



Fundação CECIERJ - Vice-Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação
Disciplina Fundamentos de Programação

AD1 – 2º 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. Respostas entregues fora do formato especificado, por exemplo, em arquivos com extensão “.pdf”, “.doc” ou outras, não serão corrigidas.
- Serão aceitos apenas soluções escritas na linguagem Python 3. Programas com erro de interpretação não serão corrigidos. Evite problemas utilizando tanto a versão da linguagem de programação (Python 3.X) quanto a IDE (PyCharm) indicadas na Aula 1.
- Quando o enunciado de uma questão inclui especificação de formato de entrada e saída, tal especificação deve ser seguida à risca pelo programa entregue. Atender ao enunciado faz parte da avaliação e da composição da nota final.
- Faça uso de boas práticas de programação, em especial, na escolha de identificadores de variáveis, subprogramas e comentários no código.
- As respostas deverão ser entregues pela atividade "Entrega de AD1" antes da data final de entrega estabelecida no calendário de entrega de ADs. Não serão aceitas entregas tardias ou substituição de respostas após término do prazo.
- As ADs são 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. Respostas plagiadas não serão corrigidas.
- **Esta versão do PDF inclui errata na Questão 3.**

1ª Questão (1,5 pontos)

Quando se é criança, muitas das disputas entre amigos acabam se decidindo pelo jogo de Par ou Ímpar. Nele, uma das crianças participantes escolhe “par” enquanto que a outra escolhe “ímpar”. Após a escolha, as duas crianças mostram simultaneamente uma certa quantidade de dedos de uma das mãos. Se a soma de dedos mostrados for par então vence a criança que escolheu “par”, caso contrário vence a criança que escolheu “ímpar”.

Joãozinho e seus amigos precisam resolver uma disputa. O problema do jogo de Par ou Ímpar é que ele só pode ser jogado por duas crianças, enquanto que o grupo de Joãozinho é formado por N crianças e N pode ser maior ou igual a 2. Para resolver o problema, Joãozinho decidiu fazer o seguinte: Cada uma das crianças foi identificada por um número de 0 a $N - 1$. Assim como no jogo de Par ou Ímpar, todas as crianças mostram simultaneamente uma certa

quantidade de dedos de uma das mãos. Dada a soma de dedos mostrados, vence a criança identificada pelo número que corresponde ao resto da divisão do total por N.

Dada uma sequência de informações sobre partidas do jogo de Joãozinho (número de crianças, seus nomes e número de dedos mostrados em cada partida), escreva um programa para indicar os vencedores em cada sequência.

Entrada

A entrada é composta por várias sequências de partidas. A primeira linha de uma sequência de partidas contém dois inteiros N e P que indicam, respectivamente, o número de crianças e o número de partidas na sequência. As N linhas seguintes contêm o nome das N crianças participantes, um por linha. Elas são identificadas sequencialmente pelos números 0 a N – 1 na ordem em que aparecem. As P linhas seguintes contêm N valores inteiros cada, indicando o número de dedos que cada criança mostrou na partida. A quantidade de dedos mostrada por cada criança vai de 0 a 5. O final da sequência de partidas é indicado por N = 0 e P = 0.

É garantido que os valores de N e P atendem às restrições: $2 \leq N \leq 100$ e $1 \leq P \leq 1000$.

Saída

Para cada sequência de partidas seu programa deverá imprimir P + 2 linhas. A primeira linha deve conter um identificador da sequência de partidas no formato “Sequência K”, onde K é numerado sequencialmente a partir de 1. As P linhas seguintes devem indicar o nome do vencedor de cada partida. A próxima linha deve ser deixada em branco.

Exemplos

Entrada	Saída
2 3 Joãozinho Paulinho 1 0 2 4 3 5 3 4 Mariazinha Paulinho Joãozinho 5 1 0 2 3 5 1 3 1 5 5 5 0 0	Sequência 1 Paulinho Joãozinho Joãozinho Sequência 2 Mariazinha Paulinho Joãozinho Mariazinha

2ª Questão (1,5 pontos)

Diferentes regiões do país apresentam diferentes variações de volume de chuva. Por exemplo, é de se esperar que em algumas cidades do Nordeste o volume médio de chuva seja mais baixo que em algumas cidades do Sul do país.

Um projeto pioneiro conduzido em parceria por vários cursos do Cederj visa estudar as variações no volume de chuva em diferentes cidades brasileiras. Você, como participante do projeto, deverá escrever um programa que ajude os pesquisadores a descobrir como se comporta a média de volume de chuva, considerada em intervalos de uma dada duração em

dias. Por exemplo, se para uma determinada cidade é registrada a sequência diária de medições de 100, 120, 0, 0, 247 e 30 mm de chuva e intervalos de 3 dias, as médias são, respectivamente, $220/3 = 73.33$, $120/3 = 40.0$, $247/3 = 82.33$ e $277/3 = 92.33$.

Mais especificamente, conhecidas a sequência de volumes de chuva medidos diariamente para uma cidade e o tamanho do intervalo desejado, em dias, seu programa deverá informar qual o maior e qual o menor volume médio registrado, considerando o tamanho do intervalo dado.

Entrada

A entrada é composta por várias cidades. A primeira linha da cidade contém dois números inteiros positivos N e M, que indicam, respectivamente, o número total de medições diárias e o tamanho dos intervalos, em dias, em que as médias devem ser calculadas. As N linhas seguintes contêm um número inteiro cada, representando a sequência de medições diárias de volume de chuva em mm (milímetros). O final da entrada é indicado por N = 0 e M = 0.

É garantido que os valores de N e M atendem às restrições: $1 \leq N \leq 1000$ e $1 \leq M \leq N$.

Saída

Para cada cidade seu programa deverá produzir três linhas. A primeira linha identifica a cidade, no formato "Cidade K", onde K é numerado a partir de 1. A segunda linha deve conter dois números de ponto flutuante X e Y, separados por um único espaço em branco, representando respectivamente os valores de menor e maior média de volume de chuva. Os valores devem ser apresentados com precisão de duas casas decimais. A terceira linha deve ser deixada em branco.

Exemplos

Entrada	Saída
6 3	Cidade 1
100	40.00 92.33
120	
0	Cidade 2
0	10.00 10.00
247	
30	Cidade 3
1 1	0.00 50.00
10	
4 2	
50	
50	
0	
0	
0 0	

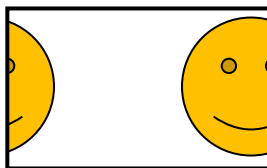
3ª Questão (2,0 pontos)

Errata: Havia um erro na versão original da AD1. A porção corrigida está marcada em azul.

Houve um tempo em que as videoaulas do Cederj eram gravadas em DVDs e enviadas para os polos pelos Correios. Nessa época os alunos se reuniam uma vez por semana para assistir os conteúdos apresentados nas disciplinas.

Os televisores disponíveis em alguns polos eram muito antigos e necessitavam de ajustes para enquadrar de maneira adequada as imagens das aulas. Sem os ajustes, as imagens podiam

ser exibidas deslocadas ciclicamente para ou direita, para a esquerda, para cima ou para baixo. Deslocamento cíclico para a esquerda significa que a parte da imagem que deixa de ser vista à esquerda da tela reaparece do lado direito:



O comportamento dos outros deslocamentos é análogo.

A imagem exibida pelos televisores pode ser tratada como uma matriz de pixels organizados em linhas e colunas. Cada comando de ajuste desloca um certo número de colunas (para a direita ou para a esquerda) e de linhas (para cima ou para baixo). Dada uma imagem e uma sequência de comandos de ajustes, seu programa deverá produzir a imagem resultante após a aplicação dos ajustes na imagem original.

Entrada

A primeira linha da entrada é composta por dois números inteiros C e L representando, respectivamente, o número de colunas e de linhas da imagem original. As L linhas seguintes contêm cada uma C valores inteiros que representam a intensidade dos pixels na linha da imagem original. As linhas que seguem a descrição da imagem original são compostas por pares de valores inteiros X e Y. Cada par de valores corresponde a um ajuste, sendo X o ajuste horizontal e Y o ajuste vertical. Valores negativos e positivos de X deslocam ciclicamente a imagem para a esquerda e direita, respectivamente. Valores negativos e positivos de Y deslocam ciclicamente a imagem para cima e para baixo, respectivamente. Os valores X = 0 e Y = 0 sinalizam o término da sequência de ajustes.

É garantido que os valores de C e L atendem às restrições: $1 \leq C \leq 1000$ e $1 \leq L \leq 1000$.

Saída

Seu programa deve produzir uma imagem de saída, emitindo a matriz com a imagem resultante após os ajustes no mesmo formato que a imagem original.

Exemplo

Entrada	Saída
4 3	1 2 3 4
6 7 8 5	5 6 7 8
10 11 12 9	9 10 11 12
2 3 4 1	
-3 2	
5 7	
0 -2	
-5 -3	
0 0	

Restrição

É proibida a importação de módulos e pacotes para manipulação de matrizes ou de estruturas de dados similares. Soluções que fazem importação não serão consideradas na correção.

4ª Questão (1,5 pontos)

O DNA de um ser vivo, ou de uma doença, pode ser representado por uma longa string composta apenas por sequência de letras "A", "C", "T", e "G". Faça um programa que:

- Leia da entrada padrão o comprimento de um DNA de um ser vivo;
- Gere uma string de DNA do ser vivo, com o comprimento lido, onde o DNA deve ser uma sequência das quatro letras, geradas aleatoriamente;
- Mostre o DNA do ser vivo na saída padrão;
- Leia da entrada padrão o comprimento de um DNA de uma doença;
- Gere uma string de DNA da doença, com o comprimento lido, onde a DNA deve ser uma sequência das quatro letras, geradas aleatoriamente;
- Mostre o DNA da doença na saída padrão;
- Escreva a contagem de todas as ocorrências da doença no ser vivo;
- Gere um DNA complementar invertido do ser vivo, correspondente a inverter o DNA do ser vivo e trocar cada ocorrência de "A" por "C", de "C" por "A", de "T" por "G" e de "G" por "T";
- Mostre o DNA complementar invertido do ser vivo;
- Escreva a contagem de todas as ocorrências da doença no DNA complementar invertido do ser vivo.

Exemplo

```
Comprimento da DNA do ser vivo: 20
DNA do ser vivo gerado aleatoriamente: TCCATGAGTTACCATCTGAG
Comprimento da DNA da doença: 4
DNA da doença gerada aleatoriamente: CCAT
Contagem das ocorrências direta da doença no ser vivo: 2
DNA do ser vivo, complementar invertido: TCTGAGCAACGGTCTGCAAG
Contagem das ocorrências da doença no DNA complementar invertido: 0
```

5ª Questão (1,5 pontos)

Utilizando subprogramação, faça um programa que leia da entrada padrão números inteiros não negativos na base decimal, até que um número negativo seja lido. Para cada número lido, exceto para o negativo, escreva as quinze strings que representam conversões do número lido para bases 2 a 16. A rotina de conversão deverá ser recursiva.

Exemplo

```
Digite um valor inteiro não negativo (negativo para sair): 13
Resultado das Conversões:
Base 2: 1101
Base 3: 111
Base 4: 31
Base 5: 23
Base 6: 21
Base 7: 16
Base 8: 15
```

```
Base 9: 14
Base 10: 13
Base 11: 12
Base 12: 11
Base 13: 10
Base 14: D
Base 15: D
Base 16: D
```

Digite um valor inteiro não negativo (negativo para sair): -5

Restrição

É proibido o uso de funções prontas do Python que realizam conversão entre bases. Ou seja, a solução recursiva deve ser escrita pelo aluno. Caso contrário, ela não será considerada na correção.

6ª Questão (2,0 pontos)

Faça um programa que leia do usuário as dimensões, quantidade de linhas e quantidade de colunas, de uma matriz bidimensional. Gere e mostre a matriz produzida onde cada célula é gerada aleatoriamente no intervalo 0 a 9. Escreva todas as diagonais possíveis da matriz, onde cada diagonal deve ser mostrada em uma linha da saída padrão.

Exemplo

```
Digite a quantidade de linhas: 3
Digite a quantidade de colunas: 4
```

Matriz Gerada Aleatoriamente:

```
8 3 2 5
0 1 9 1
2 7 1 0
```

Listagem das Diagonais:

```
8
0 3
2 1 2
7 9 5
1 1
0
2
0 7
8 1 1
3 9 0
2 1
5
```

Restrição

É proibida a importação de módulos e pacotes para manipulação de matrizes ou de estruturas de dados similares. Soluções que fazem importação não serão consideradas na correção.

Boa Avaliação!