

MAC2166 – Introdução à Computação para Engenharia

ESCOLA POLITÉCNICA

Prova Substitutiva – 24 de junho de 2013

Nome: _____

Assinatura: _____

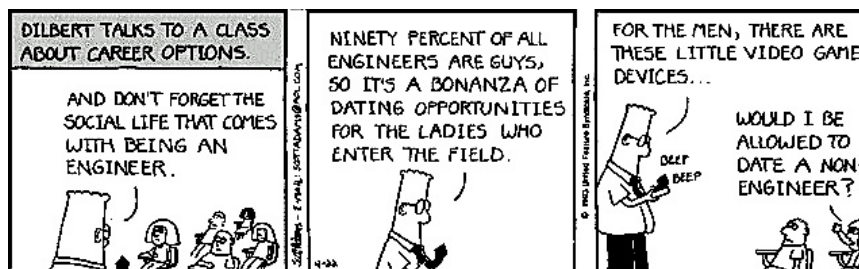
Nº USP: _____ Turma: _____

Professor: _____

Instruções:

1. Não destaque as folhas deste caderno.
2. A prova consta de 3 questões. Verifique antes de começar a prova se o seu caderno de questões está completo.
3. A prova pode ser feita a lápis. Cuidado com a legibilidade e, principalmente, com a TABULAÇÃO.
4. Qualquer questão pode ser resolvida em qualquer página. Se a questão não está na página correspondente ao enunciado basta indicar isto na página e escrever QUESTÃO X em letras ENORMES antes da solução.
5. Não é necessário apagar rascunhos no caderno de questões.
6. Não é permitido o uso de folhas avulsas para rascunho.
7. Não é permitido o uso de equipamentos eletrônicos.
8. Não é permitido a consulta a livros, apontamentos ou colegas.

DURAÇÃO DA PROVA: 2 horas



Questão	Valor	Nota
1	3,0	
2	3,0	
3	4,0	
Total	10,0	

Questão 1 (vale 3.0 pontos)

Um caixa de banco tem somente notas de **a** e **b** reais para fornecer aos seus clientes. Quando um cliente chega com um cheque no valor de **c** reais, o caixa precisa fazer umas contas para decidir se é possível descontar o cheque ou não.

Escreva um programa que resolva o problema do caixa de banco: o programa lê do teclado 3 inteiros positivos **a**, **b** e **c** e imprime **TODAS** as combinações possíveis de notas de **a** reais e notas de **b** reais, totalizando **c** reais, ou uma mensagem informando que não é possível descontar o cheque.

Exemplos:

– Para **a= 5**, **b= 3** e **c= 11** seu programa deve imprimir:

1 notas de 5 e 2 notas de 3

– Para **a= 3**, **b= 5** e **c= 30** seu programa deve imprimir:

10 notas de 3 e 0 notas de 5

5 notas de 3 e 3 notas de 5

0 notas de 3 e 6 notas de 5

– Para **a= 5**, **b= 3** e **c= 7** seu programa deve imprimir:

Não é possível descontar o cheque

– Para **a= 4**, **b= 6** e **c= 121** seu programa deve imprimir:

Não é possível descontar o cheque

– Para **a= 4**, **b= 7** e **c= 127** seu programa deve imprimir:

2 notas de 4 e 17 notas de 7

9 notas de 4 e 13 notas de 7

16 notas de 4 e 9 notas de 7

23 notas de 4 e 5 notas de 7

30 notas de 4 e 1 notas de 7

– Para **a= 6**, **b= 15** e **c= 199** seu programa deve imprimir:

Não é possível descontar o cheque

Questão 2 (vale 3.0 pontos)

Escreva um programa que leia do teclado um número inteiro $n > 0$ e uma sequência de n números reais e imprima uma tabela mostrando os números que compõem a sequência e o número de vezes que cada um deles ocorre na mesma.

Exemplo para a entrada:

8 -1.7 3.0 0.0 1.5 0.0 -1.7 2.3 -1.7

a saída deve ser:

```
-1.7 ocorre 3 vezes
3.0 ocorre 1 vezes
0.0 ocorre 2 vezes
1.5 ocorre 1 vezes
2.3 ocorre 1 vezes
```


Questão 3 (vale 4.0 pontos)

Os elementos a_{ij} de uma matriz de números reais $A_{n \times n}$ representam os custos de transporte da cidade i para a cidade j . Um itinerário que percorre k cidades é representado por uma lista C com k inteiros. Nesta questão vamos resolver o problema de calcular custos de itinerários entre as cidades, para achar o itinerário mais barato.

Por exemplo, para $A_{4 \times 4}$,

		0	1	2	3
	0	14.8	2.9	7.5	8.9
$A =$	1	5.3	8.4	18.0	14.8
	2	2.3	21.7	3.9	8.7
	3	7.3	12.1	43.7	57.8

o itinerário entre $k = 9$ cidades, representado pela lista

$C =$	0	3	1	3	3	2	1	0	2
-------	---	---	---	---	---	---	---	---	---

tem custo

$$\begin{aligned} a_{03} + a_{31} + a_{13} + a_{33} + a_{32} + a_{21} + a_{10} + a_{02} &= 8.9 + 12.1 + 14.8 + 57.8 + 43.7 + 21.7 + 5.3 + 7.5 \\ &= 171.8 \end{aligned}$$

e o itinerário entre $k = 1$ cidade, representado pela lista

$$C = [2]$$

tem custo zero.

(a) (valor: 2,0 pontos)

Escreva uma função `custo` com cabeçalho:

```
def custo(A, C):
```

que recebe uma matriz A de números reais de dimensão $n \times n$, $n > 0$, e uma lista C com $k > 0$ números inteiros.

A função deve retornar o custo do itinerário entre as k cidades.

(b) (valor: 2,0 pontos)

Escreva um programa que:

- leia o nome de um arquivo que contém uma matriz de reais **A** (matriz de custos),
- leia um número inteiro **m**, $m > 0$,
- leia **m** itinerários; cada itinerário consiste de uma sequência de **k** números inteiros, onde $k > 0$, representando as cidades (ou seja, o programa deve ler do teclado o valor de **k** e uma sequência de **k** inteiros),
- para cada um dos **m** itinerários o seu programa deve imprimir o seu custo.
- ao final, o programa deve imprimir o custo do itinerário mais barato e a sequência de cidades desse itinerário.

O seu programa **deve**, utilizar a função **custo** do item anterior, mesmo que não a tenha implementado. Além da função **custo** você pode utilizar a função com cabeçalho:

```
def le_matriz(nome_do_arquivo):
```

sem implementá-la. A função **le_matriz** recebe o nome de um arquivo e retorna uma matriz de números reais.

Exemplo: para

		0	1	2	3
A =	0	14.8	2.9	7.5	8.9
	1	5.3	8.4	18.0	14.8
	2	2.3	21.7	3.9	8.7
	3	7.3	12.1	43.7	57.8

$m = 3$,

itinerário 1 $k = 3$

0	1	2
---	---	---

 tem custo 20.9 $(= 2.9 + 18.0)$

itinerário 2 $k = 2$

2	1
---	---

 tem custo 21.7

itinerário 3 $k = 4$

1	3	0	1
---	---	---	---

 tem custo 25.0 $(= 14.8 + 7.3 + 2.9)$

A saída do programa deve ser:

0 itinerário mais barato tem custo 20.9 e passa pelas cidades: 0 1 2

