abelha avançará e obterá néctar apenas uma vez.

B. A abelha receberá apenas 2 unidades de néctar.

C. A abelha coletará todo o néctar.

D. Eu não sei.

Apenas visualizar: Mostrar

quando executar

repita 2 vezes

faça

avance

repita 2 vezes

faça

avance

obtenha néctar

vire à direita 90°

o **Exercício 5:** o aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo utilizando estruturas de repetição aninhadas.

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda, obter néctar.

Lição 7: Laços aninhados 5

Instruções

Dessa vez, ajude a abelha a coletar todo o néctar utilizando o menor número possível de blocos.

Blocos Área de trabalho: 1 / 6 blocos Começar do início Mostrar

quando executar

avance

vire à direita 90°

vire à esquerda 90°

obtenha néctar

repita 2 vezes

faça

o **Exercício 6:** o aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo utilizando estruturas de repetição aninhadas.

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda.

Lição 7: Laços aninhados 6

Instruções

“Zumbi com fome!”

Leve o zumbi até o girassol utilizando o menor número possível de blocos.

Blocos Área de trabalho: 1 / 5 blocos Começar do início Mostrar

quando executar

avance

vire à esquerda 90°

vire à direita 90°

repita 2 vezes

faça

PERCURSO FORMATIVO 4 - LIÇÃO 7

o **Exercício 7:** o aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo utilizando estruturas de repetição aninhadas.

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda,



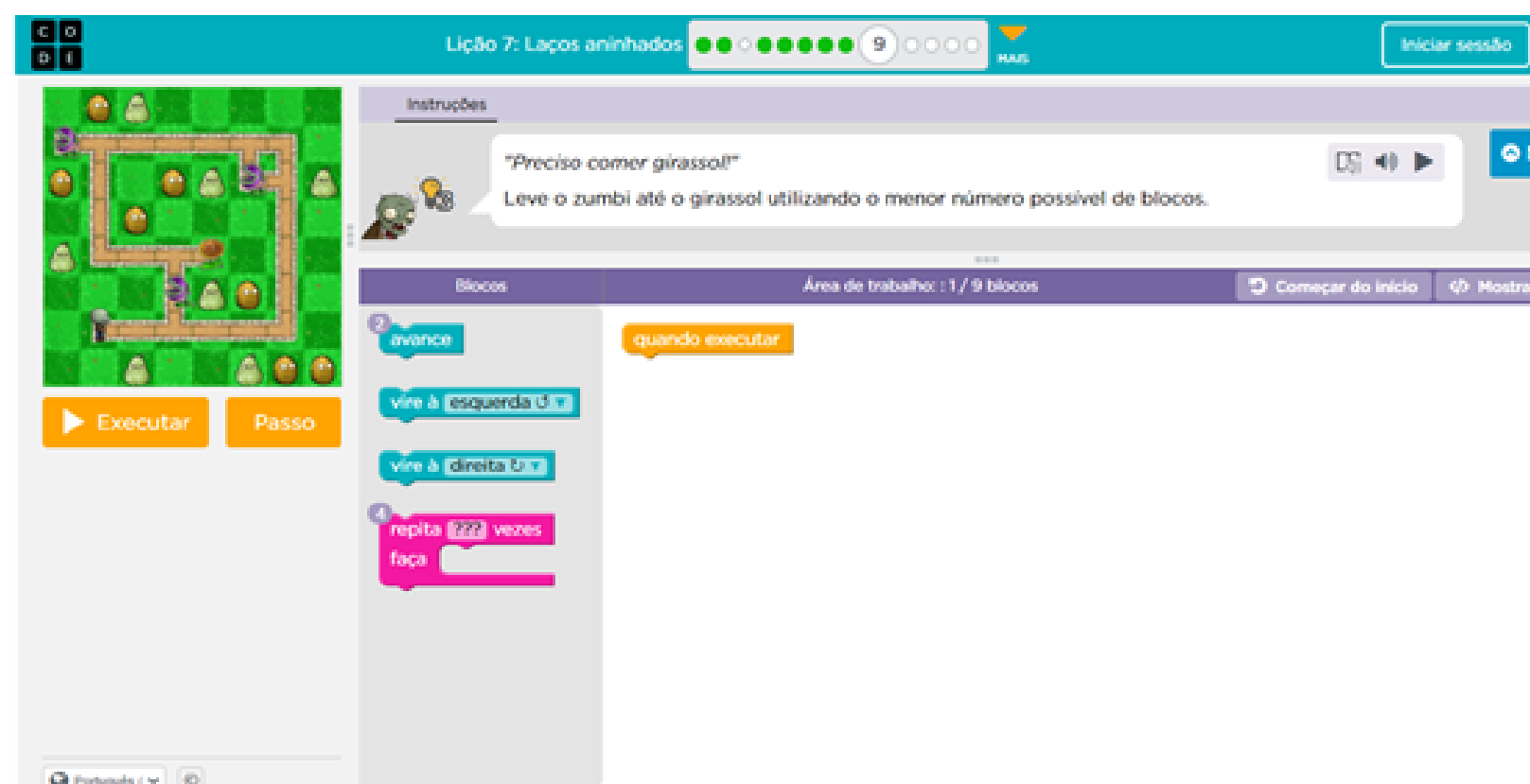
o **Exercício 8:** o aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo utilizando estruturas de repetição aninhadas. Com nível de dificuldade maior que o exercício anterior.

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda.



o **Exercício 9:** o aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo utilizando estruturas de repetição aninhadas. Com nível de dificuldade maior que o exercício anterior.

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda.



PERCURSO FORMATIVO 4 - LIÇÃO 7

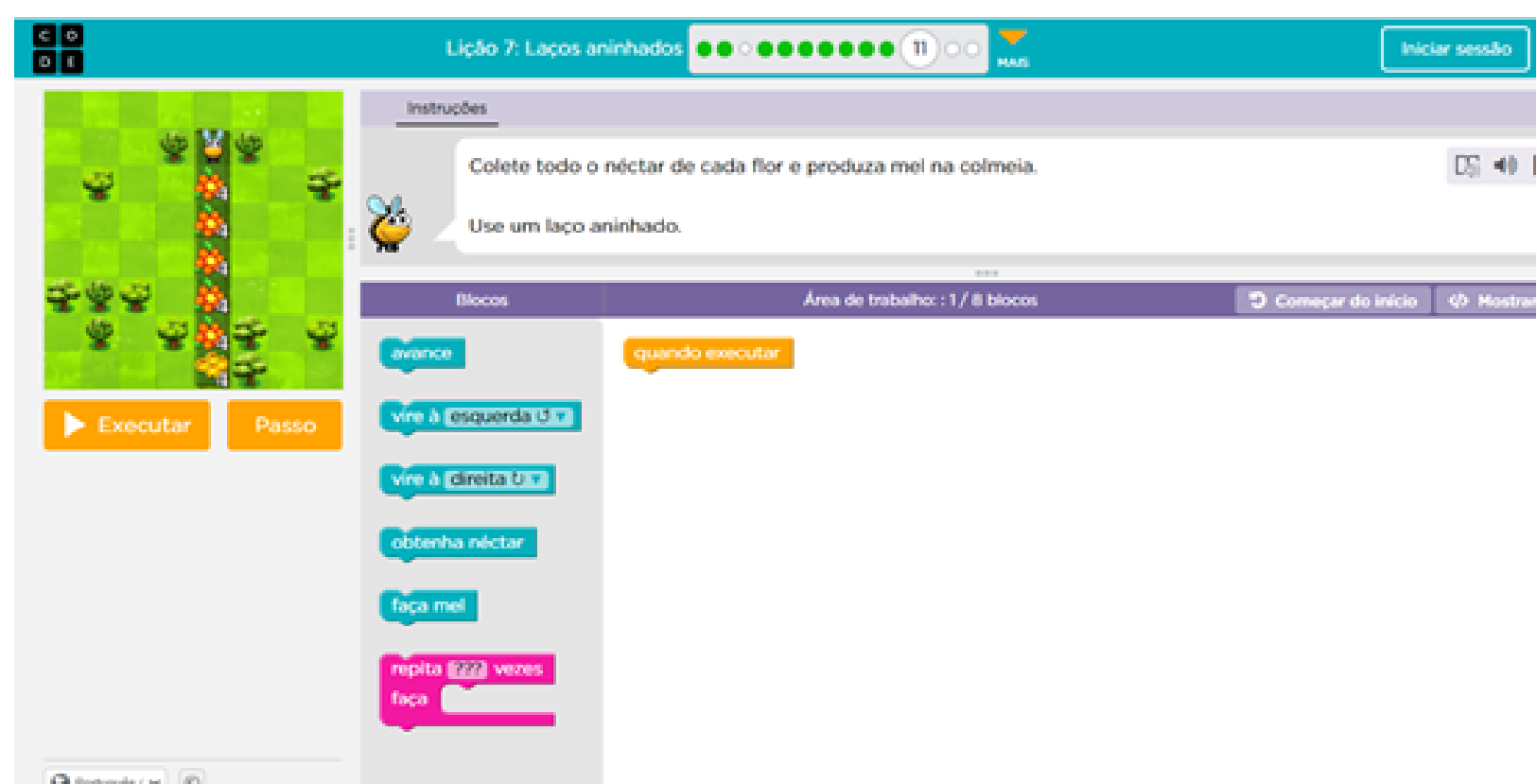
o **Exercício 10:** Desafio quebra-cabeça. O aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo utilizando estruturas de repetição aninhadas com sequência maior.

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda, obter néctar.



o **Exercício 11:** o aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo utilizando estruturas de repetição aninhadas e realizar uma nova ação (fazer mel). Além disso, o aluno deverá encontrar uma solução em que o personagem obtenha todo o néctar de cada flor a cada avanço uma vez que não há retorno para o caminho do personagem.

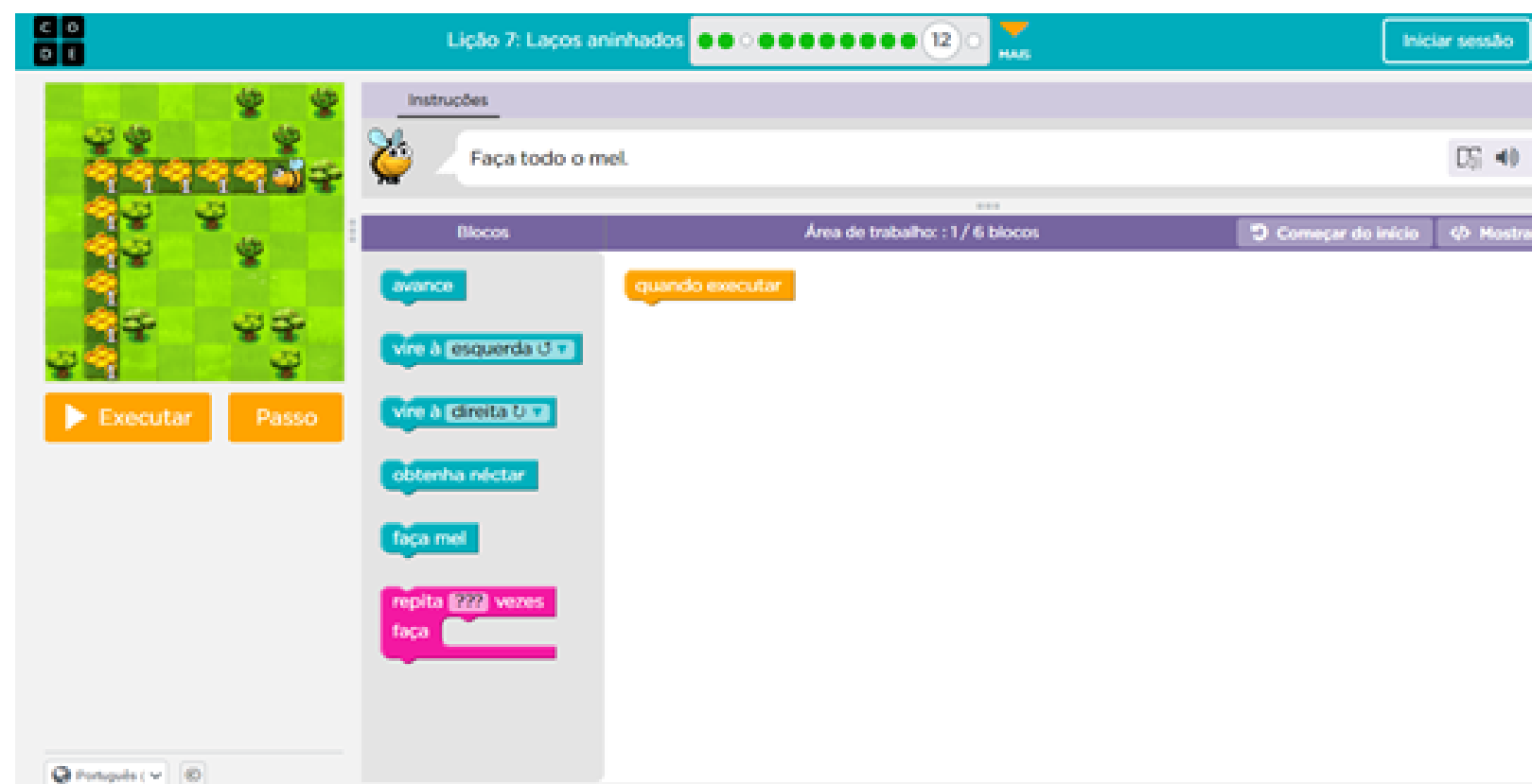
Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda, obter néctar, fazer mel.



PERCURSO FORMATIVO 4 - LIÇÃO 7

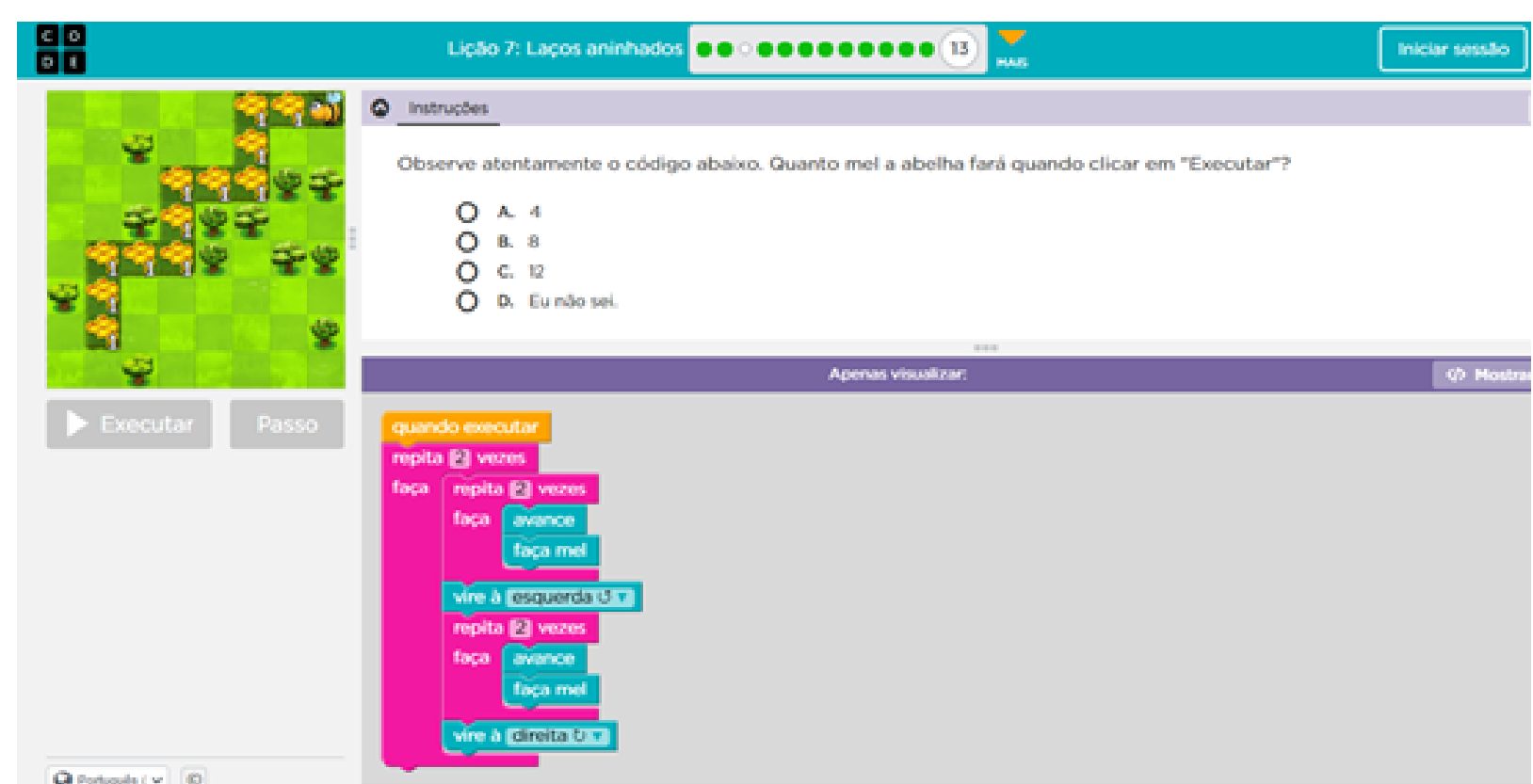
o Exercício 12: o aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo utilizando estruturas de repetição aninhadas com sequência maior e realizar uma nova ação (fazer mel).

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda, obter néctar, fazer mel.



o Exercício 13: o aluno deverá criar a estrutura do código para que o personagem atinja o seu objetivo utilizando estruturas de repetição aninhadas com sequência maior e realizar uma nova ação (fazer mel).

Ações por meio do personagem: seguir em frente, virar para a direita e para a esquerda, obter néctar, fazer mel.



PERCURSO FORMATIVO 6

As lições realizadas no Percurso Formativo 6 serão:

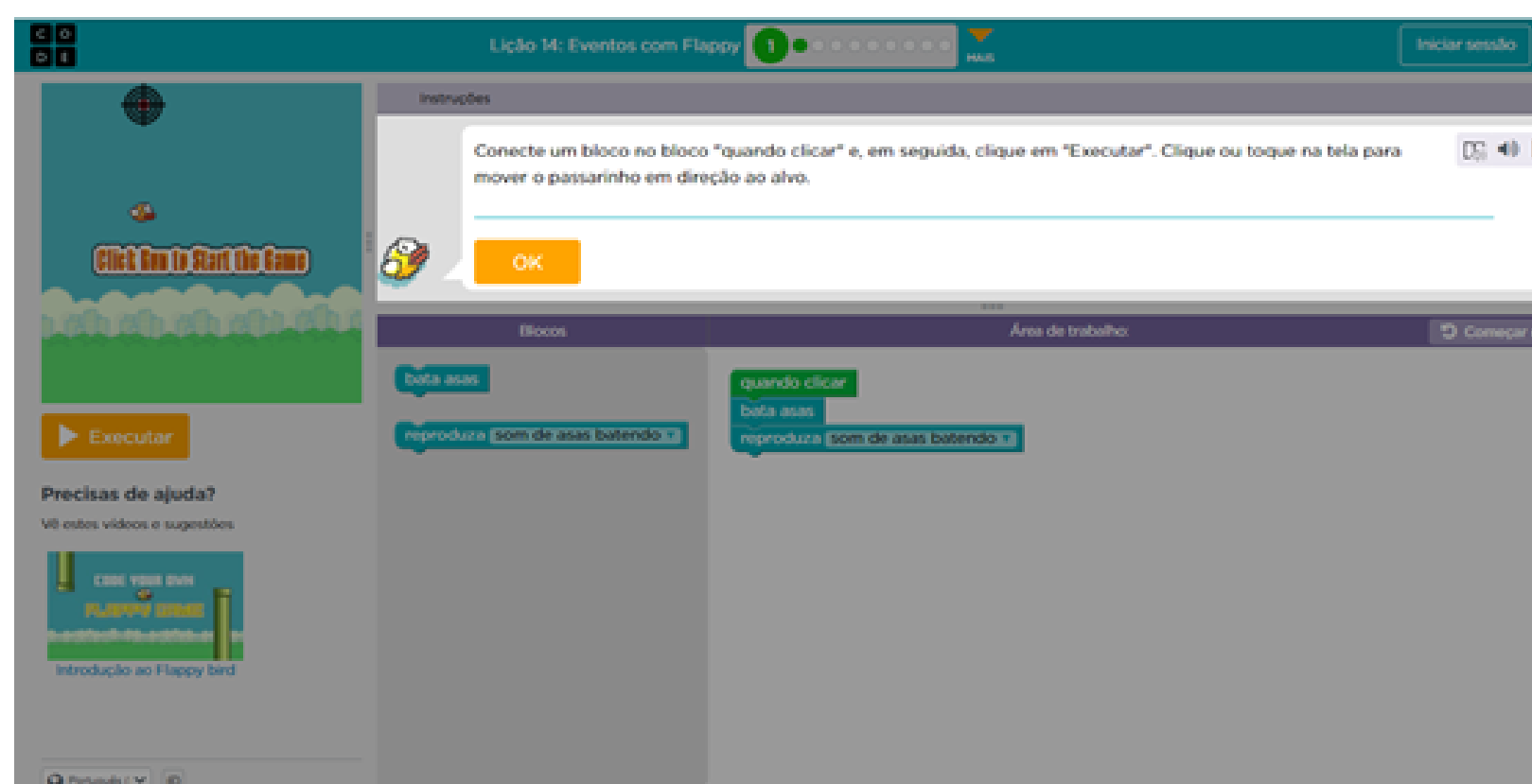
LIÇÃO 14: EVENTOS COM FLAPPY. EXERCÍCIOS: 1 A 10.

Visão geral: esta atividade aborda a manipulação de eventos utilizando o Flappy Bird como roteiro e cenário.

Objetivos: ordenar os comandos de movimento como etapas sequenciais em um programa. Manipular eventos de clique, movimento do personagem nos sentidos do plano cartesiano, inserir som aos movimentos. Criar o próprio jogo.

o Exercício 1: criar evento de movimento para o personagem.

Ações por meio do personagem: bater as asas e reproduzir som.



o Exercício 2: Criar evento de movimento para o personagem e inserir ações de estrutura de jogos na ação fazendo com que o personagem finalize o jogo se bater no chão.

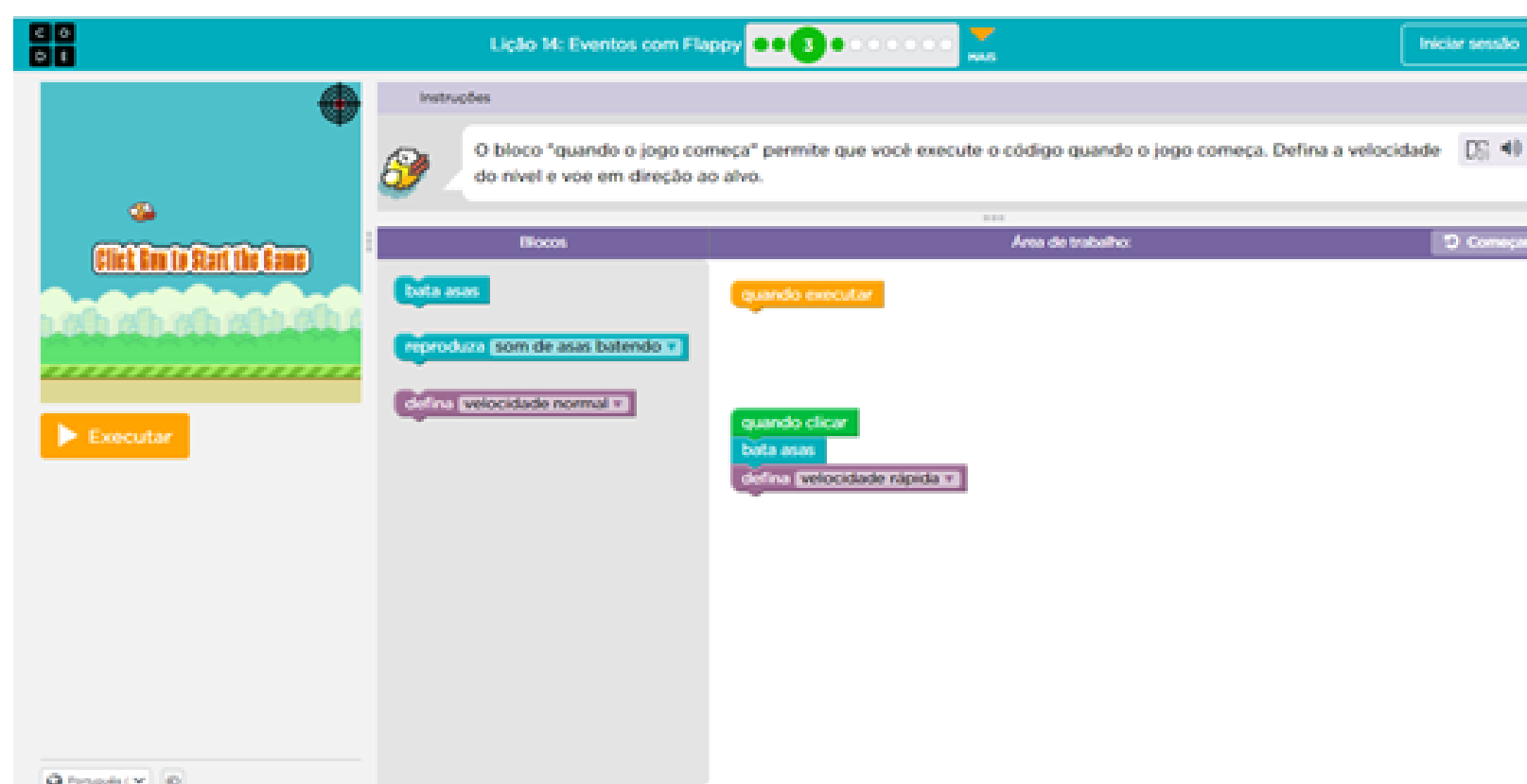
Ações por meio do personagem: bater as asas e reproduzir som, finalizar o jogo.



PERCURSO FORMATIVO 6 - LIÇÃO 14

o **Exercício 3:** Criar evento de movimento para o personagem e inserir ações de estrutura de jogos na ação do personagem.

Ações por meio do personagem: bater as asas, reproduzir som, definir velocidade.



o **Exercício 4:** criar evento de movimento para o personagem e inserir ações de estrutura de jogos na ação fazendo com que o personagem finalize o jogo se atingir um obstáculo.

Ações por meio do personagem: bater as asas e reproduzir som, finalizar o jogo, definir velocidade.



o **Exercício 5:** Criar evento de movimento para o personagem e inserir ações de estrutura de jogos na ação e pontuar se passar o obstáculo.

Ações por meio do personagem: bater as asas e reproduzir som, definir velocidade, pontuar.



PERCURSO FORMATIVO 6 - LIÇÃO 14

o **Exercício 6:** criar evento de movimento para o personagem e inserir ações de estrutura de jogos na ação fazendo com que o personagem finalize o jogo se bater no chão.

Ações por meio do personagem: bater as asas e reproduzir som, finalizar o jogo.



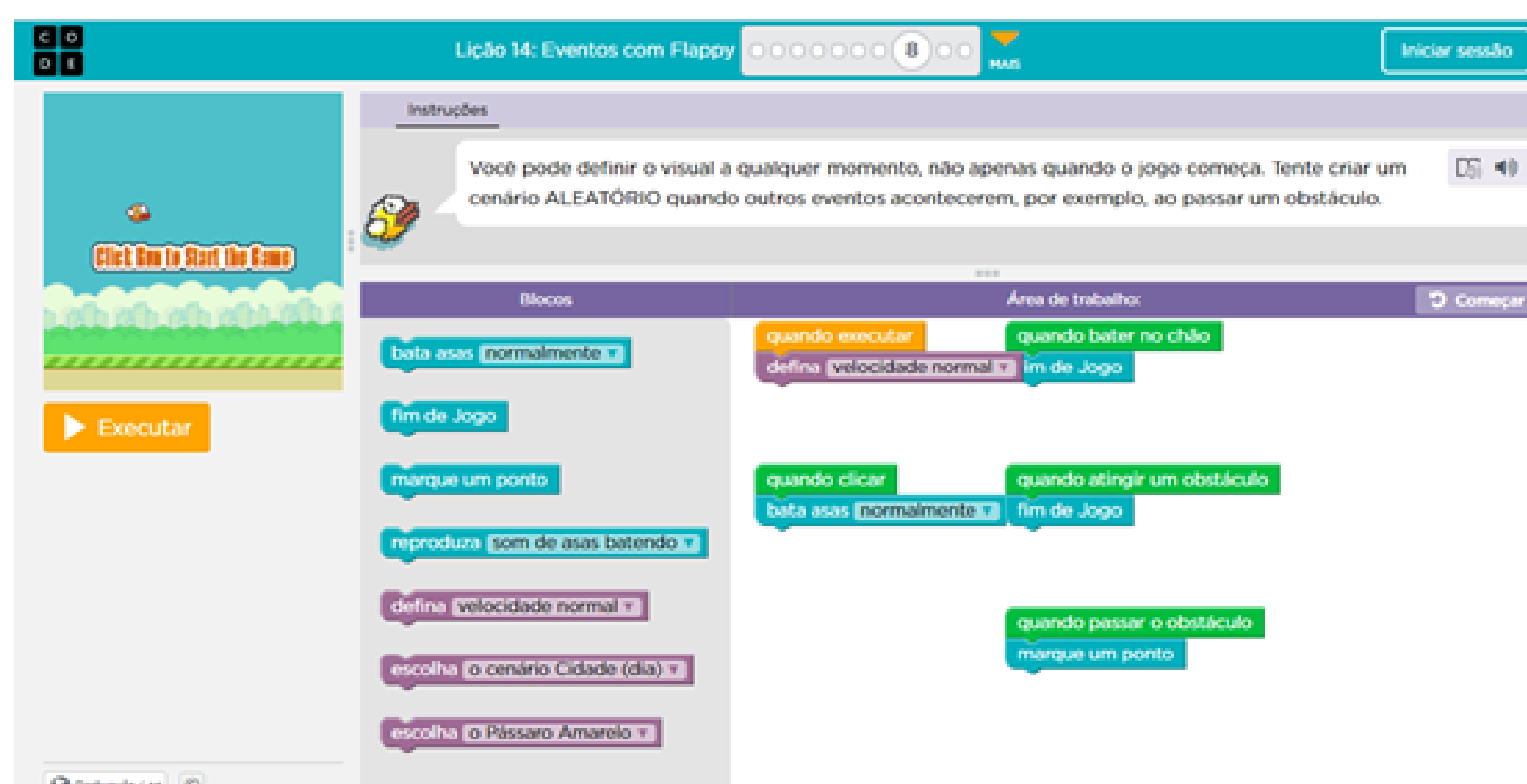
o **Exercício 7:** criar evento de movimento para o personagem inserindo ações de estrutura de jogos na ação fazendo com que o personagem se movimente, finalize o jogo se bater no chão ou atingir obstáculo e pontue se passar o obstáculo.

Ações por meio do personagem: bater as asas, reproduzir som, pontuar e finalizar o jogo.



o **Exercício 8:** criar evento de movimento para o personagem inserindo ações de estrutura de jogos na ação fazendo com que o personagem se movimente, finalize o jogo se bater no chão ou atingir obstáculo e pontue se passar o obstáculo.

Ações por meio do personagem: bater as asas, reproduzir som, pontuar, finalizar o jogo e alterar o cenário.



PERCURSO FORMATIVO 6 - LIÇÃO 14

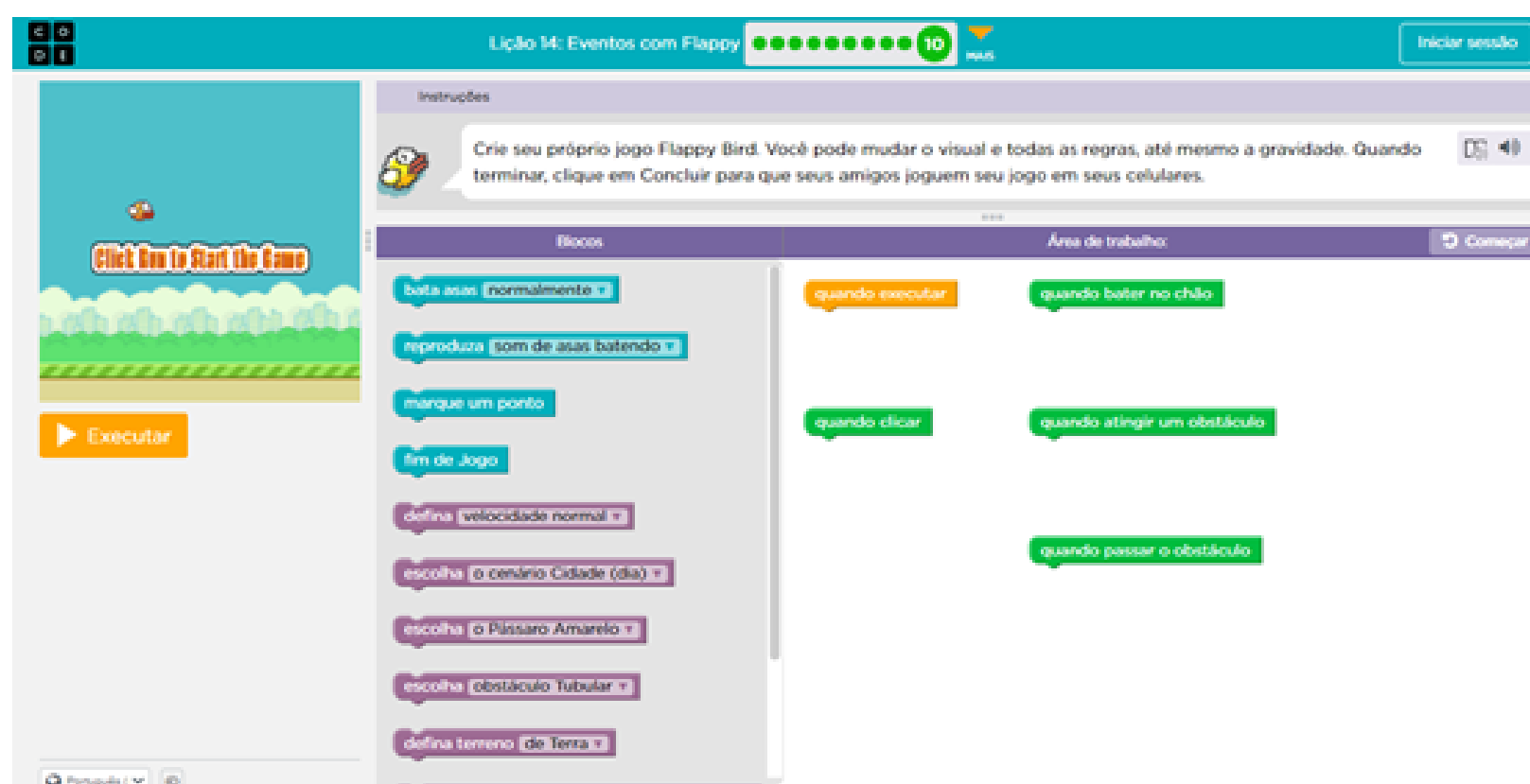
o **Exercício 9:** criar evento de movimento para o personagem e inserir ações de estrutura de jogos na ação.

Ações por meio do personagem: bater as asas e reproduzir som, finalizar o jogo ao bater no chão ou atingir obstáculo, pontuar.



o **Exercício 10:** o aluno poderá criar seu próprio jogo e compartilhar.

Ações por meio do personagem: bater as asas e reproduzir som, finalizar o jogo ao bater no chão ou atingir obstáculo, pontuar.



LIÇÃO 33: CONSTRUIR UM PROJETO.

Visão geral: esta lição deixa livre para os alunos a escolha de roteiro e cenário para a criação de um projeto utilizando programação em blocos.

Objetivos: ordenar os comandos de movimento como etapas sequenciais em um programa. Manipular eventos de clique, movimento do personagem nos sentidos do plano cartesiano, inserir som aos movimentos. Criar o próprio projeto de forma livre.

o **Exercício 2:** Revisar o seu Projeto.

