



MAC2166 - Introdução a Computação - 2018S1

Prova Substitutiva

0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9

Utilize caneta azul ou preta e preencha completamente a quadrícula.
Exemplo: ■. Não use ☒.

Turma: (somente um número; consulte a pessoa responsável se não souber)

<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6	<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9	<input type="checkbox"/> 10	<input type="checkbox"/> 11	<input type="checkbox"/> 12	<input type="checkbox"/> 20
----------------------------	----------------------------	----------------------------	----------------------------	----------------------------	----------------------------	-----------------------------	-----------------------------	-----------------------------	-----------------------------

← Marque as quadrículas ao lado para formar o seu número USP e escreva seu nome completo em letra legível na linha pontilhada abaixo. **Se seu número possui menos que 8 dígitos complete com zeros à esquerda.**

Nome: _____

.....

Esta avaliação tem duração de 120 minutos. Não desmonte este caderno.

Q1 [2 pontos] Simule o código abaixo e selecione a opção correspondente à saída impressa do programa. Ignore as quebras de linhas geradas pelo comando `print`.

```
def f (L, e, d):
    if e > d:
        return True
    c = (e+d)//2
    print (L[c])
    x = f (L, e, c-1)
    y = f (L, c+1, d)
    return True

def g (L, e, d):
    if e > d:
        return True
    c = (e+d)//2
    x = g (L, e, c-1)
    y = g (L, c+1, d)
    print (L[c])
    return True

def main ():
    f ([35, 42, 14], 0, 2)
    g ([35, 42, 14], 0, 2)

main ()
```

Rascunho

<input type="checkbox"/>	35 42 14 35 42 14
<input type="checkbox"/>	14 35 42 42 35 14
<input type="checkbox"/>	35 14 42 42 35 14
<input checked="" type="checkbox"/>	42 35 14 35 14 42
<input type="checkbox"/>	42 42
<input type="checkbox"/>	14 35 42 14 35 42
<input type="checkbox"/>	42 14 35 35 14 42

<input type="checkbox"/>	35 42 14 42 14 35
<input type="checkbox"/>	14 42 35 42 35 42
<input type="checkbox"/>	42 35 14 42 35 14
<input type="checkbox"/>	35 42 42 35
<input type="checkbox"/>	14 35 42 35 42 14
<input type="checkbox"/>	35 42 14 14 35 42
<input type="checkbox"/>	42 35 14 14 35 42

<input type="checkbox"/>	42 35 35 42
<input type="checkbox"/>	35 14 42 42 14 35
<input type="checkbox"/>	42 14 35 42 14 35
<input type="checkbox"/>	42 35
<input type="checkbox"/>	14 42 35 42 35 14
<input type="checkbox"/>	35 14 42 35 14 42



Q2 [2 pontos] A função abaixo deveria calcular, para uma lista de números previamente ordenada em ordem crescente e contendo uma quantidade ímpar de valores, a diferença entre sua média e sua mediana. Por exemplo, dada a lista $L = [2, 3, 5, 9, 11]$, sua média é dada por:

$$Md(L) = (2 + 3 + 5 + 9 + 11) / len(L) = 6.$$

A mediana é dada por:

$$Mn(L) = L[len(L)//2] = 5.$$

A diferença é dada por:

$$Dif(L) = Md(L) - Mn(L) = 1.$$

Essa função, entretanto, contém erros. Marque as quadrículas correspondentes às correções necessárias para que a função efetue o cálculo corretamente.

Observação: a cada item errado que for selecionado, desconta-se até 0.5 da nota do exercício.

```
(1) def Dif(L) :  
(2)     Md = 0  
(3)     for i in range(1, len(L)):  
(4)         Md += L[i]/i  
(5)     Md /= len(L) - 1  
(6)     return Md - L[len(L)/2]
```

Rascunho

- | | |
|---|--|
| <input type="checkbox"/> Substituir a linha (5) por <code>Md /= L[len(L)]</code> | <input checked="" type="checkbox"/> Substituir a linha (5) por <code>Md /= len(L)</code> |
| <input checked="" type="checkbox"/> Substituir a linha (3) por <code>for i in range(len(L)):</code> | <input checked="" type="checkbox"/> Substituir a linha (6) por <code>return Md - L[len(L)//2]</code> |
| <input checked="" type="checkbox"/> Substituir a linha (4) por <code>Md += L[i]</code> | <input type="checkbox"/> Substituir a linha (4) por <code>Md += L</code> |
| <input type="checkbox"/> Substituir a linha (2) por <code>Md = L[i]</code> | <input type="checkbox"/> Substituir a linha (2) por <code>Md = 1</code> |



Q3 [3 pontos] Preencha as lacunas no código abaixo (L1 até L9), de forma a obter um programa que lê vários arquivos de texto, cujos nomes são passados pelo usuário, separados por espaço (considere que o usuário sempre vai digitar pelo menos os nomes de dois arquivos) e cria uma matriz com as informações destes arquivos da seguinte forma: cada linha da matriz contém na primeira coluna uma palavra encontrada no(s) arquivo(s) e na segunda coluna a quantidade de vezes em que a palavra aparece em todos os arquivos. Depois, a matriz é ordenada pela quantidade de vezes em que as palavras aparecem em todos os arquivos (a primeira linha tem a palavra que menos aparece. A última linha tem a palavra que mais aparece).

```
def ordenaMatriz(m):
    n = len(m)
    for i in range(0,n-1):
        x = m[i+1]
        j = i
        while j >= 0 and m[j][1] > x[1]:
            m[j+1] = m[j]
            j -= 1
        L1

def incluiPalavra(p, m):
    t = len(m)
    if t == 0:
        m.append([p, 1])
    else:
        achou = False
        L2
        while not achou and i != t:
            L3
            L4
            L5
            L6
        if not achou:
            L7

def imprimeMatriz(m):
    for l in m:
        print (l[0], l[1])

def main():
    m = []
    s = input("Nomes: ")
    L8
    for n in s:
        L9
        for l in a:
            lSemEnter = l[:len(l)-1]
            l = lSemEnter.split(" ")
            for p in l:
                incluiPalavra(p,m)
        a.close()

    ordenaMatriz(m)
    imprimeMatriz(m)

main()
```

Para cada um dos 9 itens a seguir, correspondendo às lacunas no código acima, assinale todas as respostas (cada item pode ter de 1 a 6 opções corretas) que tornam o programa acima correto.

Observação: a cada item errado que for selecionado, desconta-se até 0.1 da nota do exercício.

L1:	<input checked="" type="checkbox"/> m[j+1] = x	<input type="checkbox"/> i += 1	<input type="checkbox"/> x = m[j+1][:]	<input checked="" type="checkbox"/> m[j+1] = x[0:len(x)]	<input type="checkbox"/> m[j] = x[:]	<input type="checkbox"/> m[j] = x
L2:	<input type="checkbox"/> i = -1	<input type="checkbox"/> i = 1	<input type="checkbox"/> i -= 1	<input type="checkbox"/> i = t	<input checked="" type="checkbox"/> i = 0	<input type="checkbox"/> i += 1
L3:	<input type="checkbox"/> if m[i][1] == p:	<input checked="" type="checkbox"/> if m[i][0] == p:	<input type="checkbox"/> while (m != None):	<input type="checkbox"/> if i == 0:	<input type="checkbox"/> if achou == False:	<input type="checkbox"/> if achou == True:
L4:	<input type="checkbox"/> m[i][0] = 1	<input type="checkbox"/> m[i][0] = m[i][0]+1	<input checked="" type="checkbox"/> m[i][1] = m[i][1]+1	<input type="checkbox"/> m[i][0] += 1	<input type="checkbox"/> m[i][0] = m[i][1] + 1	<input checked="" type="checkbox"/> m[i][1] += 1
L5:	<input type="checkbox"/> achou = False	<input checked="" type="checkbox"/> achou = True	<input type="checkbox"/> m[t][0] = p	<input type="checkbox"/> i = 1	<input type="checkbox"/> m[i][0] = p	<input type="checkbox"/> i = 0
L6:	<input type="checkbox"/> m[i][0] = i+1	<input checked="" type="checkbox"/> i += 1	<input type="checkbox"/> m[i][1] += 1	<input type="checkbox"/> achou = True	<input type="checkbox"/> achou = False	<input type="checkbox"/> i = i-1
L7:	<input type="checkbox"/> m.append(p,1)	<input type="checkbox"/> m += [p, 1]	<input checked="" type="checkbox"/> m.append([p, 1])	<input type="checkbox"/> m.append(p)	<input type="checkbox"/> i = 0	<input type="checkbox"/> i += 1
L8:	<input type="checkbox"/> split(s)	<input type="checkbox"/> s = s.split("\n")	<input type="checkbox"/> s.split(" ")	<input checked="" type="checkbox"/> s = s.split(" ")	<input type="checkbox"/> s.split()	<input type="checkbox"/> s.split("\n")
L9:	<input type="checkbox"/> a = open(n,"w")	<input type="checkbox"/> n.open("r")	<input type="checkbox"/> a.open(n)	<input type="checkbox"/> a.file(n,"r")	<input checked="" type="checkbox"/> a = open(n,"r")	<input type="checkbox"/> a.open()



Q4 [3 pontos] Considere o programa abaixo:

```
def f(w):  
    return w[0]  
  
def l(w):  
    return w[len(w)-1]  
  
def m(w):  
    n = len(w)  
    return w[1:n-1]  
  
def p(w):  
    if len(w) <= 1:  
        return True  
    if f(w) != l(w):  
        return False  
    return p(m(w))  
  
def main():  
    print(p("allen"))  
    print(p("bob"))  
    print(p("o"))  
    print(p("mac"))  
  
main()
```

Rascunho

Assinale entre as opções a seguir, as afirmações verdadeiras pintando as quadrículas correspondentes.

Considerações:

- As opções podem conter desde nenhuma afirmação correta até todas.
- A cada item errado que for selecionado, desconta-se até 1 ponto da nota do exercício.

- A função `f`, quando recebe uma string `w`, retorna o primeiro caracter da string
- `m("Paulo") == "aul"`
- Ao rodar, o programa imprime `False`, depois `True` duas vezes e por último `False`
- Apesar de haver 4 chamadas à função `p` na função `main`, ela é de fato executada 5 vezes
- Se houvesse uma última linha na função `main` com o conteúdo: `print(p([1,2,3,2,1]))`, o programa daria erro
- Se houvesse uma última linha na função `main` com o conteúdo: `print(p("12321"))`, o programa daria erro
- A função `l`, quando recebe uma string `w`, retorna o último caracter da string
- A função `f`, quando recebe um número inteiro `w`, devolve seu primeiro dígito
- O programa dá erro porque a linha `return w[len(w)-1]` na função `l` está errada já que o índice da lista não é um número inteiro
- A função `f`, quando recebe uma matriz `w`, devolve sua primeira linha