

Trabalho de Programação 3

Processador Intel (80x86)

1. Descrição Geral

Agora que você consegue viajar pelo espaço, sendo capaz de desviar ou destruir asteroides, é possível estabelecer rotas de navegação entre planetas da galáxia.

Para isso, entretanto, é necessário coletar as informações sobre as **coordenadas dos planetas** e a **quantidade de campos de asteroides** a serem atravessados para viajar de um planeta até outro. Tendo essas informações, será possível calcular a quantidade de combustível necessária para a viagem.

Para calcular a quantidade de combustível deve-se levar em consideração que:

- A nave gasta uma quantidade de combustível igual à distância percorrida. Por exemplo, se a distância for de 5 *parsecs*, então a quantidade de combustível será de 5 gramas de material combustível.
- Sempre que a nave tem que atravessar um campo de asteroides o consumo aumenta, de modo que se deve somar ao consumo devido à distância uma quantidade fixa de 5 gramas de material combustível para cada campo de asteroides atravessado.

As informações para seus cálculos serão fornecidas em um arquivo texto, que o seu programa deverá ler. Os resultados de distância e consumo de combustível calculados devem ser escritos em um arquivo texto de saída.

O nome do arquivo de entrada (arquivo texto com as informações dos planetas e dos campos de asteroides) deverá ser solicitado pelo programa ao usuário. Durante a leitura do nome do arquivo, o programa deverá tratar adequadamente o caractere BACKSPACE (08H). Quando o usuário digitar ENTER (0DH), a digitação do nome do arquivo de entrada estará terminada. O usuário poderá informar o nome completo do arquivo (<nome>.<sufixo>) ou apenas o nome. O programa deverá identificar esse último caso e acrescentar o sufixo, que para todos os arquivos de informações de entrada serão do tipo “.PLA”.

O nome do arquivo de saída deverá ser igual ao nome do arquivo de entrada, porém com o sufixo “.RES”. Exemplo: se o arquivo de entrada tiver o nome “TESTE.PLA”, o arquivo de resultados de saída deverá ter o nome “TESTE.RES”.

2. Especificação do Trabalho

O programa deverá realizar as seguintes tarefas:

1. Ao ser chamado, o programa deve apresentar a identificação do aluno: **nome e número do cartão**;
2. Em seguida, em uma nova linha da tela, deve pedir o **nome do arquivo de entrada**, mantendo o cursor na mesma linha da mensagem de pedido e ecoando na tela os caracteres lidos; após ter o nome completo do arquivo (nome e sufixo), abrir o arquivo; se na abertura ocorrer erro, repetir esta etapa.
3. O arquivo deve ser lido;
4. Com os dados lidos, deve-se calcular: a **distância** entre dois planetas e o **consumo de combustível** para viajar entre eles;
5. Os dados lidos do arquivo de entrada, bem como os valores calculados, devem ser escritos no arquivo de saída (criar arquivo para escrita antes de iniciar a gravação dos dados).

3. Formato do Arquivo de Entrada

As informações estarão dispostas no arquivo de entrada por linhas. Cada linha é terminada por um par de caracteres CR (0DH) e LF (0AH), nesta ordem. O programa deve descartar as linhas vazias: linhas que contenham apenas SPACES (20H) e TABs (09H) e/ou CR (0DH) e LF (0AH).

As linhas que contiverem alguma informação (linhas não vazias) terão as informações em ordem, conforme o exemplo da Figura 1, adiante.

```
3
10 20
15 30
20 65
2
0 1 2
1 2 1
```

Figura 1 – Exemplo de arquivo de entrada

Número de planetas: na primeira linha haverá um número (“3” no exemplo da Figura 1) que indica a quantidade de planetas da galáxia. Esse número é um valor inteiro no intervalo [2,10].

Lista de coordenadas dos planetas: logo após o número de planetas estarão as coordenadas de cada um deles. As coordenadas do planeta “i” são formadas por dois números separados por um ou mais SPACES (20H) ou TABs (09H). Cada linha de texto dessa lista conterá as coordenadas de um planeta, sendo que o primeiro número é a coordenada “X” e o segundo número é a coordenada “Y” deste planeta. No caso do exemplo, as coordenadas do planeta “0” são “X=10” e “Y=20”; do planeta “1” são “X=15” e “Y=30”; do planeta “2” são “X=20” e “Y=65”. Os valores válidos para “X” e “Y” são inteiros no intervalo [0,127].

Tamanho da lista de campos de asteroides: após as linhas com as coordenadas dos planetas virá uma linha com o número de elementos que descrevem a quantidade de campos de asteroides a serem atravessados, quando viajando de um planeta até outro. No caso do exemplo, esse número é “2”, mas ele pode ser um valor inteiro no intervalo [0,127].

Lista de números de campos de asteroides: após a linha com o tamanho da lista de campos de asteroides virá a lista propriamente dita. Só fazem parte dessa lista os pares de planetas *entre os quais existem um ou mais campos de asteroides*. No exemplo acima, não aparece uma linha com os planetas “0” e “2”. Portanto, não existem campos de asteroides entre eles. Cada elemento dessa lista estará em uma linha de texto e será formado por três números separados por um ou mais SPACES (20H) ou TABs (09H). O primeiro e segundo números identificam dois planetas (numerados de 0 a 9, no máximo), enquanto o terceiro número informa a quantidade de campos de asteroides a atravessar para viajar de um destes planetas até outro. A quantidade de campos de asteroides será um número inteiro no intervalo [1,10]. No caso do exemplo, na primeira linha dessa lista, está indicado que, para viajar do planeta “0” até o planeta “1”, deve-se atravessar “2” campos de asteroides e, na segunda linha, que para viajar do planeta “1” até o planeta “2” deve-se atravessar “1” campo de asteroides. Notar que o número de campos de asteroides que existe entre os planetas *i* e *j* é o mesmo que existe entre os planetas *j* e *i*.

4. Formato do Arquivo de Saída

No arquivo de saída devem ser colocadas as informações originalmente fornecidas no arquivo de entrada, acrescidas das informações calculadas. Na Figura 2 pode-se ver o resultado esperado para o arquivo de entrada apresentado na Figura 1.

```
PLANETAS: 3
PLANETA[0]: 10 20
PLANETA[1]: 15 30
PLANETA[2]: 20 65

DISTANCIAS E COMBUSTIVEL NECESSARIO
(0-1): 11 21
(0-2): 46 46
(1-2): 35 40
```

Figura 2 – Exemplo de arquivo de saída

Na primeira linha deve ser colocado o número de planetas. A seguir, deve ser colocada uma lista com as coordenadas dos planetas, um planeta em cada linha, seguida por uma linha em branco.

Logo após a linha em branco, deve vir uma lista com as distâncias entre cada par de planetas e o combustível necessário para viajar entre esses dois planetas. Essa lista deve iniciar com o texto de identificação “DISTANCIAS E COMBUSTIVEL NECESSARIO” e será seguida pelas linhas que informam os valores calculados.

Na lista de valores calculados, cada linha de texto deve apresentar as informações calculadas para um par de planetas diferentes, representados pelos respectivos números entre parênteses e separados por hífens (vide Figura 2). Além disso, os números entre parênteses deve estar dispostos de tal forma que o número da esquerda seja, sempre, menor do que o da direita. Notar que ao apresentar um par “(i,j)” não será necessário apresentar o par “(j,i)”.

Logo após a identificação dos planetas, virá o caractere ‘:’, seguido da distância entre os planetas e da quantidade de combustível necessária para viajar de um planeta até o outro, valores estes que serão calculados pelo programa. Note que estes valores deverão ser inteiros, gravados sem 0s não significativos à esquerda e precedidos de um espaço em branco.

A distância entre o planeta de coordenadas (x_0, y_0) e o planeta de coordenadas (x_1, y_1) , deve ser calculada através da seguinte expressão:

$$Distancia = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2}$$

A raiz quadrada deve ser um valor inteiro arredondado. Por exemplo, $\sqrt{27} = 5,196152$. Portanto, o valor dessa raiz quadrada deve ser informado como o valor “5”. Outro exemplo: $\sqrt{45} = 6,708204$. Logo, o valor calculado pelo programa deve ser informado como “7”. Para o cálculo da raiz quadrada poderá ser usado qualquer algoritmo da escolha do programador, desde que este forneça um resultado com a precisão necessária para obter os resultados na forma aqui descrita (teste seu algoritmo com os valores deste exemplo).

Por exemplo, para calcular a parte inteira da raiz quadrada de um número pode-se usar a Equação de Pell. Essa equação baseia-se no fato da soma dos primeiros “N” números ímpares ser sempre igual a “N²”. Veja exemplos na tabela abaixo.

N	Somatório
1	$= 1 = 1^2$
2	$= 1+3 = 4 = 2^2$
3	$= 1+3+5 = 9 = 3^2$
4	$= 1+3+5+7 = 16 = 4^2$
5	$= 1+3+5+7+9 = 25 = 5^2$

Para calcular o consumo de combustível deve-se somar a distância ao número de campos de asteroides atravessados multiplicado por 5. Então, o consumo é calculado da seguinte forma:

$$Consumo = Distancia + 5 \cdot Numero\ de\ Campos$$

5. Entregáveis: o que deve ser entregue?

Deverá ser entregue, via Moodle da disciplina, o arquivo fonte com a solução do problema apresentado, escrito *na linguagem simbólica de montagem* dos processadores 80X86 da Intel (arquivo .ASM). Além disso, esse programa fonte deverá conter comentários descritivos da implementação, no mesmo estilo dos comentários usados para documentação dos trabalhos feitos com o montador Daedalus. O nome do arquivo deverá ser formado pela inicial do primeiro nome do aluno, seguida de seu número de cartão sem zeros não significativos (exemplo: J123456.asm).

Para a correção, os programas serão montados e executados no ambiente DosBox, com diferentes arquivos de dados de entrada. A nota final do trabalho será proporcional às funcionalidades que forem atendidas pelo programa.

O trabalho deverá ser entregue até a data prevista, conforme programado no MOODLE: ver tarefa no Tópico 9 - “Entrega do terceiro trabalho (Intel)”. **Não será aceita a entrega de trabalhos após a data estabelecida.**

6. Observações

Recomenda-se a troca de ideias entre os alunos. Entretanto, a identificação de cópias de trabalhos acarretará na aplicação do Código Disciplinar Discente e a tomada das medidas cabíveis para essa situação (**tanto o trabalho original quanto os copiados receberão nota zero**).

O professor da disciplina reserva-se o direito, caso necessário, de solicitar uma demonstração do programa, onde o aluno será arguido sobre o trabalho como um todo. Nesse caso, a nota final do trabalho levará em consideração o resultado da demonstração.