



Uma visão acerca do uso da Teoria dos Grafos para a resolução de labirintos

Projeto Integrador 3B - Grupo C

Instituto de Educação Superior de Brasília - IESB

Autores: Aaron Rodrigues, Felipe Maciel e Lucas Mendonça

Orientador: Felipe Giuliani

Introdução e Objetivos Gerais



- Inspirações
 - Contexto histórico da origem de labirintos;
 - *Micromouse Competitions* - competições de mini-robôs.
- Objetivos gerais
 - Construir um aprendizado acerca da Teoria dos Grafos, sobre a linguagem de programação Python e como eles, juntos, podem ser usados para desenvolver uma aplicação que solucione labirintos.

Objetivos específicos



- Objetivos específicos
 - Realizar a chamada de uma API de forma eficiente;
 - Desenvolver algoritmos de busca em profundidade (DFS) e em largura (BFS) para tornar a aplicação capaz de encontrar o caminho com a menor quantidade de movimentações possíveis entre a entrada e a saída do labirinto;
 - Testar, com ajuda da API, a aplicação desenvolvida para garantir sua eficácia;
 - Expor as conclusões tomadas a partir dos resultados obtidos durante a pesquisa.

Metodologia



- Levantamento bibliográfico - foram verificados trabalhos correlatos para melhor embasamento da pesquisa;
- Linguagem de programação escolhida - Python;
 - Consumo de uma API (disponibilizada através da plataforma GitHub) para a criação e exploração dos labirintos (grafos);
 - Implementação de algoritmos de busca:
 - DFS;
 - BFS.
- Ampla testagem da aplicação desenvolvida para garantir sua acurácia e performance.

Referencial Teórico

- Teoria dos Grafos
 - Grafo - conjunto finito de pontos, denominados vértices, unidos por “pontes” denominadas arestas;
 - Não possui algum significado geométrico - seu propósito é representar relações de adjacência entre vértices.
- API - Interface de Programação de Aplicações
 - Funciona como um “correio”, que recebe requisições de um usuário e as transmite a um servidor que as processa e devolve ao cliente.

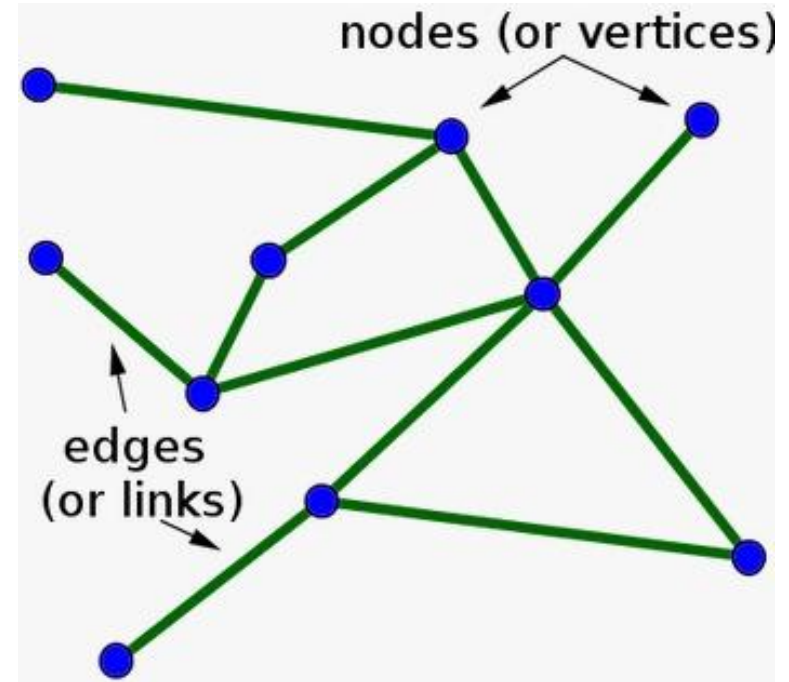


Figura 1: representação visual de um grafo. Fonte: <https://manoelcamilo.wixsite.com/analise/single-post/2017/06/25/desenho-de-grafos-fazendo-nuvens-de-dados-ganharem-significado-visual>

Referencial Teórico

- DFS - *Depth First Search*
 - Busca o nó mais fundo;
 - ideal para reconhecimento do formato do grafo;
 - Pilha LiFo (*Last in, First out*) - estrutura de dados em que o último elemento a ser inserido é sempre o primeiro a ser retirado.

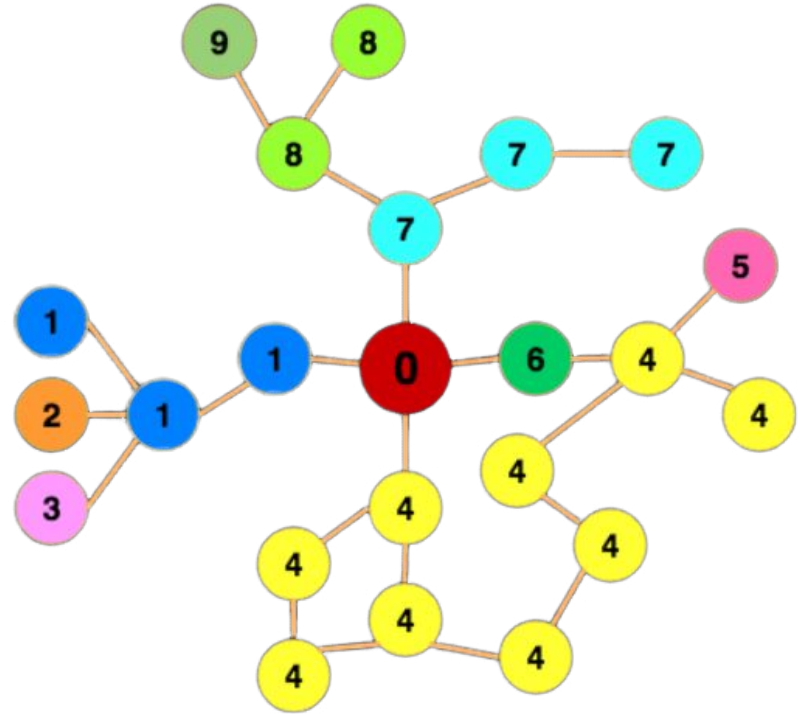


Figura 2: representação visual da atuação de um DFS. Fonte: <https://blog.pantuza.com/artigos/busca-em-profundidade>

Referencial Teórico

- BFS - *Breadth First Search*
 - Busca em largura;
 - Ideal para obter o menor caminho entre dois vértices quaisquer;
 - Fila FiFo (*First in, First out*) - estrutura de dados em que o primeiro elemento a ser inserido é sempre o primeiro a ser retirado.

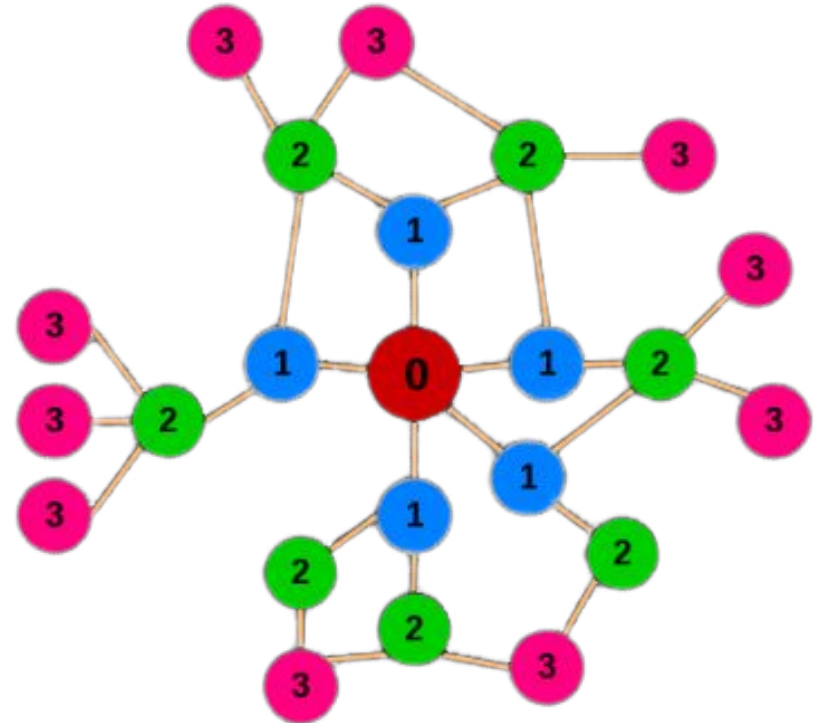


Figura 3: representação visual da atuação de um BFS. Fonte: <https://blog.pantuza.com/artigos/busca-em-largura>

Desenvolvimento



- Chamada da API
 - Endpoints
 - Iniciar - começa a exploração;
 - Movimentar - mostra as movimentações possíveis e movimenta o algoritmo dentro do grafo;
 - Valida_caminho - Imprime os movimentos executados e valida o caminho (verifica se efetivamente vai do início ao fim).

Desenvolvimento



- Foram criadas 4 funções para o desenvolvimento da aplicação:
 - Obter_labirinto - obtém os labirintos disponíveis na API, os valida e exibe seus nomes em um “**arquivo.json()**”;
 - Iniciar_labirinto - inicia a exploração do labirinto, com uma flag marcando que o usuário está no seu início;
 - Movimentar_labirinto - move o usuário dentro do labirinto selecionado;
 - Momento em que atuam o DFS e o BFS, respectivamente.
 - Validar_caminho - requisição final que verifica, junto à API, se o caminho percorrido pela aplicação é válido.

Resultados e Conclusões



- DFS - perde eficiência em situações que precisa retirar e inserir grandes quantidades de elementos de sua pilha;
 - Labirintos que possuem ramos com muitos nós que seguem em uma direção específica.
- BFS - eficiente em encontrar a saída dos labirintos com o menor número de movimentações possíveis.
 - Especialmente quando a saída se encontra em um nó próximo à entrada.



Obrigado!

Instituto de Educação Superior de Brasília - IESB

Autores: Aaron Rodrigues, Felipe Maciel e Lucas Mendonça

Orientador: Felipe Giuliani