

1º C0de_and_Conquer

UTAD, 29 de novembro de 2023

Obrigado por aceitar este desafio!

Para concluir o desafio, deve criar um programa que resolva o problema proposto e deve usar o programa para processar os três ficheiros fonte fornecidos.

Problema

Considere:

Um espaço com duas dimensões representado por um sistema de eixos de coordenadas xy , com x e y entre 0 e 1000 unidades.

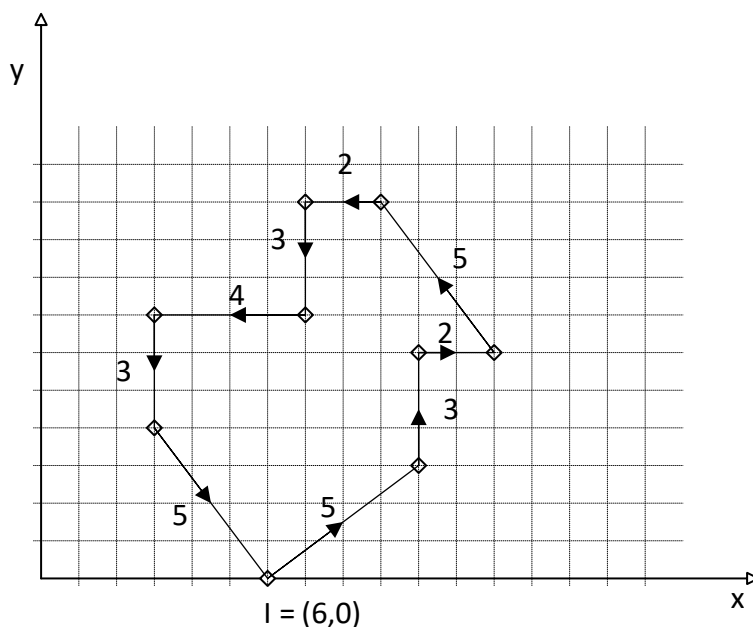
Um conjunto de pontos representados por uma lista de coordenada (x,y) .

Um ponto inicial $I = (x=500, y=0)$.

No ponto inicial I existe um veículo com autonomia para percorrer a distância de 2000 unidades

Calcule a rota que o veículo deve percorrer de forma a visitar o maior número de pontos, partindo de I e regressando a I , sem esgotar a sua autonomia.

Exemplo de rota:



Ponto de Inicial $I = (x=6, y=0)$

Distancia percorrida = $5 + 3 + 2 + 5 + 2 + 3 + 4 + 3 + 5 = 32$ unidades

Rota = $((6,0), (10,3), (10,6), (12,6), (9,10), (7,10), (7,7), (3,7), (3,4), (6,0))$

Programa e processamento dos ficheiros fonte

Deve criar um programa que implemente uma solução para o problema. O programa deve:

Receber um ficheiro fonte;

Processar o ficheiro fonte e calcular a respetiva rota;

Criar um ficheiro de resultado com a rota.

São fornecidos três ficheiros de pontos, que deve processar e calcular a melhor rota para o conjunto de pontos de cada ficheiro. Os ficheiros são: “pontos100.txt” com cerca de cem pontos; “pontos1k.txt” com cerca de mil pontos; e “pontos10k.txt” com cerca de dez mil pontos.

Do processamento dos ficheiros fonte deve guardar três ficheiros, com as designações “resultados100.txt”; “resultados1k.txt”; e “resultados10k.txt”.

Formato dos ficheiros

Os ficheiros fonte, são ficheiros de texto em que cada linha corresponde a um ponto e contém dois números (entre 0 e 1000), separados por um tab ('\t'). A ordem dos pontos é irrelevante.

Os ficheiros de resultado contêm a melhor rota e têm o mesmo formato dos ficheiros fonte, em que cada linha corresponde a um ponto. A ordem dos pontos no ficheiro representa a ordem dos pontos na rota calculada, devendo começar no ponto I e terminar com o mesmo ponto I.

Entregáveis

Deve entregar:

Ficheiro(s) de código;

Ficheiro com programa executável;

Três ficheiros designados “resultados100.txt”, “resultados1k.txt” e “resultados10k.txt” com os resultados do processamento dos ficheiros fonte fornecidos.

Relatório, até duas páginas, com:

- i) descrição do algoritmo;
- ii) descrição da implementação em código;
- iii) instruções para uso do programa executável;
- iv) indicação, para cada uma das três soluções, de:
 - a. tempo de execução gasto pelo programa para chegar à solução
 - b. contagem do número total de pontos visitados
 - c. autonomia total gasta

Submissão dos trabalhos

Os trabalhos devem ser submetidos até às 20h na página

<https://moodle.utad.pt/course/view.php?id=2522>

Nota sobre a escrita do código fonte

O código deve ser escrito e organizado de forma clara.

Deve comentar o código de acordo com as boas práticas de escrita de código, sendo valorizado por isso.

As declarações das estruturas de dados devem incluir comentários com a descrição da utilidade dessas estruturas.

Os protótipos das funções devem incluir um comentário de cabeçalho com:

Descrição do que faz a função;

Descrição dos parâmetros;

Descrição do resultado.

Exemplo:

```
/*
```

```
A função "soma" calcula a soma aritmética de dois números inteiros
```

```
Os parâmetros são os dois números inteiros a somar
```

```
O resultado é um número inteiro correspondendo à soma aritmética dos dois parâmetros
```

```
*/
```

```
int soma(int, int)
```