

PRÁCTICA 3.

Estos ejercicios se corresponden con los siguientes apartados del libro “Aprenda Matlab 7,0 como si estuviera en primero”:

- 2.11. Medida de tiempos y de esfuerzo de cálculo (pags. 21-22)
- 2.5.7. El Editor / Debugger (pags. 16-17)

Todas las funciones implementadas en los ejercicios deben realizarse de modo que el comando help sobre ellas indique qué hace la función y que parámetros le entran y devuelve.

En los ejercicios en que se pide el tiempo de ejecución, se debe añadir un comentario en el script de ejecución con el tiempo que tarda.

Ejercicio 1 (grupo a)

Construir una función a la que le entre un vector y devuelva en una matriz la distancia entre cada par de elementos del vector. La distancia será calculada como la diferencia entre cada dos elementos, siendo siempre mayor o igual que cero.

Ejemplo:

INPUT → [2 6 3 4 8 5]

OUTPUT →
$$\begin{bmatrix} 0 & 4 & 1 & 2 & 6 & 3 \\ 4 & 0 & 3 & 2 & 2 & 1 \\ 1 & 3 & 0 & 1 & 5 & 2 \\ 2 & 2 & 1 & 0 & 4 & 1 \\ 6 & 2 & 5 & 4 & 0 & 3 \\ 3 & 1 & 2 & 1 & 3 & 0 \end{bmatrix}$$

Implementar el problema de formas distintas (una de ellas se debe hacer con dos bucles for). Calcular el tiempo de ejecución, en segundos, de ambas funciones si el vector de entrada es un vector aleatorio de 1000 posiciones.

Ejercicio 2 (grupo b)

Construir una función a la que le entre un vector y devuelva en una matriz la suma entre cada par de elementos del vector.

Ejemplo:

INPUT → [2 6 3 4 8 5]

OUTPUT →
$$\begin{bmatrix} 4 & 8 & 5 & 6 & 10 & 7 \\ 8 & 12 & 9 & 10 & 14 & 11 \\ 5 & 9 & 6 & 7 & 11 & 8 \\ 6 & 10 & 7 & 8 & 12 & 9 \\ 10 & 14 & 11 & 12 & 16 & 13 \\ 7 & 11 & 8 & 9 & 13 & 10 \end{bmatrix}$$

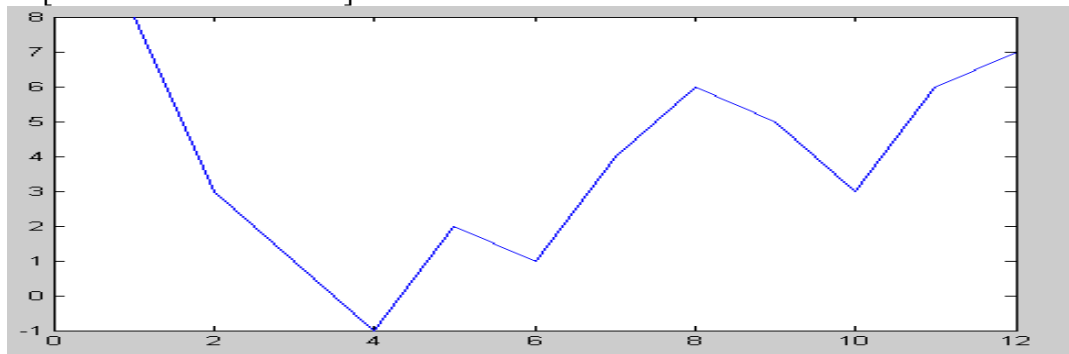
Implementar el problema de formas distintas (una de ellas se debe hacer con dos bucles for). Calcular el tiempo de ejecución, en segundos, de ambas funciones si el vector de entrada es un vector aleatorio de 1000 posiciones.

Ejercicio 3 (grupo a)

Construir una función que encuentre los mínimos locales de una función discreta. Como entrada recibe un vector que representa el valor de la función en las posiciones $x=1, x=2, \dots$. Como salida devuelve un vector con las posiciones donde se encuentran los mínimos locales de la función, así como un número indicando la posición del mínimo global.

Ejemplo:

INPUT $\rightarrow [8 \ 3 \ 1 \ -1 \ 2 \ 1 \ 4 \ 6 \ 5 \ 3 \ 6 \ 7]$



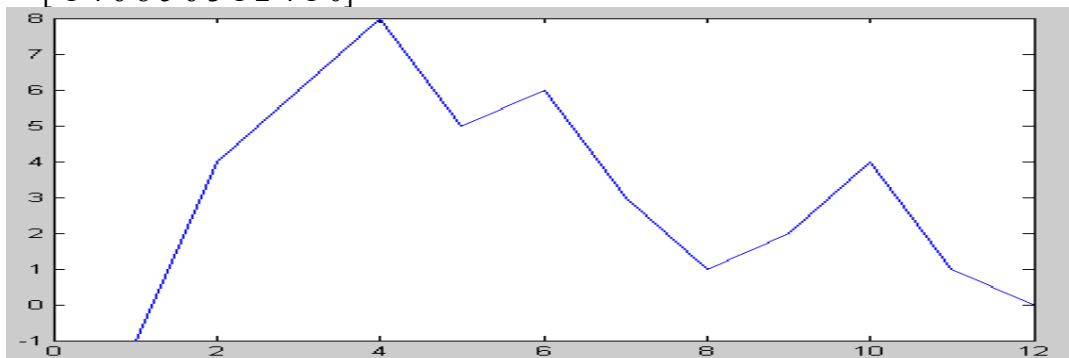
OUTPUT $\rightarrow \{[4 \ 6 \ 10], 4\}$

Ejercicio 4 (grupo b)

Construir una función que encuentre los máximos locales de una función discreta. Como entrada recibe un vector que representa el valor de la función en las posiciones $x=1, x=2, \dots$. Como salida devuelve un vector con las posiciones donde se encuentran los máximos locales de la función, así como un número indicando la posición del máximo global.

Ejemplo:

INPUT $\rightarrow [-1 \ 4 \ 6 \ 8 \ 5 \ 6 \ 3 \ 1 \ 2 \ 4 \ 1 \ 0]$



OUTPUT $\rightarrow \{[4 \ 6 \ 10], 4\}$

Ejercicio 5 (grupo a)

Implementar una función que obtenga el máximo común divisor de dos números sin utilizar la función gcd de Matlab. Implementar dos versiones de la función y comparar el tiempo que utilizan cada una de ellas cuando se le pasan como parámetros de entrada dos números aleatorios entre 100.000 y 900.000. Tomamos por convenio que $\text{m.c.d.}(x,0)=x$.

Ejercicio 6 (grupo b)

Implementar una función que obtenga el mínimo común múltiplo de dos números sin utilizar la función lcm de Matlab. Implementar dos versiones de la función y comparar el tiempo que utilizan cada una de ellas cuando se le pasan como parámetros de entrada dos números aleatorios entre 10.0000 y 90.0000.

Ejercicio 7 (grupo a)

Construir una función que calcule, dadas dos matrices cuadradas P y Q, una nueva matriz R donde cada elemento viene calculado como

$$R(x, z) = \max_y (\min(P(x, y), Q(y, z)))$$

Ejemplo:

$$\text{INPUT} \rightarrow P = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 8 & 6 \\ 4 & 1 & 0 \end{bmatrix}, Q = \begin{bmatrix} 2 & 2 & 5 \\ 2 & 3 & 4 \\ 6 & 1 & 7 \end{bmatrix}$$

$$\text{OUTPUT} \rightarrow R = \begin{bmatrix} 5 & 3 & 5 \\ 6 & 3 & 6 \\ 2 & 2 & 4 \end{bmatrix}$$

Ejercicio 8 (grupo b)

Construir una función que calcule, dadas dos matrices cuadradas P y Q, una nueva matriz R donde cada elemento viene calculado como

$$R(x, z) = \min_y (\max(P(x, y), Q(y, z)))$$

Ejemplo:

$$\text{INPUT} \rightarrow P = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 8 & 6 \\ 4 & 1 & 0 \end{bmatrix}, Q = \begin{bmatrix} 2 & 2 & 5 \\ 2 & 3 & 4 \\ 6 & 1 & 7 \end{bmatrix}$$

$$\text{OUTPUT} \rightarrow R = \begin{bmatrix} 2 & 2 & 4 \\ 2 & 2 & 5 \\ 2 & 1 & 4 \end{bmatrix}$$