



Fundação CECIERJ - Vice-Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação  
Disciplina Fundamentos de Programação**

**AD1 1º semestre de 2017**

---

**IMPORTANTE**

- As respostas (programas) deverão ser entregues pela plataforma em um arquivo ZIP contendo todos os arquivos de código fonte (extensão “.py”) necessários para que os programas sejam testados. Serão aceitos apenas soluções escritas na linguagem Python 3.
  - Faça uso de boas práticas de programação, em especial, na escolha de identificadores de variáveis e subprogramas, e comentários no código.
  - As ADs deverão ser entregues pela atividade "Entrega de AD1" antes da data final de entrega estabelecida no calendário de entrega de ADs.
  - A AD é um mecanismo de avaliação individual. As soluções podem ser buscadas por grupos de alunos, mas a redação final de cada prova tem que ser individual.
- 

**1ª Questão (1,5 pontos)**

Faça um programa que leia pares de números de ponto flutuante (x,y) do teclado, até que o par (0,0) seja digitado. Considere que cada par represente um ponto no espaço bidimensional. Desconsiderando o par (0,0), que neste caso apenas delimita o fim das entradas, escreva a média de todos os x's lidos e a média de todos os y's lidos. A este ponto damos o nome de *ponto médio*. Caso nenhum ponto além do (0,0) seja lido, informe este fato.

Entrada

A entrada é composta por várias linhas, cada uma contando um par de números de ponto flutuante, separados por um ou mais espaços em branco, representando um ponto no espaço bidimensional. A última linha contém o par 0 0, que não deve ser processado.

Saída

Uma linha deve ser emitida na saída padrão, após uma sequência de entradas de pontos ser digitada. Caso a sequência de entrada contenha apenas o par 0 0 então a mensagem “Não existem pontos!” deve ser escrita. Caso possua mais pontos, um par de números de ponto flutuante, com precisão de duas casas de decimais, deve ser escrito, separados por um espaço em branco.

### Exemplos:

Sequência de Entrada	Saída
0 0	Não existem pontos!

Sequência de Entrada	Saída
7 7 2 1.5 0 0	4.50 4.25

Sequência de Entrada	Saída
7 7 2 1.5 3.3 4.7 0 0	4.10 4.40

### **2ª Questão (1,5 pontos)**

Faça um programa que leia do usuário um número inteiro positivo representando a quantidade de pontos (x,y), com coordenadas inteiras, que deverá digitar nas linhas seguintes. Identifique os valores mínimos, xMin e yMin, e máximos, xMax e yMax, dos x's e dos y's digitados. Escreva no vídeo todos os pontos contidos no retângulo representado pelos pontos limitantes (xMin,yMin) e (xMax//2,yMax//2), e a quantidade de pontos contidos neste retângulo.

#### Entrada

A entrada é composta por uma primeira linha, que define a quantidade N de pontos (x,y) a serem lidos. As N linhas seguintes representam pares de números inteiros, representando os pontos a serem processados.

#### Saída

A primeira linha deve conter a string "xMin = " seguida do valor inteiro do menor x lido;

A segunda linha deve conter a string "yMin = " seguida do valor inteiro do menor y lido;

A terceira linha deve conter a string "xMax = " seguida do valor inteiro do maior x lido;

A quarta linha deve conter a string "yMax = " seguida do valor inteiro do maior y lido;

A quinta linha deve conter a string "Pontos dentro do retângulo:"

A linha seguinte, caso existam, mostram os pontos, que estejam dentro do retângulo especificado por (xMin,yMin) e (xMax//2, yMax//2);

A última linha possui a string "Total de Pontos: " seguida da quantidade de pontos dentro do referido retângulo.

#### Exemplo

Entrada	Saída
3 2 8 18 1 -2 27	xMin = -2 yMin = 1 xMax = 18 yMax = 27 Pontos dentro do retângulo: 2 8 Total de Pontos: 1

### 3ª Questão (2,0 pontos)

Faça um programa que peça ao usuário as dimensões de uma matriz bidimensional, chamada de `valores`, de números inteiros a ser gerada em aleatoriamente no intervalo 10 a 99. Via subprogramação:

- Mostre a matriz gerada;
- Caso existam, identifique e mostre todas as submatrizes de dimensão três por três (3x3), que possuam na sua célula central `valores[i][j]` um número menor que todos os seus oito vizinhos imediatos.

#### Entrada

A entrada é composta por uma primeira linha, que define a quantidade `L` de linhas e a quantidade `C` de colunas da matriz a ser gerada.

#### Saída

Inicialmente: `L` linhas, onde cada linha possui `C` valores inteiros no intervalo 10 a 99;

Seguida(s) de uma linha em branco;

Caso exista(m) submatriz(es) com a propriedade estabelecida, seguidas de zero ou mais repetições, separadas por linha em branco, de três linhas com três números inteiros de 10 a 99.

#### Exemplo

Entrada	Saída
5 6	13 28 45 50 26 10 27 24 22 33 88 11 90 25 85 23 76 55 77 15 31 29 13 14 66 41 50 20 47 11  28 45 50 24 22 33 25 85 23  90 25 85 77 15 31 66 41 50

### 4ª Questão (1,5 pontos)

Um número perfeito é um número inteiro positivo para o qual a soma de todos os seus divisores positivos próprios (excluindo ele mesmo) é igual ao próprio número. Por exemplo o número 6 é perfeito, pois  $1+2+3$  é igual a 6. Faça um programa que imprima se um determinado número é perfeito ou não. A verificação deverá ser feita via subprogramação. Ou seja, seu programa deverá conter uma função que, dada um número inteiro `x`, retorne `True` caso `x` seja perfeito e `False` caso contrário. A leitura dos números a serem testados e a impressão na saída padrão ficará a cargo do programa principal.

#### Entrada

A entrada contém vários casos de teste. A primeira linha da entrada contém um inteiro `N` indicando o número de casos de teste da entrada. Cada uma das `N` linhas seguintes contém um valor inteiro `x` positivo, que pode ser ou não ser um número perfeito.

### Saída

Para cada caso de teste de entrada, imprima a mensagem “X é perfeito” ou “X não é perfeito”, de acordo com a especificação fornecida.

### Exemplo

Entrada	Saída
3	6 é perfeito
6	5 não é perfeito
5	28 é perfeito
28	

### **5ª Questão (1,5 pontos)**

Você encontrou um mapa do tesouro. O mapa tem as seguintes características:

- O ponto de partida de todos os mapas é o canto superior esquerdo.
- São retangulares e em cada ponto apresenta um destes símbolos:
  - Um espaço de terreno atravessável;
  - Uma flecha, representando uma possível troca de direção; ou
  - Um baú.

Porém, você está em dúvida se ele leva a um baú cheio de ouro ou se ele leva a lugar algum ou se faz com que você ande em círculos. Escreva um programa que tire sua dúvida.

### Entrada

Na primeira linha existe um par de valores inteiros positivos que simbolizam a largura e a altura do mapa. As linhas seguintes contêm diversos caracteres respeitando as dimensões do mapa. Os caracteres válidos são:

- Uma flecha para a direita (símbolo de maior que): >
- Uma flecha para a esquerda (símbolo de menor que): <
- Uma flecha para baixo (letra v minúscula): v
- Uma flecha para cima (acento circunflexo): ^
- Um espaço de terreno atravessável (ponto): .
- Um baú (asterisco): \*

### Saída

A saída deve consistir de uma única linha com as mensagens "Esse mapa leva ao tesouro", caso o mapa seja válido, ou "Esse mapa não leva a lugar algum", o caminho indicado faça você andar em círculos ou te leve para fora do terreno.

### Exemplos

Entrada	Saída
6 3 >..v.. ..... ...>.*	Esse mapa leva ao tesouro

Entrada	Saída
7 5 >.....v ..... ..... ..... ^.....<	Esse mapa não leva a lugar algum

### **6ª Questão (2,0 pontos)**

**Errata: O segundo exemplo dessa questão foi corrigido, pois havia um erro na versão original da AD1**

A fofoca consiste no ato de descobrir uma informação sobre alguém e posteriormente contar essa informação a uma ou várias pessoas. Geralmente os fofoqueiros são bem conhecidos, de modo que é possível prever para onde uma informação sigilosa poderá propagar. Considerando que temos um ambiente cheio de fofoqueiros e algumas pessoas ingênuas que caem na armadilha de contar seus segredos, sua tarefa é escrever um programa recursivo que estime como a informação será propagada no ambiente.

Os ambientes foram discretizados em células, sendo que as células podem corresponder a uma região sem pessoas, com pessoas fofoqueiras ou com pessoas ingênuas dispostas a contar seus segredos. Devido a característica nata do ato de fofocar, uma célula com uma pessoa ingênuas repassa a informação para os fofoqueiros nas células adjacentes (esquerda, direita, acima e abaixo), que por sua vez repassam para os fofoqueiros nas células vizinhas, e assim por diante, até que o processo para quando todos os indivíduos do mesmo grupo são conhecedores do segredo.

### Entrada

A entrada é composta por vários ambientes, sendo que a descrição de cada ambiente começa com uma linha contendo dois inteiros  $N$  e  $M$ , correspondente ao número de linhas e de colunas do ambiente. As  $N$  linhas a seguir descrevem o ambiente, cada linha contendo  $M$  caracteres, além da quebra de linha. Os caracteres possíveis são:  $F$ , que representa uma célula contendo um fofoqueiro,  $x$ , que representa uma célula vazia e  $I$  que representa uma célula com uma informação a ser focada (inicialmente, é(são) nessa(s) célula(s) onde está(ão) a(s) pessoa(s) ingênuas).

A entrada termina quando  $N = M = 0$ , caso que não deve ser processado.

### Saída

Para cada ambiente, imprima na saída padrão uma estimativa de para onde a informação será focada. Esta estimativa deverá corresponder ao mapa original (como visto na entrada), porém trocando as células com fofoqueiros pelo caractere  $I$ . Deixe uma linha em branco após cada

mapa (incluindo o último mapa).

Exemplo

Entrada	Saída
6 7 xxFFxxx xxFFxFx xxxxFxx xFFFFFFx IFFxFFF xxxxxxxx 3 3 III xxx FFF 0 0	xxFFxxx xxFFxFx xxxxIxx xIIIIIx IIIxIII xxxxxxx  III xxx FFF