

Los ejercicios en adelante requieren de cumplir los Requerimientos de Valor Agregado en Código Fuente (hasta el requerimiento KK):

- Es indispensable en la entrega de cada programa, que el código fuente sea completamente en idioma Español (a excepción de lo correspondiente a la API del lenguaje).
- En adelante, entiéndase por “programa ABC”, aquel que gestione un listado de registros y cuente con las opciones de menú: “Alta”, “Baja”, “Consulta” (de un registro), “Cambio” (de datos de un registro), “Listado” (de todos los registros) y “Búsqueda” (de un registro particular o varios que coincidan con el patrón de búsqueda).

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

1. Modificar el programa ABC del ejercicio 58 de “Ejercicios PRO.pdf”, para cumplir con todo lo solicitado pero, declarar y sin definir una variable global de tipo apuntador a registro, incluir una subrutina de inicialización que se ejecute al principio del programa y ahí se solicite al (la) usuari@ la cantidad máxima de registros que podrán ser capturados; luego de lo anterior sea creado el arreglo de registros usando “malloc”; posterior a ejecutar la subrutina de inicialización, el programa funcione igual que la versión anterior del programa. Al finalizar el programa, toda la memoria solicitada sea liberada justo antes de que termine la subrutina “main”.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

2. Modificar el ejercicio 1 para que funcione tal cual se describió, pero, solicitar la memoria para el arreglo de registros usando new y delete en lugar de malloc y free. Lo anterior usando el tipificado de dato como struct y no class.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C ANSI

3. Implementar una función que haga lo mismo que strchr() de string.h recibiendo igual dos arreglos de char; usar dicha función desde main; el programa no incluya más librerías que stdio.h y stdlib.h
4. Implementar un programa que: 1) inicialice con una subrutina un arreglo de 25 enteros; 2) colocar en dicho arreglo elementos repetidos otros no repetidos; 3) al salir de dicha subrutina se imprima el contenido del arreglo; 4) otra subrutina reciba como parámetros por referencia dicho arreglo y la cantidad de elementos; 5) al salir de esta última subrutina, los elementos repetidos hayan sido eliminados y el parámetro de cantidad de elementos haya sido actualizado; 6) finalmente, se imprima el contenido actual del arreglo. Ejem: si en la lista se encuentran los elementos 1,8,7,9,2,7,3,7,3,3, la lista quede como 1,8,7,9,2,3.

IMPLEMENTAR LOS SIGUIENTES PROGRAMAS EN LENGUAJE C++ ANSI

5. Implementar un programa, donde una subrutina reciba una cadena por referencia, busque en dicha cadena un caracter, digamos el ‘j’ y si se encuentra tal caracter, removerlo de la cadena. El programa incluya dos variantes en carpetas diferentes, en la variante1, la cadena sea un arreglo de caracteres; en la variante2 sea un objeto string. Remover de la cadena tantas ocurrencias como se encuentren del citado caracter.

6. Implementar una subrutina que convierta un int en un arreglo de caracteres (no un objeto string); dicha subrutina reciba un int, el puntero a un arreglo de caracteres y el tamaño del arreglo de caracteres; dicho arreglo de caracteres sea creado antes de llamar a dicha subrutina; tal subrutina devuelva verdadero si y solo si la cantidad de caracteres a escribir en el arreglo pudieron ser almacenados en dicho arreglo; al salir de dicha subrutina, imprimir el contenido del arreglo de caracteres; por ejemplo: si la subrutina recibe un entero "2017" y el arreglo cuenta al menos con 5 celdas, el arreglo de caracteres contenga en la celda 0 el código ASCII del '2', en la celda 1 el ASCII del '0', etc. donde finalmente la celda 4 contenga el carácter nulo. Evitar usar la clase stringstream y cualquier otra clase o subrutina de librería.