

4.1



Ejercicios Simples

1. Escribir un programa que muestre en pantalla un saludo.

```
#include <iostream>
using namespace std;

int main()
{
    cout << "Bienvenido a mi primer programa en C++" << endl;
    return 0;
}
```

2. Escribir un programa que sume dos números: a=7 b=4.

```
#include <iostream>
using namespace std;

int main()
{
    int a=7, b=4;
    cout << "La suma de " << a << " + " << b << " es " << a+b << endl;
    return 0;
}
```

3. Escribir un programa que lea dos números y los sume.

```
#include <iostream>
using namespace std;

int main()
{
    int a, b;
    cout << "Ingrese el primer numero: ";
    cin >> a;
    cout << "Ingrese el segundo numero: ";
    cin >> b;
    cout << "La suma de " << a << " + " << b << " es " << a+b << endl;
    return 0;
}
```

4. Escribir un programa que sume, reste, multiplique y divida dos números: x = 10 y = 2.

5. Escribir un programa que sume, reste, multiplique y divida dos números leídos desde el teclado.



6. Escribir un programa que calcule el área de un rectángulo de lado1 = 3 y lado2 = 4.
área del rectángulo=lado1 * lado2
7. Modifique el ejercicio anterior para que los valores de los lados sean suministrados por el usuario.
8. Escribir un programa que calcule el área de un triángulo, capturando los valores de base y altura.
área del triángulo = (base * altura)/2
9. Escribir un programa que calcule la longitud y el área de una circunferencia, capturando el valor del radio.
longitud de la circunferencia = $2 * \pi * \text{radio}$
área de la circunferencia = $\pi * \text{radio}^2$
10. Escribir un programa que calcule la velocidad de un proyectil que recorre x Km en t minutos. Expresar el resultado en metros/segundo.
Velocidad = espacio/tiempo
11. Escribir un programa que calcule el volumen de una esfera de radio = r volumen de la esfera
 $= \frac{4}{3} * \pi * \text{radio}^3$
12. Escribir un programa que evalúe la siguiente expresión:
 $(a+7*c)/(b+2-a)+2*b$
13. Escribir un programa que evalúe la siguiente expresión:
 $(a+5) * 3 / 2 * b - b$
14. Escribir un programa que evalúe la siguiente expresión:
 $(-b + \sqrt{b^2 - 4*a*c})/(2*a)$.

```
#include <iostream>
#include <math.h>
using namespace std;
int main()
{
    double a, b, c;
```



```
cout << "Ingrese el valor de a: ";
cin >> a;
cout << "Ingrese el valor de b: ";
cin >> b;
cout << "Ingrese el valor de c: ";
cin >> c;
cout << endl;
cout << (sqrt(b*b) + 2*a) / (2*a);
cout << "El resultado de la expresión es: " << endl;
return 0;
```

* 15. Escribir un programa que calcule las raíces de una ecuación de 2º grado.

$$(-b + \sqrt{b^2 - 4ac}) / (2a)$$

$$(-b - \sqrt{b^2 - 4ac}) / (2a)$$

16. Escribir un programa que calcule la hipotenusa de un triángulo rectángulo.

✓ 17. Escribir un programa que calcule el equivalente en grados Fahrenheit o Celsius de una temperatura t.

$$\text{Celsius} = (\text{Fahrenheit} - 32) / 9$$

18. Escribir un programa que calcule el número de horas, minutos y segundos que hay en 3700 segundos.

19. Escribir un programa que calcule el capital producido por un capital de x de pesos, al cabo de un año depositado a un interés del y% mensual.

20. Escribir un programa que calcule el equivalente en pies de una longitud de x metros sabiendo que 1 metro equivale a 39.27 pulgadas y 12 pulgadas equivalen a 1 pie.

21. Escribir un programa que calcule el área de un rectángulo a partir de sus coordenadas x1, y1, x2, y2.

22. Escribir un programa que lea dos números enteros A y B, y obtenga los valores de la división entera de A dividido B y el residuo de esta.

23. Escribir un programa que convierta un número de segundos en su equivalente en minutos y segundos.



Condicionales

24. Escribir un programa que detecte si un número introducido desde el teclado es positivo o negativo.

```
#include <iostream>
using namespace std;

int main()
{
    int n;
    cout << "ingrese el valor de n: ";
    cin >> n;
    if (n > 0)
        cout << "el número leído es positivo" << endl;
    else
        cout << "el número leído es negativo" << endl;
    return 0;
}
```

25. Escribir un programa que detecte si se han introducido en orden creciente tres números introducidos por el usuario.

26. Escribir un programa que determine si un número leído desde el teclado es par o impar

27. Escribir un programa que detecte si un número leído desde el teclado es mayor o menor que 100.

28. Escribir un programa que dado un número del 1 a 7 escriba el correspondiente nombre del día de la semana.

29. Escribir un programa que lea un carácter e indique si es o no una vocal.

```
#include <iostream>
using namespace std;

int main()
{
    return 0;
}
```

30. Escribir un programa que lea dos números desde el teclado y si el primero es mayor que el



segundo intercambie sus valores.

31. Escribir un programa que lea desde teclado el importe bruto de una factura y determine el importe neto según los siguientes criterios.

Importe bruto menor de 20.000 -> sin descuento

Importe bruto mayor de 20.000 -> 15% de descuento

32. Escribir un programa que una vez leída una hora en formato (horas, minutos, segundos) indique cual será el tiempo dentro de un segundo.

Ciclos

33. Escribir un programa que realice un bucle con while y muestre en pantalla del 1 al 10.

34. Escribir un programa que realice un bucle con do ... while y muestre en pantalla del 1 al 10.

35. Escribir un programa que realice un bucle con For y muestre en pantalla del 1 al 10.

36. Escribir un programa que visualice en pantalla los números pares entre 1 y 25.

37. Escribir un programa que visualice en pantalla los números múltiplos de 5 comprendidos entre 1 y 100.

```
#include <iostream.h>
using namespace std;

int main()
{
    int x = 1;
    while(x<100)
    {
        if(x%5==0)
            cout << x << endl;
        x++;
    }
    return 0;
}
```

38. Escribir un programa que sume los números comprendidos entre 1 y 10. 55



39. Escribir un programa que genere la tabla de multiplicar de un número introducido por el teclado.
40. Escribir un programa que realice la pregunta ¿Desea continuar S/N? y que no deje de hacerla hasta que el usuario teclee N.
41. Escribir un programa que calcule cuantos años tarda en duplicarse un capital depositado al 5% de interés anual.
42. Escribir un programa que calcule la suma de los números hasta un número dado (introducido por el usuario).
43. Escribir un programa que pida un número y si el que se introduce por el teclado es menor de 100 que vuelva a solicitarlo.
44. Escribir un programa que calcule el factorial de un número.
45. Escribir un programa que calcule la media de 5 números introducidos por el teclado.

```
#include <iostream>
using namespace std;

int main()
{
    int acum=0;
    int n;
    for (int x=0; x<5; x++)
    {
        cout << "Ingrese el numero " << x+1 << " de 5 ";
        cin >> n;
        acum+=n;
    }
    cout << "La media de los 5 numeros es " << acum/5 << endl;
    return 0;
}
```

46. Escribir un programa que calcule la media de x cantidad números introducidos por el teclado.
47. Escribir un programa que calcule la media de números introducidos por el teclado hasta que el número ingresado sea cero.



48. Escribir un programa que detecte si un número es primo o no. Un número es primo si sólo es divisible por sí mismo y por la unidad.
49. Escribir un programa que escriba los números comprendidos entre 1 y 1000. El programa escribirá en la pantalla los números en grupos de 20, solicitando al usuario si quiere o no continuar visualizando el siguiente grupo de números.

```
#include <iostream>
using namespace std;

int main(){
    char r;
    int x=1;
    int tope = 1000;
    int paso = 10;
    do{
        cout << x << endl;
        if(x+paso-1 >= x+tope) {
            cout << endl << "desea continuar (s/n)" ;
            cin >> r;
        }
        if(r=='n') break;
        x+=paso;
    }while(x<tope);
    cout << endl;
    return 0;
}
```

50. Escribir un programa que calcule, independientemente, la suma y la media de los números pares e impares comprendidos entre 1 y 200.
51. Escribir un programa que calcule la suma de los cuadrados de los 100 primeros números enteros.
52. Escribir un programa que lea 10 datos desde el teclado y sume sólo aquellos que sean negativos.

Arreglos

53. Escribir un programa que llene un arreglo con los números enteros comprendidos entre 4 y 14.



54. Escribir un programa que llene un arreglo con los números pares comprendidos entre 1 y 100.

55. Escribir un programa que llene un arreglo con los números comprendidos entre 0 y 100 divisibles por 3 .

```
#include <iostream>
using namespace std;
int main()
{
    int x=0,y=0;
    cout<<"Input x,y : ";
    do
    {
        cin>>x>>y;
        if((x>=0)&&(x<=100)&&(y>=0)&&(y<=100))
            cout<<"Valid Input" << endl;
        else
            cout<<"Invalid Input" << endl;
    }while((x>=0)&&(y>=0));
    cout<<"Program End" << endl;
    return 0;
}
```

56. Escribir un programa que llene un arreglo con cinco números enteros consecutivos y haga una copia de ese arreglo en otro.

57. Escribir un programa que llene un arreglo de 10 números enteros aleatorios comprendidos entre 50 y 100, copie en otro arreglo esos números multiplicados por 0,5 y muestre ambos arreglos.

```
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    int n(0);
    float c[10];
    srand(time(0));
    for(int i=0;i<10;i++)
        c[i]=50+(rand()%50);
    for(int i=0,j=10;i<j;i++)
        c[i]=c[j];
    for(int i=0;i<10;i++)
        cout << n++ << " : " << c[i] << endl;
    return 0;
}
```



58. Escribir un programa que llene un arreglo con los veinte primeros números pares y calcule su suma.
59. Escribir un programa que solicite cinco números, los almacene en un arreglo y luego calcule la media aritmética de esos números.
60. Escribir un programa que tras asignar los números, 23, 45, 68, 99, 10, 15 y 4 a un arreglo, determine la posición del arreglo en la que se encuentra el máximo valor.
61. Escribir un programa que tras asignar los números, -2, 5, 8, -9, 10, 15 y -4 a un arreglo calcule, independientemente, la suma de los elementos positivos y negativos.
62. Escribir un programa que tras asignar 10 números leídos desde el teclado a un arreglo, determine las posiciones del arreglo en las que se encuentran el máximo y el mínimo valor.
63. Escribir un programa que imprima la media de los elementos que se encuentran en las posiciones pares y la media de los elementos que se encuentran en las posiciones impares de un arreglo de enteros de 20 posiciones.
64. Escribir un programa que determine la posición de la siguiente matriz en la que se encuentra el valor máximo.

32	56	50
49	99	12
78	57	89

65. Escribir un programa que sume, independientemente, los elementos positivos y negativos de la siguiente matriz:

-2	56	50
44	-12	-42
70	57	-86

66. Escribir un programa que multiplique por dos los elementos de la siguiente matriz:



5	6	13
14	2	4
21	7	6

```
#include <iostream>
using namespace std;

int main()
{
    int ols[3] = {5, 6, 13}, cl[3] = {14, 2, 4}, l[3] = {21, 7, 6};
    cout << " matriz original " << endl;
    for(int i=0; i<3; i++)
        for(int j=0; j<3; j++)
            cout << ols[i][j] << " ";
    cout << endl;
    cout << " matriz multiplicada por 2 " << endl;
    for(int i=0; i<3; i++)
        for(int j=0; j<3; j++)
            cl[i][j] = ols[i][j]*2;
        cout << cl[i][j] << endl;
    cout << endl;
    return 0;
}
```

67. Escribir un programa que llene la primera fila de una matriz de 3 filas por 10 columnas con números aleatorios entre 1 y 20, la segunda fila con los cuadrados de los datos de la primera fila; y la tercera fila con la suma de la primera y la segunda.
68. Escribir un programa que sume los datos de cada una de las filas de la siguiente matriz; el resultado se almacenará en la última posición de cada fila:

5	6	7	9	
11	8	2	15	

69. Escribir un programa que sume los datos de cada una de las columnas de la siguiente matriz; el resultado se almacenará en la última posición de cada columna:



5	6	13
14	2	4
21	7	6
12	9	5

70. Escribir un programa que sume los elementos de cada una de las filas y de las columnas de la siguiente matriz; el resultado de cada suma se almacenará en la última posición de la fila o columna correspondiente. Además la suma total de todos los elementos de la matriz se almacenará en el elemento de la esquina inferior derecha de la matriz:

2	9	11	
1	12	4	
21	17	8	
2	39	5	

71. Escribir un programa que divida todos los elementos de una matriz M (3,4) por el elemento situado en la posición 2,2 .

- 72) Escribir un programa que almacene en un arreglo los números primos comprendidos entre 1 y 100 .

- 73. Escribir un programa que genera la matriz transpuesta de una matriz de 3 filas y 4 columnas. La matriz transpuesta de una matriz M(m,n) se obtiene intercambiando filas por columnas y viceversa; el resultado se tiene que almacenar en una nueva matriz M_TRANS(n,m).
- 74. Escribir un programa que genera la inversa de una cadena de caracteres. La cadena original y la invertida deben almacenarse en arreglos independientes.
- 75. Escribir un programa que sume dos matrices bidimensionales. Las matrices para que puedan



sumarse deben tener las mismas dimensiones.

76. Escribir un programa que elimine los blancos de una cadena de caracteres. La cadena original y la transformada deben almacenarse en arreglos independientes.
77. Escribir un programa que cuente las mayúsculas de una cadena de caracteres.

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

using namespace std;

int main()
{
    char cad[30];
    int cont=0;
    for(int i=0; i<30; i++)
        cad[i]=0;
    cout << "Ingrese una cadena de texto: ";
    cin.getline(cad, 30);
    for(int i=0; i<30; i++)
        if(isupper(cad[i])) cont++;
    cout << "La cadena: " << cad << " tiene " << cont << " mayúsculas." << endl;
    return 0;
}
```

78. Escribir un programa que cambie las mayúsculas de una cadena de caracteres a minúsculas y viceversa.
- 79. Escribir un programa que encripte una cadena de caracteres sumando 2 al código ASCII de cada uno de sus caracteres.
80. Escribir un programa que encripte los caracteres de una cadena sumando 2 a los que situados en posiciones pares y 3 a los situados en posiciones impares.

```
#include <iostream>
using namespace std;

int main()
{
    char cad[30];
    char cop[30];
    for(int i=0; i<30; i++)
        cad[i]=0;
        cop[i]=0;
```



```
cout << "Ingrese una cadena de texto:";
cin.getline(cad, 30);
for (int i=0; i<30; i++) {
    if (cad[i] != '0') {
        cout << " ";
        cont++;
    }
    else {
        cop[i] = cad[i];
    }
}
cout << "La cadena : " << cad << endl;
cout << "Fue encriptada en : " << cop << endl;
return 0;
```

81. Decir el numero de vocales que aparecen en una frase introducida por el usuario.
82. Pedir que se introduzca una frase, acabada con un punto, en la que se lean las letras del abecedario introducidas y se muestre posteriormente las veces que aparece cada letra y cuales letras no aparecieron en la frase (no se debe usar cin.getline()).
83. Escribir un programa que lea una frase introducida desde el teclado y la escriba al revés.
84. Escribir un programa que compare dos arreglos de caracteres y nos diga si son idénticos o no.
85. Escribir un programa que sume los valores que hay por encima de la diagonal principal de una matriz de 4 por 4. Los valores se deben asignar aleatoriamente.
86. Decir si una frase es o no un palíndromo, es decir, si se lee igual de derecha a izquierda que de izquierda a derecha (deben ser obviados los espacios).

```
#include<iostream>
using namespace std;

int main(){
    char cad[30];
    char cop[30];
    int cont=1;
    bool pal = true;
    for(int i=0; i<30; i++) {
        cad[i] = '\0';
        cop[i] = '\0';
    }
    cout << "Ingrese una cadena de texto:";
    cin.getline(cad, 30);
```



```
for (int i=0; i<30; i++) {
    cout << "AD" << cad[i];
    cout << endl;
}

for (int i=0; i<cont; i++) {
    if (cad[i] != cop[cont-i]) {
        pal = false;
    }
}

if (pal) {
    cout << "La cadena " << cad << " es un palíndromo." << endl;
} else {
    cout << "La cadena " << cad << " no es un palíndromo." << endl;
}

return 0;
}
```

Funciones y procedimientos:

87. Escribir un programa que incremente un número usando una función.

```
#include <iostream>

using namespace std;

int incremento(int n);

int main () {
    int num;
    cout << "Ingrese un numero: ";
    cin >> num;
    num = incremento(num);
    cout << num << endl;
    return 0;
}

int incremento(int n) {
    num++;
}
```

88. Escribir un programa que calcule la potencia usando una función propia (no debe usar la función pow).

89. Escribir un programa que calcule el factorial de un numero leído desde el teclado usando funciones.

90. Escribir un programa, que con funciones, verifique si un carácter introducido es un número o no.

```

for(int i=0; i<50; i++) {
    if(cad[i] != '\0' && cad[i] != ' ')
        cont++;
    cop[cont] = cad[i];
}

for(int i=0; i<cont; i++) {
    if(cop[i] == cop[cont-i-1])
        pal = true;
    else
        pal = false;
}

if(pal)
    cout << "La cadena " << cad << " es un palíndromo." << endl;
else
    cout << "La cadena " << cad << " no es un palíndromo." << endl;

return 0;
}

```

Funciones y procedimientos:

87. Escribir un programa que incremente un número usando una función.

```
#include<iostream>
```



```
#include <iostream.h>
using namespace std;
bool verificar_numero(char c);
int main()
{
    char car;
    cout << "Ingrese un carácter" << endl;
    cin >> car;
    if(verificar_numero(car))
        cout << car << " es un número" << endl;
    else
        cout << car << " no es un número" << endl;
    return 0;
}
bool verificar_numero(char c)
{
    bool resp;
    switch(c)
    {
        case '0':
        case '1':
        case '2':
        case '3':
        case '4':
        case '5':
        case '6':
        case '7':
        case '8':
        case '9':
            resp = true;
            break;
        default:
            resp = false;
            break;
    }
    return resp;
}
```

91. Escribir un programa que, mediante una función, calcule el resultado de restar el doble de un numero a su cuadrado.
92. Escribir un programa que, utilizando funciones con parámetros, lea desde el teclado las unidades y el precio que quiere comprar, y en función de las unidades introducidas le haga un descuento o no (cuando las unidades excedan media docena se aplicará 4% y el 10% cuando excedan la docena).
93. Escribir un programa que mediante funciones obtenga la distancia entre dos puntos que se encuentran en el plano a partir de la lectura de sus coordenadas.

```
#include <iostream>
using namespace std;
bool verificar_numero(char c);
int main()
{
    char car;
    cout << "Ingrese un caracter: ";
    cin >> car;
    if(verificar_numero(car))
        cout << car << " es un numero" << endl;
    else
        cout << car << " no es un numero" << endl;
    return 0;
}

bool verificar_numero(char c)
{
    bool resp;
    switch(c){
        case '0':
        case '1':
        case '2':
        case '3':
        case '4':
        case '5':
        case '6':
        case '7':
        case '8':
        case '9':
            resp = true;
            break;
        default:
            resp = false;
            break;
    }
    return resp;
}
```



```
#include <iostream.h>
#include <math.h>
using namespace std;

void leer_cunto(int &x, int &y);
float distancia(int x1, int y1, int x2, int y2);

int main()
{
    int x1, y1, x2, y2;
    float dist;
    cout << "coordenadas primer punto" << endl;
    leer_punto(x1, y1);
    cout << "coordenadas segundo punto" << endl;
    leer_punto(x2, y2);
    dist = distancia(x1, y1, x2, y2);

    cout << "la distancia entre los puntos (" << x1 << ", " << y1 << ") -> (" << x2 << ", " << y2 << ")" << " es " << dist << endl;
    return 0;
}

void leer_punto(int &x, int &y)
{
    cout << "ingrese el valor de x" << endl;
    cin >> x;
    cout << "ingrese el valor de y" << endl;
    cin >> y;
}

float distancia(int x1, int y1, int x2, int y2)
{
    float d;
    d = sqrt(pow(x2 - x1, 2) + pow(y2 - y1, 2));
    return d;
}
```

94. Escribir un programa que reciba un numero del 1 al 12 desde el teclado y muestre el número de días correspondiente al mes que corresponda con ese día (usar funciones).

95) Escribir un programa que transforme números entre 0 y 999 a números romanos (usar funciones).

96. Escriba un programa que calcule el máximo y el mínimo de dos números usando funciones.

Con tal fin, proceda como sigue:

- Escriba una función que lea un número.
- Escriba una función que dados dos números, devuelva el máximo de ellos.
- Escriba una función que dados dos números, devuelva el mínimo de ellos.
- Escriba el programa principal que lea dos números, obtenga el mayor y el menor de ellos, y muestre el resultado en pantalla.



Estructuras:

97. Escribir un programa que almacene en un arreglo de estructuras los nombres de los alumnos, sus notas parciales y finales. Hallar la nota media y mostrar un mensaje de APTO si el alumno supera o iguala la calificación de 3.5 o NO APTO si no lo alcanza. Hacerlo para un número de 5 alumnos.

```
#include <iostream>
using namespace std;
const int TAM = 5;
struct Alumno {
    char nombre[30];
    float parcial[3];
    float final;
};
Alumno alumnos[TAM];
void leer_nombre(Alumno *a);
void leer_parciales(Alumno *a);
void calcular_final(Alumno *a);
int main() {
    system("clear");
    for(int i=0; i<TAM; i++) {
        leer_nombre(&alumnos[i]);
        for(int j=0; j<3; j++)
            cout << "para el alumno " << alumnos[i].nombre << endl;
        leer_parciales(&alumnos[i]);
        calcular_final(&alumnos[i]);
    }
    for(int i=0; i<TAM; i++) {
        if(alumnos[i].final>=3.5)
            cout << alumnos[i].nombre << " apto." << endl;
        else
            cout << alumnos[i].nombre << " no apto." << endl;
    }
    return 0;
}
void leer_nombre(Alumno *a) {
    cout << "ingrese el nombre del alumno ";
    cin.getline((a)->nombre, 30);
}
void leer_parciales(Alumno *a) {
    for(int i=0; i<3; i++) {
        cout << "ingrese la nota del parcial " << i+1 << endl;
        cin >> (a)->parcial[i];
    }
}
void calcular_final(Alumno *a) {
    (a)->final=0;
    for(int i=0; i<3; i++)
        (a)->final+= (a)->parcial[i];
    (a)->final /= 3;
}
```



98. Escribir un programa que almacene en un arreglo de estructuras las características de cada persona: nombre, sexo, edad, peso, color de pelo, color de piel, color de ojos, nacionalidad y teléfono y permita generar consultas por estas características.
99. Escriba un programa que usando estructuras implemente una lista de alumnos con sus códigos, nombres, apellidos, direcciones, teléfonos, correos y permita buscar en dicha lista.

Archivos:

- 100. Escribir un programa que escriba "hola mundo" dentro de un archivo llamado hola.txt.
- 101. Escribir un programa que lea el archivo creado en el ejercicio anterior y muestre por pantalla su contenido.
102. Escribir un programa que escriba la lista de caracteres ASCII dentro de un archivo de texto.

```
#include <iostream>
#include <iostream>

using namespace std;

int main()
{
    system("clear");
    int i = 127;

    ifstream archivo("ascii.txt", ios::out);
    if (!archivo)
        cout << "no se puede crear el archivo ascii.txt" << endl;
    else
        while (i > 0)
            archivo << i << " " << char(i) << "\t\t";
    i--;
    if (i > 0) archivo << endl;
    archivo.close();
}

return 0;
```

- 103. Escribir un programa que lea y muestre en pantalla el archivo generado en el ejercicio anterior.

salida sin formato:



```
#include <iostream>
#include <fstream>
using namespace std;

int main()
{
    system("clear");
    char c;

    ifstream archivo("ascii.txt",ios::in);
    if(archivo)
        cout << "no se puede abrir el archivo ascii.txt" << endl;
    else
    {
        archivo >> c;
        while(archivo.eof())
            cout << c;
        archivo >> c;
        archivo.close();
    }
    return 0;
}
```

salida con formato:

```
#include <iostream>
#include <fstream>
using namespace std;

int main()
{
    system("clear");
    char c;

    ifstream archivo("ascii.txt",ios::in);
    if(archivo)
        cout << "no se puede abrir el archivo ascii.txt" << endl;
    else
    {
        archivo.get(c);
        while(archivo.eof())
            cout << c;
        archivo.get(c);
        archivo.close();
    }
    return 0;
}
```

- 104. Escribir un programa que capture una lista de nombres y apellidos de los alumnos de un curso y genere un archivo de texto con estos datos.

Si



Programas varios:

105. Escribir un programa que lea un tiempo en horas, minutos y segundos y empiece a cronometrar el tiempo mostrándolo en pantalla hasta llegar al límite leído al inicio.

106. Modifique el ejercicio anterior para que el conteo se haga hacia atrás desde el tiempo leído hasta cero.

✓ → 107. Escribir un programa que ejemplifique los diferentes métodos de ordenamiento de arreglos, para esto se deben crear como mínimo las siguientes funciones:

- Función que llena el arreglo con números enteros aleatorios.
- Función que crea una copia del arreglo original.
- Función que muestra en pantalla un arreglo.
 - Función de ordenar por burbuja.
 - Función de ordenar por selección.
 - Función de ordenar por inserción.
 - Función de ordenar por quick sort.
- ✗• Función que muestra en pantalla un menú de opciones para el usuario.

nota: Recuerde que el programa no solo debe ordenar los arreglos si no mostrar a usuario como es el proceso de ordenamiento según el caso.

✓ → 108. Escribir un programa que implemente un calendario perpetuo, es decir, que dado una fecha en formato mes, día y año le retorne al usuario el día de la semana que corresponde para esta fecha.

✓ → 109. Escribir un programa que implemente una calculadora de binarios con las operaciones aritméticas básicas (suma, resta, multiplicación y división) y la implementación del complemento a uno.

✓ → 110. Escribir un programa que le permita al usuario jugar contra el computador el juego de picas y fijas, se debe implementar el juego con 3, 5 o 5 dígitos.