Departamento de Matemáticas, Universidad de Alcalá

Práctica 1

(Programación básica en Maple)

1. Definir un procedimiento que, sin utilizar el comando ! de Maple, calcule el factorial de un número $n \geq 0$.

2. Definir un procedimiento que, sin utilizar los comandos ! ni *binomial* de Maple, calcule el número combinatorio $\binom{n}{m}$ para $n \geq m$.

3. Definir un procedimiento que calcule todos los primos menores que un número dado.

4. Definir un procedimiento que, dada una lista de números reales, calcule su máximo y su mínimo. (Sin utilizar los comandos *max, min*).

5. Definir un procedimiento que, dada una lista de números reales, los ordene de menor a mayor. Modificarlo para que ordene de mayor a menor.

6. Definir un procedimiento que devuelva una matriz de la forma
$$\begin{pmatrix} a & b & b & \dots & b \\ b & a & b & \dots & b \\ b & b & a & \dots & b \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b & b & b & \dots & a \end{pmatrix}$$
con valores a en la diagonal y b fuera de ella.

7. Construir, mediante bucles y condicionales, una matriz de la forma
$$\begin{pmatrix} a & c & c & \dots & c \\ b & a & c & \dots & c \\ b & b & a & \dots & c \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b & b & b & \dots & a \end{pmatrix}$$

8. Construir, mediante bucles y condicionales una matriz $A := (a_{ij})_{i,j}$ de tamaño 6×6 con términos $a_{ij} := i + j$ si $i < j$, $a_{ij} := i * j$ si $i > j$ y $a_{ij} := i^j$ si $i = j$.

9. Dada la matriz

$$A := \begin{pmatrix} 1 & -2 & 1 & 1 & -2 & 1 & 1 \\ 0 & 3 & -1 & -1 & 2 & -1 & -1 \\ 1 & 1 & 2 & 0 & 0 & 0 & 0 \\ -1 & -1 & 2 & 4 & -2 & 1 & 1 \\ 0 & 0 & 0 & 0 & 6 & -1 & -1 \\ 1 & 1 & -2 & 1 & 1 & 5 & 0 \\ -1 & -1 & 2 & -1 & -1 & 2 & 7 \end{pmatrix}$$

   Se pide:

   (a) Construir un bucle que calcule, para cada submatriz principal de A, la suma de los cuadrados de sus autovalores.

   (b) Construir un bucle que calcule, para cada submatriz principal de A, la suma de sus autovalores elevados a sus multiplicidades.

   (c) Construir un bucle que calcule, para cada submatriz principal de A, la suma de las normas de sus autovectores.

10. La *sucesión de Fibonacci* se define de manera recursiva; un término está dado por la suma de los dos anteriores a él, salvo los dos primeros que se definen como 1.

    (a) Definir un procedimiento recursivo *Fibonacci1* que calcule el término n-ésimo de la sucesión de Fibonacci, sin usar la opción *remember*. Comprobar el resultado con el comando *combinat[fibonacci]*.

    (b) Definir *Fibonacci2* añadiendo la opción *remember* y comparar el tiempo utilizado con el del apartado anterior.

11. Definir un procedimiento que, dados tres puntos $a, b, c \in \mathbb{R}^3$, calcule la ecuación del plano que pasa por ellos. (Pista: una vez fijado un punto, por ejemplo a, la ecuación está dada por el determinante cuyas columnas contienen; (C_1) las coordenadas del vector que va desde a a un punto genérico (x, y, z), (C_2) las coordenadas del vector $\vec{ab}$, y (C_3) las coordenadas del vector $\vec{ac}$).

12. Definir un procedimiento que, dada una lista de puntos en $\mathbb{R}^3$, determine si están en *posición general*, es decir, si no hay cuatro puntos en un mismo plano.