TP - Programación en ANSI – C

1- Escribir un programa que, solicite al usuario el ingreso de un numero entero, y luego muestre su valor absoluto. Nota: para los números positivos su valor absoluto es igual al número (el valor absoluto de 52 es 52), mientras que, para los negativos, su valor absoluto es el número multiplicado por -1 (el valor absoluto de -52 es 52). (Similar al ej 5 del tp de condicionales de Introducción a la Programación)

2- Escribir un programa que solicite al usuario una letra y, si es una vocal, muestre el mensaje “es vocal”. Se debe validar que el usuario ingrese sólo un carácter. Si ingresa un string de más de un carácter, informarle que no se puede procesar el dato. (Similar al ej 8 del tp de condicionales de Introduccion a la Programación)

3- Solicitar al usuario que ingrese números enteros positivos y, por cada uno, imprimir la suma de los dígitos que lo componen. La condición de corte es que se ingrese el número -1. Al finalizar, mostrar cuántos de los números ingresados por el usuario fueron números pares. (Similar al ej 5 delen lenguaje ANSI C tp de Ciclos repetitivos - While de Introducción a la Programación)

4- Solicitar al usuario el ingreso de una frase y de una letra (que puede o no estar en la frase). Recorrer la frase, carácter a carácter, comparando con la letra buscada. Si el carácter no coincide, indicar que no hay coincidencia en esa posición (imprimiendo la posición) y continuar. Si se encuentra una coincidencia, indicar en qué posición se encontró y finalizar la ejecución. (Similar al ej 7 del tp de Ciclos repetitivos - While de Introducción a la Programación)

5- Escribir un programa que solicite el ingreso de una cantidad indeterminada de números mayores que 1, finalizando cuando se reciba un cero. Imprimir la cantidad de números primos ingresados. (Similar al ej 11 del tp de Ciclos repetitivos - While de Introducción a la Programación)

6- Requerir al usuario que ingrese un número entero positivo e imprimir todos los números correlativos entre el ingresado por el usuario y uno menos del doble del mismo. (Similar al ej 4 del tp de Ciclos repetitivos - FOR de Introducción a la Programación)

7- Solicitar al usuario que ingrese una frase de menos de 1000 caracteres y luego imprimir la cantidad de vocales que se encuentran en dicha frase. (Similar al ej 7 del tp de Ciclos repetitivos - FOR de Introducción a la Programación)

8- Dado un número entero positivo, mostrar su factorial. El factorial de un número se obtiene multiplicando todos los números enteros positivos que hay entre el 1 y ese número. (Similar al ej 10 del tp de Ciclos repetitivos - FOR de Introducción a la Programación)

9- Solicitar al usuario que ingrese su dirección email de hasta 200 caracteres. Imprimir un mensaje indicando si la dirección es válida o no, valiéndose de una función para decidirlo. Una dirección se considerará válida si contiene el símbolo "@" y no está al final ni al principio. (Similar al ej 1 del tp de Funciones de Introducción a la Programación)

10- Solicitar números al usuario hasta que ingrese el cero. Por cada uno, mostrar la suma de sus dígitos. Al finalizar, mostrar la sumatoria de todos los números ingresados y la suma de sus dígitos. Debe implementar y utilizar una función que reciba un numero de una cantidad desconocida de dígitos y devuelva la suma de ellos. (Similar al ej 3 del tp de Funciones de Introducción a la Programación)

11- Solicitar al usuario el ingreso de números primos. La lectura finalizará cuando ingrese un número que no sea primo. Por cada número, mostrar la suma de sus dígitos. También solicitar al usuario un dígito e informar la cantidad de veces que aparece en el número (frecuencia). Al finalizar el programa, mostrar el factorial del mayor número ingresado. Implementar la funcion booleana es\_primo(num) y la función frecuencia(num,digito) que retorne la cantidad de veces que el digito aparece en el número. (Similar al ej 8 del tp de Funciones de Introducción a la Programación)

12- Implementar una función eliminar(), que reciba un arreglo de numeros naturales finalizado con un 0 y un entero a ser eliminado. Si el numero se encuentra en el arreglo, se debe eliminar la primera aparición, desplazando hacia la izquierda todos los numeros siguientes, incluido el 0 que marca el fin del arreglo. Usarla en un programa que compruebe su funcionamiento.

13- Implementar una función longitud(arreglo,marca) donde arreglo es un arreglo de enteros y marca es el valor elegido para marcar el fin de los datos (puede ser un cero, un -1 o cualquier valor que convencionalmente se tome como marca de fin de datos validos). La función debe retornar cuántos valores tiene el arreglo antes de encontrar la marca. Usarla en un programa que compruebe su funcionamiento.

14- Implementar una función ordenar(pa,pb,pc) que reciba como parametros las direcciones de 3 variables float. La función debe intercambiar sus contenidos para que el malor mayor quede en la variable apuntada por pa. El valor del medio en la apuntada por pb y el valor menor en la apuntada por pc.

15- Debe implementar una función binario(num) con operaciones a nivel de bits que muestre la representación binaria de un valor entero.

16- Desarrollar un programa que solicite al usuario un numero entero mayor que 5. Implementar una función crear\_matriz() que reciba por parámetro el valor ingresado. Deberá solicitar memoria dinámica para una matriz de NxN de enteros. Debe llenarla de ceros y retornarla al main.

Luego el programa principal solicitará al usuario que indique pares de coordenadas fila y columna donde pondrá un 1 en la matriz. La carga finaliza al indicar fila=-1.

Implementar una función vecindario(matriz,fila,colu) recibirá la matriz y una posición a verificar. Si la casilla no tiene una bomba, la función retornará la cantidad de bombas que hay en casilleros vecinos (entre 0 y 8). En cambio si la casilla tiene una bomba, la funcion retornará -1.

El programa principal mostrará por pantalla un mensaje con la cantidad de bombas vecinas o bien si hemos activado una bomba, con el siguiente estilo:

“La casilla (2,4) tiene 2 bombas vecinas” o bien “BOOM”