

DESARROLLO DE ALGORITMOS 2018

Trabajo Práctico 3: Arreglos Bidimensionales.

IMPORTANTE: Diseñar algoritmos aplicando conceptos de modularización. Leer detenidamente el enunciado y tratar de pensar cuáles son los módulos que se deben definir. En cada algoritmo no olvidarse de especificar un nombre, definir comentarios apropiadamente y especificar el tipo de todas las variables utilizadas. Cada vez que se diseñe un módulo tener en cuenta en primera instancia su propósito, sus parámetros formales y valor de retorno. En cada invocación a un módulo, revisar cuidadosamente los parámetros actuales, el orden y la compatibilidad de tipos entre parámetros actuales y formales.

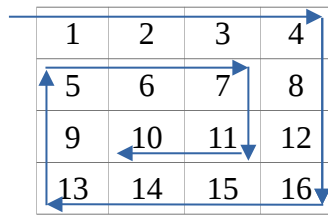
1. Diseñar un algoritmo que declare una matriz de enteros, las dimensiones de la matriz deben ser leídas (cantidad de filas y cantidad de columnas). Luego debe presentar un menú de opciones para realizar las siguientes tareas:
 - a) Cargar la fila i -ésima de la matriz
 - b) Cargar la matriz completa haciendo uso del módulo anterior
 - c) Mostrar la columna i -ésima de la matriz
 - d) Mostrar la matriz entera haciendo uso del módulo anterior
 - e) El programa debe invocar los módulos en forma adecuada para poder probarlos a todos.
2. Diseñar un algoritmo que declare una matriz de $n \times m$ de enteros y la inicialice. Luego debe presentar un menú de opciones para realizar las siguientes tareas:
 - a) Mostrar todos los elementos de una matriz.
 - b) Verificar si la matriz es cuadrada.
 - c) Verificar si la matriz es Triangular Superior
 - d) Verificar si la matriz es Matriz Diagonal
 - f) Producto por un escalar
 - g) Transponer la matriz
 - h) Sumar los elementos de una fila determinada
 - i) Sumar los elementos de una columna
3. Diseñar un algoritmo que declare dos matrices de $n \times n$ de enteros. Luego debe presentar un menú de opciones para realizar las siguientes tareas:
 - a) Cargar los elementos de las matrices.
 - b) Realizar la suma de dos matrices.
 - c) Realizar el productos de dos matrices.
4. Diseñar un algoritmo para recorrer matrices de $n \times n$ de enteros previamente cargadas y mostrar su contenido recorriendo las filas pares desde la columna 0 a la $n-1$ y las filas impares desde la columna $n-1$ a la 0 .
Por ejemplo, de la siguiente matriz

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

los valores deberán mostrarse de la siguiente forma:

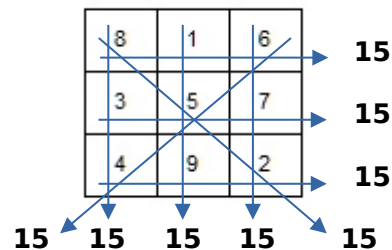
1	2	3	4
8	7	6	5
9	10	11	12
16	15	14	13

5. Diseñar un algoritmo que cargue una matriz a partir de un arreglo de 150 caracteres. La matriz debe ser de tamaño **10x15**, y se debe llenar por fila, una vez completa la primera fila se debe continuar por la segunda y así sucesivamente.
6. Diseñar un algoritmo para recorrer una matriz de enteros de **$n \times n$** previamente cargada, mostrando su contenido recorriéndola en espiral:



1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

7. Diseñar un algoritmo que verifique si una matriz de enteros de **$n \times n$** es un cuadrado mágico. Esto es cierto cuando la suma de cada fila es igual a la suma de cada columna y a la vez es igual a la suma de cada una de las diagonales.
- Por ejemplo, la siguiente matriz es un cuadrado mágico:



8	1	6
3	5	7
4	9	2