

```

1) #include <iostream>

using namespace std;
#define M 10
int main()
{
    int tab[M];
    int minimo,maximo;

    for(int i=0; i<M; i++){
        cout<<"Introduce un valor para el elemento " <<i+1<< ": ";
        cin >> tab[i];
    }

    maximo=minimo=tab[0];

    for(int i= 1; i<M; i++){
        if(tab[i]<minimo)
            minimo = tab[i];
        else if (tab[i] > maximo)
            maximo = tab[i];
    }

    cout << "El valor maximo almacenado es: "<<maximo << endl;
    cout << "El valor minimo almacenado es: "<<minimo << endl;
    return 0;
}

```

2)

```
#include <iostream>

using namespace std;
#define M 10
int main()
{
    int S;
    int tab[10];
    int maximo;
    for(int i=0; i<M; i++)
    {
        cout<<"Introduce un valor para el elemento " <<i+1<<" ": ";
        cin >> tab[i];
    }

    cout << "\nIntroduzca el numero a buscar: ";
    cin >> S;

    //esquema de busqueda

    int i= 0;
    bool encontrado = false;

    while ( i<M && !encontrado)
        if (S == tab[i]) encontrado = true;
        else i++;

    if (encontrado)
        cout <<"El valor esta en el vector";
    else
        cout << "El valor NO está en el vector";

    return 0;
}
```

3)

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

#define NFIL 10
#define NCOL 15

int main()
{
    int tabla[NFIL][NCOL];
    int X,f,c;

    srand(time(0)); // inicializar semilla generador de numeros aleatorios

    //esquema recorrido de tabla
    for(f=0; f<NFIL; f++)
        for(c=0; c<NCOL; c++)
            tabla[f][c] = rand() % 101;

    cout << "Introduce el numero a buscar: ";
    cin >> X;

    //Esquema de busqueda de tabla[f][c] == X
    bool encontrado = false;

    f=0;
    while(f<NCOL && !encontrado)
    {
        c=0;
        while(f < NFIL && !encontrado)
            if(tabla[f][c]== X)
                encontrado = true;
            else
                c++;

        if(!encontrado)
            f++;
    }
    if (!encontrado)
        cout<<" No ";
    cout<<"esta en la tabla."<<endl;
    return 0;
}
```

```

3)
#include <iostream>
#include <cstring>

using namespace std;
#define NFIL 3
#define NCOL 4

typedef char cadena[30];

int main ()
{
    cadena tabla[NFIL][NCOL], S;
    int f, c;

    //Esquema de recorrido de tabla
    for(f = 0; f < NFIL; f++)
        for(c = 0; c < NCOL; c++)
        {
            cout << "Introduce la palabra [" << f+1 << ", " << c+1 << "]:
";
            cin >> tabla[f][c];
        }

    cout << "Introduzca la palabra a buscar: ";
    cin >> S;

    //Esquema de búsqueda de tabla[f][c] == S NO ES VÁLIDO, hay que
    utilizar la función strcmp
    bool encontrado = false;
    f = 0;
    while (f < NFIL && !encontrado)
    {
        c=0;
        while (c < NCOL && !encontrado)
            if (strcmp(tabla[f][c], S) == 0)
                encontrado = true;
            else
                c++;
        if (!encontrado)
            f++;
    }

    cout << "La palabra " << S;
    if (!encontrado)
        cout << " no ";
    cout << " está en la tabla." << endl;
    return 0;
}

```

```

4)
#include <iostream>
#include <cstring>

using namespace std;
#define NFIL 3
#define NCOL 4

typedef char cadena[30];

int main ()
{
    cadena tabla[NFIL][NCOL], S;
    int f, c;

    //Esquema de recorrido de tabla
    for(f = 0; f < NFIL; f++)
        for(c = 0; c < NCOL; c++)
        {
            cout << "Introduce la palabra [" << f+1 << ", " << c+1 << "]:
";
            cin >> tabla[f][c];
        }

    cout << "Introduzca la palabra a buscar: ";
    cin >> S;

    //Esquema de búsqueda de tabla[f][c] == S NO ES VÁLIDO, hay que
    utilizar la función strcmp
    bool encontrado = false;
    f = 0;
    while (f < NFIL && !encontrado)
    {
        c=0;
        while (c < NCOL && !encontrado)
            if (strcmp(tabla[f][c], S) == 0)
                encontrado = true;
            else
                c++;
        if (!encontrado)
            f++;
    }

    cout << "La palabra " << S;
    if (!encontrado)
        cout << " no ";
    cout << " está en la tabla." << endl;
    return 0;
}

```

```

5)
#include <iostream>
#include <cstring>
#define NFIL 2
#define NCOL 4
using namespace std;

typedef char cadena[30];
struct persona
{
    int dni;
    cadena nombre;
};
int main()
{
    persona tabla [NFIL][NCOL];
    int dni_buscado;
    bool encontrado = false;

    cout<<"Rellena la tabla con los datos de las personas: "<<endl;
    for ( int f=0; f< NFIL; f++)
        for (int c=0; c<NCOL; c++)
        {

            cout << "Introduce el NOMBRE de la persona de la fila "<<f+1
<< " y columna " <<c+1<< " : ";
            cin >> tabla [f][c].nombre;
            cout << "Introduce el DNI de la persona de la fila "<<f+1 <<
" y columna " <<c+1<< " : ";
            cin >> tabla[f][c].dni;
        }

    //POR SI QUIERES MOSTRAR LA TAB:
    /*for(int f=0; f<NFIL; f++)
        for(c=0; c<NCOL; c++){
            cout<<"Nombre: "<<tabla[f][c].nombre<< " | DNI:
"<<tabla[f][c].dni<<"\t\t";

            if (NCOL==3)
                cout<<endl;
        }*/

    cout << "Introduzca el dni a buscar ";
    cin >> dni_buscado;

    int f = 0;
    cadena nombre2;

```

```

while (f<NFIL && encontrado == false)
{
    int c=0;
    while (c<NCOL && encontrado == false){
        if (dni_buscado==tabla[f][c].dni)
        {
            encontrado = true;
            strcpy(nombre2, tabla[f][c].nombre);
        }
        else
            c++;
    }
    f++;
}
if (encontrado)
    cout << "La persona con DNI "<< dni_buscado<< ", se llama "<<
nombre2<<endl;
else
    cout << "No hay nadie con el dni indicado"<<endl;
return 0;
}

```

6)

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

typedef char cadena[150];

int main()
{
    cadena frase, frase2;
    cout << "Introduce una frase pelotuo: ";
    cin.getline(frase, 150);

    int len = strlen(frase);

    int i = 0;
    for(int j=0; j<=len; j++)
        if (frase[j] != ' ')
        {
            frase2[i] = frase[j];
            i++;
        }
        else if (frase[j]==' ')
        {
            frase[j]='*';
            frase2[i] = frase[j];
            i++;
        }

    strcpy(frase, frase2);

    cout<< "La frase transformada es "<<frase<<endl;

    return 0;
}
```


7)

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

typedef char cadena[150];

int main()
{
    cadena frase, frase_tmp;

    cout << "Introduce una frase: ";
    cin.getline(frase, 150);

    int len= strlen(frase);

    int i = 0;
    for(int j=0; j<=len; j++)
        if (frase[j] != ' ')
        {
            frase_tmp[i] = frase[j];
            i++;
        }

    cout<< "La frase compactada es "<<frase_tmp<<endl;
    return 0;
}
```

8)

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

typedef char cadena[150];

int main ()
{
    cadena frase, frase_tmp;

    cout << "Introduce una frase por teclado: ";
    cin.getline(frase, 150);

    int len = strlen(frase);

    int i = 0;
    for (int j=len-1; j>=0; j--)
    {
        frase_tmp[i] = frase[j];
        i++;
    }
    frase_tmp[i] = '\0';
    strcpy(frase, frase_tmp);

    /* SOLUCIÓN MÁS RÁPIDA:

        int i = 0, j = len-1;
        char tmpc;

        while (i < j)
        {
            tmpc = frase[i];
            frase[i] = frase[j];
            frase[j] = tmpc;
            i++;
            j--;
        }
    */
    cout << "La frase invertida es: \n" << frase << endl;
    return 0;
}
```

9)

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

typedef char cadena[150];

int main ()
{
    cadena frase, frase_tmp;
    bool palindromo=false;
    cout << "Introduce una frase por teclado: ";
    cin.getline(frase, 150);

    int len = strlen(frase);

    int i = 0;
    for (int j=len-1; j>=0; j--)
    {
        frase_tmp[i] = frase[j];
        i++;
    }
    frase_tmp[i] = '\0';

    if(strcmp(frase,frase_tmp)==0)
        palindromo=true;

    if(palindromo)
        cout << "\nLa frase es un palindromo";
    if(!palindromo)
        cout<<"\nLa no es un palindromo";

    return 0;
}
```

10)

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

typedef char cadena[150];
int main()
{
    cadena nombre,apellido1,apellido2,nom_completo;

    cout << "Introduce el nombre de la persona: ";
    cin.getline(nombre, 150);
    cout<< "Introduce el primer apellido: ";
    cin.getline(apellido1, 150);
    cout<< "Introduce el segundo apellido: ";
    cin.getline(apellido2, 150);

    strcpy (nom_completo, nombre);
    strcat (nom_completo, " ");
    strcat (nom_completo, apellido1);
    strcat (nom_completo, " ");
    strcat (nom_completo, apellido2);

    cout<< "El nombre completo de la persona es " <<nom_completo<<endl;

    return 0;
}
```

```
11)
#include <iostream>
#include <cstdlib>
#include <ctime>
#include <stdlib.h>

#define M 30//MAX CARACTERES PERMITIDOS
#define L 10//N D NOM O APELLIDOS
using namespace std;
typedef char Vnombres[M],Vapellidos[M];
int main()
{
    int opcion, i;
    Vnombres nombre[L];
    Vapellidos apellido[L];
    do
    {
        cout << "          MENU " << endl
            << "    1. Pedir nombres" <<endl
            << "    2. Pedir apellidos" <<endl
            << "    3. Mostrar Vectores" <<endl
            << "    4. Generar nombres de personas"<<endl
            << "    5. Salir"<<endl
            << " Elija una opcion: ";
        cin>>opcion;
        switch(opcion)
        {
            case 1:
                for(i=0; i<10; i++)
                {
                    cout<< "Introduce el nombre " << i+1<<": ";
                    cin>> nombre[i];
                }
                break;
            case 2:
                for (i=0; i<10; i++)
                {
                    cout<<"Introduce el apellido "<<i+1<<": ";
                    cin>> apellido[i];
                }
                break;
            case 3:
                cout << "Vector Nombres: ";
                for(i=0; i<10; i++)
                {
                    cout<< nombre[i];
                    if(i<9)
                        cout<< ", ";
                }
            case 4:
                for(i=0; i<10; i++)
                {
                    cout<< "Nombre " << i+1 << ": " << nombre[i] << " ";
                    if(i%5==4)
                        cout<< "\n";
                }
                cout<< "\n";
                break;
            case 5:
                cout<< "Fin del programa\n";
                break;
        }
    } while (opcion!=5);
}
```

```

        cout<<endl<<"Vector Apellidos: ";
        for (i=0; i<10; i++)
        {
            cout<< apellido[i];
            if(i<9)
                cout<<" ";
        }
        cout<<endl;
        break;
    case 4:
        int num,randnum,randnum2,randnum3;
        srand(time(0));

        cout<<"Cuantos nombres desea generar: ";
        cin >>num;

        for(int k=1; k<= num; k++){
            randnum= rand()%L;
            randnum2= rand()%L;
            randnum3= rand()%L;

            cout<<endl<<"Nombre aleatorio "<<k<<":
"<<nombre[randnum]<<" " << apellido[randnum2]<<"
"<<apellido[randnum3]<<endl;

        }

        break;
    case 5: cout<<endl;
        break;
    default:
        cout<< "Opcion incorrecta..." <<endl;
    }
}
while (opcion != 5);

return 0;
}

```

12)

```
#include <iostream>
```

```
#include <cstring>
```

```
using namespace std;
```

```
typedef char cadena[150];
```

```
int main()
```

```
{
```

```
    cadena nombre,nombre1,nombre2,apellido1,apellido2,nom_completo;
```

```
    bool nombres=false;
```

```
    cout<< "Introduce el nombre de la persona: ";
```

```
    cin.getline(nombre, 150);
```

```
    cout<< "Introduce el primer apellido: ";
```

```
    cin.getline(apellido1, 150);
```

```
    cout<< "Introduce el segundo apellido: ";
```

```
    cin.getline(apellido2, 150);
```

```
    int len= strlen(nombre);
```

```
    int i = 0;
```

```
    for(int j=0; j<=len; j++)
```

```
    {
```

```
        if(j==0)
```

```
        {
```

```
            nombre1[i]=nombre[j];
```

```
        }
```

```
        if(nombre[j]==' ')
```

```
        {
```

```
            j=j+1;
```

```
            nombre2[i]=nombre[j];
```

```
            i++;
```

```
            nombres=true;
```

```
        }
```

```
    }
```

13)

```
#include <iostream>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    int Vector1[10], Vector2[10], opcion, i;
```

```
    bool iguales;
```

```
    do
```

```
    {
```

```
        cout << "          MENU " << endl
```

```
        << "    1. Pedir Vector 1" <<endl
```

```
        << "    2. Pedir Vector 2" <<endl
```

```
        << "    3. Mostrar Vectores" <<endl
```

```
        << "    4. Comprobar Vectores"<<endl
```

```
        << "    5. Salir"<<endl
```

```
        << " Elija una opcion: ";
```

```
    cin>>opcion;
```

```
    switch(opcion)
```

```
    {
```

```
    case 1:
```

```
        for(i=0; i<10; i++)
```

```
        {
```

```
            cout<< "Introduce el valor " << i+1<<": ";
```

```
            cin>> Vector1[i];
```

```
        }
```

```
        break;
```

```
    case 2:
```

```
        for (i=0; i<10; i++)
```

```
        {
```

```
            cout<<"Introduce el valor "<<i+1<<": ";
```

```
            cin>> Vector2[i];
```

```
        }
```

```
        break;
```

```
    case 3:
```

```
        cout << "Vector 1: ";
```

```
        for(i=0; i<10; i++)
```

```
        {
```

```
            cout<< Vector1[i];
```

```
            if(i<9)
```

```
                cout<<", ";
```

```
        }
```

```
        cout<<endl<<"Vector 2: ";
```

```
        for (i=0; i<10; i++)
```

```
        {
```

```
            cout<< Vector2[i];
```

```
            if(i<9)
```

```
                cout<<", ";
```



```

    }
    cout<<endl;
    break;
case 4:
    i=0;
    iguales=true;
    while(i<10 && iguales)
        if(Vector1[i] == Vector2[i])
            i++;
        else
            iguales = false;
    cout<< "Los vectores ";
    if(!iguales)
        cout<< "no ";
    cout <<"son iguales."<<endl;
    break;
case 5: cout<<endl;
break;
default:
    cout<< "Opcion incorrecta..." <<endl;
}
}
while (opcion != 5);
return 0;
}

```

14)

```
#include <iostream>
#define M 15
using namespace std;

int main()
{
    int p,pos,Vector[M],minimo;
    for(int i=0; i<M; i++)
    {
        cout<< "Introduce el valor " << i+1<<": ";
        cin>> Vector[i];
    }

    cout<<"\nIntroduce una posicion del vector: ";
    cin>>p;

    minimo=Vector[p];

    for( int i=p;i<M; i++){
        if(minimo>Vector[i]){
            minimo = Vector[i];
            pos=i+1;
        }
    }

    cout << "\nEl valor minimo almacenado es: "<<minimo<<" y su posicion
es el "<<pos<<endl;
    return 0;
}
```

15)

```
#include <iostream>
#define L 15
using namespace std;

int main()
{
    double vec[15];
    int valor, n,m;

    for(n=0; n<L; n++){
        cout << "Introduce un valor para el elemento "<<n+1<<": ";
        cin>>vec[n];
    }

    //Algoritmo para ordenar el vector
    for(n=1; n<L; n++)
    {
        m=0;
        while(m<n)
        {
            if(vec[n]<vec[m])
            {
                valor=vec[n];
                vec[n]=vec[m];
                vec[m]=valor;
            }
            m++;
        }
    }

    cout<<endl<<"Vector ordenado: ";
    for(n=0; n<L; n++)
    {
        cout<<vec[n];
        if(n!= L-1)
            cout<<" < ";
    }
    return 0;
}
```

```

16)
#include <iostream>
#include <cstdlib>
#include <ctime>
using namespace std;

#define M 1000

int main()
{
    char Vector[M];
    float
N,num,a,e,o,x,u,porcentaje_a,porcentaje_e,porcentaje_i,porcentaje_o,porcentaje_u;
    do
    {
        cout << "Introduce el número de elementos del vector a ser
rellenados: ";
        cin >> N;

        if(N<0 || N>1000)
            cout<<"Error,vuelva a intentarlo"<<endl;
    }
    while(N<0 || N>1000);

    srand(time(0));

    for(int i=0; i<N; i++)
    {
        num = rand() % 21;

        if(num<5)
        {
            Vector[i]='a';
            a=a+1;
        }
        else if(num>4 && num<8)
        {
            Vector[i]='e';
            e=e+1;
        }
        else if(num>7 && num<13)
        {
            Vector[i]='i';
            x=x+1;
        }
        else if(num>12 && num<17)
        {

```

```

        Vector[i]='o';
        o=o+1;
    }
    else
    {
        Vector[i]='u';
        u=u+1;
    }
}
porcentaje_a= (a*100)/N;
porcentaje_e= (e*100)/N;
porcentaje_i= (x*100)/N;
porcentaje_o= (o*100)/N;
porcentaje_u= (u*100)/N;

    cout<<"La vocal 'a' ha aparecido "<<a<<"
veces("<<porcentaje_a<<"%)"<<endl;
    cout<<"La vocal 'e' ha aparecido "<<e<<"
veces("<<porcentaje_e<<"%)"<<endl;
    cout<<"La vocal 'i' ha aparecido "<<x<<"
veces("<<porcentaje_i<<"%)"<<endl;
    cout<<"La vocal 'o' ha aparecido "<<o<<"
veces("<<porcentaje_o<<"%)"<<endl;
    cout<<"La vocal 'u' ha aparecido "<<u<<"
veces("<<porcentaje_u<<"%)"<<endl;
    return 0;
}

```

17)

```
#include <iostream>

using namespace std;
#define NFIL 10
#define NCOL 10
int main()
{
    int tab[NFIL][NCOL],opcion,f,c,impar=0,par=0,sum=0,f1,f2,c1,c2,alt;

    do
    {
        cout << "          MENU " << endl
             << "  1. Pedir datos" <<endl
             << "  2. Mostrar" <<endl
             << "  3. Analizar" <<endl
             << "  4. Sumar una fila"<<endl
             << "  5. Sumar una columna"<<endl
             << "  6. Sumar diagonal principal" <<endl
             << "  7. Sumar diagonal secundaria" <<endl
             << "  8. Intercambiar filas" <<endl
             << "  9. Intercambiar columnas"<<endl
             << " 10.Salir"<<endl<<endl
             << " Elija una opcion: ";

        cin>>opcion;
        switch(opcion)
        {
            case 1:
                for(f=0; f<NFIL; f++)
                {
                    for(c=0; c<NCOL; c++)
                    {
                        do
                        {
                            cout<<"Introduce el valor ["<<f+1<<","<<c+1<<"]:
";

                            cin>> tab[f][c];
                            if(tab[f][c]<0)
                                cout<<"Solo numeros positivos!!"<<endl;
                        }
                        while(tab[f][c]<0);
                    }
                }
                cout<<endl<<endl;
                break;
            case 2:
                cout<<endl<<"          TABLA"<<endl;
                for(f=0; f<NFIL; f++)
```

```

    {
        for(c=0; c<NCOL; c++)
        {
            cout<<tab[f][c]<<" ";
        }
        cout<<endl;
    }
    cout<<endl;
    break;
case 3:
    for(f=0; f<NFIL; f++)
    {
        for(c=0; c<NCOL; c++)
        {
            if(tab[f][c]%2==0)
                par=par+1;
            else if(tab[f][c]%2!=0)
                impar=impar+1;
        }
    }
    cout<<"De los valores introducidos, "<<par<<" de ellos son
pares y "<<impar<<" impares"<<endl<<endl;
    break;

case 4:
    cout<<endl;
    for(f=0; f<NFIL; f++)
    {
        for(c=0; c<NCOL; c++)
        {
            sum=sum+tab[f][c];
        }
        cout<<"La suma de la fila "<<f+1<<" es igual a
"<<sum<<endl;
        if(c==NCOL)
            sum=0;
    }
    cout<<endl;
    break;
case 5:
    cout<<endl;
    for(c=0; c<NCOL; c++)
    {
        for(f=0; f<NFIL; f++)
        {
            sum=sum+tab[f][c];
        }
    }

```

```

        cout<<"La suma de la columna "<<c+1<<" es igual a
"<<sum<<endl;
        if(f==NFIL)
            sum=0;
    }
    cout<<endl;
    break;
case 6:
    for(f=0; f<NFIL; f++)
    {
        for(c=0; c<NCOL; c++)
        {
            if(f==c)
                sum=sum+tab[f][c];
        }
    }
    cout<<"La suma de la diagonal principal es igual a
"<<sum<<endl<<endl;
    break;
case 7:
    for(f=0; f<NFIL; f++)
    {
        for(c=0; c<NCOL; c++)
        {
            if(f+c == NFIL-1)
                sum=sum+tab[f][c];
        }
    }
    cout<<"La suma de la diagonal secundaria es igual a
"<<sum<<endl<<endl;
    break;
case 8:
    cout<<"Introduce el indice de las filas que quieres intercambiar
[0,9]: "<<endl;
    do{
        do{
            cout<<"Fila 1: ";
            cin>>f1;

            if(f1<0 || f1>9)
                cout<<"Indice no valido"<<endl;
        }while(f1<0 || f1>9);

        do{
            cout<<"Fila 2: ";
            cin>>f2;

```



```

        if(f2<0 || f2>9)
            cout<<"Indice no valido"<<endl;
    }while(f2<0 || f2>9);

    if(f1==f2)
        cout<<"No se puede intercambiar una misma
fila."<<endl;

    }while(f1==f2);

    for(c=0; c<NCOL; c++)
    {
        alt=tab[f1][c];
        tab[f1][c]=tab[f2][c];
        tab[f2][c]=alt;
    }
    break;
case 9:
    cout<<"Introduce el indice de las columnas que quieres
intercambiar [0,9]: "<<endl;
    do{
        do{
            cout<<"Columna 1: ";
            cin>>f1;

            if(c1<0 || c1>9)
                cout<<"Indice no valido"<<endl;
        }while(c1<0 || c1>9);

        do{
            cout<<"Columna 2: ";
            cin>>c2;

            if(c2<0 || c2>9)
                cout<<"Indice no valido"<<endl;
        }while(c2<0 || c2>9);

        if(c1==c2)
            cout<<"No se puede intercambiar una misma
columna."<<endl;

        }while(c1==c2);

    for(f=0; f<NCOL; f++)
    {
        alt=tab[f][c1];
        tab[f][c1]=tab[f][c2];
        tab[f][c2]=alt;
    }

```

```
        break;
    }
    if(opcion<1 || opcion>10)
        cout<<" Del 1 al 10 gilipollas"<<endl<<endl;
}while (opcion != 10);

return 0;
}
```

```

#include <iostream>
#define N 8
#define M 5

using namespace std;

typedef int Conjunto[N];

int main()
{
    Conjunto cjtos[M];
    int op,nconjun,elementos[N], conjun1, conjun2;
    do
    {
        cout << "    MENU CONJUNTOS " << endl
            << "    1. Cargar Conjuntos" <<endl
            << "    2. Mostrar Conjuntos" <<endl
            << "    3. Mostrar interseccion de dos conjuntos"
        <<endl
            << "    4. Ver si dos conjuntos son Disjuntos"<<endl
            << "    5. Ver si dos conjuntos son iguales"<<endl
            << "    6. Salir"<<endl
            << " Elija una opcion: ";
        cin>>op;
        system("cls");

        switch(op)
        {
            case 1:
                do
                {
                    cout<<"Numero de conjuntos que desea cargar
entre 1 y 5: ";
                    cin >> nconjun;
                    system("cls");
                }
                while(nconjun<1 || nconjun>5);

                for (int i=0; i<nconjun; i++)
                {
                    do
                    {
                        cout << "Numero de elementos que tiene el
conjunto " << i+1 << ": ";
                        cin >> elementos[i];
                    }
                    while (elementos[i]<1 || elementos[i]>N);
                }
                for (int i=0; i<nconjun; i++)

```

```

        {
            for (int j=0; j<elementos[i]; j++)
            {
                cout << "Asigne un valor al elemento " <<
j+1 << " del conjunto " << i+1 << ": ";
                cin >> cjtos[i][j];
            }
        }
        break;
    case 2:
        for (int i=0; i<nconjun; i++)
        {
            for (int j=0; j<elementos[i]; j++)
            {
                cout<<"Conjunto "<<i+1<<" , elemento
"<<j+1<<": "<<cjtos[i][j]<<endl;
            }
        }

        break;
    case 3:
    {
        int inter=0;
        cout << "Introduzca un conjunto: ";
        cin >> conjun1;
        cout << "Introduzca otro conjunto: ";
        cin >> conjun2;

        cout << "Coinciden los siguientes valores: ";

        for (int i=0; i<elementos[conjun1-1]; i++)
        {
            for (int j=0; j<elementos[conjun2-1]; j++)
            {
                if (cjtos[conjun1-1][i]==cjtos[conjun2-
1][j])
                {
                    cout << cjtos[conjun1-1][i] << ", ";
                    inter++;
                }
            }
        }cout<<endl;
        if (inter==0)
        {
            system("cls");
            cout << "No coincide ningun valor."<<endl;
        }
        break;
    }
}

```

```

case 4:
{
    int inter=0;
    cout << "Introduzca un conjunto: ";
    cin >> conjun1;
    cout << "Introduzca otro conjunto: ";
    cin >> conjun2;

    cout << "No son disjuntos";

    for (int i=0; i<elementos[conjun1-1]; i++)
    {
        for (int j=0; j<elementos[conjun2-1]; j++)
        {
            if (cjtos[conjun1-1][i]==cjtos[conjun2-
1][j])
            {
                inter++;
            }
        }
    }
    if (inter==0)
    {
        system("cls");
        cout << "Son disjuntos";
    }
    break;
}

case 5:
{
    int inter=0;
    cout << "Introduzca un conjunto: ";
    cin >> conjun1;
    cout << "Introduzca otro conjunto: ";
    cin >> conjun2;

    cout << "No son iguales";

    for (int i=0; i<elementos[conjun1-1]; i++)
    {
        for (int j=0; j<elementos[conjun2-1]; j++)
        {
            if (cjtos[conjun1-1][i]==cjtos[conjun2-
1][j])
            {
                cout << cjtos[conjun1-1][i] << ", ";
                inter++;
            }
        }
    }
}

```

```

        }
    }
}
if (inter==elementos[conjun1-1] &&
inter==elementos[conjun2-1])
{
    system("cls");
    cout << "Los elementos son iguales" << endl;
}
break;
}
}

}
while (op != 6);
return 0;
}

```

18)

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

int main()
{
    int Vuno[15], Vdos[15], Vfus[30],opcion, i;
    int numuno, numdos,maximo,minimo,sum;
    do
    {
        cout << "          MENU " << endl
            << "  1. Pedir Vector 1" <<endl
            << "  2. Pedir Vector 2" <<endl
            << "  3. Mostrar Vectores" <<endl
            << "  4. Mezclar Vectores"<<endl
            << "  5. Mostrar Mezcla"<<endl
            << "  6. Salir"<<endl
            << " Elija una opcion: ";
        cin>>opcion;
        switch(opcion)
        {
            case 1:
                do{
                    cout<<"Indique el numero de elementos que desea rellenar del
Vector 1(15 max): ";
                    cin>>numuno;
                }while(numuno>15);

                for(i=0; i<numuno; i++)
                {
                    cout<< "Introduce el valor " << i+1<<": ";
                    cin>> Vuno[i];
                }
                break;
            case 2:
                do{
                    cout<<"Indique el numero de elementos que desea rellenar del
Vector 1(15 max): ";
```

```

        cin>>numdos;
    }while(numdos>15);

    for (i=0; i<numdos; i++)
    {
        cout<<"Introduce el valor "<<i+1<<" ": ";
        cin>> Vdos[i];
    }
    break;
case 3:
    cout << "Vector 1: ";
    for(i=0; i<numuno; i++)
    {
        cout<< Vuno[i];
        if(i<numuno-1)
            cout<<" ";
    }
    cout<<endl<<"Vector 2: ";
    for (i=0; i<numdos; i++)
    {
        cout<< Vdos[i];
        if(i<numdos-1)
            cout<<" ";
    }
    cout<<endl;

case 4:
    for(i=0; i<numuno; i++)
        Vfus[i]=Vuno[i];

    for(i=0; i<numdos; i++){
        Vfus[numuno+i]=Vdos[i];
    }
    break;
case 5:
    sum=numuno+numdos;
    cout << "Vector mezclado: ";
    for(i=0; i<sum; i++)
    {
        cout<< Vfus[i];
        if(i<(sum-1))
            cout<<" ";
    }cout<<endl;

    }
}while (opcion != 6);
return 0;
}

```


19)

```
#include <iostream>
#include <conio.h>
#include <cstring>
#include <cctype>

using namespace std;

int main()
{
    char palabraSecreta[50], palabra[50], letra[1];
    int puntos=9, tamano, ldescub=0, descubant=0, t, p;

    cout << "Tamano de la palabra: ";
    cin >> tamano;

    cout << "Introduzca la palabra secreta: ";
    for (int i=0; i<tamano; i++)
        {palabraSecreta[i]=getch();
        cout << "*";}

   strupr(palabraSecreta);
    for (int i=0; i<tamano; i++)
        palabra[i]='_';

    cout << endl << "Palabra a descubrir: " << palabra << endl;

    do
    {
        descubant=ldescub;
        cout << endl << "Introduzca una letra: ";
        cin >> letra[0];
        letra[0]=toupper(letra[0]);

        cout << letra[0];
        system("cls");

        cout << endl << "Palabra a descubrir: ";

        for (int i=0; i<tamano; i++)
```

```

{
if (letra[0]==palabraSecreta[i])
    {palabra[i]=palabraSecreta[i];
    ldescub=ldescub+1;}}
if (ldescub==descubant)
    {puntos= puntos-1;}

t=ldescub;
p=puntos;

cout << palabra;
}
while (t<tamano && p>0);

if (t==tamano)
    {cout << endl << "ENHORABUENA" << endl;
    cout << "Puntos conseguidos: " << puntos;}
else
    {cout << endl << "Ups, se ha quedado sin puntos." << endl;
    cout << "La palabra era: " << palabraSecreta;}

return 0;
}

```

20)

```
#include <iostream>
#include <time.h>
#include <cstdlib>
#define TAMA 25

using namespace std;

int main()
{
    int Datos[1000]; //Vector donde se almacenan los valores a analizar
    int NDatos=0; //Nº de valores almacenados en Datos
    int Valores[TAMA]; //Vector que contiene la frecuencia de aparición
    int op,i=0,numrand,valor,valormax;
    bool datosfin=false, noesta=false;
    char op2;
    srand(time(NULL));
    do
    {
        cout << "          MENU " << endl
             << "  1. Pedir datos" <<endl
             << "  2. Analizar datos" <<endl
             << "  3. Estan todos?" <<endl
             << "  4. Valor repetido"<<endl
             << "  5. Valor mas repetido"<<endl
             << "  6. Mostrar datos" <<endl
             << "  7. Mostrar analisis" <<endl
             << "  8. Salir" <<endl
             << " Elija una opcion: ";

        cin>>op;
        switch(op)
        {

            case 1:
                cout<<"Desea introducirlos manualmente (m) o de forma
aleatoria (a)? ";
                cin >> op2;

                switch(op2)
                {
```

```

        case 'm':
            cout<<"Introduce valores enteros en el intervalo[0,24].
Introduce -1 cuando quieras parar. ";
            while(i<1000 && !datosfin)
            {
                do
                {
                    cin>>Datos[i];
                    if(Datos[i]==-1)
                        datosfin= true;
                    else if(Datos[i] < -1 || Datos[i]>24)
                        cout<<"Valor no valido, introducelo de
nuevo."<<endl;
                }
                else
                    NDatos++;
            }while(Datos[i] < -1 || Datos[i]>24);
            break;
        case 'a':
            NDatos= 1+rand()%1000;

            while(i<NDatos)
            {
                numrand= rand()%25;
                Datos[i]=numrand;
                i++;
            }
            break;

        case 2:
            for(i=0; i<TAMA; i++)
                Valores[i]=0;
            for(i=0; i<NDatos; i++)
                Valores[Datos[i]]++;
            break;
        case 3:
            for(i=0; i<TAMA; i++)
            {
                if(Valores[i]==0)
                    noesta=true;
            }
            if(noesta)
                cout<<endl<<"Hay valores del intervalolo [0,24] que no
están.";
            else

```

```

        cout<<endl<<"Todos los valores del intervalo [0,24]
aparecen al menos una vez en el vector Datos";
        break;
    case 4:
        cout<<endl<<"Introduce un valor del intervalo [0,24] para ver
cuantas veces se repite: ";

        do
        {
            cin>>valor;
        }
        while(valor<0 || valor>24);

        cout<<endl<<"El valor "<<valor<<" se repite
"<<Valores[valor]<<" veces.";
        break;
    case 5:
        i=0;
        valormax=Valores[i];
        while(i<TAMA)
        {
            if(Valores[i]>valormax)
                valormax = Valores[i];
            i++;
        }
        cout<<endl<<"El valor mas repetido es el "<<valormax;
        break;
    case 6:
        for(i=0; i<NDatos; i++)
        {
            cout<< Datos[i]<<" ";
        }
        break;
    case 7:
        for(i=0; i<TAMA; i++)
            cout<<Valores[i]<<" ";
        break;
    }
}
while(op!=8);

return 0;
}

```