

## Práctica 2. Almacenamiento de matrices

En esta práctica trabajaremos con matrices de dos dimensiones.

Las matrices de dos dimensiones se almacenan en memoria como vectores consecutivos, tomando como tales, las filas o las columnas.

Supongamos que una matriz de dos dimensiones, con elementos enteros de 32 bits, se almacena por filas. En este caso, si la dirección del comienzo de la matriz es  $A$  y sus dimensiones son  $n$  y  $m$  (es decir, que tiene  $n$  filas y  $m$  columnas) la dirección del elemento  $A[i][j]$  vendrá dada por:

Dirección de  $A[i][j] = A + 4 * m * i + 4 * j$  ya que cada elemento es un entero que ocupa 4 bytes.

Recuerde que en Informática los índices comienzan en 0.

### Ejercicios entregables (en un único fichero):

1. Escriba una función en lenguaje ensamblador de MIPS con tres parámetros: dos direcciones ( $A$  y  $B$ ) y un número entero ( $n$ ). La función debe escribir a partir de la dirección  $B$  la matriz traspuesta de la matriz cuadrada de  $n$  filas que se encuentra a partir de la dirección  $A$ .
2. Escriba una función en lenguaje ensamblador de MIPS con cuatro parámetros: tres direcciones ( $A$ ,  $B$  y  $C$ ) y un número entero ( $n$ ). La función debe escribir a partir de la dirección  $C$  la suma de las matrices cuadradas de  $n$  filas que se encuentran a partir de las direcciones  $A$  y  $B$ .
3. Escriba una función en lenguaje ensamblador de MIPS que reciba como parámetros la dirección de una matriz cuadrada y su número de filas e imprima en la consola la matriz de forma tabulada.
4. Empleando la funciones construidas en los apartados anteriores, escriba un programa en lenguaje ensamblador de MIPS que pida por teclado una matriz 5x5 e imprima por pantalla de forma tabulada la matriz resultante de sumarla con su traspuesta.