

Examen - 19 de febrero de 2021
Parte 2: Iteración – 42 puntos

- Duración de esta parte de la prueba: 75 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. Por ejemplo: se controlará el uso correcto de `for` y `while`.
- No se deben realizar más iteraciones que las necesarias para resolver cada uno de los problemas

Problema 4	14 pts	
-------------------	--------	--

Implementar en *Octave* una función **iterativa** *comienzaCon* que, dado dos vectores **v1** y **v2**, devuelva 1 si el vector **v1** comienza con el vector **v2**, y 0 en caso contrario. Notar que ambos vectores pueden tener cualquier tamaño. El vector vacío es comienzo de cualquier vector.

Ejemplos:

```
comienzaCon([], []) devuelve 1
comienzaCon([1,7,2,4], []) devuelve 1
comienzaCon([1,7,2,4], [1]) devuelve 1
comienzaCon([1,7,2,4], [7]) devuelve 0
comienzaCon([1,7,2,4], [1,7]) devuelve 1
comienzaCon([1,7,2,4], [1,2]) devuelve 0
comienzaCon([1,7,2,4], [1,7,4]) devuelve 0
```

Problema 5	14 pts	
-------------------	--------	--

Implementar en *Octave* la función **iterativa** *contarOcurrencias* que, reciba un vector **valores** que contiene números enteros mayores o iguales a 1 y un número entero **n** que contiene el máximo valor que está almacenado en el vector, devuelva un vector **resultado** de tamaño **n** que contenga la cantidad de ocurrencias de cada uno de los valores que aparecen en el vector **valores**. Es decir, en **resultado(i)** se debe almacenar la cantidad de ocurrencias del valor **i** en el vector **valores**. Asuma que el vector **valores** no es vacío.

```
contarOcurrencias([3, 1, 4, 2, 3, 6],6) devuelve [1,1,2,1,0,1]
contarOcurrencias([1, 1, 1, 2, 1],2) devuelve [4,1]
contarOcurrencias([8],8) devuelve [0,0,0,0,0,0,0,1]
```

Problema 6	14 pts	
-------------------	--------	--

Implementar en *Octave* la función **iterativa** *pares* que, dada una matriz **M** que tiene números devuelva un vector conteniendo todos los elementos de la matriz que cumplan que sean múltiplos de 2. Si un elemento está repetido en la matriz y cumple con la condición deberá aparecer tantas veces como esté en la matriz.

```
pares([1,2,3,5;4,1,7,6;8,8,8,8]) devuelve [2,4,6,8,8,8,8] o [4,8,2,8,8,6,8].
```