Ejercicio No. 1

El siguiente ejercicio muestra que los parámetros pasados por valor se pueden modificar dentro de la función sin cambiar la variable que representa al parámetro en el llamador. Digite (póngale extensión .cpp a su archivo), compile y ejecute el programa.

#include <stdio.h>

#include <conio.h>

void Intercambio(int, int);

void Intercambio(int \*, int \*);

void Intercambio(int, int &, int &);

int main() {

int x = 22, y = 33;

printf("x = %d\t y = %d\n", x, y);

Intercambio(x,y); // llamada por valor

printf("x = %d\t y = %d\n", x, y);

Intercambio(1, x, y); // llamada por referencia

printf("x = %d\t y = %d\n", x, y);

Intercambio(&x, &y); // llamada por puntero

printf("x = %d\t y = %d\n", x, y);

getch();

}

void Intercambio( int a, int b)

{

int aux;

aux = a;

a = b;

b = aux;

}

void Intercambio(int \*q, int \*e)

{

int aux; aux = \*q;

\*q = \*e;

\*e = aux;

}

void Intercambio( int i, int &a, int &b)

{

int aux=i;

aux = a;

a = b;

b = aux;

}

Como resultado de la llamada a Intercambio por valor (la primera llamada de Intercambio ()), ¿se cambian los valores de a y b? \_\_si\_ ¿por qué? \_porque a toma el valor de b y b toma el valor de aux.\_

Después de la segunda llamada a Intercambio () (llamada por referencia), ¿hubo modificación de a y b \_\_\_si \_\_\_\_ ¿por qué? \_ya que \*a apunta al valor de \*b y \*b apunta al valor de aux, asi que cada uno muestra el valor de donde apuntan.\_

¿Es equivalente la tercer llamada a Intercambio () con la segunda llamada a la misma función? \_\_\_no\_\_\_ ¿por qué? Porque el segundo intercambio es de los valores atraves del puntero, mientra que el tercer intercambio se obtiene la posición de la memoria.\_

Ejercicios Propuestos:

1. Escribir una función **mayor()** que intercambie dos valores cuando el primero sea mayor que el segundo. Hacer un programa que la utilice.

2. Escribir una función que multiplique un valor por 10 y devuelva el valor modificado. Hacer un programa que la utilice.

3. Escribir una función **potencia()**, que calcule la potencia de un número n (tipo double) elevado a un exponente p (tipo int). Escribir un programa que haga uso de esta función.

4. Diseñar una función **area()** que pueda calcular el área de un cuadrado y un cubo. Escribir un programa que haga uso de esta función.

5. Escribir una función **menorAcero()** al que se pasan dos argumentos int por referencia y a continuación fijar el menor de los dos números a 0. Escribir un programa que utilice esta función.