

UNIVERSIDADE FEDERAL DE LAVRAS
INSTITUTO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
DEPARTAMENTO DE CIÊNCIAS DA COMPUTAÇÃO
GCC218 — ALGORITMOS EM GRAFOS
Prof. Mayron César de O. Moreira

Nome: Gilson dos Santos Júnior	14A
Gustavo Henrique Marques Magalhães	14A
Raul Souza Lima	14A

Para a resolução do problema foi definido um sistema de classe utilizado para a leitura de dados direto do arquivo texto. Os dados então são armazenados em vetores de tamanho 8, sendo cada posição:

- 0: latitude;
- 1: longitude;
- 2: demanda do local ($dem > 0$ para coleta, $dem < 0$ para entrega);
- 3: tempo mais cedo possível para iniciar o serviço (janela de tempo);
- 4: ultima hora possível para iniciar o serviço (janela de tempo);
- 5: a duração do serviço neste local;
- 6: o par de coleta se $<id>$ for uma entrega e 0 caso contrário;
- 7: o par de entrega se $<id>$ for uma coleta e 0 caso contrário.

Foi também criada uma matriz que armazena o tempo de viagem entre cada par de vértices da instância.

A função “verificar_solucão” verifica se a solução, passada por parâmetro, é uma solução viável para o problema, respeitando as condições pré-impostas de precedência de coleta e entrega, origem e horário de serviço, janelas de tempo, obrigatoriedade e exclusividade de visita, atendimento de pedido e capacidade de veículo.

Uma solução é gerada aleatoriamente para fins de teste, e que possa ser inicialmente verificada pela função “verificar_solução”.

Para encontrar o caminho mais curto que o caminhão pode percorrer em um limite de tempo será utilizado o algoritmo de Warshall. A estratégia seria utilizar a saída gerada pelo algoritmo para definir qual seria o melhor caminho possível a ser percorrido para que todos os pontos sejam percorridos, todas as entregas realizadas e o caminhão pare no depósito. Esta análise também deverá definir se será necessário alterar o caminho de uma variável x para uma variável y se o caminho de x para i e de i para y seja menor.

```
para cada i de 1 até n
  para cada j de 1 até n
    para cada k de 1 até n
      se distância[i,j] > distância[i,k] + distância[k,j] então
        distância[i,j] = distância[i,k] + distância[k,j]
        camMaisCurto[i,j] = camMaisCurto[k,j]
retorne distância
```

Dado o resultado é possível montar uma matriz que analise os diversos caminhos gerados por todas as rotas de todos os caminhões e conferir a viabilidade de cada rota, para que o caminhoneiro consiga realizar todas as entregas dentro do limite de tempo.