

**Universidade Federal do Agreste de Pernambuco**

Claudierio Baltazar Barra Nova

Henrique de Almeida Silva

Robert Freire de Melo

Sistema de recomendação de rotas utilizando grafos

Garanhuns/PE

2023

Claudierio Baltazar Barra Nova

Henrique de Almeida Silva

Robert Freire de Melo

Sistema de recomendação de rotas utilizando grafos

Trabalho apresentado ao professor Igor Medeiros da disciplina de Algoritmos e estrutura de dados 2, com o intuito de aprofundar nosso conhecimento em Grafos.

Garanhuns/PE

2023

## **1.Definição**

Grafo é um tipo de estrutura de dados utilizada para representar relacionamentos entre pares de objetos. A utilização dessa estrutura e modelagem para a área da computação foi essencial para soluções de diversos problemas teóricos e práticos, como otimização de rotas em sistemas de transporte, mecanismos de pesquisas e de classificação e análise de redes sociais, pois podem ser reduzidos em propriedades dos grafos e ser aplicados em algoritmos que utilizam a definição dessa estrutura.

## **2.Contextualização**

A teoria de grafos é uma ferramenta matemática que pode ser aplicada em diversas áreas, inclusive na análise do valor das rotas entre cidades. Por meio da representação de um conjunto de cidades e das conexões entre elas, podemos criar um modelo de grafo para encontrar a menor rota possível entre a cidade de origem e destino. Os vértices do grafo representam as cidades e as arestas representam a relação entre elas.

Utilizando esse modelo, é possível criar um sistema de recomendação de rotas que, por meio da análise da distância entre as cidades, é capaz de comparar as rotas possíveis e recomendar a de menor caminho. Nesse caso, a distância entre as cidades pode ser utilizada para calcular o valor da passagem, considerando que o preço é de 0,50 centavos por km percorrido.

Ao encontrar a menor rota, o sistema exibe todo o percurso que a rota fará, incluindo as cidades que serão visitadas e a distância entre elas, bem como o valor final da passagem. Essa análise pode ser muito útil para empresas de transporte e agências de viagem, que podem utilizar essas informações para otimizar suas rotas e oferecer um serviço mais eficiente e econômico para seus clientes.

### **3.Características do Grafo**

Um grafo é uma estrutura de dados fundamental na programação, sendo amplamente utilizado em diversos campos, como algoritmos de redes, algoritmos de busca, algoritmos de caminho mais curto, entre outros. O grafo é composto por um conjunto de vértices e arestas que os conectam, e cada vértice pode ser visto como um objeto ou entidade, enquanto as arestas representam as relações ou conexões entre esses objetos.

Algumas das principais características dos grafos na programação incluem:

**Conectividade:** A conectividade de um grafo se refere à capacidade de se mover de um vértice para outro por meio de uma ou mais arestas. Um grafo pode ser completamente desconectado, possuindo várias componentes isoladas, ou pode ser totalmente conectado, o que significa que é possível percorrer de qualquer vértice para qualquer outro vértice através de uma ou mais arestas.

**Direcionalidade:** Os grafos podem ser direcionados ou não-direcionados. Em um grafo não-direcionado, as arestas não têm uma direção definida e podem ser percorridas em ambas as direções. Já em um grafo direcionado, as arestas têm uma direção definida, e é possível percorrer apenas na direção especificada.

**Peso:** Algumas aplicações exigem que cada aresta em um grafo tenha um peso associado. O peso de uma aresta pode representar uma variedade de fatores, como a distância física entre dois vértices, o custo de percorrer uma aresta ou o tempo necessário para percorrer uma aresta. Os pesos das arestas podem ser usados em algoritmos de caminho mais curto, algoritmos de rede, entre outros.

**Ciclicidade:** Um ciclo em um grafo é um caminho que começa e termina no mesmo vértice, percorrendo várias arestas e vértices no caminho. Um grafo é considerado cíclico se contiver pelo menos um ciclo. Caso contrário, é considerado acíclico. Grafos acíclicos são úteis em diversas aplicações, como em algoritmos de ordenação topológica.

**Representação:** Os grafos podem ser representados de diversas formas, como por exemplo, matrizes de adjacência, listas de adjacência, matriz de incidência, entre outras. A representação escolhida dependerá do objetivo da aplicação e das operações que serão realizadas no grafo.

#### 4. Grafo escolhido para resolução do problema

Na ciência da computação, o algoritmo de Floyd-Warshall é um algoritmo que resolve o problema de calcular o caminho mais curto entre todos os pares de vértices em um grafo orientado e valorado. No contexto do problema de recomendação de rotas, o grafo representa as cidades e as estradas que ligam cada uma delas, e os pesos nas arestas representam as distâncias entre as cidades. Utilizando o algoritmo de Floyd, é possível encontrar o menor caminho entre todas as cidades do grafo, o que permite encontrar a rota mais curta entre duas cidades quaisquer.

#### 5. Algoritmo em Execução

Para a execução do algoritmo, foi criado um repositório com 50 cidades, e essas cidades são adicionadas como os vértices do grafo, logo após, são criadas as arestas que estabelecem a relação entre as cidades utilizando a distância como o peso do grafo. Com todo o grafo preenchido, chamamos a função Floyd Warshall, onde será a comparação entre as cidades, para que se ache a menor distância entre elas e salvar em uma outra matriz o menor caminho utilizado entre as cidades.

```
Sistema de Recomendação de rotas - CHR Rotas
1 - Imprimir cidades
2 - Imprimir a Matriz
3 - Escolher a viagem
0 - Sair
```

Imagem 1: print da função menu (classe View)

0 - Garanhuns	25 - Ibirajuba
1 - Paranatama	26 - Cachoeirinha
2 - Saloá	27 - Altinho
3 - Bom Conselho	28 - Cupira
4 - Meladinho	29 - Bonito
5 - Iati	30 - Batateira
6 - Aguas Belas	31 - Canhotinho
7 - Santo Antônio	32 - Angelim
8 - Garcia	33 - São João
9 - Japocanga	34 - Brejão
10 - Tarci	35 - Catende
11 - Venturosa	36 - Jaqueira
12 - Pedra	37 - Serra Grande
13 - Arcoverde	38 - União dos Palmares
14 - Alagoinha	39 - Branquinha
15 - Buíque	40 - Santana do Mundaú
16 - Caetés	41 - Chá Preta
17 - Capoeiras	42 - Correntes
18 - São Bento do Una	43 - Lagora do Ouro
19 - Queimada Grande	44 - Barra do Brejo
20 - Lajedo	45 - Palmerina
21 - Jupi	46 - Iratama
22 - Jacaré	47 - Quipapá
23 - Jurema	48 - Panelas
24 - Batinga	49 - Agrestina
25 - Ibirajuba	

Imagem 2: saída do código que representa o banco de dados

## 6.Exemplo de execução do Algoritmo

Usando como exemplo, será posto como origem a cidade “Garanhuns”, e a cidade de destino como “Cachoeirinha”, teremos como saída:

```
Informe o nome da cidade de origem:
Garanhuns
Informe o nome da cidade de destino:
Cachoeirinha
Viagem de Garanhus para Cachoeirinha

Caminho: Garanhus -> Batinga -> Jupi -> Lajedo -> Cachoeirinha
Distância: 59,70 Km
Valor da viagem: R$ 29,85
```

Outro exemplo, onde como origem a cidade é “Buíque”, e a cidade de destino “Brejão”, teremos como saída:

```
Informe o nome da cidade de origem:
Buíque
Informe o nome da cidade de destino:
Brejão
Viagem de Buíque para Brejão

Caminho: Buíque -> Arcoverde -> Pedra -> Venturosa -> Tarci -> Caetés -> Garanhus -> Brejão
Distância: 146,00 Km
Valor da viagem: R$ 73,00
```

E por fim, será posto que a cidade origem é “Saloá”, e a cidade de destino “Venturosa”, e terá como saída:

```
Informe o nome da cidade de origem:
Saloá
Informe o nome da cidade de destino:
Venturosa
Viagem de Saloá para Venturosa

Caminho: Saloá -> Paratama -> Garanhus -> Caetés -> Tarci -> Venturosa
Distância: 101,80 Km
Valor da viagem: R$ 50,90
```

## **7. Conclusão**

O algoritmo de Floyd mostrou-se bastante útil na resolução do problema de recomendação de rotas, possibilitando encontrar o menor caminho entre as cidades do grafo. A escolha deste algoritmo se deu pela sua eficiência e simplicidade de implementação. Além disso, a utilização de uma matriz de adjacência permitiu a representação clara do grafo e a manipulação das informações necessárias para a recomendação de rotas.

Em suma, o algoritmo de Floyd é uma ferramenta valiosa para problemas que envolvem a análise de rotas, como em sistemas de transporte ou de logística, permitindo encontrar o caminho mais curto entre dois pontos em um grafo ponderado. A aplicação desse algoritmo pode trazer benefícios significativos em termos de eficiência e economia de tempo e recursos, sendo uma alternativa eficaz para solução de problemas desse tipo.