



Algoritmos e Programação de Computadores

Correção da Prova 2

Profa. Sandra Avila

Instituto de Computação (IC/Unicamp)

MC102, 22 Junho, 2018

AVISO: Temos 3 tipos de provas.

Questão 1

(Tipo 1)

Questão 1 (a)

Programa	O que será exibido na tela?
<pre># (0.2 ponto) def invert_sinal(a): print("-a =", -a) invert_sinal(-3)</pre>	<p>-a = 3</p>

Questão 1 (b)

Programa	O que será exibido na tela?
<pre><i># (0.3 ponto)</i> def inverte_sinal(a): a = -a return a a = inverte_sinal(5) print("-a =", -a)</pre>	<p style="text-align: center;">-a = 5</p>

Questão 1 (c)

Programa	O que será exibido na tela?
<pre><i># (0.3 ponto)</i> def invert_sinal(a): a = -a invert_sinal(8) print("-a =", -a)</pre>	<p>Erro na linha 5 (<code>print("-a =", -a)</code>), a não foi definido.</p>

Programa

```
# (0.3 ponto)
def inverte_sinal(a):
    c = -c
    print("c invertido = ", c)

a = -1
inverte(a)
inverte(a)
inverte(a)
if a == 1:
    print("Ficou invertido.")
elif a == -1:
    print("Não ficou invertido.")
else:
    print("-a =", -a)
```

O que será exibido na tela?

Resposta 1:

Erro na linha 6 (inverte(a)), a função inverte não foi definida.

Resposta 2 (caso inverte_sinal(a) nas linhas 6, 7 e 8):

c invertido = 1

c invertido = 1

c invertido = 1

Não ficou invertido.

As duas respostas serão aceitas.

Questão 1 (e)

Programa	O que será exibido na tela?
<pre># (0.4 ponto) def inverte_lista(l): for i in range(1,l): l[i] = -l[i] print(l) return l l = [5, 4, 3] inverte_lista(l.copy()) print(l)</pre>	<p>Erro na linha 2 (for i in range(1,l):), não é possível iterar sobre a lista.</p> <p>O correto seria:</p> <pre>for i in range(1,len(l)):</pre>

Questão 1 (f)

As duas respostas serão aceitas.

Programa	O que será exibido na tela?
<pre># (0.5 ponto) def inverte_misterio(m): misterio_m = m.copy() for i in range(len(m)): misterio_m[i][0] = -m[i][0] print(misterio_m) m = [[-1, 3, 5], [6, 3, 2], [7, -2, 0]] misterio_matriz(m) print(m)</pre>	<p>Resposta 1: Erro na linha 8 (<code>misterio_matriz(m)</code>), a função <code>misterio_matriz</code> não foi definida.</p> <p>Resposta 2 (caso <code>inverte_misterio(m)</code> na linha 8): [[<code>-1, 3, 5</code>], [<code>6, 3, 2</code>], [<code>7, -2, 0</code>]] [[<code>-1, 3, 5</code>], [<code>6, 3, 2</code>], [<code>7, -2, 0</code>]]</p>

Questão 1

(Tipo 2)

Questão 1 (a)

Programa	O que será exibido na tela?
<pre># (0.2 ponto) def inverte_sinal(a): print("-a =", -a) inverte_sinal(-5)</pre>	<p>-a = 5</p>

Questão 1 (b)

Programa	O que será exibido na tela?
<pre data-bbox="104 500 966 873"># (0.3 ponto) def invert_sinal(a): a = -a print("invert_sinal: -a =", -a) invert_sinal(15) print("-a =", -a)</pre>	<p data-bbox="1051 642 1789 732">Erro na linha 6 (<code>print("-a =", -a)</code>), a não foi definido.</p>

Questão 1 (c)

Programa	O que será exibido na tela?
<pre><i># (0.3 ponto)</i> def invert_sinal(a): a = -a return a a = invert_sinal(15) print("-a =", -a)</pre>	<p style="text-align: center;">-a = 15</p>

Programa

```
# (0.3 ponto)
def inverte_sinal(a):
    c = -c
    print("c invertido = ", c)

a = -1
inverte(a)
inverte(a)
if a == 1:
    print("Ficou invertido.")
elif a == -1:
    print("Não ficou invertido.")
else:
    print("-a =", -a)
```

O que será exibido na tela?

Resposta 1:

Erro na linha 6 (inverte(a)), a função inverte não foi definida.

Resposta 2 (caso inverte_sinal(a) nas linhas 6 e 7):

c invertido = 1

c invertido = 1

Não ficou invertido.

As duas respostas serão aceitas.

Questão 1 (e)

Programa	O que será exibido na tela?
<pre># (0.6 ponto) def inverte_lista(l): for i in range(1,l): l[i] = -l[i] print(l) return l l = [8, 2, 7] inverte_lista(l.copy()) print(l)</pre>	<p>Erro na linha 2 (for i in range(1,l):), não é possível iterar sobre a lista.</p> <p>O correto seria:</p> <pre>for i in range(1,len(l)):</pre>

Questão 1 (f)

As duas respostas serão aceitas.

Programa	O que será exibido na tela?
<pre># (0.5 ponto) def invert_e_misterio(m): misterio_m = m.copy() for i in range(len(m)): misterio_m[0][i] = -m[0][i] print(misterio_m) m = [[-1, 3, 5], [6, 3, 2], [7, -2, 0]] misterio_matriz(m) print(m)</pre>	<p>Resposta 1: Erro na linha 8 (<code>misterio_matriz(m)</code>), a função <code>misterio_matriz</code> não foi definida.</p> <p>Resposta 2 (caso <code>invert_e_misterio(m)</code> na linha 8): [[1, -3, -5], [6, 3, 2], [7, -2, 0]] [[1, -3, -5], [6, 3, 2], [7, -2, 0]]</p>

Questão 1

(Tipo 3)

Questão 1 (a)

Programa	O que será exibido na tela?
<pre># (0.2 ponto) def invert_sinal(a): print("-a =", -a) invert_sinal(3)</pre>	<p>-a = -3</p>

Questão 1 (b)

Programa	O que será exibido na tela?
<pre><i># (0.3 ponto)</i> def invert_sinal(a): a = -a invert_sinal(10) print("-a =", -a)</pre>	<p>Erro na linha 5 (<code>print("-a =", -a)</code>), a não foi definido.</p>

Questão 1 (c)

Programa	O que será exibido na tela?
<pre><i># (0.3 ponto)</i> def invert_sinal(a): a = -a return a a = invert_sinal(5) print("-a =", -a)</pre>	<p style="text-align: center;">-a = -5</p>

Programa

```
# (0.3 ponto)
def inverte_sinal(a):
    c = -c
    print("c invertido = ", c)

a = -1
inverte(a)
b = -2
inverte(b)
if a == -1 and b == -2:
    print("Não foram invertidos")
elif a == 1 and b == 2:
    print("Foram invertidos")
else:
    print("a =", a, "b =", b)
```

O que será exibido na tela?

Resposta 1:

Erro na linha 6 (inverte(a)), a função inverte não foi definida.

Resposta 2 (caso inverte_sinal(a) na linha 6 e inverte_sinal(b) na linha 8):

c invertido = 1

c invertido = 2

Não foram invertidos

As duas respostas serão aceitas.

Questão 1 (e)

Programa	O que será exibido na tela?
<pre># (0.4 ponto) def inverte_lista(l): for i in range(1,l): l[i] = -l[i] return l l = [7, 3, 5] inverte_lista(l) print(l)</pre>	<p>Erro na linha 2 (for i in range(1,l):), não é possível iterar sobre a lista.</p> <p>O correto seria:</p> <pre>for i in range(1,len(l)):</pre>

Questão 1 (f)

As duas respostas serão aceitas.

Programa	O que será exibido na tela?
<pre># (0.5 ponto) def inverte_misterio(m): misterio_m = m.copy() for i in range(len(m)): misterio_m[i][i] = -m[i][i] print(misterio_m) m = [[-1, 3, 5], [6, 3, 2], [7, -2, 0]] misterio_matriz(m) print(m)</pre>	<p>Resposta 1: Erro na linha 8 (<code>misterio_matriz(m)</code>), a função <code>misterio_matriz</code> não foi definida.</p> <p>Resposta 2 (caso <code>inverte_misterio(m)</code> na linha 8): [[1, 3, 5], [6, -3, 2], [7, -2, 0]] [[1, 3, 5], [6, -3, 2], [7, -2, 0]]</p>

Questão 2

(Tipo 1)

Questão 2

(2.0 pontos) Carla estava se preparando para estudar ... anotações de aula para um algoritmo:

- 1. Definir uma lista com n elementos
- 2. Definir uma variável i como $\neq 2$
- 3. Para cada elemento da posição 0 até a posição $n-i$
 - Se o elemento subsequente ao elemento atual for maior que o elemento atual, troque esses elementos
- 4. Escrever na tabela de iterações o conteúdo da lista

Nota: Os passos 3 e 4 devem ser executados no total $n-1$ vezes. Após cada execução, lembrar de incrementar o valor de i em 1.

Questão 2 (a)

(0.5 ponto) Qual o algoritmo de ordenação a que se referem as anotações de Carla?

Ordenação por bolha ou Bubble sort.

Questão 2 (b)

(1.5 ponto) Considere a seguinte lista de números inteiros:

3	7	9	2	5
---	---	---	---	---

Preencha a tabela com os resultados de cada iteração do algoritmo de ordenação que estudamos nesta aula.

Lista Original	3	7	9	2	5
Iteração 1	3	7	2	5	9
Iteração 2	3	2	5	7	9
Iteração 3	2	3	5	7	9

Questão 2

(Tipo 2)

Questão 2

(2.0 pontos) Carla estava se preparando para estudar ... anotações de aula para um algoritmo:

- 1. Definir uma lista com n elementos
- 2. Divida a lista em duas partes: uma ordenada (com o primeiro elemento da lista) e uma não ordenada (com os demais elementos)
- 3. Selecione e remova o primeiro elemento da parte não ordenada (deixe um buraco nessa posição da lista)
- 4. Desloque para a direita os elementos da parte ordenada (do último ao primeiro) até encontrar a posição onde encaixar o elemento selecionado no item 3
- 5. Atualize a divisão da lista: agora a parte ordenada contém um elemento a mais
- 6. Repita os itens 3, 4, e 5 até que a parte ordenada contenha ~~$n-1$~~ n elementos

Questão 2 (a)

(0.5 ponto) Qual o algoritmo de ordenação a que se referem as anotações de Carla?

Ordenação por inserção ou Insertion sort.

Questão 2 (b)

(1.5 ponto) Considere a seguinte lista de números inteiros:

3	7	9	2	5
---	---	---	---	---

Preencha a tabela com os resultados de cada iteração do algoritmo de ordenação que estudamos nesta aula.

Lista Original	3	7	9	2	5
Iteração 1	3	7	9	2	5
Iteração 2	3	7	9	2	5
Iteração 3	2	3	7	9	5
Iteração 4	2	3	5	7	9

Questão 2

(Tipo 3)

Questão 2

(2.0 pontos) Carla estava se preparando para estudar ... anotações de aula para um algoritmo:

- 1. Definir a lista
- 2. Para cada elemento da lista (percorrer a lista do primeiro ao ~~último~~ penúltimo elemento)
 - Descubra a posição do menor elemento a partir do elemento atual
 - Se o elemento encontrado for menor que o elemento atual, troque esses valores
 - Escreva na tabela de iterações o conteúdo atual da lista

Questão 2 (a)

(0.5 ponto) Qual o algoritmo de ordenação a que se referem as anotações de Carla?

Ordenação por seleção ou Selection sort.

Questão 2 (b)

(1.5 ponto) Considere a seguinte lista de números inteiros:

3	7	9	2	5
---	---	---	---	---

Preencha a tabela com os resultados de cada iteração do algoritmo de ordenação que estudamos nesta aula.

Lista Original	3	7	9	2	5
Iteração 1	2	7	9	3	5
Iteração 2	2	3	9	7	5
Iteração 3	2	3	5	7	9
Iteração 4	2	3	5	7	9

Questão 3

(Tipo 1, 2 e 3)

Questão 3 (a)

Como exemplo de chamada da função `busca_sequencial`, considere o seguinte programa:

`programa_item_c.py`

```
lista = [11, 19, 12, 13, 19, 43, 32, 41, 11, 12, 24]
chave = 12
pos = busca_sequencial(lista, chave)
if (pos == -1):
    print("Não existe")
else:
    print("Chave localizada na posição = ", pos)
```

Questão 3 (a)

(0.8 ponto) Implemente aqui a função `busca_sequencial`:

```
def busca_sequencial(lista, chave):  
    for i in range(len(lista)):  
        if lista[i] == chave:  
            return i  
    return -1
```

Podemos ter várias soluções.
O programa apresentado é uma
possível resposta.

Questão 3 (b)

(0.6 ponto) Você considera que é possível implementar a Busca Binária na situação descrita no item a? Ou seja, a 3a linha do `programa_item_c.py` seria substituída por: `pos = busca_binaria(lista, chave)`.

Responda SIM ou NÃO e **justifique** a sua escolha.

NÃO, porque para a busca binária funcionar a lista precisa estar ordenada, e a lista não está ordenada.

(0.8 ponto) Vamos ajudar Mário a compreender o algoritmo de Busca Binária? Explique aqui (use português, não código de programação) o algoritmo da busca binária. Indique (e explique!) se João estava certo ao afirmar que a busca binária é mais eficiente que a busca sequencial.

A busca binária parte do pressuposto de que a lista está ordenada. A ideia é a seguinte: primeiro verifique se a chave de busca é igual ao valor da posição do meio da lista. Caso seja igual, devolva esta posição. Caso o valor desta posição seja maior, então repita o processo mas considere que a lista tem metade do tamanho, indo até posição anterior a do meio. Caso o valor desta posição seja menor, então repita o processo mas considere que a lista tem metade do tamanho e inicia na posição seguinte a do meio.

João estava certo. A busca binária é mais eficiente que a busca sequencial porque faz menos comparações, uma vez que a cada iteração o espaço de busca está sendo reduzido pela metade.

Podemos ter várias respostas para essa questão.

Questão 3 (d): Tipo 1

(1.0 ponto) Considere a seguinte lista $[3, 5, 7, 9, 11, 12, 13, 14]$. A chave a ser procurada é o **número 3**. Complete cada uma das iterações da busca binária, seguindo o modelo abaixo. ... Ao final, indique se a chave procurada foi ou não encontrada. Em caso positivo, qual é a sua posição na lista?

	3	5	7	9	11	12	13	14
Iteração 1	I			M				F
Iteração 2	I	M	F					
Iteração 3	IMF							

A chave foi encontrada na posição 0 (zero).

Questão 3 (d): Tipo 2

(1.0 ponto) Considere a seguinte lista [3, 5, 7, 9, 11, 12, 13, 14]. A chave a ser procurada é o **número 15**. Complete cada uma das iterações da busca binária, seguindo o modelo abaixo. ... Ao final, indique se a chave procurada foi ou não encontrada. Em caso positivo, qual é a sua posição na lista?

	3	5	7	9	11	12	13	14
Iteração 1	I			M				F
Iteração 2					I	M		F
Iteração 3							IM	F
Iteração 4								IMF

A chave não foi encontrada.

Questão 3 (d): Tipo 3

(1.0 ponto) Considere a seguinte lista [3, 5, 7, 9, 11, 12, 13, 14]. A chave a ser procurada é o **número 14**. Complete cada uma das iterações da busca binária, seguindo o modelo abaixo. ... Ao final, indique se a chave procurada foi ou não encontrada. Em caso positivo, qual é a sua posição na lista?

	3	5	7	9	11	12	13	14
Iteração 1	I			M				F
Iteração 2					I	M		F
Iteração 3							IM	F
Iteração 4								IMF

A chave foi encontrada na posição 7.

Questão 4

(Tipo 1)

Questão 4

(3.0 pontos) Continuando seus estudos sobre funções, João implementou a seguinte função recursiva:

```
1: def func(n):  
2:     print("n = ", n)  
3:     if n == 0:  
4:         return 0  
5:     return n + func(n-1)
```

Questão 4 (a)

As duas respostas serão aceitas.

(0.4 ponto) O que o código `print("r = ", func(4))` irá imprimir na tela quando executado?

Resposta 1:

Erro na linha 3. A função não será executada porque está faltando os dois pontos depois do comando `if`.

Resposta 2 (caso `if n == 0:`)

```
n = 4
n = 3
n = 2
n = 1
n = 0
r = 10
```

Questão 4 (b)

As duas respostas serão aceitas.

(0.2 ponto) Do ponto de vista matemático, pode-se afirmar que a função implementada por João resolve qual problema?

Resposta 1:

A função não será executada porque está faltando os dois pontos depois do comando `if` na linha 3.

Resposta 2:

Soma dos números de 0 a n.

Questão 4 (c)

(0.4 ponto) Caso as linhas 3 e 4 fossem removidas da função, o que ocorreria quando a mesma fosse executada?

O programa entra em loop infinito.

Questão 4 (d)

(1.0 ponto) Implemente a função *func* com uma solução iterativa (sem usar recursão).

```
def func(n):  
    soma = 0  
    for i in range(n+1):  
        soma = soma + i  
    return soma
```

Podemos ter várias soluções.
O programa apresentado é uma possível resposta.

Questão 4 (e)

(1.0 ponto) Implemente uma função recursiva chamada `fat_n_(n)` que recebe como parâmetro um número natural n ($n \geq 0$) e retorna o fatorial de n . Por exemplo, a função irá retornar: 1 se $n == 0$; 1 se $n == 1$; 2 se $n == 2$; 6 se $n == 3$; 24 se $n == 4$ e assim por diante.

```
def fat_n_(n):  
    if n == 0:  
        return 1  
    elif n == 1:  
        return 1  
    else:  
        return n * fat_n_(n-1)
```

Podemos ter várias soluções.
O programa apresentado é uma possível resposta.

Questão 4

(Tipo 2)

Questão 4

(3.0 pontos) Continuando seus estudos sobre funções, João implementou a seguinte função recursiva:

```
1: def func (n):
2:     print("n = ", n)
3:     if n == 1
4:         return 0
5:     if n == 2
6:         return 1
5:     return func(n-1) + func(n-2)
```

Questão 4 (a)

As duas respostas serão aceitas.

(0.4 ponto) O que o código `print("r = ", func(4))` irá imprimir na tela quando executado?

Resposta 1:

Erro nas linhas 3 e 5. A função não será executada porque está faltando os dois pontos depois do comando `if`.

Resposta 2 (caso `if n==1:` e `if n==2:`)

```
n = 4
n = 3
n = 2
n = 1
n = 2
r = 2
```

Questão 4 (b)

As duas respostas serão aceitas.

(0.2 ponto) Do ponto de vista matemático, pode-se afirmar que a função implementada por João resolve qual problema?

Resposta 1:

A função não será executada porque está faltando os dois pontos depois do comando `if` nas linhas 3 e 5.

Resposta 2:

Números de Fibonnaci.

Questão 4 (c)

As duas respostas serão aceitas.

(0.4 ponto) Caso as linhas 3, 4, 5 e 6 fossem removidas da função, o que ocorreria quando a mesma fosse executada?

Resposta 1:

O programa entra em loop infinito.

Resposta 2:

O programa executará apenas a linha 2 (`print("n = ", n)`). Nesse caso estamos considerando que a linha 5 (que deveria ser a linha 7) também foi removida.

Questão 4 (d)

(1.0 ponto) Implemente uma função **não** recursiva chamada `fat_n_(n)` que recebe como parâmetro um número natural n ($n \geq 0$) e retorna o fatorial de n . Por exemplo, a função irá retornar: 1 se $n == 0$; 1 se $n == 1$; 2 se $n == 2$; 6 se $n == 3$; 24 se $n == 4$ e assim por diante.

```
def fat_n_(n):  
    total, k = 1, 1  
    while k <= n:  
        total, k = total * k, k + 1  
    return total
```

Podemos ter várias soluções.
O programa apresentado é
uma possível resposta.

Questão 4 (e)

(1.0 ponto) Implemente a função do item d) utilizando recursão.

```
def fat_n_(n):  
    if n == 0:  
        return 1  
    elif n == 1:  
        return 1  
    else:  
        return n * fat_n_(n-1)
```

Podemos ter várias soluções.
O programa apresentado é uma
possível resposta.

Questão 4

(Tipo 3)

Questão 4

(3.0 pontos) Continuando seus estudos sobre funções, João implementou a seguinte função recursiva:

```
1: def func(n):  
2:     print("n = ", n)  
3:     if n == 0 or n == 1:  
4:         return 1  
5:     return n * func(n-1)
```

Questão 4 (a)

(0.4 ponto) O que o código `print("r = ", func(4))` irá imprimir na tela quando executado?

```
n = 4
n = 3
n = 2
n = 1
r = 24
```

Questão 4 (b)

(0.2 ponto) Do ponto de vista matemático, pode-se afirmar que a função implementada por João resolve qual problema?

Fatorial.

Questão 4 (c)

(0.4 ponto) Caso as linhas 3 e 4 fossem removidas da função, o que ocorreria quando a mesma fosse executada?

O programa entra em loop infinito.

Questão 4 (d)

(1.0 ponto) Implemente a função *func* com uma solução iterativa (sem usar recursão).

```
def func(n):  
    total, k = 1, 1  
    while k <= n:  
        total, k = total * k, k + 1  
    return total
```

Podemos ter várias soluções.
O programa apresentado é uma possível resposta.

Questão 4 (e)

(1.0 ponto) Implemente uma função recursiva chamada `soma_n_naturais(n)` que recebe como parâmetro um número natural n ($n \geq 0$) e retorna a soma de todos os naturais de 0 até n (inclusive). Por exemplo, a função irá retornar: 0 se $n == 0$; 1 se $n == 1$; 3 se $n == 2$; 6 se $n == 3$ e assim por diante.

```
def soma_n_naturais(n):  
    print("n = ", n)  
    if n == 0  
        return 0  
    return n + soma_n_naturais(n-1)
```

Podemos ter várias soluções.
O programa apresentado é uma possível resposta.

Obrigada,
Pessoal!

AMO
PROGRAMAR



! ! !
D - - -
X - - -
T - - -
- - -
o - - -
n - - -
! ! !