

Primer Parcial - 24 de Septiembre de 2011 - 1ª parte

- 1) La programación estructurada se basa en la utilización de las estructuras de control:**
 - a. Selección, iteración y secuencia.

 - 2) De acuerdo a lo dictado durante el curso, si se quiere hacer un programa que sume los números naturales del 1 al 53, el código deberá incluir necesariamente un:**
 - a. for

 - 3) Indique cual de las siguientes afirmaciones sobre sistemas operativos es correcta:**
 - a. El sistema operativo se encarga de la interacción entre los programas y el hardware.

 - 4) De acuerdo a lo dictado durante el curso, si se quiere hacer un programa que sume los números naturales mayores que 0 hasta que la suma sea mayor o igual que 53, el código deberá incluir necesariamente un:**
 - b. while

 - 5) Indique cuál de las siguientes afirmaciones es incorrecta.**
 - c. El alcance de las variables de una función es local y se incorporan como nuevas variables al entorno iterativo (consola).

 - 6) De acuerdo al enfoque seguido en el curso, al terminar la ejecución del siguiente código:**
 - c. El valor de i está indeterminado.

 - 7) Al terminar la ejecución del siguiente código:**
 - b. x vale 2, t vale 8 y z vale 5.

 - 8) Una característica de la programación estructurada es que los programas:**
 - c. tienen un único punto de entrada y un único punto de salida

 - 9) Indique cual de las siguientes afirmaciones es incorrecta:**
 - d. Un lenguaje interpretado puede ser ejecutado en cualquier plataforma siempre y cuando exista disponible un link editor para dicha plataforma.

 - 10) Indique cual de las siguientes afirmaciones es correcta:**
 - c. La memoria caché es extremadamente rápida.
-

Primer Parcial - 24 de Septiembre de 2011 - 2ª parte**Problema 1 (8 puntos)**

```
function raiz = calcularRaiz(x, e)

    s = x / 2;
    r = s - (s^2 - x) / (2*s);

    while (abs(s-r) >= e)
        s = r;
        r = s - (s^2 - x) / (2*s);
    end

    raiz = r;
```

Problema 2 (12 puntos)

```
function c = intercalarRaro(a, b)

    n = length(a);
    m = length(b);
    h = ceil(n / (m+1));

    s = 0;
    ib = 1;
    c = [];

    for ia = 1:n
        c = [c a(ia)];
        s = s + 1;
        if s == h && ib <= m
            s = 0;
            c = [c b(ib)];
            ib = ib + 1;
        end
    end

    c = [c b(ib:m)];
```

Problema 3 (4, 2 y 4 puntos, respectivamente)**Parte a)**

```
function R = repeticiones(M, x)

    [m, n] = size(M);

    R = [];

    for i = 1:m
        for j = 1:n
            if M(i,j) == x
                R = [R; i j];
            end
        end
    end

end
```

**Parte b)**

```
function pertenece = perteneceColumna(M, x, c)

    [m, n] = size(M);
    pertenece = 0;
    i = 1;

    while i <= m & ~pertenece
        if M(i,c) == x
            pertenece = 1;
        end

        i = i + 1;
    end
```

Parte c)

```
function C = contarRepeticiones(M)

    [m, n] = size(M);
    C = [];

    for i = 1:m
        for j = 1:n
            x = M(i,j);
            pertenece = perteneceColumna(C, x, 1);

            if ~pertenece
                R = repeticiones(M, x);
                C = [C; x size(R,1)];
            end
        end
    end
```