

Febrero 2007

Cada ejercicio se deberá realizar teniendo en cuenta las especificaciones que se indican. Cualquier cuestión de diseño en la que no se dé una especificación concreta, podrá ser realizada de la manera que se desee. No se permite el uso de variables globales. Los ejercicios solo se consideran bien realizados si el programa ejecutable funciona correctamente y el diseño se adecua a lo especificado. En cada ejercicio los ficheros y el programa ejecutable deberán denominarse como se indica.

1.- Dados 3 vectores podemos obtener el producto escalar de cada par de 2 vectores. El vector tridimensional cuyas componentes son los productos escalares calculados es denominado vector proyección. Por ejemplo, dados los vectores de 3 dimensiones $a = (1,1,1)$, $b = (2,3,1)$ y $c = (1,2,3)$ se tiene que $a \times b = 1 \times 2 + 1 \times 3 + 1 \times 1 = 6$, $a \times c = 1 \times 1 + 1 \times 2 + 1 \times 3 = 6$ y $b \times c = 2 \times 1 + 3 \times 2 + 1 \times 3 = 11$ y entonces $a \times b \times c = (6,6,11)$. Implementar un programa que permita introducir 3 vectores y obtenga el vector Proyección. Definir en el programa principal 4 vectores enteros (3 para los vectores dato y 1 para el vector proyección) que tengan dimensión 4 e inicializarlos todos para que todas sus componentes sean -1. En el programa principal se definirá una variable entera denominada dimensión, en la cual se pedirá al usuario la dimensión de los vectores que se vayan a usar, que podrá ser 2, 3 ó 4. Por ejemplo, si la dimensión es 2, sólo serán usadas las 2 primeras componentes de cada vector.

- (a) Para ello diseñar una función llamada Introducción, que reciba como parámetros un vector y la dimensión, que sea llamada desde el programa principal 3 veces para introducir los 3 vectores dato. Diseñar una función llamada Imprimir que reciba un vector y la dimensión y muestre cuales son sus elementos, llamando a esta función desde el programa principal 3 veces para mostrar cada uno de los elementos de los vectores dato. **(1 punto)**.
- (b) Diseñar también una función que reciba 2 vectores y la dimensión como parámetros y devuelva el producto escalar de los 2 vectores. Esta función será también llamada desde el programa principal 3 veces para calcular los 3 productos escalares necesarios y obtener el vector llamado “proyección”. Usar la función Imprimir para mostrar el vector proyección. **(2 puntos)**

El programa principal y las funciones construidas estarán en un archivo que se llamara producto.c, existirá un archivo de cabecera denominado producto.h para los prototipos y el programa ejecutable se llamará producto.exe.

2.- Dada una matriz se desea saber cuantos de sus elementos cumplen la condición de ser mayores que el triple de la media de todos ellos. En el programa principal se ha de declarar una matriz de tipo double de dimensión 4x4, inicializando todos sus valores a cero. También en el programa principal se declararán dos variables enteras para preguntar al usuario el número de filas y de columnas de la matriz con la que desea trabajar (dimensiones de la matriz). En el programa principal deberá comprobarse que la dimensión de la matriz respeta el máximo posible (4 x 4) indicándole al usuario que tiene un error si esa condición no se cumple **(0.5 puntos)**.

- a) Diseñar una función llamada Introducción, que reciba la matriz y sus dimensiones, que permita introducir datos en la matriz y una función llamada Imprimir que muestre los valores de la matriz **(1 punto)**.
- b) Diseñar una función llamada media que reciba la matriz y sus dimensiones, y devuelva el valor medio de los elementos de la matriz que deberá ser mostrado desde el programa principal **(1 punto)**.
- c) Diseñar una función llamada calcular_mayores que reciba la matriz, sus dimensiones y el valor medio, y devuelva cuantos elementos de la matriz son mayores que el triple de la media, mostrando el resultado en el programa principal **(1 punto)**.

El programa principal y las funciones construidas estarán en un archivo que se llamara matriz.c, existirá un archivo de cabecera denominado matriz.h para los prototipos y el programa ejecutable se llamará matriz.exe.

3.- Se quiere almacenar en una matriz de caracteres de 5 filas y 10 columnas, 5 cadenas de longitud máxima 9 cada una (dentro de las cadenas no aparecerán espacios en blanco). Diseñar un programa que invoque secuencialmente a las siguientes funciones:

- a) Función para introducir 5 cadenas en la matriz **(0.5 puntos)**.
- b) Función que muestre todas las cadenas introducidas en la matriz **(0.75 puntos)**,
- c) Función que obtenga y muestre la cadena introducida que tenga mayor longitud **(0.75 puntos)**.
- d) Función que calcule y devuelva cuantas veces se cumple la condición “la secuencia consecutiva de caracteres **ca** se encuentra en la cadena” para todas las cadenas introducidas. Por ejemplo, las cadenas “casimiro” o “bacalao” cumplen la condición, la cadena “ancla” no la cumple, la cadena “cacao” la cumple 2 veces, y la cadena “acta” no la cumple, por lo que el total de veces que se cumple la condición es 4 **(1.5 puntos)**.

Las funciones necesarias se denominaran Introducir, Imprimir, Mostrar_cadena y Comprobar_condicion y estarán en un archivo denominado funciones.c para el que existirá un archivo llamado funciones.h con los prototipos, y el programa principal se denominará cadenas.c y el programa ejecutable se denominará cadenas.exe.