



UNIVERSIDADE FEDERAL DO MARANHÃO
BACHARELADO INTERDISCIPLINAR EM CIÊNCIA E TECNOLOGIA
EECP0008 - INTELIGENCIA ARTIFICIAL

JOSE NUNES DE SOUSA NETO

ALGORITMO DE LABIRINTO E BFS

SÃO LUÍS - MA
DEZEMBRO/2024

JOSE NUNES DE SOUSA NETO (2022003263)

ALGORITMO DE LABIRINTO E BFS

Documento apresentado como requisito parcial de avaliação da disciplina Inteligência Artificial - Turma 02, no curso Bacharelado Interdisciplinar em Ciência e Tecnologia da Universidade Federal do Maranhão.

Orientador: Prof. Dr. Thales Levi Azevedo Valente.

SÃO LUÍS - MA

DEZEMBRO/2024

SUMÁRIO

1 INTRODUÇÃO.....	4
2 OBJETIVOS.....	4
2.1 GERAÇÃO DE LABIRINTOS.....	4
2.2 BUSCA DE CAMINHOS COM BFS.....	4
2.3 SIMULAÇÃO DE AGENTE.....	5
2.4 EDUCAÇÃO E APRENDIZADO.....	5
3 TECNOLOGIAS UTILIZADAS.....	5
4 ESTRUTURA DO PROJETO.....	5
5 DETALHAMENTO TÉCNICO.....	6
5.1 CONSTANTES DO SISTEMA.....	6
5.2 CLASSE CELL.....	6
5.3 ALGORITMO DE GERAÇÃO DE LABIRINTO.....	6
5.4 ALGORITMO BREADTH-FIRST SEARCH (BFS).....	7
5.5 SIMULAÇÃO DO AGENTE.....	7
6 RESULTADOS ESPERADOS.....	7
7 DESAFIOS ENCONTRADOS.....	7
8 CONCLUSÃO.....	8

1 INTRODUÇÃO

Este projeto visa desenvolver um labirinto gerado de forma programática utilizando Python e aplicar o algoritmo Breadth-First Search (BFS) para que um agente explore o labirinto. O projeto aborda conceitos fundamentais de Inteligência Artificial (IA) relacionados à geração procedural, algoritmos de busca e visualização interativa.

Este projeto tem como objetivo criar um labirinto gerado programaticamente em Python, utilizando o algoritmo de Backtracking Recursivo, e aplicar o algoritmo Breadth-First Search (BFS) para encontrar caminhos no labirinto. Ademais, um agente será simulado para percorrer o labirinto de acordo com o caminho gerado pelo BFS. O projeto abrange conceitos fundamentais de Inteligência Artificial, algoritmos de busca e visualização interativa, proporcionando uma abordagem prática e visual para aprendizado.

A geração procedural de conteúdo, como labirintos, é amplamente utilizada em jogos e aplicações educacionais. A utilização de algoritmos de busca, como o BFS, para encontrar caminhos em ambientes complexos é um pilar importante da Inteligência Artificial. Este projeto une essas duas áreas para oferecer um exemplo funcional que pode ser usado em vários contextos.

O trabalho consiste em:

1. Gerar um labirinto utilizando o algoritmo de Backtracking Recursivo.
2. Aplicar o algoritmo BFS para encontrar o caminho entre dois pontos do labirinto.
3. Simular um agente que percorre o caminho encontrado pelo BFS.

2 OBJETIVOS

2.1 GERAÇÃO DE LABIRINTOS

Implementar a criação de labirintos utilizando o algoritmo de Backtracking Recursivo e oferecer flexibilidade na escala do labirinto por meio de parâmetros ajustáveis.

2.2 BUSCA DE CAMINHOS COM BFS

Aplicar o algoritmo Breadth-First Search para encontrar o menor caminho entre dois pontos do labirinto.

2.3 SIMULAÇÃO DE AGENTE

Desenvolver a movimentação visual de um agente seguindo o caminho encontrado pelo BFS.

2.4 EDUCAÇÃO E APRENDIZADO

Fornecer um exemplo didático e interativo que auxilie no entendimento de algoritmos de busca e geração procedural.

3 TECNOLOGIAS UTILIZADAS

- Linguagem de programação: Python
- Bibliotecas:
 1. Pygame para a visualização gráfica do labirinto e do agente.
 2. Estruturas de dados básicas, como listas e filas, para implementação dos algoritmos.

4 ESTRUTURA DO PROJETO

O projeto está dividido nas seguintes etapas principais:

1. **Configuração inicial:**
 - Definição de constantes como o tamanho da janela, tamanho das células (tiles) e o número de linhas e colunas do labirinto.
2. **Classe Cell:**
 - Representa cada célula da grade do labirinto.
 - Contém informações sobre coordenadas, paredes (topo, direita, baixo e esquerda) e o estado de visitação.
3. **Algoritmo de geração de labirinto:**
 - Utiliza Backtracking Recursivo para gerar o labirinto.
 - Remove paredes entre células adjacentes de forma controlada.
4. **Busca de caminhos com BFS:**
 - Implementação do algoritmo BFS para encontrar o caminho mais curto entre dois pontos no labirinto.
 - Garante que o caminho encontrado respeite as paredes do labirinto.

5. Simulação do agente:

- O agente percorre o caminho encontrado pelo BFS com animações e destaques visuais.

5 DETALHAMENTO TÉCNICO

5.1 CONSTANTES DO SISTEMA

- **Tamanho da janela:**
 - Define as dimensões da área de desenho do labirinto.
 - Por exemplo: 800x800 pixels.
- **Tamanho das células:**
 - Define o tamanho de cada quadrado do labirinto (ex.: 40x40 pixels).
- **Número de linhas e colunas:**
 - Calculado dividindo o tamanho da janela pelo tamanho das células.

5.2 CLASSE CELL

A classe `Cell` representa cada unidade do labirinto e possui as seguintes propriedades e métodos:

- **Propriedades:**
 - `x, y`: Coordenadas da célula na grade.
 - `walls`: Um dicionário que indica se as paredes (topo, direita, baixo, esquerda) estão presentes.
 - `visited`: Um booleano que indica se a célula foi visitada.
- **Métodos:**
 - `draw()`: Desenha as paredes da célula com base no dicionário `walls`.
 - `highlight()`: Destaca a célula atual.

5.3 ALGORITMO DE GERAÇÃO DE LABIRINTO

O algoritmo de Backtracking Recursivo segue os seguintes passos:

1. Começa na célula inicial da grade.
2. Marca a célula atual como visitada.
3. Seleciona um vizinho não visitado de forma aleatória.
4. Remove as paredes entre a célula atual e o vizinho.
5. Repete o processo para o vizinho.
6. Retorna para a célula anterior caso todos os vizinhos tenham sido visitados.

5.4 ALGORITMO BREADTH-FIRST SEARCH (BFS)

O BFS é usado para encontrar o caminho mais curto no labirinto. Ele funciona assim:

1. Adiciona a célula inicial a uma fila.
2. Explora as células adjacentes, verificando se as paredes permitem a passagem.
3. Marca as células visitadas e registra o caminho.
4. Para quando a célula destino é encontrada.

5.5 SIMULAÇÃO DO AGENTE

A simulação visual do agente inclui:

1. Animação da movimentação do agente ao longo do caminho.
2. Destaque do caminho percorrido pelo agente.
3. Ajustes visuais para tornar a experiência interativa.

6 RESULTADOS ESPERADOS

- Um labirinto gerado de forma aleatória e visualizado graficamente.
- Caminho corretamente calculado pelo algoritmo BFS.
- Simulação clara e animada do agente percorrendo o labirinto.

7 DESAFIOS ENCONTRADOS

- Ajustar a remoção de paredes para garantir a consistência do labirinto.
- Garantir a eficiência do BFS em labirintos de maior escala.
- Sincronizar a animação do agente com os tempos de renderização do pygame.

8 CONCLUSÃO

Este projeto apresentou uma integração prática entre geração procedural de labirintos, algoritmos de busca e visualização interativa, demonstrando como conceitos teóricos de Inteligência Artificial podem ser aplicados de forma criativa. A implementação do algoritmo de Backtracking Recursivo permitiu a criação de labirintos únicos, enquanto o Breadth-First Search destacou sua eficiência ao encontrar o menor caminho em um ambiente com restrições. Por fim, a simulação do agente navegando pelo labirinto proporcionou uma visualização clara do funcionamento dos algoritmos, reforçando seu valor educacional e ilustrativo. Este trabalho não apenas consolida conhecimentos em IA e programação, mas também abre portas para explorações futuras, como a aplicação de outros algoritmos de busca ou a adaptação do projeto a contextos mais complexos.