

### Trabalho 3

Prezados alunos,

nesse 3o trabalho vocês devem:

1) Baixar os arquivos:

1. [bayg29.tsp.gz](#) (TSP simétrico com 29 cidades),
2. [ch130.tsp.gz](#) (TSP simétrico com 130 cidades),
3. [br17.atsp.gz](#) (TSP assimétrico com 17 cidades) e
4. [ftv70.vrp.gz](#) (TSP assimétrico com 70 cidades).

- Arquivos TSP Simétrico:

O arquivo vem com as cidades separadas pelas coordenadas X e Y.

Each line is made of three numbers:

Node\_id x y

Node\_id is a unique *integer* ( $\geq 1$ ) node identifier and (x,y) are Cartesian coordinates unless otherwise stated. The coordinates don't have to be integers and can be any real numbers.

Por exemplo:

NAME : att48 → NOME DO ARQUIVO E DESCRIÇÃO DO PROBLEMA

COMMENT : 48 capitals of the US (Padberg/Rinaldi) → INFORMA QUE SÃO 48 CIDADES

TYPE : TSP

DIMENSION : 48

EDGE\_WEIGHT\_TYPE : ATT

NODE\_COORD\_SECTION

→ O ARQUIVO CONTÉM: O NÚMERO\_DA)CIDADE

VALOR\_DA\_COORDENADA\_X VALOR\_DA\_COORDENADA\_Y

1 6734 1453

2 2233 10

3 5530 1424

4 401 841

5 3082 1644

...

47 5185 3258

48 3023 1942

EOF → APÓS A ÚLTIMA CIDADE É MARCADO O SINAL EOF (END OF FILE)

A distância entre as cidades i e j dos problemas simétricos deve ser calculada como distância euclidiana.

- Arquivos TSP Assimétrico (ATSP)

A distância para ir da cidade i para a cidade j é diferente da distância para ir da cidade j para a cidade i. O arquivo vem com a matriz das cidades e a distância entre elas, marcando com um valor extremo a posição na matriz que coincide linha e coluna com i.

Por exemplo:

NAME: br17 → NOME DO ARQUIVO E DESCRIÇÃO DO PROBLEMA

TYPE: ATSP

COMMENT: 17 city problem (Repetto)

DIMENSION: 17

EDGE\_WEIGHT\_TYPE: EXPLICIT  
EDGE\_WEIGHT\_FORMAT: FULL\_MATRIX  
EDGE\_WEIGHT\_SECTION

→ A MATRIZ É A DISTANCIA DA CIDADE DA LINHA I COM A CIDADE DA COLUNA J

→ COMEÇA COM LINHA I (QUE VAI DE 1 ATÉ 38) COM CADA CIDADE J (QUE VAI DE 1 ATÉ 38)

→ QUANDO I COINCIDE COM J MARCA O VALOR 9999

```
9999 3 5 48 48 8 8 5 5 3 3 0 3 5 8 8
5
3 9999 3 48 48 8 8 5 5 0 0 3 0 3 8 8
5
5 3 9999 72 72 48 48 24 24 3 3 5 3 0 48 48
24
48 48 74 9999 0 6 6 12 12 48 48 48 48 74 6 6
.....
8 8 50 6 6 0 0 8 8 8 8 8 8 50 0 9999
8
5 5 26 12 12 8 8 0 0 5 5 5 5 26 8 8
9999
```

EOF → APÓS A ÚLTIMA CIDADE É MARCADO O SINAL EOF (END OF FILE)

- 2) Usando um algoritmo evolutivo para encontrar a melhor rota para os problemas listados no item 1.

Pode ser AG, ACO, PSO, etc.

Qualquer algoritmo escolhido deve limitar em 20000 cálculos de função objetivo para os problemas com instâncias menores que 50, e 70000 cálculos de função objetivo para os problemas com instâncias maiores que 50.

- 3) Comparar a distância da melhor rota encontrada pelo seu algoritmo pela melhor rota disponível na TSPLIB, através do Erro Relativo, que indica quão próxima estão a solução encontrada pelo seu problema e a melhor solução conhecida:

$$\text{Relative Error (\%)} = ((R_{\text{Prevista}} - R_{\text{Conhecida}}) / R_{\text{Conhecida}}) \times 100$$

Rotas Conhecidas:

- a. bayg29 : 1610 tem o arquivo [bayg29.opt.tour.gz](#)
- b. ch130 : 6110 tem o arquivo [ch130.opt.tour.gz](#)
- c. br17: 39
- d. ftv70: 1950