



UNIVERSIDADE
FEDERAL
DE PERNAMBUCO



**Centro de
Informática**
UFPE

**Universidade Federal de Pernambuco
Centro de Informática - CIn**

Disciplina: Algoritmo e Estrutura de Dados

Docente: Sérgio Ricardo de Melo Queiroz

Discentes: Erick Daniel Alves de Lima
Maiara da Silva Lira

Relatório do projeto: Grupo #2.6
Encontrando o menor tempo com Bellman-Ford

Recife, 19 de abril de 2023

1. Contexto do problema

Nesse projeto utilizamos o algoritmo de Bellman-Ford, que consiste em ser um algoritmo de caminho que gera a árvore de custo mínimo ou de menor distância em grafos ponderados que podem possuir arestas negativas e positivas. Basicamente, ele encontra o caminho mais curto de um vértice de origem para todos os outros vértices do grafo. Se o grafo contiver um ciclo de peso negativo, o algoritmo detectará esse ciclo. Escolhemos como base de dados a “CollegeMsg temporal network” que é uma base de dados composta por mensagens privadas enviadas em uma rede social online da Universidade da Califórnia, em Irvine. Os usuários podem procurar outras pessoas na rede e iniciar uma conversa com base nas informações do perfil. Uma borda (u, v, t) significa que o usuário u enviou uma mensagem privada para o usuário v no tempo t . Dessa forma, usamos u e v como vértice ou nó do nosso grafo e t como o peso dele.

Dados relevantes sobre a base de dados:

Estatísticas do conjunto de dados	
Nós	1899
bordas temporais	59835
Bordas no gráfico estático	20296
Intervalo de tempo	193 dias

Figura 1. Tabela referente aos valores da Base de Dados

Exemplo de leitura dos dados: A primeira coluna traz o usuário que enviou a mensagem e a segunda o usuário que recebeu, a terceira, por sua vez, traz o tempo em segundos:

```
1 2 1082040961
3 4 1082155839
5 2 1082414391
```

Figura 2. Representação da ordenação dos dados

Nesse caso o usuário 1 enviou uma mensagem ao usuário 2 em 1082040961 o que corresponde a cerca de 12.53 dias, aproximadamente.

2. Implementação

2.1 Algoritmo utilizado: Bellman-Ford em um contexto real de aplicação

2.2 Desenvolvimento

Primeiramente, o projeto foi desenvolvido utilizando o Google Colab para um melhor desempenho devido a base de dados ser relativamente grande. A ideia por trás do projeto consistiu em encontrar o menor tempo possível, através da árvore de custo mínimo que é gerada pelo algoritmo, em que um usuário A enviou uma mensagem para outro usuário sabendo que eles podem ter trocado mensagens em múltiplas vezes e que o usuário A enviou mensagens para diversas pessoas.

Somando a isso, elaboramos o código objetivando a implementação do algoritmo Bellman-Ford para encontrar o caminho mais curto em um grafo direcionado com pesos, em que os pesos são representados pelo tempo e o caminho mais curto entre eles significa o tempo mais curto de envio de uma mensagem entre os usuários.

```
Digite o vértice de início: 99
Digite o vértice de destino: 8
o Peso dessa aresta é 1082630832
```

Figura 3. Exemplo de valores esperados ao final da execução do algoritmo

1. A def "bellman_ford" é a função que implementa o algoritmo de Bellman-Ford para encontrar o menor caminho entre dois vértices. Ele recebe o grafo, o vértice de início e o vértice de destino como entrada e retorna o menor caminho e o peso da aresta que os conecta.

2. O segundo bloco do código é relativo a abertura do arquivo de texto, que é a base de dados, que é utilizado na construção do grafo. Ele lê cada linha do arquivo e armazena os vértices e arestas em uma lista.
3. Ao usarmos “import sys”, “import networkx as nx” e “import matplotlib.pyplot as plt”, estamos importando as bibliotecas necessárias para executar o código: sys para definir o valor máximo para a distância inicial e networkx e matplotlib para visualizar o grafo.
4. Basicamente, o programa realiza três tarefas principais: calcular o menor caminho entre dois vértices em um grafo ponderado e exibir uma visualização gráfica do grafo usando as bibliotecas NetworkX e Matplotlib. Já utilizando a Tkinter conseguimos construir a GUI, O método `__init__` é o construtor da Classe Application e ele define os elementos da GUI, incluindo labels, rótulos, campos de entrada e botões.
5. O método “programa” é o responsável por ler os dados do arquivo de entrada, no caso nosso banco de Dados, “CollegeMsg.txt”, e por executar o algoritmo de Bellman-Ford para encontrar o menor caminho entre os vértices inseridos pelo usuário. Ele também atualiza a GUI com o resultado da execução do algoritmo.

3. Bibliotecas Utilizadas

- Matplotlib

Utilizada como auxílio para criação e visualização dos dados em grafo.

- Networkx

Utilizada em conjunto com a biblioteca matplotlib para criação da representação do grafo.

- Tkinter

Utilizada para elaboração da interface de execução do programa.

Conclusão

O programa trata o banco de dados e recebe como entrada através da interface gráfica o vértice de início e o vértice de destino. A saída retorna o menor percurso entre os dois e seu respectivo peso, que no caso é o menor tempo entre estes dois vértices.

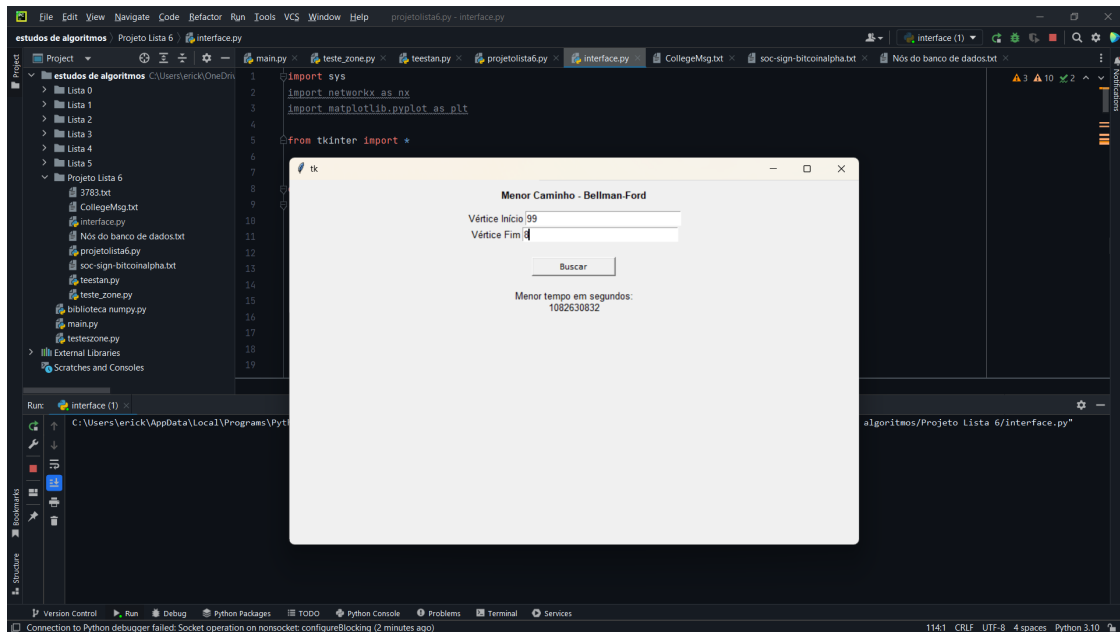


Figura 4. Execução do programa no Pycharm

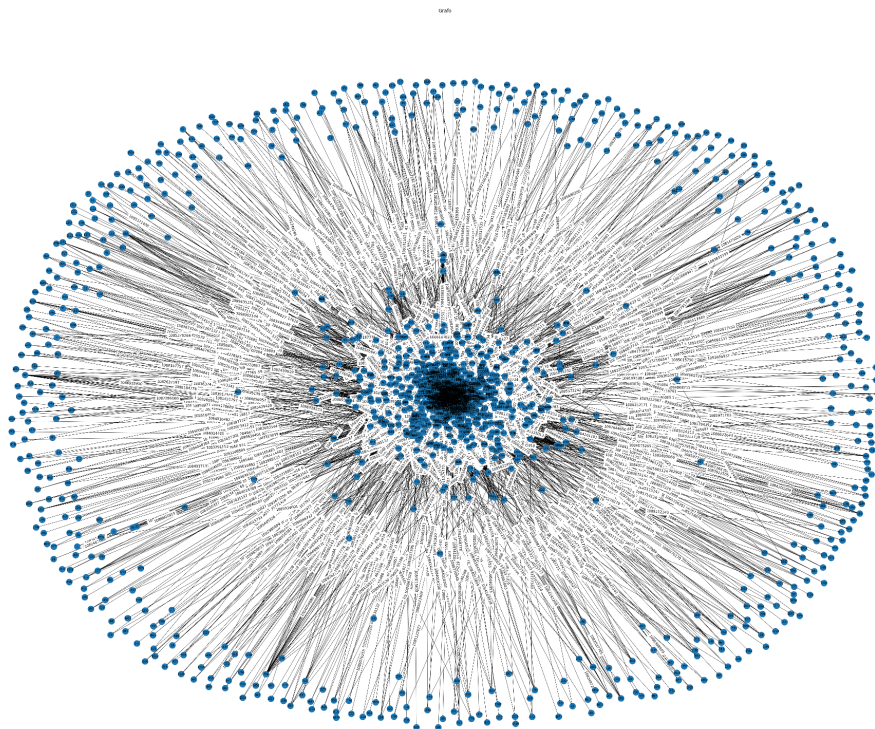
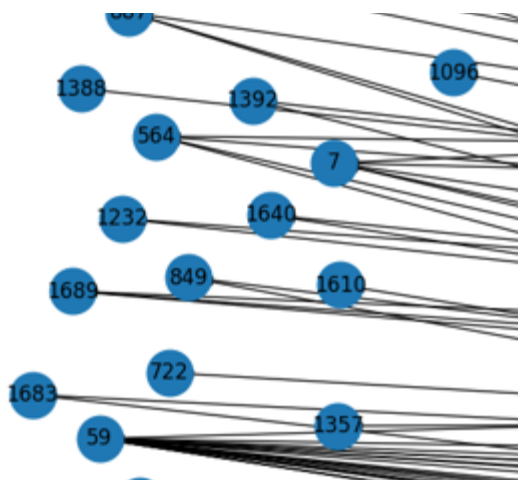
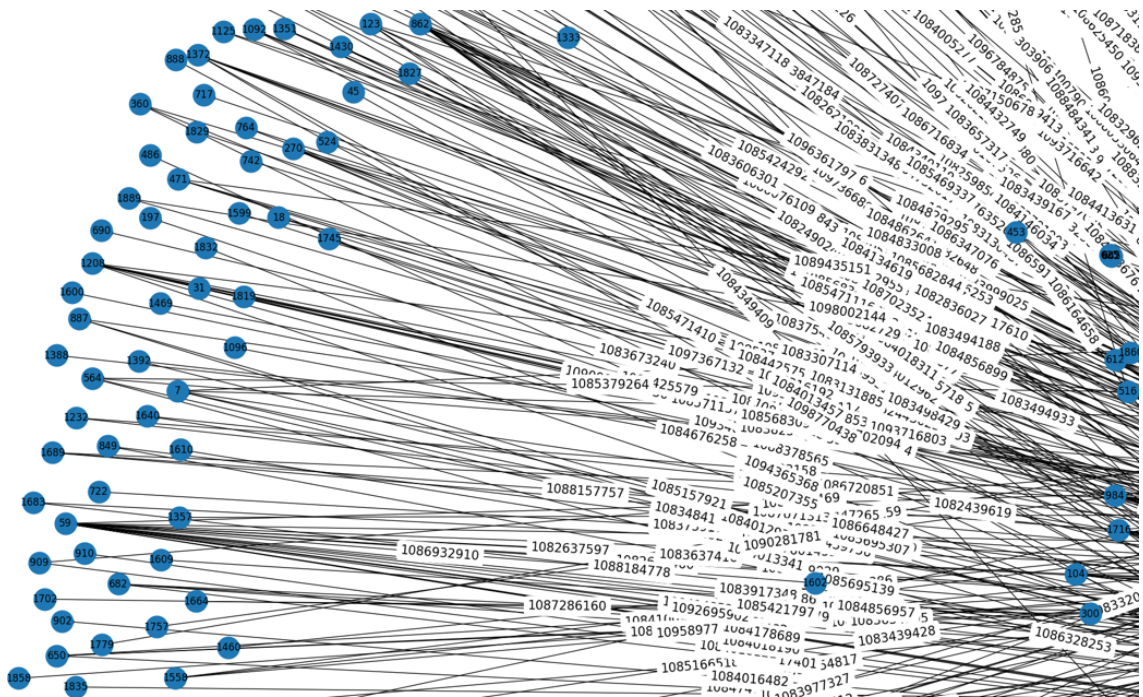


Figura 5. Representação do Grafo - Execução no Google Colab



Referências

- [1] "Matplotlib Documentation," 2020. <https://matplotlib.org/stable/contents.html>. [Acessado: Abr. 12, 2023].
- [2] Site oficial networkx. *networkx.github.io*. «[Introduction — NetworkX 1.7 documentation](#)». [Acessado: Abr. 12, 2023].
- [3] Site oficial tkinter. [tkinter — Python interface to Tcl/Tk — Python 3.11.3 documentation](#). [Acessado: Abr. 12, 2023].
- [4] Site do banco de dados. [SNAP: Network datasets: CollegeMsg temporal network \(stanford.edu\)](#). [Acessado: Abr. 12, 2023].