

Ejercicios parte 1

Instrucciones:

1. Leer cada problema las veces que sea necesario hasta comprender el problema, pueden realizar dibujos o utilizar Excel para comprender el problema.
2. Ya analizado el problema, empezar a programar. Si en dado caso no logran resolver el problema pueden ver la solución (recordar que cada problema tiene más de 1 posible solución, por lo que pueden llegar al mismo resultado de diferentes maneras).
3. Al programar deben de tomar en cuenta posibles errores que pueda tener un programa. Estos errores pueden ser, por ejemplo: Ingresar un número mayor al rango que le solicita el programa (Ingrese un numero entre 1-5 ¿qué pasa si se ingresa -5 o 10?). Si pide el número mayor entre 3 números ¿Qué pasa si los 3 son iguales?, ¿Qué pasa si 2 son iguales y 1 diferente? Etc.
4. No realizar copy paste del programa. Para iniciar, el profe puede saber cuántas veces presionaron la tecla al realizar el programa, por lo que se va a dar cuenta si hicieron un copy paste. Mínimo cambiar las variables. Además, deben estudiar a conciencia y comprender cada uno de los problemas para garantizar el aprendizaje.

Contenido

Ejercicio 1:	3
Ejercicio 2:	4
Ejercicio 3:	6
Ejercicio 4:	7
Ejercicio 5:	10
Ejercicio 6:	11
Ejercicio 7:	13
Ejercicio 8:	15
Ejercicio 9:	16
Ejercicio 10:	18
Ejercicio 11:	20
Ejercicio 12:	21
Ejercicio 13:	22
Ejercicio 14:	23
Ejercicio 15:	25
Ejercicio 16:	28
Ejercicio 17:	29
Ejercicio 18:	33
Ejercicio 19:	35
Ejercicio 20:	36
Ejercicio 21:	38
Ejercicio 22:	40
Ejercicio 23:	43
Ejercicio 24:	45
Ejercicio 25:	48
Ejercicio 26:	49
Ejercicio 27:	52
Ejercicio 28:	55
Ejercicio 29:	56
Ejercicio 30:	58
Ejercicio 31:	60
Ejercicio 32:	62

Ejercicio 33:	63
Ejercicio 34:	65
Ejercicio 35:	65
Ejercicio 36:	67
Ejercicio 37:	68
Ejercicio 38:	70

Ejercicios:

Ejercicio 1:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int n1, n2, suma;

    cout << "Ingrese el numero 1: ";
    cin >> n1;
    cout << "Ingrese el numero 2: ";
    cin >> n2;

    suma = n1 + n2;

    cout << " La suma de ambos numeros es: " << suma;

    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}
```

Explicación:

En este caso el problema pide específicamente que se realice la suma 2 cantidades, por lo que no es necesario utilizar un “for” para realizar la suma. Solo se necesita de 2 variables para almacenar los números y una tercera variable para realizar la suma.

“/n” vendría a tener una función similar que endl. Sirve para saltar de línea y lo utilizo para que el mensaje “presione una tecla para continuar salga debajo de la última línea del programa.

```
Ingrese el numero 1: 10
Ingrese el numero 2: 8
La suma de ambos numeros es: 18

Presione una tecla para continuar . . . _
```

Sino se vería así:

```
Ingrese el numero 1: 5
Ingrese el numero 2: 5
La suma de ambos numeros es: 10Presione una tecla para continuar . . . _
```

Ejercicio 2:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    double n1, n2, operaciones;

    cout << "Ingrese el numero 1: ";
```

```

cin >> n1;
cout << "Ingrese el numero 2: ";
cin >> n2;

operaciones = n1 + n2;
cout << "La suma de los numeros es:  " << operaciones << "\n";

operaciones = n1 - n2;
cout << "La resta de los numeros es:  " << operaciones << "\n";

operaciones = n1 * n2;
cout << "La multiplicacion de los numeros es:  " << operaciones << "\n";

operaciones = n1/ n2;
cout << "La division de los numeros es:  " << operaciones << "\n";

cout << "\n\n\n\n";

system("PAUSE");
return 0;
}

```

Explicación:

En ejercicio nos pide el mismo que el anterior, realizar diferentes operaciones con 2 valores ingresados. Una manera de hacer seria crear las variables para los números ingresados y crear una variable para cada operación: suma, resta, multiplicación y división.

Pero para cambiar el ejercicio solo cree una variable llamada “Operaciones”. Lo que hago con esta variable es realizar las diferentes operaciones y actualizar su resultado luego de imprimir el resultado.

```

operaciones = n1 + n2;
cout << "La suma de los numeros es:  " << operaciones << "\n";

operaciones = n1 - n2;
cout << "La resta de los numeros es:  " << operaciones << "\n";

operaciones = n1 * n2;
cout << "La multiplicacion de los numeros es:  " << operaciones << "\n";

operaciones = n1/ n2;
cout << "La division de los numeros es:  " << operaciones << "\n";

```

En esta parte del código se puede observar que primero realizo la operación, luego la imprimo y después de imprimirla actualizo la variable “operaciones” con la nueva operación. (Recordar utilizar double)

```
Seleccinar C:\Progra 1\Ejercicio_1_1\x64\Debug\Ejercicio_1_1.exe
Ingrese el numero 1: 10.5
Ingrese el numero 2: 2.3
La suma de los numeros es: 12.8
La resta de los numeros es: 8.2
La multiplicacion de los numeros es: 24.15
La division de los numeros es: 4.56522

Presione una tecla para continuar . . .
```

Ejercicio 3:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int n;

    cout << "Ingrese el numero: ";
    cin >> n;

    if (n<0)
    {
        cout << "El numero ingresado es negativo";
    }
    else
    {
        cout << "El numero ingresado es positivo";
    }
}
```

```

        cout << "\n\n\n\n";

        system("PAUSE");
        return 0;
}

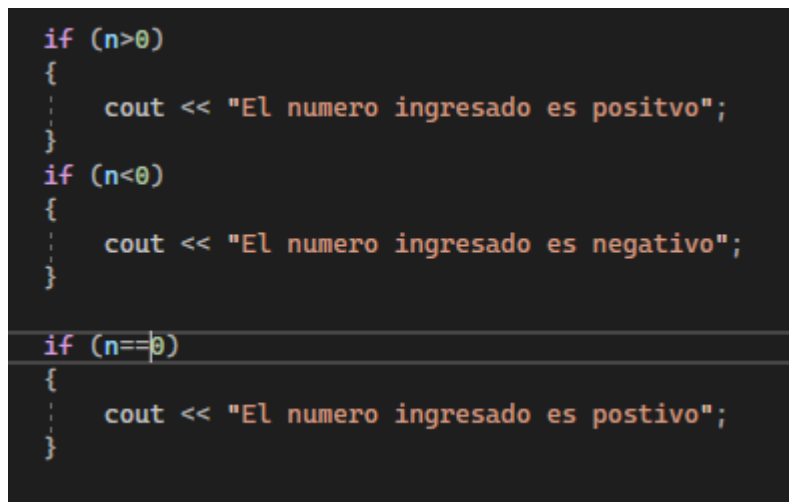
```

Explicación:

Este ejercicio nos pide que evaluemos si un número es positivo o negativo. Realizar esto es fácil, pero se puede realizar de distintas maneras. En el código utilizo un if para verificar si el número es menor que 0. Si en caso de que el numero sea menor que 0 imprimirá el texto: “

El numero ingresado es negativo”. En caso contrario imprimirá el texto: “El numero ingresado es positivo.

Pero también se puede realizar de otras maneras como, por ejemplo:



```

if (n>0)
{
    cout << "El numero ingresado es positivo";
}
if (n<0)
{
    cout << "El numero ingresado es negativo";
}

if (n==0)
{
    cout << "El numero ingresado es postivo";
}

```

La desventaja de hacer este método es que tenemos que crear un “if” extra en dado caso que el usuario ingrese 0. Pero el resultado final sería el mismo.

Ejercicio 4:

Código:

```

#include<iostream>
#include<cstdlib>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {

```

```

COORD punto;
punto.X = x; punto.Y = y;
SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int n;

    cout << "Ingrese el ano de nacimiento: ";
    cin>> n;

    cout << "La persona tiene: " << 2022 - n << " anos de edad";

    cout << "\n\n\n\n";

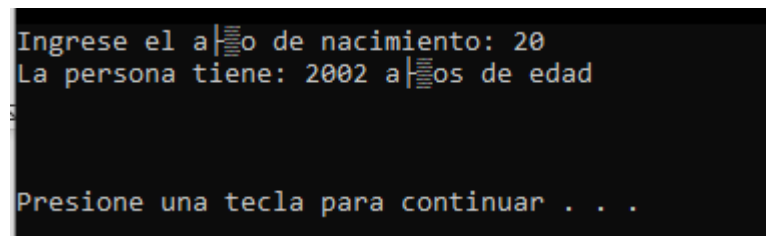
    system("PAUSE");
    return 0;
}

```

Explicación:

En este ejercicio lo que hay que hacer es una resta simple. Solo hay que restar de año actual (2022) con el año en el cual nació la persona y se obtendría el resultado. Otra manera de hacer este ejercicio crear una variable aparte para realizar la resta. Pero en mi ejemplo la realizo directamente en el "cout".

No escribo la letra "ñ" debido a que para que el programa acepte caracteres especiales se debe agregar una librería extra. Si escribo la letra "ñ" se vería así:



```

Ingrese el año de nacimiento: 20
La persona tiene: 2002 años de edad

Presione una tecla para continuar . . .

```

Solucionar este problema es fácil, solo se necesita realizar los siguientes pasos:


```

#include <stdio.h>
#include <wchar.h>
#include <locale.h>
#include <stdlib.h>

int main(void){

    setlocale(LC_ALL, "");

    printf("ó á");

    return EXIT_SUCCESS;
}

```

Lo que hacemos es incluir las librerías `wchar.h`, que incluye un soporte para amplios tipos de caracteres y la librería `locale.h` con la que se incluye el soporte para distintos tipos de formatos de fecha, moneda, texto, etc.

Solo basta usar la funcion:

```
setlocale(LC_ALL, "");
```

La que nos permitira usar en nuestro programa distintos tipos de caracteres que no pertenezcan al "estandar", que se puede imprimir normalmente.

Mucho ojo en este problema porque se podría dar el caso en que una persona escriba un numero negativo y como saben, no existen años negativos. Por lo que tendría que utilizar un "while".

```

while (n<0)
{
    cout << "Ingresar un dato correcto";
    cin >> n;
}

```

Este while lo que hace es que si detecta que alguien ingresa un año negativo le vuelve a pedir que ingrese un numero correcto.

```

Ingrese el año de nacimiento: -12
Ingresar un dato correcto2001
La persona tiene: 21 años de edad

Presione una tecla para continuar . . .

```

También se puede dar el caso que una persona ingrese, por ejemplo: 2025. Lo que daría como resultado que tiene -3 años. Todos estos son posibles problemas que puede tener el programa.

Ejercicio 5:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int n1,n2;

    cout << "Ingrese primer dato: ";
    cin>> n1;

    cout << "\nIngrese segundo dato: ";
    cin >> n2;

    if (n1 > n2)
    {
        cout << "\nEl numero mayor es: " << n1;
    }
    else if (n1==n2)
    {
        cout << "\nAmbos numeros son iguales";
    }
    else
    {
        cout << "\nEl numero mayor es: " << n2;
    }

    cout << "\n\n\n\n";
```

```

        system("PAUSE");
        return 0;
    }

```

Explicación:

En este ejercicio primero que todo hay que analizar los posibles resultados que puede tener el ejercicio, los cuales serían: $n1 > n2$, $n2 > n1$ y $n1 = n2$. Son 3 posibles casos, por lo que hay que programar cada uno de los casos. En mi programa de ejemplo evalué primero si el número 1 es mayor al número 2. En caso de que sea verdadero ya sabremos cual es el número mayor. En caso contrario evalué si ambos valores son iguales. En caso de que sea verdad imprimo como resultado que ambos valores son verdaderos.

Por último, si no se cumple ninguna de las condiciones anteriores solo quedaría una opción, que el número 2 sea el mayor. Por ese motivo utilizo else.

Al igual que los programas anteriores, se puede realizar en diferente orden el if. Por ejemplo, primero se podría evaluar si $n2 > n1$, luego $n1 > n2$ y si no se cumple ninguna de las condiciones querría decir que ambos números son iguales.

Ejercicio 6:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int n1, n2, n3;

    cout << "\nIngrese primer dato: ";
    cin >> n1;

    cout << "\nIngrese segundo dato: ";

```

```

cin >> n2;

cout << "\nIngrese el tercer dato: ";
cin >> n3;

if (n1>n2 && n1>n3)
{
    cout<<"\nEl numero mayor seria: " << n1;
}
else if(n2>n1 && n2>n3)
{
    cout << "\nEl numero mayor seria: " << n2;
}
else if(n3>n1 && n3>n2)
{
    cout << "\nEl numero mayor seria: " << n3;
}
else if (n1==n2 && n1==n3)
{
    cout << "Todos los numeros son iguales";
}

cout << "\n\n\n\n";

system("PAUSE");
return 0;
}

```

Explicación:

A comparación de ejercicios anteriores, este pide que ingresemos 3 datos. Por lo que, para encontrar el número mayor, se debe cumplir la condición de ser más grande que los otros números. Hay que evaluar todos los posibles casos. Se puede dar el caso que el usuario ingrese 5,3,5. Lo que daría que se repite el número 5. En este programa no afecta que se repita el número, porque, aunque el usuario haya ingresado dos veces el número 5, 5 sigue siendo el mayor número. Así que el único caso especial que se puede dar es que el usuario ingrese: 1,1,1. Lo que daría que todos los números son iguales. En los "if" utilizo el operador " && " que vendría a significar "and".

```

if (n1>n2 and n1>n3)
{
    cout<<"\nEl numero mayor seria: " << n1;
}
else if(n2>n1 and n2>n3)
{
    cout << "\nEl numero mayor seria: " << n2;
}
else if(n3>n1 and n3>n2)
{
    cout << "\nEl numero mayor seria: " << n3;
}
else if (n1==n2 and n1==n3)
{
    cout << "Todos los numeros son iguales";
}
}

```

También pueden escribir como tal “and”.

Ejercicio 7:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int n1,n2,n3,n4;

    cout << "\nIngrese primer dato: ";
    cin>> n1;

    cout << "\nIngrese segundo dato: ";
    cin >> n2;

    cout << "\nIngrese el tercer dato: ";
    cin >> n3;

    cout << "\nIngrese el cuarto valor: ";
}

```

```

cin >> n4;
// Numero mayor
if (n1>=n2 && n1>=n3 && n1>=n4)
{
    cout<<"\nEl numero mayor seria: " << n1;
}
else if(n2>=n1 && n2>=n3 && n2>=n4)
{
    cout << "\nEl numero mayor seria: " << n2;
}
else if(n3>=n1 && n3>=n2 && n3>=n4)
{
    cout << "\nEl numero mayor seria: " << n3;
}
else if (n4>=n1 && n4>=n2 && n4>=n3)
{
    cout << "\nEl numero mayor es: " << n4;
}

// numero menor
if (n1 <= n2 && n1 <= n3 && n1 <= n4)
{
    cout << "\nEl numero menor seria: " << n1;
}
else if (n2 <= n1 && n2 <= n3 && n2 <= n4)
{
    cout << "\nEl numero menor seria: " << n2;
}
else if (n3 <= n1 && n3 <= n2 && n3 <= n4)
{
    cout << "\nEl numero menor seria: " << n3;
}
else if (n4 <= n1 && n4 <= n2 && n4 <= n3)
{
    cout << "\nEl numero menor es: " << n4;
}

if (n1 == n2 && n1 == n3 && n1 == n4)
{
    system("cls");
    cout << "\nLos 4 numeros son iguales";
}

cout << "\n\n\n\n";

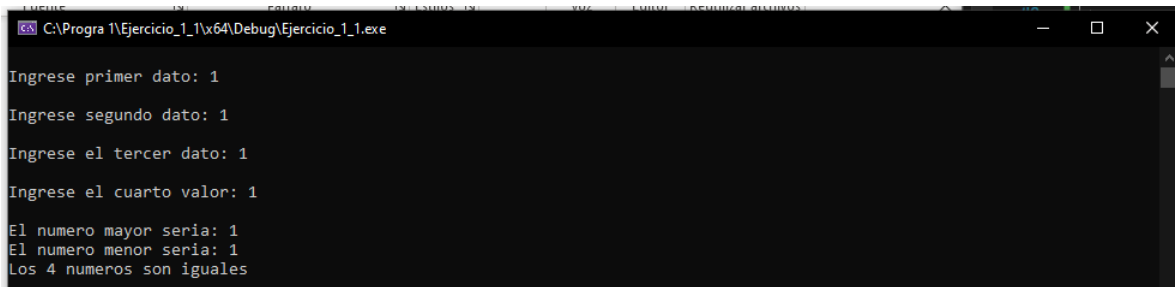
system("PAUSE");
return 0;
}

```

Explicación:

La primera parte del ejercicio (la de encontrar el número mayor) vendría a ser igual que el ejercicio anterior. Solo hay que agregar la nueva variable. Se utiliza el símbolo >= debido a que se pueden dar combinaciones como, por ejemplo: 9,1,1,9. En las que el programa no detectaría la respuesta correcta. Porque no se cumplen todas las condiciones del if.

Compruébenlo ustedes mismos. Lo malo de utilizar este método, es que al final del programa se vería así:



```
C:\Progra 1\Ejercicio_1_1\Debug\Ejercicio_1_1.exe
Ingrese primer dato: 1
Ingrese segundo dato: 1
Ingrese el tercer dato: 1
Ingrese el cuarto valor: 1
El numero mayor seria: 1
El numero menor seria: 1
Los 4 numeros son iguales
```

Debido a que se cumplen las 3 condiciones, por utilizar `>=`. Así que, para solucionar este problema, lo que hago es utilizar `system("cls")`. Lo que hace este código es que borra todo lo que se encuentra en la consola. Entonces en el caso de que todos los datos sean iguales. Primero borro los mensajes del número mayor y número menor para hacer que solo aparezca el mensaje de que los 4 números son iguales.

Ejercicio 8:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int n,i,suma;

    suma = 0;
    cout << "Ingrese 25 datos para calcular la sumatoria\n";

    for (i = 0; i < 25; i++)
    {
        cout << "\nIngrese el numero " << i + 1 << ": ";
        cin>> n;
        suma+= n;
    }

    cout << "\nLa sumatoria de los valores es: " << suma;
```

```

        cout << "\n\n\n";

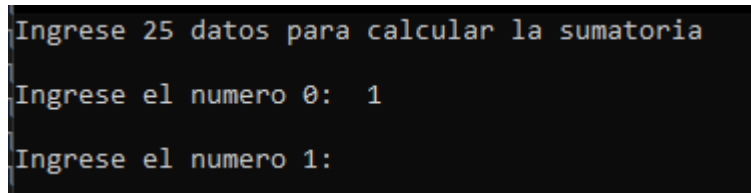
        system("PAUSE");
        return 0;
}

```

Explicación:

En este ejercicio nos pide que ingresemos 25 datos. Cuando son múltiples datos los que nos pide el problema es mejor utilizar un for. Lo que hacemos es poner “for (i = 0; i < 25; i++)”. En otras palabras, dice: El valor inicia en 0, seguir haciendo este ciclo hasta que el valor sea 25, y por cada ciclo que se realice, sumarle “1” a “i”.

Entonces este ciclo se repetirá un total de 25 veces. En el “cout” escribo “i + 1” para mostrar el número actual, además se escribe de esta manera porque si escribiera solamente “i” iniciaría con el número cero y se vería así:



```

Ingrese 25 datos para calcular la sumatoria
Ingrese el numero 0: 1
Ingrese el numero 1:

```

Después de que el usuario ingresa el numero lo que hago es sumar el numero ingresado a la variable suma. En mi ejemplo escribo “suma +=n”. Pero otra manera de escribirlo sería: suma = suma + n, vendrían a significar lo mismo. Por último, solo se muestra la sumatoria final.

Ejercicio 9:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
}

```



```

    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    double i,n,suma,multiplicacion,promedio;

    suma = 0;
    multiplicacion = 1;

    cout << "Ingrese 25 datos para calcular la sumatoria\n";

    for (i = 0; i < 30; i++)
    {
        cout << "\nIngrese el numero " << i + 1 << ": ";
        cin>> n;
        suma+= n;
        multiplicacion = multiplicacion * n;
    }

    promedio = suma / 30;
    cout << "\nLa sumatoria de los valores es: " << suma;
    cout << "\n La multiplicacion de los valores es: " << multiplicacion;
    cout << "\nEl promedio es: " << promedio;

    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}

```

Explicación:

Este ejercicio vendría a ser similar al anterior. Lo único que cambia es que nos pide que agreguemos la multiplicación y promedio de los valores ingresados. Mucho ojo con la multiplicación. Generalmente al inicializar una variable (ejemplo `a=0;`). Se pone que el numero sea 0, pero en este caso queremos que se inicialice en 1 ¿Por qué? Debido a que, si se inicializa en cero, cualquier número multiplicado por 0 da 0

	0 (*)	1 (*)	10 (=)			0

Así que ponemos “multiplicación = 1”. Para que la primera operación que haga sea $1 * \text{el número ingresado}$. Para el promedio hay que recordar que es la sumatoria dividido el número de datos ingresado.

Ejercicio 10:

Código:

```
#include<iostream>
```

```
#include<cstdio>
```

```
#include<string>
```

```
#include<dos.h>
```

```
#include<Windows.h>
```

```
using namespace std;
```

```
int main() {
```

```
    int valores[20],i,nummayor,nummenor;
```

```
    cout << "Ingrese 20 valores:" << endl;
```

```
    for (i = 0; i < 20; i++)
```

```
    {
```

```
        cout << "Ingrese el valor " << i + 1 << endl;
```

```
        cin>> valores[i];
```

```
    }
```

```
    nummayor = valores[0];
```

```
    nummenor = valores[0];
```

```
    for (i = 1; i < 20; i++)
```

```
    {
```

```
        if (valores[i]>nummayor)
```

```
        {
```

```

        nummayor = valores[i];
    }

    if (valores[i]<nummenor)
    {
        nummenor = valores[i];
    }
}

cout << "El numero mayor era:" << nummayor << endl;
cout << "El numero menor era:" << nummenor << endl;


system("PAUSE");

return 0;

}

```

Explicación:

En este ejercicio nos pide ingresar 20 datos, por lo que no nos conviene crear 20 variables para almacenar cada dato. Así que para este ejemplo utilice un vector. Un vector es un tipo de variable que puede almacenar múltiples datos en él. Hay que recordar que la primera posición del vector es el número 0.

Así que lo primero que hace el programa es ingresar los 20 datos en el vector. Luego de eso utilizo 2 variables nummayor y nummenor y les asigno el primer valor del vector, recordar que el primer espacio del vector es la casilla 0.

Matriz							
Numero de comparacion inicial:		10					
Espacio:	0	1	2	3	4	5	
Contenido:	10	5	2	3	20	21	

En este ejemplo, el primer número que ingrese fue el número 10, así que, lo que hace el “for” es ir analizando casilla por casilla si hay un número que sea mayor o menor que el número actual. En la casilla 1, se encuentra el numero 5 así que es un nuevo número menor. Al cumplirse esta condición se actualiza el nuevo número menor que pasaría a ser 5.

Así hasta llegar al final y encontrar el número mayor y el número menor del vector.

Ejercicio 11:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {

    int i,n;

    cout << "Ingrese un numero" << endl;
    cin >> n;

    for (i = 1; i < n; i++)
    {
        cout << "\n" << i + 1;
    }

    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}
```

Explicación:

En este ejercicio lo que nos pide es básicamente que el programa se quede en un bucle hasta que se cumpla una condición. En este caso es imprimir números hasta llegar al número que el usuario ingrese. Por esa razón pedimos que el usuario ingrese el número y en el for ponemos de condición que el programa se siga ejecutando hasta llegar al número deseado.

Ejercicio 12:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i,n;

    for (i = 1; i < 200; i++)
    {
        cout << "\n5 * " << i << " = " << i * 5;

        if ((i*5)%2==0)
        {
            cout << "      El numero es par";
        }
        else
        {
            cout << "      El numero es impar";
        }

    }

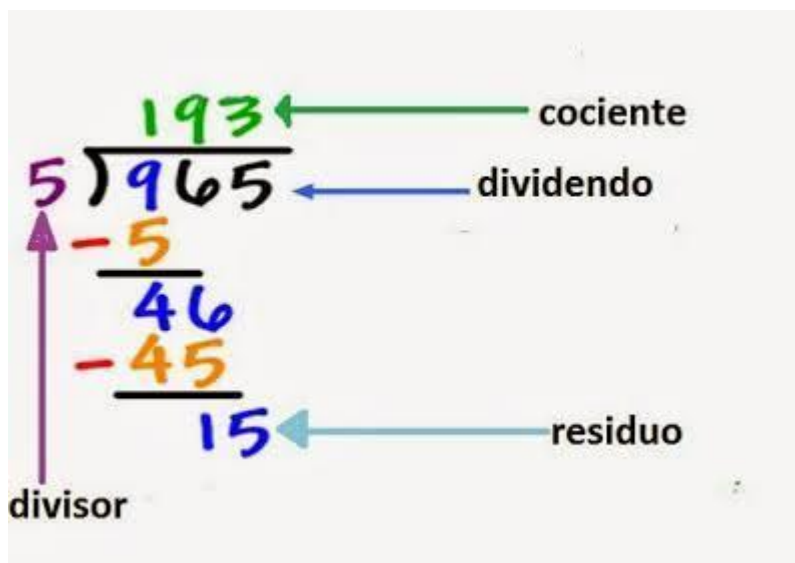
    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}
```

Explicación:

En este ejercicio lo que nos solicita el profesor es que mostremos la tabla del 5, desde el número 1 hasta el número 200 y que mostremos si el resultado par o impar.

Para realizar las multiplicaciones lo podremos hacer fácil con un for. Simplemente multiplicamos 5 por el valor de i, así obtendremos el resultado. El problema viene cuando nos preguntan el número par e impar. ¿Cómo sabemos si un número es par o impar? Para realizar esto utilizaremos la función modulo en c++ que se vendría a escribir como "%". ¿Para qué nos sirve el módulo? Este operador lo que realiza es la división de un número, pero en lugar de darnos como resultado cociente de la operación, obtendremos el residuo de la operación.



En este ejemplo podemos observar las partes de la división. Si por ejemplo ponemos $2\%2$, obtendríamos 0 como respuesta, porque $2/2$ es 1 y su residuo 0. Entonces cualquier número par tendrá como resultado un residuo 0 y cualquier número impar un residuo diferente de 0.

Ejercicio 13:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;
```

```

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    float r, perimetro, area;

    cout << "Ingrese el radio del circulo:";
    cin >> r;
    perimetro = 2 * 3.1416 * r;
    area = 3.1416 * (r * r);

    cout << "\nEl area del circulo es: " << area;
    cout << "\nEl perimetro del circulo es: " << perimetro;

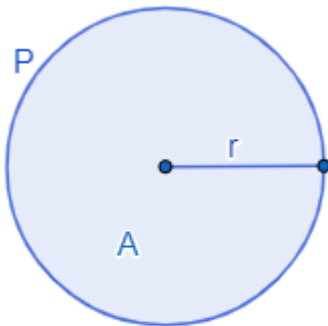
    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}

```

Explicación:

Para realizar este ejercicio debemos recordar cual es la fórmula del perímetro y el área del triángulo.



$$P = 2 \cdot \pi \cdot r$$

$$A = \pi \cdot r^2$$

Deberemos solicitar que el usuario ingrese el radio del círculo, luego eso realizar las operaciones necesarias. Para escribir pi, simplemente se escribe como 3.1416.

Ejercicio 14:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {

    double b,a,area;

    cout << "Ingrese la altura del triangulo: ";
    cin >> a;
    cout << "\nIngrese la altura del triangulo: ";
    cin >> b;

    area = (a * b) / 2;

    cout << "\nEl area del triangulo es: " << area;

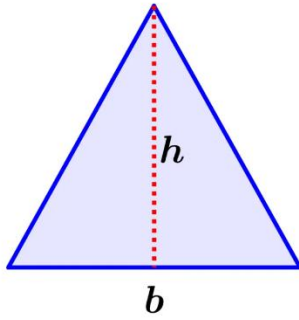
    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}

```

Explicación:

En este ejercicio nos piden encontrar el área de un triángulo. La fórmula del área de un triángulo es la siguiente:



$$A = \frac{1}{2} \times b \times h$$

Lo único que debemos de hacer es que el usuario pueda ingresar 2 números y realizar la operación. Tener siempre cuidado con los paréntesis.

Ejercicio 15:

Código:

Explicación:

Para este ejercicio utilizaremos de ejemplo la siguiente ecuación.

A photograph of a handwritten equation $x^2 + 5x + 6 = 0$ in blue ink on a yellow grid background.

Lo que nos pide el ejercicio es que utilizemos la formula general para encontrar las raíces de la ecuación. La fórmula seria la siguiente:

A photograph of the quadratic formula written in green ink on a yellow grid background. The formula is
$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2 \cdot a}$$

Por lo que tenemos que pedirle al usuario que ingrese los 3 números para encontrar el resultado. Utilizamos 2 variables porque en una utilizamos el valor positivo de la ecuación y en la otra el negativo.

$$\frac{-5 \pm \sqrt{5^2 - 4 \cdot 1 \cdot 6}}{2 \cdot 1}$$

$$\frac{-5 \pm \sqrt{25 - 24}}{2}$$

$$\frac{-5 \pm \sqrt{1}}{2}$$

$$x_1 = \frac{-5 + 1}{2}$$

$$x_2 = \frac{-5 - 1}{2}$$

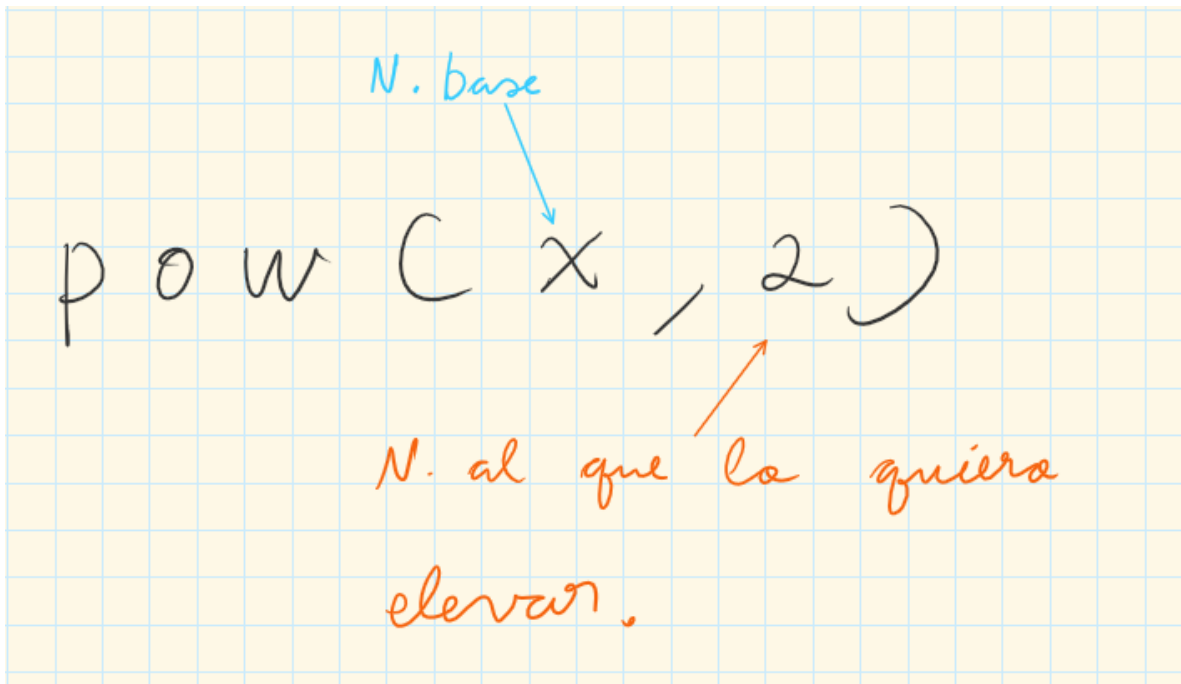
$$x_1 = -2 //$$

$$x_2 = -3 //$$

El problema es cuando intentamos utilizar una raíz en c++. Utilizaremos la función sqrt(). Lo que hace esta función es realizar la operación “raíz” de lo que tenga en su interior.

```
raiz1 = (-b + ( sqrt( (pow(b,2)) - (4*a*c)) ) ) / (2*a);  
raiz2 = (-b - (sqrt( (b * b) - (4 * a * c)))) / (2 * a);
```

Mucho cuidado con los paréntesis en estos problemas, también pueden realizar la operación con más de 1 variable y dividir las diferentes partes de la operación para que se les haga más sencillo de entender. Por ejemplo, crear una variable solo para la raíz, otra para realizar -b + el resultado anterior y por último otra variable para que a ese resultado le dividan el 2*a. También utilizo el operador pow el cual funciona de la siguiente manera:



El numero “X” vendría a ser el numero base que quieren elevar y después de la coma deberán escribir el numero al cual quieren elevar ese número “X”.

En este ejemplo debido a que es al cuadrado podrán observar que lo escribí de diferentes maneras, pero el resultado sigue siendo el mismo.

```
raiz1 = (-b + ( sqrt( (pow(b,2)) - (4*a*c)) ) ) / (2*a);  
raiz2 = (-b - (sqrt( (b * b) - (4 * a * c)))) / (2 * a);
```

En raíz 1 utilizo pow, mientras que en raíz 2 utilizo simplemente b*b.

```
En este prograba debera ingresar 3 valores (a,b,c)
de una ecuacion tipo: ax^2 +bx +c para encontrar
las raices de la solucion
Ingrese el numero 'a': 1

Ingrese el numero 'b': 5

Ingrese el numero 'c': 6

El primer resultado es: -2
El segundo resultado es: -3

Presione una tecla para continuar . . .
```

Ejercicio 16:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {

    int i,n,mayores,menores;
    cout << "Este programara clasificara los numeros ingresados que sean
menores y los que sean mayores o iguales que 20";
    mayores = 0;
    menores = 0;

    for (i = 0; i < 30; i++)
    {
        cout << "\nIngrese el numero " << i + 1<<": ";
        cin >> n;
        if (n<20)
        {
            menores = menores + 1;
        }
        else
        {

```

```

        mayores += 1;
    }

}

cout << "\nLos numeros mayores o iguales a 20 son: " << mayores;
cout << "\nLos numeros menores son: " << menores;

cout << "\n\n\n\n";

system("PAUSE");
return 0;
}

```

Explicación:

Este programa es un poco la combinación de anteriores programas. Deberemos ingresar 30 números, por lo que utilizaremos un for. Luego nos pide clasificar los números 2 maneras diferentes. Los números que sean ≥ 20 y los < 20 (mayores o iguales que 20 y los que sean menores que 20.) Aquí podríamos hacer de diferentes maneras el “if” podríamos poner, por ejemplo: `if (n>=20)` también podríamos poner simplemente $n > 20$, pero no nos conviene, debido a que tendríamos que crear un if extra en el cual deberíamos de clasificar los valores que sean iguales que 20: $n = 20$. Siempre inicializar las variables fuera del for para que no ocurran errores. Por último, simplemente se le suma “1” a cada contador según va clasificando los números. Se puede escribir de las siguientes 2 maneras, pero significarían lo mismo:

```

if (n>=20)
{
    menores = menores + 1;
}
else
{
    mayores += 1;
}

```

En la segunda escribo el operador += que vendría a cumplir la misma función que el de arriba “sumarse a sí mismo una nueva cantidad”.

Ejercicio 17:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>

```

```

#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int x, y, i, z;
    //figura 1
    for (i = 0; i < 7; i++)
    {
        gotoxy(20,i+20); cout << "*";
    }
    for (i = 0; i < 5; i++)
    {
        gotoxy(21, i + 21); cout << "*";
    }
    for (i = 0; i < 3; i++)
    {
        gotoxy(22, i + 22); cout << "*";
    }
    gotoxy(23, 23); cout << "*";

    //figura 2
    for (i = 0; i < 5; i++)
    {
        gotoxy(40 + i, 24); cout << "*";
    }

    for (i = 0; i < 3; i++)
    {
        gotoxy(41 + i, 23); cout << "*";
        gotoxy(41 + i, 25); cout << "*";
    }
    gotoxy(42, 22); cout << "*";
    gotoxy(42, 26); cout << "*";

    //figura 3

    for (i = 0; i < 7; i++)
    {
        gotoxy(60, 20+i); cout << "*";
    }
    for (i = 0; i < 3; i++)
    {
        gotoxy(59+i, 21); cout << "*";
    }
}

```

```

    }
    for (i = 0; i < 5; i++)
    {
        gotoxy(58 + i, 22); cout << "*";
    }

    gotoxy(5, 40); system("PAUSE");

    return 0;
}

```

Explicación:

Para este ejercicio deberán comprender como utilizar la función gotoxy. Recordar que la función gotoxy contiene 2 números en su interior "gotoxy(x,y)" En la "x" iría el valor de la fila, mientras que en "y" el valor de la columna. Lo que nos pide el ejercicio es que hagamos los siguientes dibujos con ciclos:

Dibujo 1:						Posicion 1:				
*						0,0	0,1	0,2	0,3	
*	*					1,0	1,1	1,2	1,3	
*	*	*				2,0	2,1	2,2	2,3	
*	*	*	*			3,0	3,1	3,2	3,3	
*	*	*				4,0	4,1	4,2	4,3	
*	*					5,0	5,1	5,2	5,3	
*						6,0	6,1	6,2	6,3	

Dibujo 2:						Posicion 2:				
						0,0	0,1	0,2	0,3	0,4
		*				1,0	1,1	1,2	1,3	1,4
	*	*	*			2,0	2,1	2,2	2,3	2,4
*	*	*	*	*		3,0	3,1	3,2	3,3	3,4
	*	*	*			4,0	4,1	4,2	4,3	4,4
		*				5,0	5,1	5,2	5,3	5,4
						6,0	6,1	6,2	6,3	6,4

Dibujo 2:						Posicion 2:					
		*				0,0	0,1	0,2	0,3	0,4	
		*				1,0	1,1	1,2	1,3	1,4	
		*				2,0	2,1	2,2	2,3	2,4	
	*	*	*			3,0	3,1	3,2	3,3	3,4	
*	*	*	*	*		4,0	4,1	4,2	4,3	4,4	
		*				5,0	5,1	5,2	5,3	5,4	
		*				6,0	6,1	6,2	6,3	6,4	

Recomendaciones: primero pueden utilizar Excel tal como en el ejemplo de arriba para que puedan darse una idea de cómo debería quedar el dibujo. Como segunda recomendación, utilicen comentarios en el código (deberán escribir “//” y se convertirá en un comentario). Para poder guiarse en su programa.

Lo que debemos hacer es que, por cada ciclo en el for, vaya sumando “1” ya sea en el espacio de X o en el espacio de Y para que en cada ciclo vaya cambiando su posición.

En la figura uno decidí pintar la columna, ósea, lo vertical. Así que la posición en X siempre seguirá siendo la misma, lo único que debe ir cambiando es su posición en Y. En total cree 3 fors, cada uno pinta una diferente columna y solo deben ajustar un poco los valores para que quede bien la figura.

Y así sucesivamente con cada figura, en alguna decidí pintar las filas y en otras las columnas. En otras palabras, en algunos dibujos varia la posición solamente en X y en otros dibujos varia la posición simplemente en Y.



Ejercicio 18:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
```

```

int vector[15],i;

for (i = 0; i < 15; i++)
{
    cout << "\nIngrese el numero "<<i+1<<": ";
    cin >> vector[i];
}

for (i = 0; i <15; i++)
{
    cout << "\nEl numero en el espacio " << i << " es: " << vector[i];
}

cout << "\n\n\n\n";
system("PAUSE");

return 0;
}

```

Explicación:

Este ejercicio nos pide que ingresemos valores a un vector, para realizar un for. Recordar que un vector tendría una forma similar a esta:

Vector de 5							
			0	1	2	3	4
0	x		x	x	x	x	x
1	x						
2	x			posicion			
3	x			Contenido en esa posicion			
4	x						

En este ejemplo utilizaremos un vector de tamaño 5. Hay que recordar que la primera posición inicia en 0. Este vector debido a que es de una dimensión, se lo pueden imaginar de manera horizontal o vertical.

Así que lo único que hará el for en este programa es ir navegando posición por posición en el vector para ir rellenando los espacios necesarios.

```

Ingrese el numero 1: 10
Ingrese el numero 2: 10
Ingrese el numero 3: 23

El numero en el espacio 0 es: 10
El numero en el espacio 1 es: 10
El numero en el espacio 2 es: 23

```

Ejercicio 19:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    double nb1[15],nb2[15],nb3[15],nb4[15], nf[15],promedio[15];
    int i;
    string nombre[15], resultado[15];

    cout << "          NOMBRE          NB1          NB2          NB3          NB4          NOTA FINAL
Resultado";

    for (i = 0; i < 15; i++)
    {
        gotoxy(5, i + 2); cin >> nombre[i];
        gotoxy(17, i + 2); cin >> nb1[i];
        gotoxy(25, i + 2); cin >> nb2[i];
        gotoxy(33, i + 2); cin >> nb3[i];
        gotoxy(41, i + 2); cin >> nb4[i];

        promedio[i] = (nb1[i] + nb2[i] + nb3[i] + nb4[i]) / 4;

        if (promedio[i] >= 60)
        {
            nombre[i] = "APROBADO";
        }else
        {
            nombre[i] = "REPROBADO";
        }

        gotoxy(49, i + 2); cout << promedio[i];
        gotoxy(64, i + 2); cout << nombre[i];

    }

    cout << "\n\n\n\n";
```

```

        return 0;
    }

```

Explicación:

Para este ejercicio deberemos crear un diferente vector por los diferentes valores y utilizar un for para ir navegando a lo largo del vector. Deberemos utilizar conceptos anteriores como el gotoxy para que quede la información de manera más visual para el usuario.

Las operaciones tales como sumas, multiplicaciones etc, se trabaja de la misma manera con los vectores. Entonces en cada ciclo lo que haremos es introducir la información de cada usuario y utilizar un if para saber si el usuario aprobó o reprobó el curso en base al promedio final.

NOMBRE	NB1	NB2	NB3	NB4	NOTA FINAL	Resultado
BYron	60	60	60	60	60	APROBADO
Josue	50	50	30	20	37	REPROBADO

Ejercicio 20:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    double sb[15], bonif[15], horasex[15], sueldoorasex[15],
    iggs[15], promedio[15], isr[15], sueldoliquido[15], sueldoxhora[15];
    int i;
    string nombre[15];

    cout << "      EMPLEADO      SUELDO BASE      BONIF      HORASEX
SUELDOHORASEX      IGGS      ISR      SUELDO LIQUIDO";

```

```

for (i = 0; i < 10; i++)
{
    gotoxy(5, i + 2); cin >> nombre[i];
    gotoxy(19, i + 2); cin >> sb[i];
    gotoxy(35, i + 2); cin >> bonif[i];
    gotoxy(45, i + 2); cin >> horasex[i];

    sueldoxhora[i] = ((sb[i]) / (8 * 30));
    sueldoorasex[i] = sueldoxhora[i] * 2*horasex[i];

    iggs[i] = sb[i]*(4.83/100);

    if (sb[i] >=3000)
    {
        isr[i] = sb[i] * 0.05;
    }
    else
    {
        isr[i] = 0;
    }

    sueldoliquidado[i] = sb[i] + bonif[i] + sueldoorasex[i]-iggs[i]-
isr[i];

    gotoxy(57, i + 2); cout << sueldoorasex[i];
    gotoxy(74, i + 2); cout << iggs[i];
    gotoxy(84, i + 2); cout<< isr[i];
    gotoxy(92, i + 2); cout<< sueldoliquidado[i];
}

cout << "\n\n\n\n";

return 0;
}

```

Explicación:

Para realizar este ejercicio es cuando ya se puede complicar un poco el asunto. Debido a que a pesar de que los procedimientos son solamente sumas, multiplicaciones etc, debemos hacerlas de manera ordenada, sino no funcionara el programa. Primero deberán leer la información que les brinda el problema y comprenderla, como por ejemplo que para calcular el sueldo por hora seria: sueldo por hora * 2. Pero para sacar el sueldo por hora hay que hacer otro procedimiento. Distinguir los valores que deben ser ingresados por el usuario de los valores que calcula el problema.

Con eso en mente, este ejercicio vendría a ser muy similar al anterior, en el cual el usuario deberá ingresar información y el programa debe realizar los procesos para calcular los resultados.

EMPLEADO	SUELDO BASE	BONIF	HORASEX	SUELDOHORASEX	IGGS	ISR	SUELDO LIQUIDO
hernan	4000	250	4	133.333	193.2	200	3990.13

Ejercicio 21:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i,z,vector[4][4],nummayor;

    for (i = 0; i < 4; i++)
    {
        for (z = 0; z < 4; z++)
        {
            cout << "\nIngrese el numero en el espacio [" << i << "] ["
<< z << "] : ";
            cin >> vector[i][z];

        }

    }

    nummayor = vector[0][0];

    for (i = 0; i < 4; i++)
    {
        for (z = 0; z < 4; z++)
        {
            if (vector[i][z] > nummayor)
            {
                nummayor = vector[i][z];
            }

        }

    }

}
```

```

        cout << "\nEl numero mayor es:" << nummayor;

        cout << "\n\n\n\n";

        return 0;
}

```

Explicación:

Para realizar este ejercicio primero debemos de entender como “rellenar una matriz de 4x4”. Esto hace referencia a que nos está solicitando un vector de 2 dimensiones... Vector[x][y]. No es complicado, imagínenlo como el siguiente ejemplo:

POSICION			
0,0	0,1	0,2	0,3
1,0	1,1	1,2	1,3
2,0	2,1	2,2	2,3
3,0	3,1	3,2	3,3

Esto sería un vector 4x4, al igual que con los de una dimensión, el valor inicial es 0. Así que el primer valor hace referencia a la fila, mientras que el segundo a la columna. Ejemplo [2][1] fila 2, columna 1.

Ahora ¿cómo se escribiría esto en código?

```

for (i = 0; i < 4; i++)
{
    for (z = 0; z < 4; z++)
    {
        cout << "\nIngrese el numero en el espacio [" << i << "] [" << z << "] : ";
        cin >> vector[i][z];
    }
}

```

Aquí lo estaríamos haciendo es navegar al vector en el siguiente orden: [0][0], [0][1], [0][2], [0][3], [1][0], [1][1] etc. El for que se encuentra hasta afuera (el que tiene valor “i”) es el valor de la fila, mientras que en el for que contiene “z” es el valor de la columna.

En otras palabras, lo que hace este for es: Valor inicial 0, luego se inicia el siguiente for, en el que ira aumentando su valor z, para ir navendo toda la fila: [0][0], [0][1], [0][2], [0][3].

Cuando finaliza el for “z”. simplemente sale del bucle y termina de realizar las siguientes instrucciones que se encuentran el for “i”. Luego el valor de “i” cambia a uno y vuelve a realizar el for “z” pero ahora tendrá el valor “1” en lugar de “0”. Así sucesivamente hasta terminar la navegación de todo el vector.

Para encontrar el número más grande simplemente debemos hacer lo mismo que el ejercicio 5, pero adaptándolo a este nuevo ejercicio.

Ejercicio 22:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int
    i,z,vector_a[5][5],vector_b[5][5],suma[5][5],resta[5][5],columnas,filas;

    columnas = 0;
    filas = 0;

    cout << "Ingrese los valores del vector A";

    for (i = 0; i < 5; i++)
    {
        for (z = 0; z < 5; z++)
        {
            cout << "\nIngrese el numero en el espacio [" << i << "] ["
<< z << "] : ";
            cin >> vector_a[i][z];
        }
    }
}
```



```

    }
    cout << "Ingrese el valor del Vector B";

    for (i = 0; i < 5; i++)
    {
        for (z = 0; z < 5; z++)
        {
            cout << "\nIngrese el numero en el espacio [" << i << "] ["
<< z << "] : ";
            cin >> vector_b[i][z];
        }
    }

    //suma
    for (i = 0; i < 5; i++)
    {
        for (z = 0; z < 5; z++)
        {
            suma[i][z] = vector_a[i][z] + vector_b[i][z];
        }
    }

    //resta

    for (i = 0; i < 5; i++)
    {
        for (z = 0; z < 5; z++)
        {
            resta[i][z] = vector_a[i][z] - vector_b[i][z];
        }
    }

    // imprimir suma
    system("cls");

    for (i = 0; i < 5; i++)
    {
        for (z = 0; z < 5; z++)
        {
            gotoxy(5 + columnas, 5 + filas); cout << suma[i][z];
            columnas += 8;
        }
        filas += 2;
        columnas = 0;
    }

    filas = 0;

    //imprimir resta

    for (i = 0; i < 5; i++)
    {
        for (z = 0; z < 5; z++)
        {
            gotoxy(5 + columnas, 25 + filas); cout << resta[i][z];

```

```

        columnas += 8;
    }
    filas += 2;
    columnas = 0;
}

cout << "\n\n\n\n";

return 0;
}

```

Explicación:

Para realizar este ejercicio deberemos comprender como se realiza una suma de matrices.

$$A = \begin{pmatrix} 2 & 0 & 1 \\ 3 & 0 & 0 \\ 5 & 1 & 1 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

$$A+B = \begin{pmatrix} 2+1 & 0+0 & 1+1 \\ 3+1 & 0+2 & 0+1 \\ 5+1 & 1+1 & 1+0 \end{pmatrix} = \begin{pmatrix} 3 & 0 & 2 \\ 4 & 2 & 1 \\ 6 & 2 & 1 \end{pmatrix}$$

$$A-B = \begin{pmatrix} 2-1 & 0-0 & 1-1 \\ 3-1 & 0-2 & 0-1 \\ 5-1 & 1-1 & 1-0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & -2 & -1 \\ 4 & 0 & 1 \end{pmatrix}$$

Luego tendremos que llenar la información de 2 matrices diferentes, tal como hemos hecho en anteriores ejercicios. Una vez ingresada la información deberemos hacer la suma entre matrices. Deberemos observar al detalle cómo hacemos el recorrido de nuestro for en la matriz. En el caso de la suma y resta no es muy complicado, básicamente es sumar o resta el número que se encuentran en la misma posición en ambas matrices. Por último, tendremos que imprimir el resultado.

5	6	25	8	440
44	437	348	501	88
46	3446	7	8	100
10	67	6	5	7
6	37	396	78	469

1	2	-21	2	-428
-42	-431	-338	-367	2
-42	-3440	1	2	34
4	61	0	-1	1
0	31	-290	-8	-399

Este ejercicio deberá tener la siguiente forma:

El programa debe verse similar a lo siguiente (solamente recordar que las matrices son de 5x5):

MATRIZ A			MATRIZ B			SUMA			RESTA		
2	0	1	1	0	1	3	0	2	1	0	0
3	0	0	1	2	1	4	2	1	2	-2	1
5	1	1	1	1	0	6	2	1	4	0	1

Para ver si estas comprendiendo el tema, deberás agregar la impresión de la matriz A Y B. y acomodar todo de tal manera que sea agradable de ver para el usuario.

Ejercicio 23:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>
```

```

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int
    i,z,h,vector_a[4][4],vector_b[4][4],multiplicacion[4][4],columnas,filas;

    filas = 0;
    columnas = 0;
    cout << "Ingrese los valores del vector A";

    for (i = 0; i < 4; i++)
    {
        for (z = 0; z < 4; z++)
        {
            cout << "\nIngrese el numero en el espacio [" << i << "] ["
<< z << "] : ";
            cin >> vector_a[i][z];
        }
    }

    cout << "Ingrese los valores del vector B";

    for (i = 0; i < 4; i++)
    {
        for (z = 0; z < 4; z++)
        {
            cout << "\nIngrese el numero en el espacio [" << i << "] ["
<< z << "] : ";
            cin >> vector_b[i][z];
        }
    }

    for (i = 0; i < 4; i++)
    {
        for (z = 0; z < 4; z++)
        {
            multiplicacion[i][z] = ((vector_a[i][0] * vector_b[0][z]) +
(vector_a[i][1] * vector_b[1][z]) + (vector_a[i][2] * vector_b[2][z]) +
(vector_a[i][3] * vector_b[3][z]));
        }
    }

    system("cls");

```

```

        for (i = 0; i < 4; i++)
        {
            for (z = 0; z < 4; z++)
            {
                gotoxy(5 + columnas, 5 + filas); cout <<
multiplicacion[i][z];
                columnas += 8;
            }
            filas += 2;
            columnas = 0;
        }

        cout << "\n\n\n\n";

        return 0;
}

```

Explicación:

Para este ejercicio se deberá realizar la multiplicación de 2 matrices ingresadas por el usuario. Este tema como tal no es difícil de entender, pero si algo extensa la explicación. Aquí te dejo un video donde explican de manera sencilla el problema: [\(44\) Reto #12 | Multiplicación de matrices - YouTube](#)

Ejercicio 24:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i,opcion;
    double n,t,d;

```

```

do
{
    system("cls");
    gotoxy(5, 5); cout << "          MENU";
    gotoxy(5, 6); cout << "1. Cuadrado de un numero";
    gotoxy(5, 7); cout << "2. Raiz cuadrada de un numero";
    gotoxy(5, 8); cout << "3. Velocidad de un automovil";
    gotoxy(5, 9); cout << "4. Area de un circulo";
    gotoxy(5, 10); cout << "5. Salir";
    gotoxy(5, 12); cin >> opcion;

    switch (opcion)
    {
    case 1:
        system("cls");
        cout << "Ingrese el valor que desea elevar al cuadrado: ";
        cin >> n;
        cout << "\nEl numero elevado al cuadro es: " << n * n;
        cout << "\n\n\n\n";
        system("PAUSE");
        break;

    case 2:
        system("cls");
        cout << "Ingrese el valor que desea encontrar la raiz: ";
        cin >> n;
        cout << "\nLa raiz del numero es: " << sqrt(n);
        cout << "\n\n\n\n";
        system("PAUSE");

        break;

    case 3:
        system("cls");
        system("cls");
        cout << "Este programa calculara la velocidad de un
automovil ingresando el tiempo y la distancia";
        cout << "\nIngrese el tiempo: ";
        cin >> t;
        cout << "\nIngrese la distancia: ";
        cin >> d;
        cout << "\nLa velpcidad del automovil es: " << d / t << "
metros por segundo";
        cout << "\n\n\n\n";
        system("PAUSE");
        break;

    case 4:
        system("cls");
        cout << "Este programa calculara el area de un circulo
sabiendo su radio";
        cout << "\nIngrese el radio del circulo:";
        cin >> n;

        cout << "\nEl area del circulo es: " << 3.1416 * (n * n);;

```

```

        cout << "\n\n\n\n";
        system("PAUSE");

        break;

    case 5:
        break;

    default:
        system("cls");
        cout << "Ingrese un dato valido";
        cout << "\n\n\n\n";
        system("PAUSE");
        break;

    break;

}

} while (opcion != 5);

cout << "\n\n\n\n";

system("PAUSE");
return 0;
}

```

Explicación:

En este ejercicio utilizaremos nuevos operadores que no hemos utilizado, así que explicare cada uno de estos. Primero explicare que es el do while, lo que hace esta función es básicamente como si fuera un for. Lo que hace es, primero hacer todas las instrucciones que tiene la función, una vez hechas todas instrucciones que tiene la función, verifica la condición (lo que se encuentra en la parte de abajo). Si se cumple la condición, volverá a hacer todas las instrucciones que tiene el do while y así sucesivamente hasta que se deje de cumplir la condición.

En este ejemplo para crear menús, lo que hago es crear una variable que se llame opción, ahí el usuario ingresará el numero de la opción que desea realizar, si el usuario escoge el numero 5 el programa deberá detenerse porque es la opción de salir. ¿Como escribimos esa condición? En este caso escribo la siguiente condición:

```

} while (opcion != 5);

```

Esto en otras palabras dice "Haz este bucle siempre y cuando el numero sea diferente de 5", porque el bucle se debe de repetir hasta que el usuario ingrese el número 5. Cuando se ingresa el número 5 se rompe la condición y así finaliza el programa.

Ahora la función switch, el switch básicamente es un "if" pero más bonito, viene a cumplir una función similar, primero escribimos switch, luego dentro del switch escribimos la variable la cual vamos a utilizar (en este ejemplo es la opción), luego escribimos la palabra "Case". El "Case" vendría a sustituir el "if", ya que en otras palabras dice "Si en CASO que la variable "opción" sea el numero 1 entonces...". Después de cada "case" pondremos las instrucciones que se realicen en caso que la condición se cumpla, al final de las instrucciones recordar poner "break;".

Como te podrás dar cuenta, dentro del switch hay una opción que se llama "Default". ¿Qué es default?, el default viene a cumplir la misma función que el case, pero este sirve específicamente cuando se ingresa un dato el cual no está especificado. En este ejemplo escribí un case para los números 1,2,3,4 y 5. Si se ingresara un numero fuera de ese rango se ejecutaría la opción default, en este caso hago que la consola imprima el mensaje "Escribir un dato valido", haciendo referencia a que la persona tiene que escribir un número que se encuentre 1 y 5 .

Ejercicio 25:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i,nit[7];
    float subtotal[7], iva[7], total[7],precio[7], cantidad[7];
    string nombre[7],direccion[7],nombreproducto[7];

    cout << "
    nombre      NOMBRE      DIRECCION      NIT      Producto-
    nombre      Cantidad      PRECIO      Subtotal      IVA      TOTAL";

    for (i = 0; i < 7; i++)
    {
```



```

        gotoxy(1, 2 + i); cin >> nombre[i];
        gotoxy(21, 2 + i); cin >> direccion[i];
        gotoxy(40, 2 + i); cin >> nit[i];
        gotoxy(53, 2 + i); cin >> nombreproducto[i];
        gotoxy(79, 2 + i); cin >> cantidad[i];
        gotoxy(93, 2 + i); cin >> precio[i];

        subtotal[i] = cantidad[i] * precio[i];
        iva[i] = subtotal[i] * 0.12;
        total[i] = subtotal[i] - iva[i];

        gotoxy(108, 2 + i); cout<<subtotal[i];
        gotoxy(122, 2 + i); cout << iva[i];
        gotoxy(131, 2 + i); cout << total[i];

    }
    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}

```

Explicación:

Este ejercicio es la combinación de ejercicios ya anteriormente vistos. Como recomendación, siempre verificar cuales variables pueden ser int y cuáles pueden ser double o float. Como reto, te propongo que al mostrar la información tabules la información, debe aparecer la información de esta manera:

NOMBRE	DIRECCION	NIT	Producto-nombre	Cantidad	PRECIO	Subtotal	IVA
miguel	2da.avenida	32142	Manzana	2	10	20	2.4
Ana	3.avenida	5543	jugo	10	10	100	12
xavier	zona12	234	peras	1	2	2	0.24

Ejercicio 26:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {

```

```

COORD punto;
punto.X = x; punto.Y = y;
SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i,nit,z,contador,tarifa,derecho;
    float subtotal, iva,
total,cuota,cantidad,kws_gastados,kws_demas,kw_adicional;
    string nombre,direccion,nombreproducto;

    gotoxy(5, 3); cout << "NOMBRE:";
    gotoxy(5, 5); cout << "DIRECCION:";
    gotoxy(5, 7); cout << "NO. CONTADOR:";
    gotoxy(5, 9); cout << "TARIFA A LA QUE PERTENECE:";
    gotoxy(5, 11); cout << "KWS GASTADOS EN EL MES:";

    gotoxy(35, 3); cin>>nombre;
    gotoxy(35, 5); cin >>direccion;
    gotoxy(35, 7); cin >> contador;
    gotoxy(35, 9); cin >> tarifa;

    while (tarifa != 3 && tarifa != 2 && tarifa != 1)
    {
        gotoxy(35, 9); cout << "ERROR Ingrese un numero de tarifa valido
(1,2 o 3)";
        gotoxy(5, 25); system("PAUSE");
        gotoxy(5, 25); cout << "
";
        gotoxy(35, 9); cout << "
";
        gotoxy(35, 9); cin >> tarifa;
    }

    gotoxy(35, 11); cin >> kms_gastados;
    system("cls");

    if (tarifa == 1)
    {
        derecho = 300;
        cuota = 200;
        kw_adicional = 1.5;
        if (kws_gastados>derecho)
        {
            kms_demas = kms_gastados - derecho;
        }
        else
        {
            kms_demas = 0;
        }
        subtotal = cuota + (kws_demas * kw_adicional);
    }
    else if (tarifa == 2)
    {
        derecho = 500;

```

```

        cuota = 365;
        kw_adicional = 1;
        if (kws_gastados > derecho)
        {
            kws_demas = kws_gastados - derecho;
        }
        else
        {
            kws_demas = 0;
        }
        subtotal = cuota + (kws_demas * kw_adicional);
    }
    else
    {
        derecho = 800;
        cuota = 500;
        kw_adicional = 0.5;
        if (kws_gastados > derecho)
        {
            kws_demas = kws_gastados - derecho;
        }
        else
        {
            kws_demas = 0;
        }
        subtotal = cuota + (kws_demas * kw_adicional);
    }
}

```

```
iva = subtotal * 0.12;
```

```
total = iva + subtotal;
```

```

gotoxy(8, 3); cout << "NOMBRE:";
gotoxy(8, 5); cout << "DIRECCION:";
gotoxy(8, 7); cout << "NO. CONTADOR:";
gotoxy(8, 9); cout << "TARIFA A LA QUE PERTENECE:";
gotoxy(8, 11); cout << "KWS A LOS QUE TIENE DERECHO:";
gotoxy(8, 13); cout << "KWS GASTADOS DE MAS";
gotoxy(8, 15); cout << "CUOTA:";
gotoxy(8, 17); cout << "PAGO EXTRA:";
gotoxy(8, 19); cout << "SUBTOTAL:";
gotoxy(8, 21); cout << "IVA:";
gotoxy(8, 25); cout << "TOTAL A PAGAR:";

```

```

gotoxy(40, 3); cout<<nombre;
gotoxy(40, 5); cout <<direccion;
gotoxy(40, 7); cout <<contador;
gotoxy(40, 9); cout <<tarifa;
gotoxy(40, 11); cout << derecho;
gotoxy(40, 13); cout <<kws_demas ;
gotoxy(40, 15); cout << cuota;
gotoxy(40, 17); cout << kw_adicional*kws_demas;
gotoxy(40, 19); cout << subtotal;
gotoxy(40, 21); cout << iva;
gotoxy(40, 25); cout << total;

```

```

        cout << "\n\n\n\n";

        system("PAUSE");
        return 0;
    }

```

Explicación:

Para este ejercicio, no nos piden que podamos ingresar más de 1 usuario. Así que no es necesario utilizar un for. En este ejemplo cambie el modo en el cual se visualizan los datos, para variar un poco.

En este programa se debe prestar mucha atención a los datos finales que nos pide el programa, debido a que existen 3 tarifas diferentes, en el cual cada tarifa tiene diferentes valores en su cuota a pagar, lo kws derecho etc.

Así que recomiendo primero leer el problema y analizar desde el inicio las variables que se deben crear. Una vez analizado el problema, procedemos a programar. En este ejercicio tuve que crear un if para cada tarifa. ¿Qué sucede si el usuario ingresa una tarifa que no existe?

```

while (tarifa != 3 && tarifa != 2 && tarifa != 1)
{
    gotoxy(35, 9); cout << "ERROR Ingrese un numero de tarifa valido (1,2 o 3)";
    gotoxy(5, 25); system("PAUSE");
    gotoxy(5, 25); cout << " ";
    gotoxy(35, 9); cout << " ";
    gotoxy(35, 9); cin >> tarifa;
}

```

Para solucionar este error se utiliza un while, se verifica la condición y si se cumple realiza las instrucciones que tiene el while. En este ejemplo, el while se cumplirá si el usuario ingresa cualquier número que no sea 1,2 o 3. Cuando sucede esto salta un mensaje de error y le pide al usuario que vuelva a ingresar información hasta que sea correcta. Mucho cuidado con la información, creo varios cout vacíos debido a que se utilizan para borrar de la consola, la información que estaba ahí antes. Sino se realiza este paso, quedara en la consola información previa y podría ser confuso para el usuario.

Además, al realizar este while garantizamos que las únicas posibles respuestas sean 1,2,3. Así que solo deberemos crear un if para cada tarifa. En cada if se deberá introducir la diferencia de precio de cuota a pagar, precio por kw adicional etc.

Ejercicio 27:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i, opcion, n;
    double suma, t, d, mayor, menor, numero;

    do
    {
        system("cls");
        gotoxy(5, 5); cout << "-----MENU-----";
        gotoxy(5, 6); cout << "1. Mayor y menor de una lista de numeros";
        gotoxy(5, 7); cout << "2. Sumatoria de N numeros";
        gotoxy(5, 8); cout << "3. Salir";

        gotoxy(5, 12); cin >> opcion;

        switch (opcion)
        {
            case 1:
                system("cls");
                cout << "Ingrese la cantidad de numeros que desea ingresar: ";
                cin >> n;

                mayor = 0;
                menor = 0;

                for (i = 0; i < n; i++)
                {
                    cout << "\nIngrese el numero " << i + 1 << ": ";
                    cin >> numero;

                    if (i == 0)
                    {
                        mayor = numero;
                        menor = numero;
                    }
                    else
                    {
                        if (numero > mayor)
                        {

```

```

        mayor = numero;
    }

    if (numero < menor)
    {
        menor = numero;
    }
}

cout << "\nEl numero mayor es: " << mayor;
cout << "\nEl numero menor es: " << menor;
cout << "\n\n\n\n";
system("PAUSE");
break;

case 2:
    system("cls");
    cout << "Ingrese la cantidad de numeros que desea sumar: ";
    cin >> n;
    suma = 0;

    for (i = 0; i < n; i++)
    {
        cout << "\nIngrese el numero " << i + 1 << ": ";
        cin >> numero;
        suma = suma + numero;
    }
    cout << "\nLa sumatoria es de: " << suma;
    cout << "\n\n\n\n";
    system("PAUSE");
    break;

case 3:
    break;

default:
    system("cls");
    cout << "Ingrese un dato valido";
    cout << "\n\n\n\n";
    system("PAUSE");
    break;

    break;
}

} while (opcion != 3);

cout << "\n\n\n\n";

system("PAUSE");
return 0;
}

```

Explicación:

Este ejercicio contiene varios temas vistos anteriormente, pero ahora nos piden que agreguemos un menú. Utilizamos un do while para hacer el menú del programa. La única diferencia con los for de este programa es que el usuario decide cuantos números va a ingresar, en otras palabras, decide cuantos ciclos hará el for. En la opción 1, la opción de encontrar el mayor y el menor número, la realice de una manera diferente a anteriores ejercicios, pero el resultado es el mismo. Cuando la "i" vale 0 hago que las variables mayor y menor tomen el valor del primer número ingresado. Entonces cuando se encuentra en el siguiente ciclo: "1". Lo que hace es comparar el valor ingresado anteriormente, con el nuevo valor y así verificar si es mayor o menor. Si se cumple la condición simplemente se actualiza el nuevo número mayor o menor encontrado.

Ejercicio 28:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

#include <chrono>

#include <thread>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i, contador1, contador2, contador3;

    gotoxy(5, 5); cout << " HORA      MINUTOS      SEGUNDOS";
    contador1 = 0;
    contador2 = 0;
    contador3 = 0;
    for (i = 1; contador3<=24 ; i++)
    {
        contador1 += 1;
        Sleep(1000);
        if (contador1==60)
        {
```

```

        contador1 = 0;
        contador2 += 1;
    }

    if (contador2==60)
    {
        contador3 += 1;
        contador2 = 0;
        contador1 = 0;
    }

    gotoxy(7, 7); cout << contador3;
    gotoxy(11, 7); cout << contador2;
    gotoxy(23, 7); cout << contador1;

}

cout << "\n\n\n\n";

system("PAUSE");
return 0;
}

```

Explicación:

En este ejercicio utilizaremos una nueva función llamada "Sleep()" (Para que funcione esta función necesitaremos las siguientes librerías: `#include <chrono>` y `#include <thread>`).

Lo que hace la función sleep es pausar la ejecución del programa por un determinado tiempo, el tiempo se introduce dentro del sleep y se escribe en milisegundos, por lo que si quieren pausar el programa 1 segundo se escribiría de la siguiente manera: `Sleep(1000)`. Este programa nos pide que emulemos un reloj, por lo que tendremos que crear 3 contadores, horas, minutos y segundos. Como sabrán un minuto tiene 60 segundos, 1 hora 60 minutos y el día 24 horas.

Así que lo que hacemos es lo siguiente: El contador de segundos cuando llegue a 60, quiere decir que ese tiempo ya es un minuto, así que reinicia el contador de segundos y se le suma 1 al contador de minutos. Lo mismo aplicara para las horas, cuando el contador de minutos llegue a 60. Se reiniciará el contador de minutos y se sumará 1 al contador de horas. Este proceso seguirá así hasta llegar a las 24 horas del día. En mi ejemplo el for se detiene al llegar a las 24 horas, pero un reloj real lo que haría es reiniciar todos los contadores a 0 para iniciar el nuevo día.

Ejercicio 29:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>

```



```

#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i;
    double n, factorial;

    gotoxy(5, 5); cout << "Ingrese un numero entre 1 y 20";
    gotoxy(5, 7); cin >> n;
    factorial = 1;
    while (n<1 || n>20)
    {
        gotoxy(5, 7); cout << "ERROR, DEBE INGRESAR UN NUMERO QUE SEA
MAYOR A 1 Y MENOR A 20";
        gotoxy(5, 10); system("PAUSE");
        gotoxy(5, 7); cout << "
";
        gotoxy(5, 10); cout << "
";
        gotoxy(5, 7); cin >> n;
    }

    for (i = 1; i <n+1 ; i++)
    {
        factorial = factorial * i;
    }
    cout << "\nEl factorial del numero " << n << " es: " << factorial;

    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}

```

Explicación:

Para este ejercicio primero debemos saber que es un numero factorial, aquí dejo unos ejemplos de números factoriales:

$$\begin{aligned}
 0! &= 1 \\
 1! &= 1 \\
 2! &= 1 \cdot 2 = 2 \\
 3! &= 1 \cdot 2 \cdot 3 = 6 \\
 4! &= 1 \cdot 2 \cdot 3 \cdot 4 = 24 \\
 5! &= 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 = 120 \\
 6! &= 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 = 720
 \end{aligned}$$

Con esto en mente empezamos a programar, en las instrucciones nos piden que salte un mensaje de error en caso de que la persona ingrese un número que sea menor a 1 o si es mayor a 20. Tal como en ejercicios anteriores, deberemos usar un while y declarar ahí la condición para que se cumpla la condición.

```
while (n<1 || n>20)
{
    gotoxy(5, 7); cout << "ERROR, DEBE INGRESAR UN NUMERO QUE SEA MAYOR A 1 Y MENOR A 20";
    gotoxy(5, 10); system("PAUSE");
    gotoxy(5, 7); cout << "
    gotoxy(5, 10); cout << "
    gotoxy(5, 7); cin >> n;
}
```

En este caso utilizamos el operador “or” también se puede escribir “||”. Aquí estamos diciendo en otras palabras “Ejecutar el while si el número es menor a 1 o si es mayor a 20”. No utilizamos “and” en este caso porque si ingresamos, por ejemplo -1, el número será menor a -1... Pero no será mayor a 20. Si utilizamos “and” no se ejecutaría el while, porque se deben de cumplir ambas condiciones para ejecutarse el while.

Para el número factorial basta con crear un for e ir multiplicando cada ciclo con la variable i.

Ejercicio 30:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;
```

```

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i;
    double lado1, lado2, lado3, perimetro, area;

    cout << "\nIngrese los 3 numeros para saber si forman un triangulo";
    cout << "\nIngrese el lado1: ";
    cin >> lado1;
    cout << "\nIngrese el lado2: ";
    cin >> lado2;
    cout << "\nIngrese el lado3: ";
    cin >> lado3;

    if (lado1+lado2>lado3 && lado1+lado3>lado2 && lado2 + lado3 > lado1 )
    {
        cout << "\nLos 3 lados si forman un triangulo";

        perimetro = lado1 + lado2 + lado3;
        area = sqrt(perimetro * (perimetro - lado1) * (perimetro - lado2) *
(perimetro - lado3));

        cout << "\nEl area del triangulo es: " << area;
        cout << "\nEl perimetro del triangulo es: " << perimetro;

    }
    else
    {
        cout << "\nLos 3 lados no forman un triangulo";
    }

    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}

```

Explicación:

El programa inicia solicitando al usuario tres datos, estos tres datos son el tamaño de cada lado respectivamente de un triángulo, luego en el if verificamos que los lados del triángulo si forme un triángulo, para eso suma dos lados y verifica si la suma de los dos lados es mayor que el tercer lado, y va verificado lo mismo para cada lado, si se cumple la condición el programa concluye que si es un triángulo y si no se cumple concluye que no es un triángulo.

formula:

$$\begin{aligned}a + b &> c \\a + c &> b \\b + c &> a\end{aligned}$$

Given:

$$a = 7 \quad b = 10 \quad c = 5$$

Si se cumple que los lados si forman un triángulo rectángulo el programa suma todos los lados para obtener el perímetro del triángulo, luego para sacar el área utiliza la formula:

$$\sqrt{P * (P - L1) * (P - L2) * (P - L3)}$$

Ejercicio 31:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
```

```

#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i, edad;

    gotoxy(5, 5); cout << "Ingrese la edad de la persona: ";
    gotoxy(5, 7); cin >> edad;

    while (edad<0)
    {
        gotoxy(5, 7); cout << "ERROR la edad no puede ser negativa";
        gotoxy(5, 10); system("PAUSE");
        gotoxy(5, 7); cout << "
";
        gotoxy(5, 10); cout << "
";
        gotoxy(5, 7); cin >> edad;
    }
    system("cls");
    if (edad>= 18)
    {
        cout << "Es mayor de edad, puede ingresar a la fiesta";
    }
    else
    {
        cout << "Es menor de edad, debe irse a dormir";
    }

    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}

```

Explicación:

El programa inicia pidiendo al usuario la edad de una persona, luego verifica con un While que la edad ingresada no sea menor a cero, si la edad es menor a cero vuelve a pedir la edad de la persona hasta que esta cumpla la condición de que no sea menor a cero, luego con un if verifica si la edad es mayor a 18, si la edad es mayor a 18 el programa manda un mensaje diciendo, que es mayor de edad, y si la edad es menor de 18 manda un mensaje diciendo, que es menor de edad.

Ejercicio 32:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i, contador, suma;

    cout << "Los primeros 30 numeros impares son:";
    suma = 0;
    contador = 0;
    for ( i = 0; contador<=30; i++)
    {
        if (i%2 != 0)
        {
            cout << "\n" << i;
            contador += 1;
            suma += i;
        }
    }

    cout << "\n La suma de los nuemros es: " << suma;

    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}
```

Explicación:

El programa va a ir iterando del 1 al 30, es decir que va a ir contando de uno en uno hasta llegar a 30, luego con el if va a ir dividiendo el número de cada iteración en 2 para sacar el residuo de la división, es decir que cuando el contador este en uno 1 va a dividir ese 1 en 2 para verificar si es igual a cero, si es diferente a cero el número es impar, si es igual a cero

es par. Si es impar muestra en pantalla el número que determino que es impar, luego a la variable “contador” se le suma uno en cada iteración, luego en la variable “suma” suma el valor actual de cada iteración de la variable “suma” (en cada iteración) con el valor actual de cada iteración de la variable.

Ejercicio 33:

Código:

```
#include<iostream>
```

```
#include<cstdio>
```

```
#include<string>
```

```
#include<Windows.h>
```

```
#include<dos.h>
```

```
#include<conio.h>
```

```
#include<math.h>
```

```
using namespace std;
```

```
void gotoxy(int x, int y) {
```

```
    COORD punto;
```

```
    punto.X = x; punto.Y = y;
```

```
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
```

```
}
```

```
int main() {
```

```
    int i,contador,suma,n,numero,promedio;
```

```

cout << "Ingrese la cantidad de numeros que desea ingresar: ";

cin >> n;

suma = 0;

for (i = 0; i < n; i++)
{
    cout << "\nIngrese el numero: " << i + 1<<": ";

    cin >> numero;

    suma = suma + numero;

}

promedio = suma / n;

cout << "\nEl promedio es de: " << promedio;


cout << "\n\n\n\n";


system("PAUSE");

return 0;

}

```

Explicación:

Se le solicita al usuario la cantidad de valores que desea ingresar, la cantidad de valores será igual a la cantidad de ciclos que hará el for, adentro del for va pidiendo el valor de cada dato que desea ingresar, y va sumando todos los valores de los datos ingresados, luego de tener la suma saca el promedio multiplicando la suma total con la cantidad de datos ingresado, para después mostrar en pantalla un mensaje indicando el promedio.

Ejercicio 34:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i, contador, suma, n, numero, promedio;

    cout << "Ingrese el numero que desee para mostrar su tabla de
multiplicar: ";
    cin >> n;

    for (i = 1; i < 51; i++)
    {
        cout << "\n      " << n << " * " << i << " = " << n * i;
    }
    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}
```

Explicación:

El programa solicita a usuario que tabla de multiplicar desea, el for va a ciclar hasta 50, adentro del for va a ir mostrando en la pantalla el numero de la tabla ingresada e ira multiplicando cada número del for hasta llegar a 50 y mostrando el resultado de cada multiplicación.

Ejercicio 35:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
```

```

#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    double utilidad, n, n_ventas, iva, comision, venta, suma_ventas,
    suma_iva, suma_comisiones, suma_utilidad;
    int i;

    cout << "Ingrese cuantas ventas realizo: ";
    cin >> n_ventas;
    suma_ventas = 0;
    suma_iva = 0;
    suma_comisiones = 0;
    suma_utilidad = 0;

    for (i = 0; i < n_ventas; i++)
    {
        cout << "\nIngrese el valor de la venta numero " << i + 1<<" : ";
        cin >> venta;
        iva = venta * 0.12;
        comision = venta * 0.15;
        utilidad = venta - (venta * 0.7);

        suma_ventas = suma_ventas+venta;
        suma_iva = suma_iva+iva;
        suma_comisiones = suma_comisiones+comision;
        suma_utilidad = suma_utilidad+utilidad;
    }

    system("cls");
    gotoxy(5, 5); cout << "TOTAL VENTAS          TOTAL IVA          TOTAL
COMISION          TOTAL UTILIDAD";
    gotoxy(5, 7); cout << suma_ventas;
    gotoxy(30, 7); cout << suma_iva;
    gotoxy(47, 7); cout << suma_comisiones;
    gotoxy(68, 7); cout << suma_utilidad;

    cout << "\n\n\n\n";

    system("PAUSE");
    return 0;
}

```

Explicación:

Se le solicita al usuario la cantidad de ventas realizadas que desea ingresar, el for ira iterando por esa cantidad, adentro del for pide el valor de cada venta. Luego sacara el valor del iva, la comisión para el vendedor y la utilidad, luego ira sumando los valores antiguos con los nuevos en cada ciclo del for. Para luego mostrar todo en pantalla.

Ejercicio 36:

Código:

```
#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    double numero, mayor, menor, diferencia;
    int i, n;

    cout << "Cuantos numeros desea ingresar: ";
    cin >> n;

    mayor = 0;
    menor = 0;

    for (i = 0; i < n; i++)
    {
        cout << "\nIngresa el numero " << i + 1 << ": ";
        cin >> numero;

        if (i == 0)
        {
            mayor = numero;
            menor = numero;
        }
        else
        {
            if (numero > mayor)
            {
```

```

        mayor = numero;
    }

    if (numero < menor)
    {
        menor = numero;
    }
}

diferencia = mayor - menor;
cout << "\nEl numero mayor es: " << mayor;
cout << "\nEl numero menor es: " << menor;
cout << "\nLa diferencia es de: " << diferencia;
cout << "\n\n\n\n";

system("PAUSE");
return 0;
}

```

Explicación:

Se le solicita al usuario la cantidad de valores que desea ingresar, la cantidad de valores será igual a la cantidad de ciclos que hará el for, adentro del for pide asignarle un valor a una variable, si el número de ciclo es igual a cero asigna el valor del número ingresado a dos variables “Menor” y “Mayor”, si el nuevo ciclo es diferente a cero, otro if verifica si el nuevo número ingresado es mayor a el valor de la variable “Mayor” si es mayor se asigna el valor del nuevo número a la variable “Mayor” pero si es menor al valor de la variable “Menor”, se le asigna el valor del nuevo número a la variable “Menor”, al final se resta el mayor con el menor y se muestra en pantalla el valor del menor y la del mayor y la diferencian entre ellos.

Ejercicio 37:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>

using namespace std;

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

```

```
}
```

```
int main() {
    int i, opcion, n;
    double suma, t, d, mayor, menor, numero, contador;

    do
    {
        system("cls");
        gotoxy(5, 5); cout << "-----MENU-----";
        gotoxy(5, 6); cout << "1. Tabla de multiplicacion de un numero
'n';
        gotoxy(5, 7); cout << "2. Suma de los numeros impares de 1 a n";
        gotoxy(5, 8); cout << "3. Salir";

        gotoxy(5, 12); cin >> opcion;

        switch (opcion)
        {
            case 1:
                system("cls");
                gotoxy(5, 5); cout << "Ingrese el numero que desee para
mostrar su tabla de multiplicar: ";
                gotoxy(5, 7); cin >> n;

                while (n < 0)
                {
                    gotoxy(5, 7); cout << "ERROR Ingresar un numero que
no sea negativo";
                    gotoxy(5, 10); system("PAUSE");
                    gotoxy(5, 7); cout << "
";
                    gotoxy(5, 10); cout << "
";
                    gotoxy(5, 7); cin >> n;
                }
                system("cls");
                for (i = 1; i < 21; i++)
                {
                    cout << "\n      " << n << " * " << i << " = " << n *
i;

                }
                cout << "\n\n\n\n";

                system("PAUSE");

                break;

            case 2:
                system("cls");
                cout << "Ingrese un numero: ";
                cin >> n;
                suma = 0;
                contador = 0;
                for (i = 0; i < n; i++)
                {
```

```

        if (i % 2 != 0)
        {
            cout << "\n" << i;
            contador += 1;
            suma += i;
        }

    }

    cout << "\n La suma de los nuemros es: " << suma;
    cout << "\n\n\n\n\n";

    system("PAUSE");
    break;

case 3:
    break;

default:
    system("cls");
    cout << "Ingrese un dato valido";
    cout << "\n\n\n\n\n";
    system("PAUSE");
    break;

    break;
}

} while (opcion != 3);

cout << "\n\n\n\n\n";

system("PAUSE");
return 0;
}

```

Explicación:

Ejercicio 38:

Código:

```

#include<iostream>
#include<cstdio>
#include<string>
#include<Windows.h>
#include<dos.h>
#include<conio.h>
#include<math.h>
#include <chrono>
#include <thread>

using namespace std;

```

```

void gotoxy(int x, int y) {
    COORD punto;
    punto.X = x; punto.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), punto);
}

int main() {
    int i, opcion, n;
    double suma, t, d, mayor, menor, numero;

    do
    {
        system("cls");
        gotoxy(5, 5); cout << "-----MENU-----";
        gotoxy(5, 6); cout << "1. Numeros impares de 0 a 200";
        gotoxy(5, 7); cout << "2. Numeros pares de 0 a 200";
        gotoxy(5, 8); cout << "3. Salir";

        gotoxy(5, 12); cin >> opcion;

        switch (opcion)
        {
            case 1:
                system("cls");
                suma = 0;
                contador = 0;
                for (i = 0; i < 201; i++)
                {
                    if (i % 2 != 0)
                    {
                        cout << "\n" << i;

                        Sleep(100);
                        suma += i;
                    }
                }

                cout << "\n La suma de los numeros impares es: " << suma;
                cout << "\n\n\n\n\n";

                system("PAUSE");

                break;

            case 2:
                system("cls");
                suma = 0;
                contador = 0;
                for (i = 0; i < 201; i++)
                {

```

```

        if (i % 2 == 0)
        {
            cout << "\n" << i;
            Sleep(100);

            suma += i;
        }

    }

    cout << "\n La suma de los numeros pares es: " << suma;
    cout << "\n\n\n\n\n";

    system("PAUSE");
    break;

case 3:
    break;

default:
    system("cls");
    cout << "Ingrese un dato valido";
    cout << "\n\n\n\n\n";
    system("PAUSE");
    break;

    break;
}

} while (opcion != 3);

cout << "\n\n\n\n\n";

system("PAUSE");
return 0;
}

```

Explicación:

Tenemos que basado en la selección de opción el programa elige un switch donde al seleccionar el caso número 1, obtenemos el comando para la limpieza de pantalla, aparte de ellos, tenemos dos contadores los cuales se utilizaran para la suma y el almacenamiento del número. Utilizamos un for donde el entero i empieza a sumarse hasta llegar a todos los números menores que 201, luego de ello obtenemos otro if donde se verifica que si el residuo de i entre 2 es diferente de 0 entonces es un número impar, el cual las opciones de cout lo muestra, aparte de mostrar, también la variable de suma se empieza a sumar, siendo que la suma es igual a la suma más i, dado esto al finalizar el ciclo inicial de números menores a 201 muestra la suma de todos aquellos impares entre el ciclo y despliega los números impares

Para el caso 2 tenemos que el sistema borra pantalla, luego declaramos nuestras variables que utilizaremos como contadores, posterior a ello, se dictamina otro ciclo for donde vemos una variable i que irá sumando cada vez hasta llegar a todos los números menores a 201, luego al entrar al ciclo vemos que se encuentra un if donde se declara que si el residuo del número dividido 2 es igual a 0 entonces se muestra la variable i, además de ello en la variable suma va sumando entre suma más i, al finalizar el ciclo se muestra la suma de los números pares así como los números pares.