

MAC2166 – Introdução à Computação
Grande áreas Civil, Mecânica, Petróleo e Química
ESCOLA POLITÉCNICA
Segunda Prova – 10 de maio de 2016

Nome: _____

Assinatura: _____

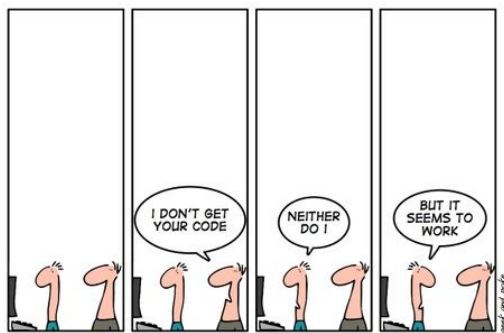
Nº USP: _____ Turma: _____

Professor: _____

Instruções:

1. Não destaque as folhas deste caderno. A prova pode ser feita a lápis.
2. A prova consta de 3 questões. Verifique antes de começar a prova se o seu caderno está completo.
3. As questões podem ser resolvidas em qualquer página. Ao escrever uma solução (ou parte dela) em página diferente do enunciado, escreva **QUESTÃO X** em letras **ENORMES** junto da solução.
4. As soluções devem ser em Python. Você pode usar apenas recursos de Python vistos em aula. Você pode definir funções auxiliares e usá-las à vontade. Cuidado com a legibilidade e, principalmente, com a **TABULAÇÃO**.
5. As soluções não precisam verificar consistência de dados.
6. Não é permitido o uso de folhas avulsas para rascunho, a consulta a livros, apontamentos, colegas ou equipamentos eletrônicos. Desligue o seu celular e qualquer equipamento que possa perturbar o andamento da prova.

DURAÇÃO DA PROVA: 2 horas



THE ART OF PROGRAMING

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

QUESTÃO 1 (vale 4,0 pontos)

Um polinômio de uma variável pode ser representado pela lista de seus coeficientes. Por exemplo, o polinômio $3x^4 + 2x^2 + x + 5$ é representado pela lista `[5, 1, 2, 0, 3]`. Nesta representação:

1. O tamanho da lista é o grau do polinômio mais 1.
2. O número na posição 0 é o coeficiente do monômio de grau 0, o número na posição 1 é o coeficiente do monômio de grau 1. Em geral, o número na posição k é o coeficiente do monômio x^k .
3. O último elemento da lista é sempre diferente de zero. Assim, o polinômio zero é representado pela lista vazia `[]`. Lembre-se também de que o grau do polinômio zero é -1 por convenção, assim o item 1 acima também se aplica ao polinômio zero.

Esta questão consiste na implementação de 2 funções.

Item 1(a) (vale 2,0 pontos)

Escreva uma função `calcula_derivada` que recebe uma lista que representa um polinômio p e devolve a lista que representa sua derivada p' .

```
def calcula_derivada(p):  
    '''list -> list
```

```
  
    Recebe uma lista p que representa um polinômio e devolve a lista que  
    representa sua derivada.
```

```
  
    Exemplos:
```

```
    calcula_derivada([ 1 ]) devolve [ ].
```

```
    calcula_derivada([ 1, 2, 3 ]) devolve [ 2, 6 ].
```

```
    ...
```

ESCREVA A SUA SOLUÇÃO NESTA PÁGINA

Item 1(b) (vale 2,0 pontos)

Escreva uma função `calcula_polinomio` que recebe a lista que representa um polinômio p e um número real x e devolve $p(x)$.

```
def calcula_polinomio(p, x):  
    '''list, float -> float
```

```
  
    Recebe uma lista p que representa um polinômio e um número real x  
    e devolve o valor do polinômio calculado em x.
```

```
  
    Exemplos:
```

```
    calcula_polinomio([ ], 5) devolve 0.
```

```
    calcula_polinomio([ 1, 2, 3 ], 2) devolve 17.
```

```
    ...
```

ESCREVA A SUA SOLUÇÃO AQUI

QUESTÃO 2 (vale 2,0 pontos)

Seqüências de DNA são strings de caracteres A, C, G e T. Um *alinhamento* de duas seqüências de DNA s e t é obtido adicionando-se *gaps*, representados pelo caractere ‘_’ (*underscore*), a s e t , de modo que as strings resultantes tenham o mesmo tamanho. Por exemplo, se $s = \text{TCGTAC}$ e $t = \text{ATCG}$, então um alinhamento pode ser

$$\begin{aligned}s' &= \text{T_CGTAC} \\ t' &= \text{ATCG_}\end{aligned}$$

Dados números inteiros não-negativos m , d e g , a *pontuação* de um alinhamento é calculada da seguinte forma: duas letras iguais alinhadas contam m pontos, duas letras diferentes alinhadas contam $-d$ pontos e uma letra alinhada com um gap ou dois gaps alinhados contam $-g$ pontos. Assim, se $m = 5$, $d = 5$ e $g = 3$, a pontuação do alinhamento acima é

$$-5 - 3 + 5 + 5 - 3 - 3 - 3 = -7.$$

Uma tarefa importante em biologia computacional é calcular um alinhamento entre duas seqüências de DNA que tenha pontuação máxima. Nesta questão você dará um primeiro passo para resolver este problema, escrevendo uma função que, dados números inteiros não-negativos m , d e g e duas strings de mesmo tamanho contendo apenas as letras A, C, G e T e o caractere underscore ‘_’, devolve a pontuação do alinhamento.

```
def pontuacao(m, d, g, s, t):
    '''int, int, int, str, str -> int

    Recebe inteiros não-negativos m, d e g e duas strings s e t de
    mesmo tamanho contendo apenas os caracteres A, C, G, T e _ e
    devolve a pontuacao do alinhamento representado pelas strings.

    Exemplos:
    pontuacao(5, 5, 3, 'T_CGTAC', 'ATCG_') devolve -7.
    '''
```

ESCREVA A SUA SOLUÇÃO NESTA PÁGINA

QUESTÃO 3 (vale 4,0 pontos)

Considere o jogo de PacMan como no EP2, no qual o labirinto é representado por uma matriz com os seguintes caracteres:

- '+' representa uma parede;
- '.' representa um *pac-dot* (a pastilha que quando comida pelo PacMan dá 1 ponto a ele);
- um espaço em branco representa uma posição vazia.

As seguintes regras devem ser respeitadas no jogo:

- paredes não podem ser atravessadas;
- se o PacMan chegar numa posição com um *pac-dot*, ele ganha 1 ponto e no lugar do *pac-dot* é colocado um espaço em branco;
- se o PacMan se chocar com um fantasma ele morre. Se havia um *pac-dot* naquela posição, ele não é comido;
- o labirinto é cíclico.

Escreva a função `movimentaPacMan` com protótipo apresentado na próxima página. A função recebe uma matriz representando o labirinto, uma lista que representa o PacMan e uma matriz com as posições dos fantasmas, realiza o movimento do PacMan de acordo com as regras do jogo, atualizando o labirinto e a lista que representa o PacMan, e devolve `True` se o PacMan não morreu após o movimento; `False` caso contrário.

A lista que representa o PacMan tem o formato

[linha, coluna, direção, pontos]

onde:

- `linha` é um inteiro representando a linha da matriz onde o PacMan se encontra;
- `coluna` é um inteiro representando a coluna da matriz onde o PacMan se encontra;
- `direção` é um inteiro representando para onde o PacMan deve se mover (Esquerda = 0, Direita = 1, Cima = 2 e Baixo = 3);
- `pontos` é um inteiro representando a quantidade de pontos do PacMan na partida.

A matriz que representa os fantasmas contém para cada fantasma sua posição no labirinto, no formato

[[linha, coluna], [linha, coluna], ...],

de modo que cada linha da matriz contém a posição de um dos fantasmas.

ESCREVA A SUA SOLUÇÃO NESTA PÁGINA

```
def movimentaPacMan(lab, pacman, fantasmas):  
    '''(matriz, lista, matriz) -> bool
```

```
Recebe uma matriz lab que representa um labirinto, uma lista pacman que  
representa o PacMan e que contém a direção na qual ele deve ser movido  
e uma matriz fantasmas que contém as posições dos fantasmas no labirinto,  
uma por linha. A função realiza o movimento do PacMan, atualizando  
a lista que o representa e o labirinto, e devolve True se o PacMan não  
morreu após seu movimento; False caso contrário.
```

```
'''
```