

COMPUTACIÓN 2002 – ASTRONOMÍA

PRACTICA 7

El objetivo de esta práctica es construir una pequeña biblioteca de subrutinas para las principales operaciones con matrices. Esto permite reutilizar código, sin tener que estar reescribiendo subrutinas cada vez que las necesitemos.

1. Para comenzar nos pondremos de acuerdo en como guardar los elementos de una matriz en forma eficiente. En principio, podemos utilizar un arreglo bidimensional. Sin embargo, esto no es adecuado por diversas razones técnicas. En vez de ésto, guardaremos las matrices fila por fila en un arreglo unidimensional (vector), como sigue.

Sea $\mathbf{A} = (a_{ij})$ una matriz $(n \times m)$ (esto es, de n filas y m columnas),

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix}$$

entonces los elementos de $\mathbf{A}$ son guardados en el arreglo unidimensional $\mathbf{A}$ de tamaño $N * M$ fila por fila:

$$\begin{aligned} \mathbf{A}(1) &= a_{11} \\ \mathbf{A}(2) &= a_{12} \\ &\dots \\ \mathbf{A}(M) &= a_{1m} \\ \mathbf{A}(M + 1) &= a_{21} \\ \mathbf{A}(M + 2) &= a_{22} \\ &\dots \\ \mathbf{A}(2 * M) &= a_{2m} \\ &\dots \\ \mathbf{A}(N * M) &= a_{nm} \end{aligned}$$

Así, si los índices de un elemento a_{ij} son representados por las variables FORTRAN I y J , el índice K del correspondiente elemento del arreglo unidimensional $\mathbf{A}$ está dado por

$$K = (I - 1) * M + J$$

¡Verifíquelo!

2. En lo que sigue llamaremos $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ a las matrices sobre las cuales se realiza una operación y $\mathbf{R}$ a la matriz resultante de esta operación. Los símbolos N , M , L representan los tamaños de estas matrices.

Construya entonces las siguientes subrutinas:

- (a) La subrutina `MTXTRA` transfiere la matriz $\mathbf{A}$ ($n \times m$) en el arreglo $\mathbf{R}$, esto es realiza una copia de $\mathbf{A}$ en $\mathbf{R}$.

- (b) La subrutina MTXMSC multiplica la matriz A ($n \times m$) por un escalar S y coloca el resultado en R .
- (c) La subrutina MTXUNT construye la matriz unidad de orden n en el arreglo R .
- (d) La subrutina MTXTRP coloca la traspuesta de la matriz A ($n \times m$) en R .
- (e) La subrutina MTXADD realiza la suma de dos matrices ($n \times m$) A y B y pone el resultado en R .
- (f) La subrutina MTXSUB resta la matriz ($n \times m$) B de la matriz ($n \times m$) A colocando el resultado en R .
- (g) La subrutina MTXMLT multiplica la matriz ($n \times l$) A por la matriz ($l \times m$) B y coloca la matriz producto ($n \times m$) en R .
- (h) La subrutina MTXWRT escribe la matriz ($n \times m$) guardada en A fila por fila por pantalla.

Verifique que cada subrutina funcione correctamente.

3. Utilizando las subrutinas anteriores construya un programa tal que dadas dos matrices ($n \times n$) A y B y el escalar λ realice los siguientes cálculos:

$$AB, \quad A - \lambda I, \quad AB^t, \quad A^t B, \quad AB^t - A^t B$$

Nota: Para utilizar las subrutinas no es necesario insertar el código dentro del programa (aunque, por supuesto, puede hacer esto). Puede compilar por separado las subrutinas y luego unir (*link*) las necesarias con el programa principal en un único ejecutable como sigue.

Supongamos que mi programa `matriz.f` llama a la subrutina MTXMLT, la cual está guardada en el archivo `mtxmlt.f`. Entonces primero compilamos `mtxmlt.f` como sigue

```
$ g77 -c mtxmlt.f
```

Esto genera un archivo `mtxmlt.o` conocido como código objeto. A continuación unimos el código objeto de nuestra subrutina con el programa principal y generamos el ejecutable,

```
$ g77 -o matriz matriz.f mtxmlt.o
```