

Programação I 2016/2017

Departamento de Informática, Universidade de Évora

2º teste(a) – 29 de novembro de 2016

Observações: Teste sem consulta. Justificar as respostas, apresentando **todos** os cálculos efectuados.

Relembre a semântica de algumas funções:

`ord(x)`: devolve a codificação decimal de `x`

`chr(x)`: devolve o carácter cuja codificação é `x`

`max(x)`: devolve o maior valor de `x`

`x.index(y)`: devolve o índice de `x` onde se encontra a primeira ocorrência de `y`

`x.split(s,n)`: divide `x` utilizando o separador `s`, devolvendo as partes

`x.append(e)`: adiciona `e` no final de `x`

1. Quando o nº de vezes de um determinado conjunto de acções a executar é desconhecido, que tipo de estruturas repetitivas podem ser usadas? Justifique.

2. Para cada um dos seguintes problemas, indique, **justificando**, se é ou não necessária a utilização de uma variável multi-valor (`list`):

- (a) Ler uma sequência de números e mostrar os números que estão num determinado intervalo de valores.
- (b) Ler uma sequência de números e calcular a sua média
- (c) Ler uma sequência de números e mostrar essa sequência ordenada por ordem crescente
- (d) Ler uma sequência de números e mostrar o segundo maior número da sequência.

3. Indique o tipo e o resultado da avaliação das seguintes expressões:

- (a) `range(1,10,3)[2]`
- (b) `a=[1]; a.append([2,5]); a`
- (c) `a='programacao'; a.upper()+ '16'`
- (d) `s="ana:bill,celia:dinis"; s.split(":")`

4. Implemente a função `altera(L)` que, dada a lista `L` onde os elementos estão ordenados por ordem crescente, devolve uma nova lista ordenada por ordem decrescente e com o menor e maior valores retirados (serão valorizadas soluções eficientes).

5. Implemente a função `sub_factorial(n)` que calcula o sub-factorial de um número `n`, sabendo que:

$$sub_factorial(n) = \frac{n!}{2!} - \frac{n!}{3!} + \frac{n!}{4!} - \dots (-1)^n \frac{n!}{n!}$$

6. Numa empresa os funcionários são pagos a 12,5€ à hora. Pretende-se calcular o valor que um empregado deve receber na referida semana sabendo o tempo de laboração em cada um dos 6 dias da semana, tendo em conta que os descontos para a segurança social e IRS representam 22,5% do vencimento bruto. O programa deve solicitar primeiro o nº de horas de trabalho em cada dia da semana e só depois calcular o valor a receber.

7. Implemente a função `mix(A,B)` que dadas duas listas A e B (com o mesmo número de elementos), devolve uma nova lista com os elementos intercalados de cada uma das listas A e B. Por exemplo, se `A=[1,2,3,4]` e `B=[5,6,7,8]`, a função deverá devolver a lista `[1,5,2,6,3,7,4,8]`.

8. Implemente a função `minimax(A)` que devolve o elemento minimax da matriz A. O elemento minimax de uma matriz, é o menor elemento de uma linha onde se encontra o maior elemento da matriz.

9. Considere a função `qualquer_coisa(S)`.

```
1 def qualquer_coisa(S):  
2     a = [0]*256  
3     for b in S:  
4         if 'a' <= b <= 'z':  
5             a[ord(b)] += 1  
6     for i in range(4):  
7         d = max(a)  
8         j = a.index(d)  
9         print (chr(j), ":", d)  
10        a[j] = 0
```

(a) Indique, justificando, o que faz a função.

(b) Indique o que será escrito na consola se `S='c pppp jik def jji'`. Justifique.