

**CURSO: ALGORITMIA Y ESTRUCTURA DE DATOS (SI-205)**  
**EXAMEN PARCIAL**

**NOTA:**

1. Responder cualquier pregunta de forma continua y completa. No se calificará trozos de respuestas en diferentes partes.
2. No hacer borrones en el limpio. Se descontará por borrones. Use su hoja de borrador o líquido.
3. Sin copias, apuntes. Solo use lapicero de tinta azul o negro.
4. Sírvase facilitar su identificación presentando su carné.

1. Crear un programa (algoritmo) que permita generar una matriz  $M$  de  $5 \times 5$  de elementos aleatorios del 0 al 7 que corresponde a una imagen en la escala de grises. Vamos a aplicar el filtro de la media para obtener una matriz  $N$  de  $5 \times 5$ , donde cada elemento  $N[i][j]$  se determina de la suma de los elementos:

$$M[i-1][j-1], M[i-1][j], M[i-1][j+1],$$

$$M[i][j-1], M[i][j], M[i][j+1],$$

$$M[i+1][j-1], M[i+1][j], M[i+1][j+1] \text{ (si existen) y promediándolos entre 9.}$$

Ejemplo: Matriz  $M$  generada:

$$M = \begin{matrix} & \begin{matrix} 0 & 1 & 2 & 3 & 4 \end{matrix} \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ 2 & 0 & 7 & 1 & 2 \\ 1 & 7 & 0 & 1 & 0 \\ 7 & 0 & 7 & 1 & 7 \\ 1 & 1 & 1 & 0 & 4 \end{bmatrix} \end{matrix}$$

Matriz  $N$  generada a partir de  $M$ :

Para  $N[0][0]$  consideramos los valores son: 1, 0, 2, 0 (los otros no existen)

La media será:  $(1 + 0 + 2 + 0)/9 = 1/3 = 0,333 \approx 0$  (redondeado)

Luego:  $N[0][0] = 0$

Para  $N[3][2]$  consideramos los valores son: 7, 0, 1, 0, 7, 1, 1, 1, 0.

La media será:  $(7 + 0 + 1 + 0 + 7 + 1 + 1 + 1 + 0)/9 = 2$

Luego:  $N[3][2] = 2$

Continuando de esta forma para cada elemento de la matriz  $N$  se obtiene:

$$N = \begin{matrix} & \begin{matrix} 0 & 1 & 2 & 3 & 4 \end{matrix} \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 & 1 \\ 2 & 3 & 3 & 3 & 1 \\ 2 & 3 & 2 & 2 & 1 \\ 1 & 2 & 1 & 2 & 1 \end{bmatrix} \end{matrix}$$

```
#include <stdlib.h>
```

```
#include <time.h>
```

```
...
```

```
main(){
```

```
  srand(time(NULL));
```

```
  ...
```

```
  M[i][j]= 0 + rand() % (8); //para generar valores de 0 a 7
```

(7 puntos)

2. Escriba un algoritmo que reciba un texto no mayor de 256 caracteres que contiene algunas palabras con símbolos no alfabéticos (signos, comas, puntos, guiones, etc.) colocados antes o después de la palabra (ejemplo: "HOla!!", "¡HOla!" o "HOla,"). El algoritmo debe eliminar los símbolos no alfabéticos de las palabras que los contengan. Por tanto, el resultado para los ejemplos mostrados será "HOla". Si una palabra contiene dígitos seguidos de un símbolo y luego una palabra (ejemplo: "0-amigos"), separe los dígitos de la palabra eliminando el símbolo, creando dos palabras independientes. Por tanto, para el ejemplo el resultado será "0 amigos"). Conserve las palabras o números (solo dígitos). Para el caso de las palabras alfabéticas que tienen letras mayúsculas en un lugar diferente a la primera letra de la palabra, el algoritmo debe cambiarlas por minúsculas. Su algoritmo debe mostrar:

El texto corregido

La cantidad de palabras que tenían mayúsculas incorrectas (corregidas)

La cantidad de números presentes en el texto final

Ejemplo, entrada:

**HOla!! tengo 12-perros y 7.gatos, PERO 0-amigos :(**

Resultado

**Hola tengo 12 perros y 7 gatos Pero 0 amigos**

Palabras con mayúsculas corregidas: 2 (Hola y Pero)

Cantidad de números: 3 números (12, 7 y 0)

(7 puntos)

3. Se le pide que escriba un programa en C++ para llevar un registro de puntuaciones en un torneo de múltiples juegos, para ello use un arreglo tridimensional llamado Puntaje[m][n][p] donde: "m" número de jugadores, "n" rondas y "p" número de juegos.  
Inicializa el arreglo con valores enteros aleatorios, cada valor de 0 y 20, los que representan las puntuaciones de los diferentes jugadores en las diferentes rondas y en los diferentes juegos.  
El programa debe calcular e imprimir:
- El índice y el puntaje total del jugador promedio (el que tiene la puntuación total más cercana a la media de las puntuaciones totales de todos los jugadores)
  - El puntaje total acumulado de un jugador específico solicitado por pantalla.

```
#include <stdlib.h>
#include <time.h>
...
main(){
srand(time(NULL));
...
Puntaje[i][j][k]= 0 + rand() % (21); //para generar valores de 0 a 20
```

(6 puntos)

TIEMPO: 112 minutos

LOS PROFESORES