Relatório T5 - INF1010 - Estrutura de Dados Avançados

Componentes do grupo:

Diego Miranda -2210996

Eric Goulart - 2110878

**INTRODUÇÃO:**

Neste trabalho fizemos um programa que implementa na linguagem C um grafo e utilizando o algoritmo de Dijkstra calcula o caminho mais curto para vértices a partir do vértice A. Depois o programa faz uma busca em profundidade a partir do vértice A.

**DESCRIÇÃO:**

Primeiro definimos as três estruturas. A struct **“no**” representa um vértice do grafo. A struct “**lista”** representa a lista de adjacências e a struct “**grafo**” que representa o grafo.

Nossas funções são:

* **inicializarGrafo**

Inicializa o grafo, recebe como parâmetro a struct Grafo, o número de vértices e os vértices (representados em um array de char).

* **indiceVertice**

Utilizamos essa função como uma busca, para verificar se determinado vértice está no grafo, caso ele esteja a função retorna o índice, senão retorna -1.

* **adicionarAresta**

Função para adicionar uma aresta entre dois vértices.

* **distanciaMinima**

Função que é responsável pelas distâncias dos vértices até o vértice escolhido. Todas começam como infinito e à medida que percorremos o grafo vamos atualizando as distâncias.

* **dijkstra**

Função principal para o algoritmo de dijkstra, utiliza da função ‘distanciaMinima’ para percorrer o grafo e ir calculando as distâncias..

* **profundidadeRecursivo**

Função auxiliar para a busca em profundidade que vai percorrendo os vértices e marcando eles como visitados.

* **buscaEmProfundidade**

Essa função inicializa um array de visitados e chama a função ‘profundiadeRecursivo’ para realizar a busca em profundidade.

* **main**

Aqui primeiro criamos a nossa struct grafo, definimos o número de vértices e definimos quais serão esses vértices por um array de char. Depois disso chamamos a função para inicializar o grafo e chamamos várias vezes a função de adicionar aresta passando os vértices a serem conectados e o peso dessa aresta. Por fim chamamos as funções de dijkstra para calcular a menor distância dos vértices para o vértice A e a função busca em profundidade para realizar uma busca em profundidade a partir do vértice A..

**OBSERVAÇÕES:**

Conseguimos entender bem a ideia desse trabalho e realizar o que foi pedido sem muitas dificuldades. Um aspecto a ser destacado é que no começo nos embolamos um pouco pelo vértices irem de F para H pulando o G, mas conseguimos resolver rapidamente.

**Lista de adjacências:**

A-> D-2 / C-4 / B-5

B-> H-9 / E-6 / C-6 / A-5

C-> E-4 / D-3 / B-6 / A-4

D-> F-9 / E-5 / C-3 / A-2

E-> H-6 / F-2 / D-5 / C-4 / B-6

F-> H-3 / E-2 / D-9

H-> F-3 / E-2 / B-9