

2^{do} Parcial - 22 de Noviembre de 2012

- 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 hasta 30 minutos antes de la finalización del mismo.
- Escriba las hojas de un solo lado
- Las partes no legibles del examen se considerarán no escritas
- En la primer hoja a entregar ponga con letra clara, en el ángulo superior derecho, salón en el cual desarrolló la prueba, su nombre, número de cédula de identidad y cantidad de hojas -en ese orden-; las demás hojas es suficiente con nombre, número de cédula y número de página.

Problema 1	12 (2,2,2,3,3) ptos	
-------------------	---------------------	--

- a) ¿Qué son y para qué se usan las matrices dispersas? Justifique su uso.
- b) Nombre y describa al menos 2 formatos para almacenar matrices dispersas.
- c) ¿Qué es un protocolo de comunicación en una red?
- d) Determine qué número decimal representa la siguiente tira en formato de punto flotante de simple precisión de la IEEE

0 10000001 110100000000000000000000

- e) Represente en sistema de punto flotante de la IEEE simple precisión el número decimal -13,5.

Problema 2	15 (6, 6, 3) ptos	
-------------------	-------------------	--

Dada una matriz en formato disperso elemental, se desea obtener otra matriz dispersa (en formato disperso elemental) con los elementos pertenecientes a la parte triangular superior de la misma y que los datos sean mayores que un parámetro dado.

- a) Implemente una función **iterativa** en *Matlab* que permita resolver la problemática antes descrita.
- b) Implemente una función **recursiva** en *Matlab* que permita resolver la problemática antes descrita.
- c) Implemente una función **iterativa** en *Matlab* que devuelva 1 (verdadero) si una matriz dispersa en formato elemental es triangular superior y 0 (falso) en caso contrario.

Problema 3	19 (6, 5, 4, 4) ptos	
-------------------	----------------------	--

- a) Implemente una función **recursiva** $ultOcurr(v,x)$ que devuelva la posición de la última ocurrencia (de izquierda a derecha) de un elemento x en el vector v . En caso que no haya ninguna ocurrencia de dicho elemento, se debe retornar -1.
- b) Implemente una función **iterativa** $ultOcurrIter(v,x)$ que devuelva la posición de la última ocurrencia (de izquierda a derecha) de un elemento x en el vector v . En caso que no haya ninguna ocurrencia de dicho elemento, se debe retornar -1.
- c) Implemente una función **recursiva** $borrarPrim(v,x)$ que devuelva un vector igual a v sin la primera ocurrencia (de izquierda a derecha) del elemento x . En caso que no haya ninguna ocurrencia de dicho elemento, se debe retornar v .
- d) Implemente una función **iterativa** $borrarTodos(v,x)$ que devuelva un vector igual a v sin todas las ocurrencias de un elemento x . En caso que no haya ninguna ocurrencia de dicho elemento, se debe retornar v .

Problema 4	14 (3, 4, 7) pts	
-------------------	------------------	--

El número áureo o proporción áurea se representa por la letra griega φ y se suele atribuir un carácter estético a los objetos cuyas medidas guardan dicha proporción. φ es un número irracional y puede calcularse mediante la siguiente fórmula directa: $\varphi = (1 + \sqrt{5})/2$

Otra forma de calcular φ es mediante la siguiente recurrencia:

$$\begin{cases} \varphi_1 = 1 \\ \varphi_n = 1 + \frac{1}{\varphi_{n-1}}, n > 1 \end{cases}$$

- a) Implemente en *Matlab* una función **iterativa** *proporciónAureaI* que reciba como parámetro de entrada la cantidad de pasos y calcule el número áureo a partir de la definición de la recurrencia.
- b) Implemente en *Matlab* una función **recursiva** *proporciónAureaR* que reciba como parámetro de entrada la cantidad de pasos y calcule el número áureo usando la recurrencia.
- c) Implemente en *Matlab* una función **recursiva** *enQuePaso* que reciba como parámetro una tolerancia **tol** y devuelva para que paso la diferencia en valor absoluto entre el valor calculado mediante la recurrencia y la fórmula directa está por debajo de la tolerancia.

Sugerencia: (parte c) Usar una función cabecal y una función auxiliar recursiva.

Sugerencia: (parte c) No usar la función de la parte b (ni a).