

Examen – 15 de febrero de 2023

- **Duración del examen: 3 Hs.**
- **No** se podrá utilizar ningún tipo de material (apuntes, libro, calculadora, etc). **Apague** su teléfono celular.
- **Escriba las hojas de ambos lados con buena letra.** Las partes no legibles se considerarán no escritas.
- Sólo se contestarán preguntas sobre interpretación de la letra.
- **En la primera hoja, en el ángulo superior derecho, ponga su nombre, número de cédula y cantidad de hojas. En las demás hojas ponga nombre y número de cédula. En todas las páginas ponga número de página.**

Para la resolución de los diferentes ejercicios **solamente** podrá utilizar las siguientes funciones brindadas por **Octave**:

- `length()` y `size()`
- `mod()` y `rem()`
- `fix()`, `floor()`, `ceil()` y `round()`
- `zeros()`, `ones()` y `M'` (la transpuesta)

Notas: - En todos los ejercicios se deben usar las estructuras de control adecuadas para cada caso. Por ejemplo: se controlará el uso correcto de `for` y `while`.
- No se deben realizar más iteraciones o invocaciones recursivas que las necesarias para resolver el problema

Problema 1	10 (2,2,2,2,2) ptos	
-------------------	---------------------	--

En todos los siguientes ejercicios deben justificarse los resultados:

- $a=6, b=0$ y $c=4$ o $a=10, b=4, c=0$ o $a=9, b=3, c=1$ o $a=8, b=2, c=2 \dots etc$
- 01111 (ocurre overflow dado que el resultado correcto debería tener signo negativo).
- 00000 (no ocurre overflow).
- $123021_4 = 11011001001_2$.
- $1\ 1001\ 0100000 = -1,01_2 * 2^{(9-7)} = -101_2 = -5$

Problema 2	20 (10,10) ptos	
-------------------	-----------------	--

a)

```
function vout = acumVec(vin)
    n = length(vin);
    vout = zeros(1, n);
    acum = 0;

    for i = n:-1:1
        acum = acum + vin(i);
        vout(i) = acum;
    end
end
```

b) Opción 1:

```
function i = corteVec(vin)
    n = length(vin);
    sumaDer = 0;
    for j = 2:n
        sumaDer = sumaDer + vin(j);
    end
    i = 1;
    sumaIzq = 0;
    while i <= n && sumaIzq < sumaDer
        sumaIzq = sumaIzq + vin(i);
        sumaDer = sumaDer - vin(i+1);
        i = i + 1;
    end
end
```

Opción 2 (usando la parte anterior):

```
function i = corteVec(vin)
    n = length(vin);
    sumaDer = acumVec(vin);
    i = 1;
    sumaIzq = 0;
    while i < n && izq < sumaDer(i+1)
        sumaIzq = sumaIzq + vin(i);
        i = i + 1;
    end
end
```

Problema 3	10 ptos	
-------------------	---------	--

```
function vout = cortarBilatX(vin, x)
    n = length(vin);
    if n==0
        vout=[];
    else
        if vin(1)~=x && vin(n)~=x
            vout=cortarBilatX(vin(2:n-1), x);
        elseif vin(1)~=x
            vout=cortarBilatX(vin(2:n), x);
        elseif vin(n)~=x
            vout=cortarBilatX(vin(1:n-1), x);
        else
            vout=vin;
        end
    end
end
```

Problema 4	15 (5,10) ptos.
-------------------	-----------------

a)

```
function t = trazaDisp(f,c,v)
    t = 0;
    n = length(f);
    for i = 1:n
        if f(i)==c(i)
            t = t + v(i);
        end
    end
end
```

b)

```
function cf = comienzoFilas(f,c,v)
    n = length(f);
    if n == 0
        cf = [];
    elseif n == 1
        cf = [1];
    else
        cf = comienzoFilas(f(1:n-1),c,v);
        if f(n)~=f(n-1)
            cf=[cf,n];
        end
    end
end
```

Problema 5	15 ptos
-------------------	---------

```
function dif = difRec(v1, v2)

    lv1 =length(v1);
    lv2 =length(v2);

    if lv1==0 || lv2 ==0
        dif=[v1,v2];
    elseif v1(1) < v2(1)
        dif = [v1(1), difRec(v1(2:lv2), v2)];
    elseif v1(1) > v2(1)
        dif = [v2(1), difRec(v1, v2(2:lv2))];
    else
        dif = difRec(v1(2:lv2), v2(2:lv2));
    end
end
```

Problema 6	28 (10,8,10) pts
-------------------	------------------

a)

```
function es = esTriangInf(M)
    [n,m] = size(M);
    es = 1 ;
    i = 1;
    while i <= n && es
        j=i+1;
        while j <= m && es
            es = M(i,j)==0;
            j=j+1;
        end
        i=i+1;
    end
```

b)

```
function es = esCeroVec(vec)
    n = length(vec);
    if n==0
        es = 1;
    elseif vec(1)~=0
        es = 0;
    else
        es = esCeroVec(vec(2:n));
    end
end
```

c)

```
function es = esTriangInfRec(M)
    n = size(M,2);
    if n<=1
        es = 1;
    else
        es = esCeroVec(M(1,2:n)) && esTriangInfRec(M(2:n,2:n));
    end
end
```