

Universidade Católica Dom Bosco

Engenharia da Computação

Luan Henrique Santos Miranda
João Henrique Schweitzer Rezende

Projeto Caminho Mínimo
Google Maps

Campo Grande – MS

2023

Luan Henrique Santos Miranda
João Henrique Schweitzer Rezende

Projeto Caminho Mínimo Google Maps

Projeto desenvolvido para apresentação na
disciplina Matemática Discreta do curso de
Engenharia da Computação, da
Universidade Católica Dom Bosco.

Professor: Hemerson Pistori

Campo Grande – MS

2023

Relatório de Desenvolvimento – Google Maps

Introdução

O código Python fornecido é uma implementação de um sistema para manipulação e análise de grafos não direcionados. Ele utiliza a biblioteca Pyvis e Tkinter para criar, visualizar e calcular informações sobre grafos.

Desenvolvimento

O Projeto foi desenvolvido com o auxílio de inteligências artificiais como Chat GPT 3.5 e Black Box, que juntas foram de extrema importância para o desenvolvimento do projeto.

No início do projeto foi analisado o código disponibilizado pelo professor regente [Hemerson Pistori](#) para termos um caminho a seguir, após isto podemos rapidamente entender o que falta no algoritmo para torná-lo mais eficiente e intuitivo. Por meio da biblioteca *Tkinter* foi construído uma interface simples e intuitiva, onde o usuário pode informar a “*Origem*” e o “*Destino*”, e rapidamente já é mostrado o caminho mínimo.

Após isto, percebendo que faltava algo, por meio da biblioteca *Pyvis* é mostrado o grafo já previamente montado, para facilitar a visualização do caminho mínimo e a demonstração do grafo.

Estrutura do Grafo

O grafo é representado por duas estruturas principais: uma lista de vértices (*vertices*) e uma lista de arestas (*arestas*). Cada aresta é uma tupla contendo o par de vértices conectados e o peso associado àquela conexão. A inclusão de pesos nas arestas permite a realização de análises de caminhos mínimos.

Funcionalidades Principais

Visualização Gráfica: Utilizando a biblioteca Pyvis, o código gera representações visuais do grafo completo com seus vértices e arestas. A visualização inclui os pesos das arestas para uma melhor compreensão.

Encontrar Caminho Mínimo: Implementa o algoritmo de Dijkstra para encontrar um caminho mínimo entre dois vértices fornecidos pelo usuário. A função *caminho()* calcula o caminho mínimo entre dois vértices do grafo, retornando uma lista de vértices que formam esse caminho.

Componentes Principais do Código

Estruturas de Dados: O código utiliza listas para armazenar informações sobre vértices e arestas. Também faz uso de dicionários para representar o custo associado a cada vértice.

Manipulação do Grafo: Funções como *vizinhos()* retornam os vértices vizinhos a um vértice dado. *custo()* encontra o peso de uma aresta que liga dois vértices. *recupera_caminho()* retorna o caminho percorrido entre dois vértices a partir de uma lista de antecessores.

Interface Gráfica: Há uma interface gráfica simples criada com Tkinter. Ela permite que o usuário insira os vértices de origem e destino para encontrar o caminho mínimo.

Cronograma de Desenvolvimento

Foi utilizado o algoritmo de Dijkstra, para encontrar um caminho mínimo, porém, acrescentamos uma interface para o programa, utilizando a biblioteca **tkinter**, e foram adicionadas outras funcionalidades, sendo elas:

- def imprimir_caminho_minimo(): desenvolvida por João
 - 09/11/2023
- def eh_conexo():desenvolvida por João
 - 09/11/23
- def criar_interface_grafica(): desenvolvida por Luan
 - 13/11/23
- def encontrar_caminho(): desenvolvida por João
 - 13/11/23
- def desenhar_grafo():desenvolvida por Luan
 - 13/11/23

Modo de uso

Para utilizar o projeto, siga os passos abaixo:

Certifique-se de ter o “*conda*” instalado, após isto de seguinte comando no terminal (base) para permitir que o arquivo bash seja executado de maneira correta:

```
chmod +x caminho.sh
```

Ainda no terminal o usuário deve dar o seguinte comando, iniciando assim, todo o processo de maneira automática, desde a criação do ambiente conda, instalação das dependências, até o processamento do código:

```
./caminho.sh
```

Já na execução do programa, ele é bem intuitivo, precisando somente que o usuário dê dois dados, um de entrada e outro de saída, de acordo com os vértices que estão disponíveis para verificação, caso o usuário clique na opção “Mostrar Grafo”, possibilitando assim, que o usuário decida, qual será a origem e destino.

Conclusão

O Projeto apresenta uma gama de funcionalidades para manipulação e análise de grafos. Ele oferece uma visualização gráfica dos grafos completos e dos caminhos mínimos encontrados, além de permitir interação por meio de uma interface gráfica básica.

O tempo estimado para a finalização de todas as etapas do projeto é de 24 horas.

Link do vídeo do programa sendo executado:

<https://www.youtube.com/watch?v=QtmdB-5WVpw>