



Processamento da Informação

Simulado 1

Prof. Jesús P. Mena-Chalco
CMCC/UFABC

Q2/2018

Questão 1 – a

```
static int enigmaA ( int num ) {  
    int i, t=0;  
  
    for (i=0; i<num; i=i+1) {  
        t = t+i;  
        i = i-1;  
    }  
  
    return t;  
}
```

```
int x;  
x = enigmaA(10);
```

Laço infinito:
O método não devolve resposta

Questão 1 – b

```
static int enigmaB ( int p, int q ) {  
    int i, j, soma=0;  
  
    for (i=p; i<=q; i=i+1) {  
        for (j=1; j<=i; j=j+1) {  
            soma = soma+j;  
        }  
    }  
    return soma - (soma/10)*10;  
}
```

```
int x;  
x = enigmaB(3, 5);  
  
x = 1
```

Questão 2

```
static void triangulo(int n) {
    int i, j;

    for (i=1; i<=n; i=i+1) {
        for(j=n; j>=1; j=j-1) {
            if (i>j) {
                System.out.printf("o");
            }
            else {
                System.out.printf("*");
            }
        }
        System.out.printf("\n");
    }
}
```

Para n = 5

1,5	1,4	1,3	1,2	1,1
2,5	2,4	2,3	2,2	2,1
3,5	3,4	3,3	3,2	3,1
4,5	4,4	4,3	4,2	4,1
5,5	5,4	5,3	5,2	5,1

Questão 3

```
static int FFF ( int n ) {  
    int i, j, k, soma=0;  
  
    for (i=0; i<n; i=i+1) {  
        soma = soma+1;  
  
        for (j=0; j<n; j=j+1) {  
            for (k=0; k<n; k=k+1) {  
                soma = soma+1;  
            }  
        }  
    }  
  
    return soma;  
}
```

Qual das seguintes opções é correta?

- (a) FFF, não devolve valor inteiro.
- (b) FFF, dado um inteiro positivo n , devolve $1+n^2$.
- (c) FFF, dado um inteiro positivo n , devolve $1+2n$.
- (d) FFF, dado um inteiro positivo n , devolve $n+n^2$.
- (e) FFF, dado um inteiro positivo n , devolve $n+n^3$.
- (f) Nenhuma das anteriores.

Questão 4

O método novoPI promete calcular uma boa aproximação para o valor de PI (3.1415...) considerando somente os primeiros 100 termos da equação:

$$\frac{\pi^2}{8} = \frac{1}{1^2} + \frac{1}{3^2} + \frac{1}{5^2} + \frac{1}{7^2} + \frac{1}{9^2} + \dots$$

```
1. static double novoPI () {
2.     int i;
3.     double soma=0;
4.
5.     for ( _____ ) {
6.         soma = soma + 1.0/( _____ );
7.     }
8.
9.     return Math.sqrt(soma*8);
10. }
```

Versão I: linha 5: (i=1; i<=100; i=i+1)

linha 6: (2*i-1)*(2*i-1)

Versão II: linha 5: (i=1; i<=100; i=i+2)

linha 6: (i)*(i)

Versão III: linha 5: (i=1; i<=200; i=i+2)

linha 6: (i)*(i)

Questão 4

```
static double novoPI () {  
    int i;  
    double soma=0;  
  
    for (i=1; i<=100; i=i+1) {  
        soma = soma + 1.0/( (2*i-1)*(2*i-1) );  
    }  
  
    return Math.sqrt(soma*8);  
}
```

```
static double novoPI2 () {  
    int i;  
    double soma=0;  
  
    for (i=1; i<=200; i=i+2) {  
        soma = soma + 1.0/( i*i );  
    }  
  
    return Math.sqrt(soma*8);  
}
```

Selecione uma opção:

- (a) Somente a versão I está correta.
- (b) Somente a versão II está correta.
- (c) Somente a versão III está correta.
- (d) A versão I e II estão corretas.
- (e) A versão I e III estão corretas.
- (f) A versão II e III estão corretas.
- (g) As 3 versões estão corretas.
- (h) Nenhuma versão está correta.

Questão 5

```
static int enigmaRA (int ra) {
    int aux, i=1, t=5;

    while (i<=5) {
        aux = ra%10;
        ra = ra/10;
        i = i+1;
        if (aux>t) {
            t = aux;
        }
    }

    return t;
}
```

O método devolve
O maior dígito entre 5 e
os últimos 5 dígitos
de seu RA

Exemplos:

ra = 123456789 → 9

ra = 987654321 → 5

ra = 323434324 → 5