

MAC 110 - Introdução à Computação
IMEUSP - BMAC – 1º Semestre de 2017
Primeira Prova – 03/05/2017

Questão	1	2	3	4	5	Total
Valor	2	2	2	2	2	10.0
Nota						

Nome: _____ NUSP: _____

Questão 1 - 2 pontos:

Diga o que será impresso pelo programa abaixo. O número a ser digitado, isto é, o valor solicitado pelo programa é o seu número USP:

```
nusp = int(input("Entre com o seu número USP: "))
p = 1
s = 0

while nusp > 0:
    k = nusp % 10
    s = p * k + s
    p = p * 10
    nusp = nusp // 10
    print(nusp, " * ", s)
```

Exemplos de saída:

```
Entre com o seu número USP: 1234567
123456 * 7
12345 * 67
1234 * 567
123 * 4567
12 * 34567
1 * 234567
0 * 1234567
```

```
Entre com o seu número USP: 2030405
203040 * 5
20304 * 5
2030 * 405
203 * 405
20 * 30405
2 * 30405
0 * 2030405
```

```
Entre com o seu número USP: 987045
```

98704 * 5
9870 * 45
987 * 45
98 * 7045
9 * 87045
0 * 987045

Questão 2 - 2 Pontos

Faça um programa que leia triplas de números inteiros representando datas e imprima ao final a data mais recente. O programa deve parar quando encontrar a data 0 0 0 (três zeros). Pode supor que as datas são válidas e que há uma ou mais.

Exemplo – Se os dados forem:

```
3 5 2009
15 10 2010
23 12 2003
13 9 2010
30 10 2003
0 0 0
```

O programa deve imprimir:

A data mais recente é 15/10/2010

```
def data_mais_recente():

    # le a primeira data para considerá-la como a maior
    dia_maior = int(input("Entre com o dia:"))
    mes_maior = int(input("Entre com o mes:"))
    ano_maior = int(input("Entre com o ano:"))

    # le cada data e compara com a maior até o momento
    while True:
        dia = int(input("Entre com o dia:"))
        mes = int(input("Entre com o mes:"))
        ano = int(input("Entre com o ano:"))

        # verifica se é a última
        if dia == 0 and mes == 0 and ano == 0: break

        # compara com a maior e troca se for o caso
        if ano > ano_maior:
            dia_maior, mes_maior, ano_maior = dia, mes, ano
        elif ano == ano_maior:
            if mes > mes_maior:
                dia_maior, mes_maior, ano_maior = dia, mes, ano
            elif mes == mes_maior:
                if dia > dia_maior:
                    dia_maior, mes_maior, ano_maior = dia, mes, ano

    # imprime a maior data
    print("A data mais recente é ", dia_maior, "/", mes_maior, "/",
          ano_maior)

data_mais_recente()
```

Outra solução – transformando os 3 números em um só (aammdd) na ordem correta de comparação:

```
def data_mais_recente():
```

```

# le a primeira data para considerá-la como a maior
dia = int(input("Entre com o dia:"))
mes = int(input("Entre com o mes:"))
ano = int(input("Entre com o ano:"))
# transforma num inteiro: aammdd
data_maior = dia + mes * 100 + ano * 10000

# le cada data e compara com a maior até o momento
while True:
    dia = int(input("Entre com o dia:"))
    mes = int(input("Entre com o mes:"))
    ano = int(input("Entre com o ano:"))
    # transforma num inteiro: aammdd
    data = dia + mes * 100 + ano * 10000

    # verifica se é a última
    if data == 0: break

    # compara com a maior e troca se for o caso
    if data > data_maior:
        data_maior = data

# transforma data_maior em 3 números dd mm aa
dd = data_maior % 100
mm = data_maior // 100 % 100
aa = data_maior // 10000
# imprime
print("A data mais recente é ", dd, "/", mm, "/", aa)

```

data_mais_recente()

Outra solução – comparando tuplas

```
def data_mais_recente():
```

```

    # lê a primeira - supondo a maior até o momento
    diaM = int(input("Entre com o dia:"))
    mesM = int(input("Entre com o mes:"))
    anoM = int(input("Entre com o ano:"))
    # le cada data e compara com a maior até o momento
    while True:
        dia = int(input("Entre com o dia:"))
        mes = int(input("Entre com o mes:"))
        ano = int(input("Entre com o ano:"))
        # verifica se é a última
        if dia == 0: break
        # compara a tupla - troca se for o caso
        if (ano, mes, dia) > (anoM, mesM, diaM):
            diaM, mesM, anoM = dia, mes, ano

    # imprime
    print("A data mais recente é ", diaM, "/", mesM, "/", anoM)

```

data_mais_recente()

Questão 3 - 2 Pontos

Dado $N \geq 0$, inteiro e par, faça um programa que calcule a seguinte somatória:

$0! + 2! + 4! + \dots + N!$, ou seja a soma dos fatoriais dos números pares de 0 a N .

Usando repetição encaixada:

```
N = int(input("Entre com o valor de N - par e inteiro: "))

soma = 0

for k in range(0, N + 1, 2):
    # calcula o fatorial de k
    kfat = 1
    for i in range(2, k + 1): kfat = kfat * i
    # adiciona o fatorial de k à soma
    soma = soma + kfat

# resultado
print("A soma dos fatoriais pares de 0 até ", N, " = ", soma)
```

Sem repetição encaixada – usando o fato de que:

$(k + 2)! = k! * (k + 1) * (k + 2)$

```
N = int(input("Entre com o valor de N - par e inteiro: "))

soma = 0
kfat = 1

for k in range(0, N + 1, 2):
    # adiciona o fatorial de k à soma
    soma = soma + kfat
    # calcula o fatorial do próximo k
    kfat = kfat * (k + 1) * (k + 2)

# resultado
print("A soma dos fatoriais pares de 0 até ", N, " = ", soma)
```

Outra forma – Calcular todos os fatoriais de 0 a N e só somar os fatoriais pares.

Calcula todos os fatoriais de 0 a N e soma apenas os pares

```
N = int(input("Entre com o valor de N - par e inteiro: "))
```

```
# inicia com o fatorial de 0
```

```
soma = 1
```

```
kfat = 1

# todos de 1 a N
for k in range(1, N + 1):
    # calcula o fatorial de k
    kfat = kfat * k
    # soma apenas os fatoriais pares
    if k % 2 == 0: soma = soma + kfat

# resultado
print("A soma dos fatoriais pares de 0 até ", N, " = ", soma)
```

Questão 4 - 2 Pontos

Escreva trechos de programa equivalentes aos trechos abaixo usando o comando **for** em vez do comando **while**. Pode supor que os valores de n e k são inteiros positivos.

a)

```
i = 1
while i < n:
    print(i)
    i = i + 1
```

```
for i in range(1, n):
    print(i)
```

b)

```
i = 2
while i <= n:
    print(i)
    i = i + k
```

```
for i in range(2, n + 1, k):
    print(i)
```

c)

```
i = n
while i >= 0:
    print(i)
    i = i - k
```

```
for i in range(n, -1, -k):
    print(i)
```

Questão 5 - 2 Pontos

Escreva trechos de programa equivalentes aos trechos abaixo usando o comando **while** em vez do comando **for**. Pode supor que os valores de n e k são inteiros positivos.

a)

```
for i in range(-k, k):  
    print(i)
```

```
i = -k  
while i < k:  
    print(i)  
    i = i + 1
```

b)

```
for i in range(3, n, k):  
    print(i)
```

```
i = 3  
while i < n:  
    print(i)  
    i = i + k
```

c)

```
for i in range(n, 2, -k):  
    print(i)
```

```
i = n  
while i > 2:  
    print(i)  
    i = i - k
```