

Práctica 6 de C: 17-02-2009

Objetivos:

- Asentar los conceptos asociados a la definición y acceso a los vectores (arrays) en C.
- Profundizar los conceptos del manejo de los esquemas condicionales y repetitivos en C.
- Continuar con la definición y manejo de funciones en C.
- Conocer cómo se pasan vectores como parámetros en C.

Enunciado:

1. Crea un nuevo programa principal practica6_1.c que utilice los siguientes algoritmos:
 - Lectura de N datos para un vector de números reales, debiendo solicitar N al usuario. Se puede dar al usuario la opción de no introducir los datos, sino de generarlos de forma aleatoria mediante la función `rand()` (Mira en el manual cuál es la sintaxis y la semántica de `rand()`). Utiliza la función de ahormado (`cast`) para conseguir generar números reales.
 - Mostrar el contenido del vector, si es que ya hay datos leídos.
 - Dado el vector de números reales leído en el apartado anterior, queremos hacer una función que calcule cuántas veces aparece un número real X dentro de ese vector, y si aparece al menos una vez que muestre por pantalla las posiciones en las que aparece. El valor de X debe solicitarse al usuario.

La función debe responder a las siguientes especificaciones en código algorítmico:

Algoritmo cuenta_apariciones (X, datos, N, cuantas, existe) es

x: numérico; {real; parámetro dato; dato a buscar en el vector}

Datos: vector[1..100] de numérico; {reales; parámetro dato; vector con los datos donde hay que buscar las ocurrencias de X}

N: numérico; {parámetro dato; entero; posiciones ocupadas del vector datos, $1 \leq N \leq 100$ }

Cuantas: numérico; {entero; parámetro resultado; número de veces que aparece el elemento X en V; $0 \leq cuantas \leq N$ }

Existe: lógico; {parámetro resultado; será cierto si X aparece al menos una vez en Datos; falso en caso contrario}

- Dado el vector de números reales leído en el apartado anterior, queremos hacer una función que calcule cuáles son los valores: más alto, más bajo y promedio del vector. Además, debe indicarnos mediante un mensaje por pantalla las posiciones de los

elementos más alto y más bajo del vector. La función debe responder a las siguientes especificaciones en código algorítmico:

Algoritmo varias_operaciones (datos, N, mas_alto, mas_bajo, media) es

datos:vector[1..100] de numérico; {parámetro dato}

N: numérico; {entero; parámetro dato; número real de posiciones ocupadas del vector}

mas_bajo, mas_alto: numérico; {parámetros resultado; valores numéricos de los valores más bajo y más alto, respectivamente}

media: numérico; {parámetro resultado; valor medio de los datos}

Utiliza los dos algoritmos, *leer_datos(VR, T)* y *mostrar_datos(VR, T)*, de la práctica anterior para que sean capaces de leer un máximo de T datos de un vector VR de números reales y a continuación mostrarlos por pantalla, de forma similar a como funciona *mostrar_v()*.

- Crea un nuevo programa principal practica6_2.C que solicite al usuario dos números enteros positivos, P y Q , y que a continuación le muestre al usuario aquellos números que estén comprendidos entre P y Q y que además sean múltiplos de 3, de 7 ó de ambos. En el caso de que no haya alguno, también debe indicarlo. Además, sólo deberá hacer las comprobaciones si $Q > P$.
- Crea un nuevo programa principal practica6_3.C que solicite al usuario un número entero positivo válido, R , y que a continuación le muestre al usuario aquellos números menores o iguales que R que verifiquen que la suma de sus divisores es un número primo. El programa puede realizar llamadas a algoritmos con nombre para realizar aquellas operaciones que consideres relevantes. En caso de disponer de estas funciones, debes incluirlas en el programa. Intenta introducir estas funciones mediante un fichero cabecera (*.h) o como un fichero fuente adicional en el proyecto.

NOTA: No es necesario entregar esta práctica, pero puedes revisarla en horario de tutorías.