

APÊNDICE G – Criando um Algoritmo Genérico para Resolver Labirintos no Minecraft  
- Plano de Aula

**PLANO DE AULA**

**DADOS DE IDENTIFICAÇÃO**

**Descrição:** Criando um Algoritmo Genérico para Resolver Labirintos no Minecraft.

**Nome(s) do(s) participante(s):** Jonnhy Moraes Marques

**Descrição da Aula:** Os alunos aprenderão a identificar as entradas (recursos necessários) e as saídas (itens criados) em um processo de criação automática no Minecraft.

**Duração:** 50 minutos

**OBJETIVO DA AULA**

Nesta atividade, os alunos devem comparar diferentes labirintos no Minecraft e programar o *Agent* para resolvê-los de forma genérica. Eles devem identificar as semelhanças e diferenças entre os labirintos e criar um algoritmo que usa variáveis para que o *Agent* possa resolver qualquer labirinto.

**HABILIDADES TRABALHADAS**

(EF06CO06) Comparar diferentes casos particulares (instâncias) de um mesmo problema, identificando as semelhanças e diferenças entre eles, e criar um algoritmo para resolver todos, fazendo uso de variáveis (parâmetros) para permitir o tratamento de todos os casos de forma genérica.

**METODOLOGIA**

**Introdução**

Hoje, vamos nos aventurar em um desafio emocionante no Minecraft: resolver labirintos! Vocês já se perguntaram como é possível encontrar o caminho certo em um labirinto complexo? Pois bem, nossa missão de hoje será criar um algoritmo que ajude o *Agent* a encontrar a saída, não importa quão difícil seja o labirinto. Vamos aprender a identificar as características dos labirintos e, mais importante, como podemos programar o *Agent* para resolver qualquer labirinto de forma eficiente.

Ao longo da atividade, vamos explorar como diferentes labirintos podem ser resolvidos usando variáveis que ajustam o comportamento do *Agent*. Juntos, vamos descobrir como fazer o *Agent* tomar decisões em cada bifurcação e garantir que ele chegue à saída, ajustando parâmetros no código. A cada etapa, entenderemos as semelhanças e diferenças entre os labirintos e como podemos usar isso a nosso favor.

Preparem-se, pois nossa jornada de hoje vai desafiar sua capacidade de resolver problemas de maneira genérica, criando um algoritmo flexível que permite ao *Agent* superar qualquer labirinto!

## **Desenvolvimento**

No início da atividade, os alunos serão apresentados a um **quadro de instruções** (Figura 1) que explicará o objetivo da tarefa e como ela será realizada. O objetivo principal é resolver dois labirintos com diferentes níveis de dificuldade, criando um algoritmo que possa ser generalizado para resolver ambos.

### **Etapa 1: Resolvendo o Primeiro Labirinto**

Os alunos começam com um labirinto simples (Figura 2). Esse labirinto tem poucas bifurcações e um caminho mais direto até a saída, sendo ideal para a introdução ao problema. Os alunos devem usar o Criador de Código no Minecraft para programar o *Agent* e guiá-lo até a saída. Na Figura 3, há um exemplo que foi implementado para resolver esse primeiro labirinto. Esse código contém instruções básicas, como mover o *Agent* em direções específicas (frente, esquerda, direita) e lidar com pequenas bifurcações.

### **Etapa 2: Resolvendo o Segundo Labirinto**

Após concluir o primeiro labirinto, os alunos avançam para o segundo labirinto (Figura 4), que é mais complexo e apresenta mais bifurcações e caminhos sem saída. Nesse ponto, a tarefa é criar um algoritmo para resolver o segundo labirinto. Os passos se assemelham ao do primeiro labirinto. Na Figura 5, há um exemplo que foi implementado para resolver esse primeiro labirinto.

### **Etapa 3: Resolvendo o Ambos Labirintos**

Após concluir o primeiro e o segundo labirinto, os alunos avançam para a terceira etapa. Nesse ponto, a tarefa é criar um algoritmo genérico que possa ser adaptado para resolver ambos os labirintos. A habilidade de generalização será fundamental aqui, pois o código inicial deve ser modificado para se ajustar às novas condições do labirinto.

Eles podem usar variáveis para definir o tamanho dos caminhos e a lógica de decisão nas bifurcações. Assim, o mesmo algoritmo pode ser usado para resolver labirintos diferentes. Outra alternativa é juntar os dois códigos, modificando a chamada da função, definindo uma chamada para o labirinto 1 e outra chamada para o labirinto 2, além do comando para chamar o *Agent* para a posição inicial de ambos os labirintos (Figura 6).

## **Fechamento**

Chegamos ao final desta atividade de resolução de labirintos no Minecraft, onde os alunos atingiram com sucesso o objetivo principal da aula: criar um algoritmo genérico capaz de resolver diferentes tipos de labirintos, utilizando variáveis para ajustar direções e distâncias. Durante esta jornada no mundo virtual do Minecraft, os alunos aplicaram conceitos de generalização, análise e resolução de problemas, ao compreender como adaptar um único código para diferentes situações, desenvolvendo um raciocínio mais eficiente e flexível para a solução de desafios.

## Imagens

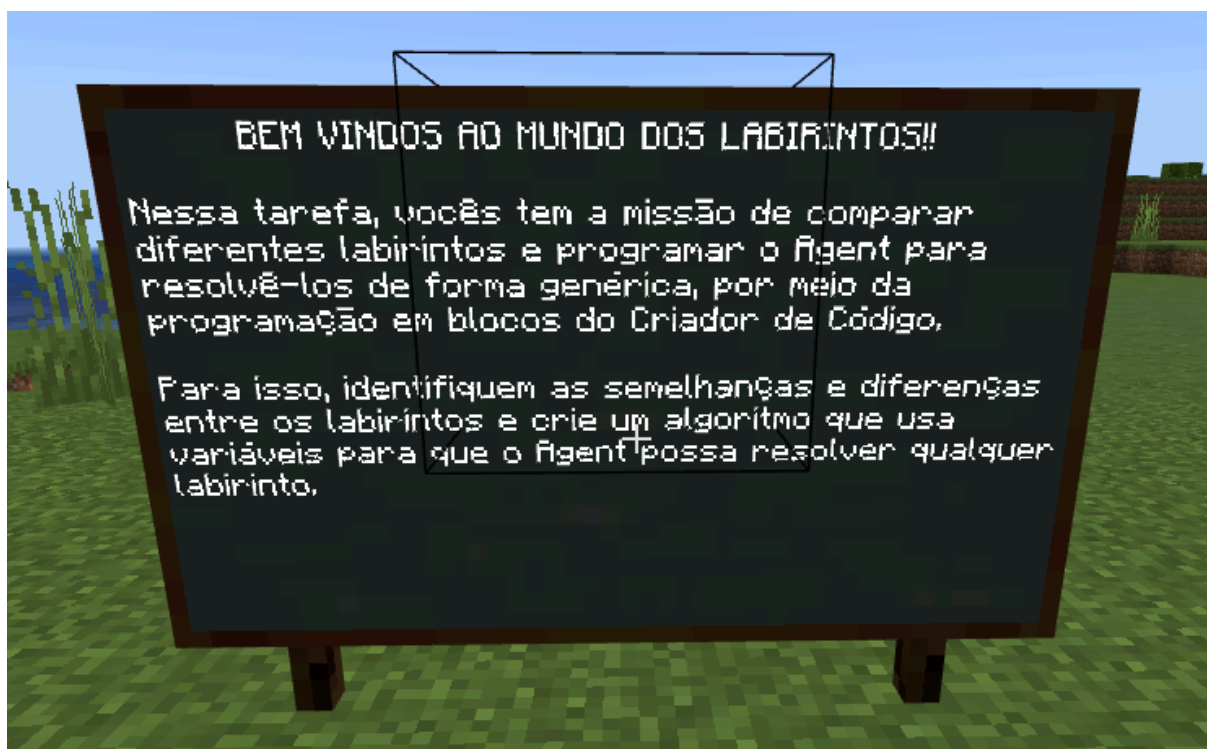


Figura 1 - Quadro informativo com instruções e objetivo da aula.

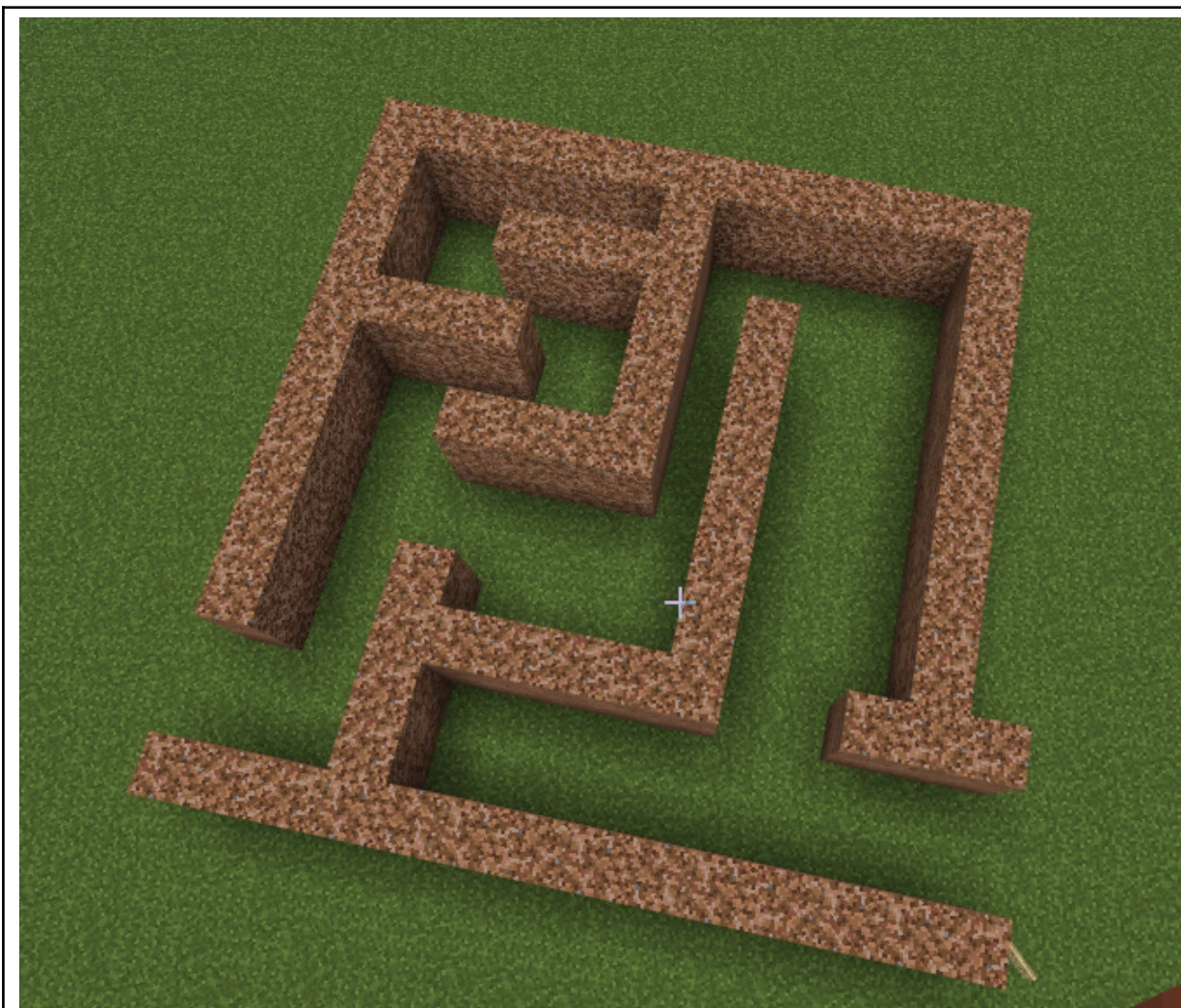


Figura 2 - Primeiro labirinto da tarefa.

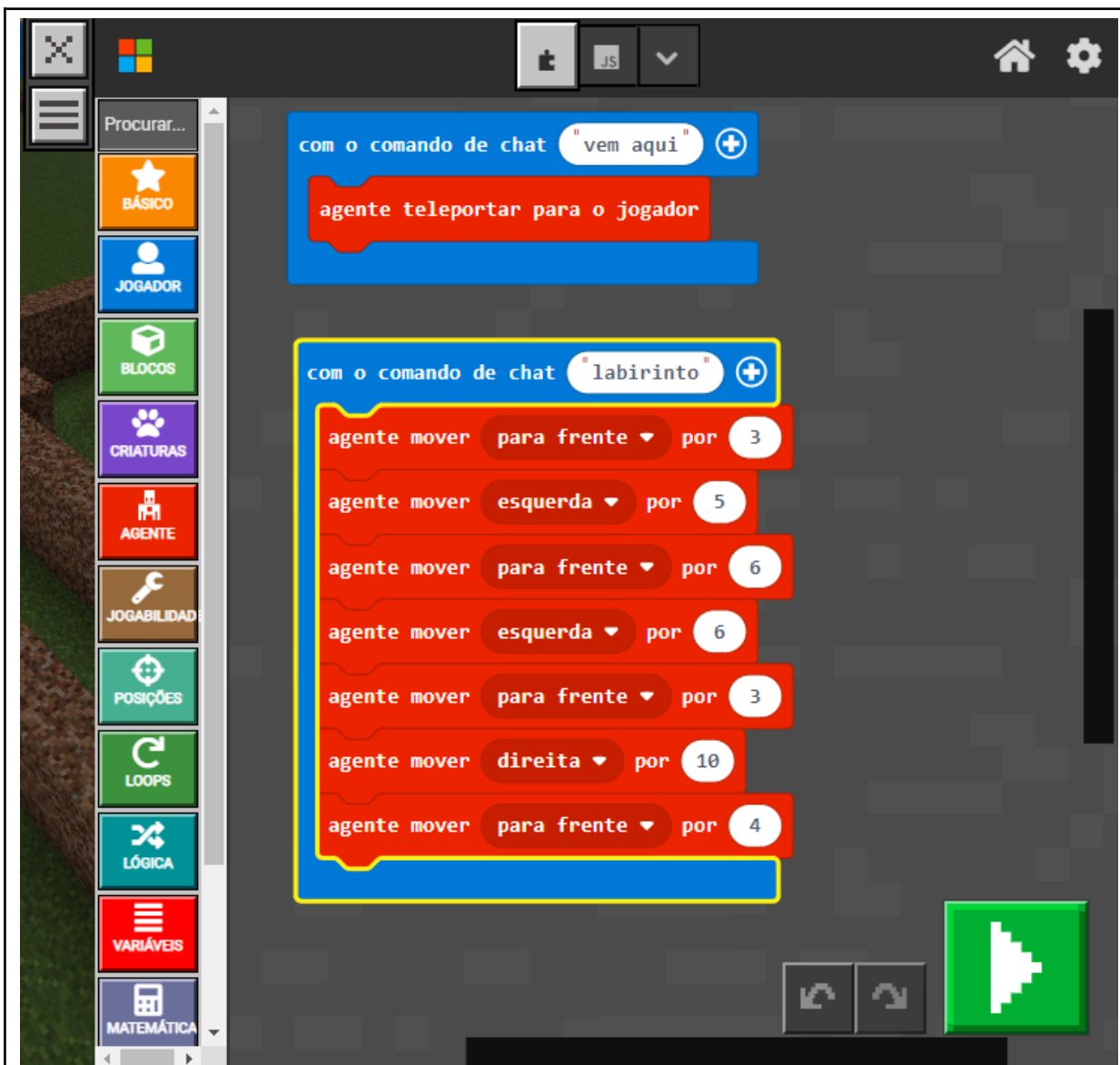


Figura 3 - Exemplo de código para a execução do primeiro labirinto.





Figura 4 - Segundo labirinto da tarefa.

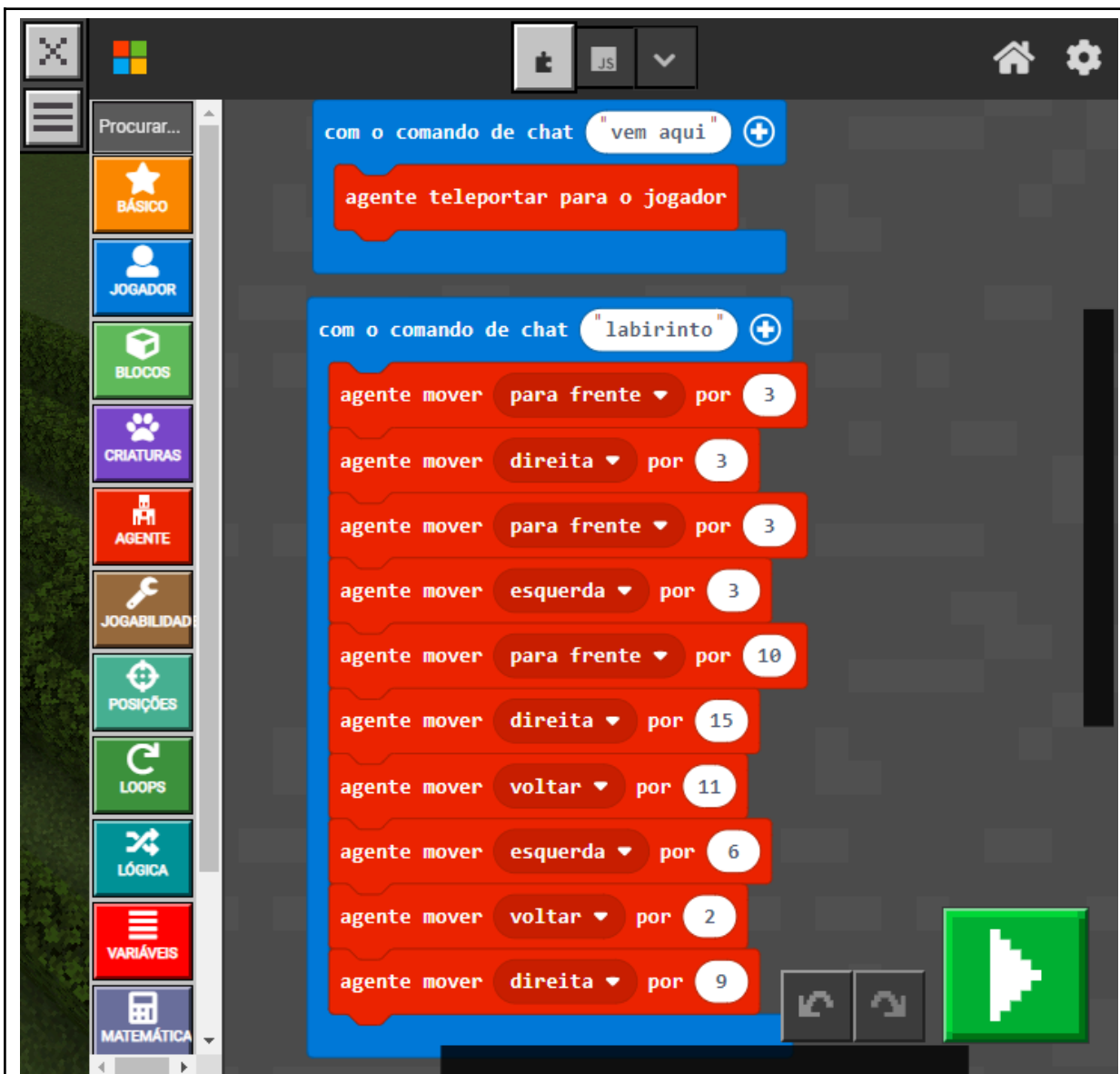


Figura 5 - Exemplo de código para execução do segundo labirinto.



Figura 6 - Exemplo de código para execução de qualquer um dos labirintos.

#### MATERIAL E RECURSOS NECESSÁRIOS

- Um computador com a ferramenta do Minecraft Educacional instalada e em pleno funcionamento, utilizando o Mundo da Aula dos Labirintos.

#### REFERÊNCIAS

WIKI. Minecraft Wiki. Disponível em <https://minecraft.fandom.com/pt/wiki/>