

COMPUTACIÓN 1
Instituto de Computación

Examen - 22 de julio de 2022
Parte 1 – 22 puntos

- Duración de esta parte de la prueba: 45 minutos.
- Duración total incluyendo descansos: 200 min (45' – 10' – 75' – 10' - 60') = 3h:20min
- Puntaje total de la prueba: 100 puntos.
- No se podrá utilizar ningún tipo de material (apuntes, libro, calculadora, etc). Apague su teléfono celular.
- Sólo se contestarán preguntas sobre interpretación de la letra.

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

- length() y size()
- mod() y rem()
- floor(), ceil() y round()
- zeros() y ones()

Notas: - En todos los ejercicios se deben usar las estructuras de control adecuadas para cada caso.
- No se deben realizar más iteraciones o invocaciones recursivas que las necesarias para resolver el problema

Problema 1	8 (1, 1, 1, 1, 4) ptos	
-------------------	------------------------	--

- a) Representar el número $10010000111010000010001_2$ en base 8.
- b) Representar en base 6 el resultado de $1234_6 * 100_6$
- c) ¿En qué sistemas de representación de 4 bits la operación $1110 + 0010 = 0001$ es correcta?
- d) Convierta el negativo del dígito de verificación de su cédula, es decir $-x$ donde x es el dígito verificador de su cédula, a Complemento a 2 de 4 bits.
- e) Represente el número 2,5 utilizando un sistema de punto flotante con 1 bit de signo, 4 bits de exponente (con exceso $M=7$) y 7 bits de mantisa.

Problema 2	4 ptos	
-------------------	--------	--

Determine el valor de las variables a y b luego de ejecutar `miscript.m` desde la línea de comandos de octave.

<pre>% funcAux.m function b = funcAux(a) if a<5 b = 1; else b = a + funcAux(a-2); end end</pre>	<pre>% miscript.m b = 8; a = 3; b = funcAux(b); a = a - 1;</pre>
--------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------

COMPUTACIÓN 1
Instituto de Computación

Problema 3	10 (2, 8) pts	
-------------------	---------------	--

Se dispone de la función `[pares, impares]=separar(v)`, que dado un vector `v` devuelve los elementos pares del vector `v` en el primer parámetro de salida y los elementos impares en el segundo parámetro.

El siguiente código intenta resolver dicha función de forma recursiva.

```
1 - function [pares,impares] = separar(v)
2 -     largo = length(v);
3 -     if (largo == 0)
4 -         pares = [];
5 -         impares = [];
6 -     else
7 -         if (mod(v(1),2) == 0)
8 -             pares = [v(1) separar(v(2:largo))];
9 -         else
10 -            impares = [v(1) separar(v(2:largo))];
11 -        end
12 -    end
```

a) Indique que resultado produciría hacer una invocación a la función como por ejemplo (justifique su respuesta):

```
[x,y]=separar([1,2,3,4,5,6]);
```

b) Corrija el código para que la función cumpla su cometido.