



Solución de Examen de Diciembre – 20 de Diciembre de 2014

Problema 1	10 (3,2,3,1,1) pts	
-------------------	--------------------	--

- a) 27
- b) 0 10000010 1011000000000000000000
- c) 0 11100010 0000001100000000000000
- d) [2 0 0 0 1]
- e) No se puede representar.

Problema 2	25 (13,12) pts	
-------------------	----------------	--

a)

```
function res = esMayorSuma(M,valor)
    [fil,col] = size(M);
    suma = 0;
    i = 1;
    while i <= fil && suma <= valor
        j = 1;
        while j <= col && suma <= valor
            suma = suma + M(i,j);
            j = j + 1;
        end
        i = i + 1;
    end

    if (suma > valor)
        res = 1;
    else
        res = 0;
    end
end
```

b)

```
function res = sumaMayor(M,valor)
    [fil,col] = size(M);
    if (fil == 1)
        if (M(i,j) > valor)
            res = M(i,j);
        else
            res = 0;
        end
    else
        res = sumaMayor(M(1:fil/2,1:col/2),valor) +
            sumaMayor(M(1:fil/2,col/2 + 1:col),valor) +
            sumaMayor(M(fil/2 + 1:fil,1:col/2),valor) +
            sumaMayor(M(fil/2 + 1:fil,col/2 + 1:col),valor);
    end
end
```



Problema 3	25 (5, 10, 10) pts	
-------------------	--------------------	--

a)

```
function res = fibonacci_recursivo(n)
    if (n==1)
        res = 1;
    elseif (n==2)
        res = 2;
    else
        res = fibonacci_recursivo(n-1) + fibonacci_recursivo(n-2);
    end
end
```

b)

```
function res = fibonacci_iterativo(n)
    if (n==1)
        res = 1;
    elseif (n==2)
        res = 2;
    else
        penult=1;
        ult=2;
        for i=3:n
            nue = penult + ult;
            penult = ult;
            ult = nue;
        end
        res = ult;
    end
end
```

c)

Alternativa 1

```
function res = fibonacci(n)
    res = fib_cabecal(1,2,n-2)

function res = fib_cabecal(ant,act,n)
    nuevo = ant + act;
    if (n == 1)
        res = nuevo;
    else
        res = fib_cabecal(act,nuevo,n-1)
    end
end
```



Alternativa 2

```
function res = fibonacci(n)
    res = fib_cabecal(1,2,3,n)

function res = fib_cabecal(ant,act,proximo,n)
    nuevo = ant + act;
    if (proximo == n)
        res = nuevo;
    else
        res = fib_cabecal(act,nuevo,proximo+1,n)
    end
```

Problema 4	20 (5, 5, 10) ptos	
-------------------	--------------------	--

a)

```
function res=productoEscalarIt(u, v)
    n = length(u);
    res = 0;
    for i = 1:n
        res = res + v(i) * u(i);
    end
```

b1)

```
function res=productoEscalarRec(u, v)
    n=length(u);
    if n==0
        res = 0;
    else
        res = v(1)*u(1) + productoEscalarRec(u(2:n),v(2:n));
    end
```

b2)

```
function res=productoEscalarRec(u, v)
    n=length(u);
    if n==1 %no se considera el caso en que n==0.
        res = v(1)*u(1);
    else
        res = v(1)*u(1) + productoEscalarRec(u(2:n),v(2:n));
    end
```

c)

```
function w=multiplicacionPolinomiosIt(u,v)
    n = length(u);
    m = length(v);
    grado_w = n-1 + m-1;
    w=zeros(1,grado_w+1);
    for i = 1:n
        for j = 1:m
            w(i+j-1) = w(i+j-1) + u(i)*v(j);
        end
    end
end
```



Problema 5	20 (10, 10) ptos	
-------------------	------------------	--

a)

```
function mc=MaxColsIter(D,F,C,ncols)
    LD=length(D);
    mc=zeros(1,ncols);
    for i=1:LD
        if D(i)>mc(C(i))
            mc(C(i))=D(i);
        end
    end
end
```

b)

```
function mc=MaxColsRec(D,F,C,ncols)
    LD=length(D);
    if LD==0
        mc=zeros(1,ncols);
    else
        mc=MaxColsRec(D(1:LD-1),F(1:LD-1),C(1:LD-1),ncols);
        if D(LD)>mc(C(LD))
            mc(C(LD))=D(LD);
        end
    end
end
```