Trabalho Prático 02 – AEDS 1 – Em Dupla

Professora: Thais R. M. Braga Silva

Valor: 10 pontos

Data de Entrega:

Forma de Entrega: PVANet (formato .zip ou .tar.gz)

O objetivo deste trabalho prático é permitir a avaliação do impacto causado pelo desempenho dos algoritmos em sua execução real. Vimos em sala de aula que existem problemas e algoritmos de complexidade exponencial, chamados de intratáveis. Nesses casos, os programas, ao serem executados podem demorar uma quantidade de tempo não razoável para encontrar uma solução, dependendo do tamanho da entrada. Vamos observar, portanto, como isso ocorre na prática. Para tanto, cada dupla fará uma implementação para o Problema de Roteamento de Veículos (PRV) – versão clássica. Esse é um problema intratável, pois sua solução exata somente é possível através do cálculo e avaliação de todas as possíveis saídas, o que chamamos de força bruta. Em seguida, essa implementação deverá ser executada para diferentes valores de entrada N, e o tempo gasto para que o programa termine, em cada caso, deverá ser medido por meio de comandos do sistema operacional.

PROBLEMA DE ROTEAMENTO DE VEÍCULOS (PRV)

O problema de roteamento de veículos (PRV) é um problema clássico da literatura que pode ser definido como segue. Dado um conjunto de cidades, cada qual com uma demanda por um produto, e um depósito com veículos de capacidade Q_v, encontrar as rotas para os veículos minimizando as distâncias percorridas.

Em linhas gerais, esse problema recebe como entrada N cidades e um conjunto de distâncias (não negativas) entre pares de cidades C_i e C_j (considere que a distância entre C_i e C_j representa a existência de uma estrada entre elas, sendo que a distância entre C_i e C_j é a mesma daquela entre C_j e C_i). As cidades são identificadas por números, que vão de 0 a N-1. Assim o conjunto de cidades é dado por $\{C_0,C_1,C_2,...,C_{N-1}\}$. A cidade C_0 representa o depósito, sendo este a base de uma frota de veículos idênticos de capacidade Q_v. Cada cidade C_i possui uma demanda não negativa Q_i e $Q_0=0$. Desta forma, o problema também recebe como entrada um vetor Q de tamanho N, contendo em cada i-ésima posição a demanda não negativa da cidade i. Neste trabalho supõe-se que o somatório das demandas Q_i de todas as cidades deve ser um múltiplo da capacidade Q_v de um caminhão. Dessa forma, o número de caminhões necessários para atender às demandas de todas as cidades é dado pela divisão entre esse somatório e a capacidade de um caminhão.

O Problema de Roteamento de Veículos consiste em determinar o conjunto de rotas que deverão ser feitas pelos veículos minimizando os custos de transporte, dado pela distância percorrida e respeitando as seguintes restrições:

a) Cada rota começa e termina no depósito;

b) Toda cidade, com exceção do depósito, é visitada somente uma vez por somente um veículo;

c) A demanda total de qualquer rota não deve superar a capacidade Q_v de um veículo.

Uma solução do PRV é representada por meio de uma permutação de cidades, numeradas de 1 a N-1, separadas em tantas partições quantos forem o número de veículos usados. O elemento separador é representado pelo valor zero e indica o depósito.

Cada rota percorrida por um único caminhão é denominada pétala. Por exemplo, se existem 9 cidades para serem atendidas e 4 caminhões disponíveis, então uma possível solução S seria S=[0 7 2 6 0 3 9 5 0 1 0 4 8 0], onde [0 7 2 6 0], [0 3 9 5 0], [0 1 0] e [0 4 8 0] são as pétalas desta rota.

SOLUÇÃO EXATA PRV

Existem diversas possíveis implementações para o PRV. Entretanto, as mais utilizadas requerem recursos de programação que ainda não foram estudados por vocês. Dessa forma, adotaremos uma estratégia mais direta, visto que o objetivo principal do trabalho é a avaliação de desempenho, e não o desenvolvimento do algoritmo. Cada dupla deverá implementar um programa, em linguagem C, para o PRV da seguinte forma:

- Procurar na Web um algoritmo de arranjo para um conjunto de N elementos, X a X. Esse algoritmo deve, portanto, produzir todos os possíveis arranjos de tamanho X entre os N elementos. Você deverá utilizar a linguagem C, portanto, se o algoritmo encontrado estiver em outra linguagem de programação, vocês deverão convertê-lo para linguagem C.

- Para armazenar as distâncias entre as N cidades do PRV, utilizar uma matriz M [NxN]. Para a ligação entre duas cidades i e j, adicionar o valor da distância entre elas nas posições M[i][j] e M[j][i]. As posições M[i][i] da matriz devem receber 0. Exemplo: para N = 3, com distância(0,1) = 5, distância(0,2) = 10, distância (1,2) = 2, a matriz seria:

0	5	10
5	0	2
10	2	0

- Cada cidade deve ser representada pela sua demanda Q_i e por um marcador que indica se a mesma já foi atendida por um caminhão ou não. Lembre-se de que cada cidade deve ser atendida apenas 1 vez.

- O somatório de todas as demandas Q_i deve ser um múltiplo da capacidade de um caminhão Q_v.

- Exclua 0 do conjunto de N cidades, visto que ela será considerada o depósito. Utilize o algoritmo de arranjo para gerar todos os arranjos, de todos os tamanhos possíveis (de tamanho Y até tamanho 1), entre as Y cidades ainda não atendidas por nenhum caminhão.

- Para cada arranjo resultante do passo anterior, utilize os valores do vetor Q para calcular o somatório total de demanda das cidades envolvidas e a matriz M para calcular o custo do caminho. Considere a distância de 0 à primeira cidade e a distância entre a última cidade e 0 nesse cálculo. Exclua os arranjos cuja demanda total for diferente da capacidade de um caminhão e, ao final, escolha o arranjo que possua o menor custo de caminho. Faça isso para cada um dos caminhões necessários para se atendar à demanda de todas as N-1 cidades do problema.

- Execute o programa para entradas com valor N começando em 3, e vá fazendo incrementos de 1. Ao executar o programa, utilize uma ferramenta para medição do tempo de execução, como o comando *time* do Unix.

- Faça um relatório final contendo o código implementado, uma breve explicação do mesmo (indicando como funciona o algoritmo de permutação utilizado, de onde foi obtido, como o seu programa calcula os custos dos caminhos e escolhe o menor). O relatório também deverá conter os resultados dos tempos de execução para os valores de N conforme demandado acima.

- Coloque no seu relatório uma análise sobre os tempos obtidos para cada valor de N testado e, ao fim, faça um gráfico que ilustre o ocorrido. Comente sobre o último valor de N testado por vocês e justifique o motivo pelo qual vocês não avaliaram valores mais altos.

- O arquivo do relatório final deverá ser entregue até a data limite através do PVANet. Lembrem-se de colocar os nomes e matrículas dos dois integrantes da dupla.