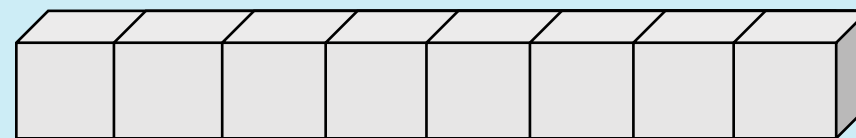
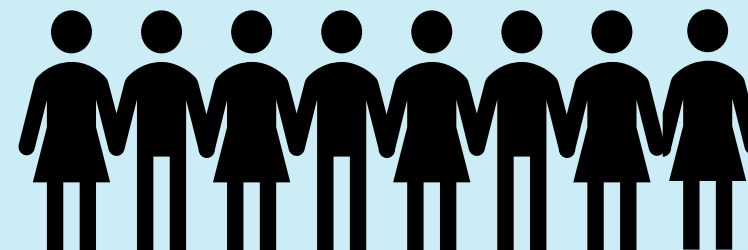




UNIVERSIDAD NACIONAL DE TRUJILLO

UNT

ARREGLOS



0 1 2 3 4 5 6 7



06. Ingrese los nombres y las notas de n alumnos y reportar una lista en orden alfabético y otra lista en orden de mérito.

Ingreso de datos

	Alumnos		notas
A[0]	Pepe	N[0]	13
A[1]	Arturo	N[1]	09
A[2]	Miguel	N[2]	20
A[3]	Vizcarra	N[3]	15
A[4]	Claudia	N[4]	18

Orden alfabético

Alumnos	notas
Arturo	09
Claudia	18
Miguel	20
Pepe	13
Vizcarra	15

Orden mérito

Alumnos	notas
Miguel	20
Claudia	18
Vizcarra	15
Pepe	13
Arturo	09



PSEUDOCODIGO

numDatos(entero n(R))

Hacer

escribir " Número de alumnos: "

leer n

mientras $n \leq 0$

fin_numDatos



PSEUDOCODIGO

ingresoDatos(cadena nombres[], real notas[], entero n)

entero i

para i ← 0 hasta n - 1 inc 1 hacer

escribir "Nombre: "

leer nombres[i]

Hacer

escribir "nota: "

leer notas[i]

mientras notas[i] < 0 o

notas[i] > 20

fin_para

fin_ingresoDatos



PSEUDOCODIGO

reporteDatos(cadena nombres[], real notas[], entero n)

entero i

para i \leftarrow 0 hasta n - 1 inc 1 hacer

 escribir nombres[i], ", ", notas[i]

fin_para

fin_reporteDatos



PSEUDOCODIGO

ordenAlfabetico(cadena nombres[], real notas[], entero n)

```
entero i,j
cadena temp1
real temp2
para i ← 0 hasta n - 2 inc 1 hacer
    para j ← i+1 hasta n - 1 inc 1 hacer
        si nombres[i] > nombres[j] entonces
            temp1 ← nombres[i]
            nombres[i] ← nombres[j]
            nombres[j] ← temp1
            temp2 ← notas[i]
            notas[i] ← notas[j]
            notas[j] ← temp2
        fin_si
    fin_para
fin_para
```

fin_ordenAlfabetico



PSEUDOCODIGO

ordenMerito(cadena nombres[], real notas[], entero n)

```
entero i,j
cadena temp1
real temp2
para i ← 0 hasta n - 2 inc 1 hacer
    para j ← i+1 hasta n - 1 inc 1 hacer
        si notas[i] < notas[j] entonces
            temp1 ← nombres[i]
            nombres[i] ← nombres[j]
            nombres[j] ← temp1
            temp2 ← notas[i]
            notas[i] ← notas[j]
            notas[j] ← temp2
        fin_si
    fin_para
fin_para
```

fin_ordenAlfabetico



PSEUDOCODIGO

Algoritmo arreglos_06

```
entero n
cadena nombres[100]
real notas[100]
numDatos(n)
ingresoDatos(nombres,notas,n)
ordenAlfabetico(nombres,notas,n)
escribir "Datos en orden Alfabetico"
reporteDatos(nombres,notas,n)
ordenMerito(nombres,notas,n)
escribir "Datos en orden de Merito"
reporteDatos(nombres,notas,n)
```

fin_algoritmo



CODIFICACION

```
#include <iostream>
#include <string.h>
using namespace std;
#define MAX 100

void numDatos(int &n);
void ingresoDatos(char nombres[][60], float notas[],int n);
void reporteDatos(char nombres[][60], float notas[],int n);
void ordenAlfabetico(char nombres[][60], float notas[],int n);
void ordenMerito(char nombres[][60], float notas[],int n);

int main(int argc, char *argv[]) {
    int n;
    char nombres[MAX][60];
    float notas[MAX];
    numDatos(n);
    ingresoDatos(nombres,notas,n);
    ordenAlfabetico(nombres,notas,n);
    cout<<"Datos en orden Alfabetico"<<endl;
    reporteDatos(nombres,notas,n);
    ordenMerito(nombres,notas,n);
    cout<<"Datos en orden de Merito"<<endl;
    reporteDatos(nombres,notas,n);
    return 0;
}

void numDatos(int &n)
{
    do{
        cout<<"Número de alumnos : ";
        cin>>n;
    }while(n<=0);
}
```

```
void ingresoDatos(char nombres[][60], float notas[],int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        cin.get();
        cout<<"Nombre : ";
        cin.getline(nombres[i],60);
        do{
            cout<<"Nota : ";
            cin>>notas[i];
        }while(notas[i]<0 || notas[i]>20);
    }
}

void reporteDatos(char nombres[][60], float notas[],int n)
{
    int i;
    for(i=0;i<n;i++)
        cout<<nombres[i]<<" , "<<notas[i]<<endl;
}

void ordenAlfabetico(char nombres[][60], float notas[],int n)
{
    int i,j;
    char temp1[60];
    float temp2;
    for(i=0;i<n-1;i++)
        for(j=i+1;j<n;j++)
            if(strcmp(nombres[i],nombres[j])>0)
            {
                strcpy(temp1,nombres[i]);
                strcpy(nombres[i],nombres[j]);
                strcpy(nombres[j],temp1);
                temp2=notas[i];
                notas[i]=notas[j];
                notas[j]=temp2;
            }
}
```



CODIFICACION	SALIDA DE PANTALLA
<pre>void ordenMerito(char nombres[][60], float notas[],int n) { int i,j; char templ[60]; float temp2; for(i=0;i<n-1;i++) for(j=i+1;j<n;j++) if(notas[i]<notas[j]) { strcpy(templ,nombres[i]); strcpy(nombres[i],nombres[j]); strcpy(nombres[j],templ); temp2=notas[i]; notas[i]=notas[j]; notas[j]=temp2; } }</pre>	<p>Número de alumnos : 5</p> <p>Nombre : Pepe Nota : 13</p> <p>Nombre : Arturo Nota : 09</p> <p>Nombre : Miguel Nota : 20</p> <p>Nombre : Vizcarra Nota : 15</p> <p>Nombre : Claudia Nota : 18</p> <p>Datos en orden Alfabético</p> <p>Arturo, 9 Claudia,18 Miguel, 20 Pepe, 13 Vizcarra,15</p> <p>Datos en orden de Merito</p> <p>Miguel, 20 Claudia,18 Vizcarra,15 Pepe, 13 Arturo, 9</p>



07. Ingresar los nombres y las estaturas de n personas. Calcular el promedio de las estaturas y reportar las personas que son mayores que el promedio de las estaturas.

Ingreso de datos

Nombres		estaturas	
N[0]	Pedro	E[0]	1.80
N[1]	Roberto	E[1]	1.70
N[2]	Ana	E[2]	1.67
N[3]	Fernanda	E[3]	1.65
N[4]	Miguel	E[4]	1.76



PSEUDOCODIGO

numDatos(entero n(R))

Hacer

 escribir " Número de alumnos: "

 leer n

 mientras $n \leq 0$

fin_numDatos



PSEUDOCODIGO

```
ingresoDatos(cadena nombres[], real estaturas[], entero n)
    entero i
    para i ← 0 hasta n - 1 inc 1 hacer
        escribir "nombre: "
        leer nombres[i]
        Hacer
            escribir "estatura: "
            leer estaturas[i]
            mientras estaturas[i] ≤ 0
        fin_mientras
    fin_para
fin_ingresoDatos
```



PSEUDOCODIGO

reporteDatos(cadena nombres[], real estaturas[], entero n)

entero i

para $i \leftarrow 0$ hasta $n - 1$ inc 1 hacer

 escribir nombres[i], ", ",

 estaturas[i]

fin_para

fin_reporteDatos



PSEUDOCODIGO

real promedioEstaturas(real estaturas[], entero n)

 real s

 entero i

 s \leftarrow 0

 para i \leftarrow 0 hasta n - 1 inc 1 hacer

 s \leftarrow s + estaturas[i]

 fin_para

 retornar s/n

fin_promedioEstaturas



PSEUDOCODIGO

personasMayoresAlPromedio(cadena nombres[], real estaturas[], entero n)

```
    real pe
    entero i
    pe ← promedioEstaturas(estaturas,n)
    escribir "Promedio de Estaturas ", pe
    escribir "Personas con estaturas mayores
    al promedio"
    para i ← 0 hasta n – 1 inc 1 hacer
        si estaturas[i] > pe entonces
            escribir nombres[i]
        fin_si
    fin_para
```

fin_personasMayoresAlPromedio



PSEUDOCODIGO

Algoritmo arreglos_07

```
entero n
cadena nombres[100]
real estaturas[100]
numDatos(n)
ingresoDatos(nombres,estaturas,n)
escribir "Datos ingresados"
reporteDatos(nombres, estaturas,n)
personasMayoresAlPromedio(nombres,
estaturas,n)
fin_algoritmo
```



CODIFICACION

```
#include <iostream>
using namespace std;
#define MAX 100

void numDatos(int &n);
void ingresoDatos(char nombres[][60], float estaturas[],int n);
void reporteDatos(char nombres[][60], float estaturas[],int n);
float promedioEstaturas(float estaturas[], int n);
void personasMayoresAlPromedio(char nombres[][60], float estaturas[], int n);

int main(int argc, char *argv[]) {
    int n;
    char nombres[MAX][60];
    float estaturas[MAX];
    numDatos(n);
    ingresoDatos(nombres,estaturas,n);
    cout<<"Datos ingresados"<<endl;
    reporteDatos(nombres,estaturas,n);
    personasMayoresAlPromedio(nombres,estaturas,n);

    return 0;
}

void numDatos(int &n)
{
    do{
        cout<<"Número de alumnos : ";
        cin>>n;
    }while(n<=0);
}
```



CODIFICACION

```
void ingresoDatos(char nombres[][60], float estaturas[],int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        cin.get();
        cout<<"Nombre : ";
        cin.getline(nombres[i],60);
        do{
            cout<<"Estatura : ";
            cin>>estaturas[i];
        }while(estaturas[i]<0);
    }
}

void reporteDatos(char nombres[][60], float estaturas[],int n)
{
    int i;
    for(i=0;i<n;i++)
        cout<<nombres[i]<<" , "<<estaturas[i]<<endl;
}
```



CODIFICACION

```
float promedioEstaturas(float estaturas[], int n)
{
    float s=0;
    int i;
    for(i=0;i<n;i++)
        s=s+estaturas[i];
    return s/n;
}

void personasMayoresAlPromedio(char nombres[][60], float estaturas[], int n)
{
    float pe=promedioEstaturas(estaturas,n);
    int i;
    cout<<"Promedio de Estaturas"<<pe<<endl;
    cout<<"Personas con estaturas mayores al promedio"<<endl;
    for(i=0;i<n;i++)
    {
        if(estaturas[i]>pe)
            cout<<nombres[i]<<endl;
    }
}
```



SALIDA DE PANTALLA

Número de alumnos : 5

Nombre : Pedro

Estatura : 1.80

Nombre : Roberto

Estatura : 1.70

Nombre : Ana

Estatura : 1.67

Nombre : Fernanda

Estatura : 1.65

Nombre : Miguel

Estatura : 1.76

Datos ingresados

Pedro, 1.8

Roberto, 1.7

Ana, 1.67

Fernanda, 1.65

Miguel, 1.76

Promedio de Estaturas 1.716

Personas con estaturas mayores al promedio

Pedro

Miguel



08. Programa para ingresar el nombre, teléfono, fechaNacimiento de n personas.se pide hacer un menú con las siguientes opciones:

AGENDA

1. Registrar Persona
2. Buscar Persona
3. Modificar datos de Persona
4. Eliminar Persona
5. Mostrar Personas
6. Salir

Ingreso de datos

	Nombres		teléfono		fechaNacimiento
N[0]	Jose	E[0]	555	F[0]	25/12/1977
N[1]	Roxana	E[1]	320569	F[1]	20/01/2004
N[2]	Felicia	E[2]	852693	F[2]	05/03/1999
N[3]	Solares	E[3]	7777	F[3]	15/12/2000
N[4]	Ignacio	E[4]	201745	F[4]	28/07/1981



PSEUDOCODIGO

```
ingresoPersona(cadena nombre[], cadena teléfono[], cadena fechaNac[], entero n)
    escribir "Ingrese nombre: "
    leer nombre[n]
    escribir "Ingrese teléfono: "
    leer teléfono[n]
    escribir "Ingrese fecha de Nacimiento: "
    leer fechaNac[n]
    n ← n + 1
fin_ingresoPersonas
```



PSEUDOCODIGO

entero busqueda(cadena nombre[], entero n, cadena nomBus)

entero i

para i \leftarrow 0 hasta n - 1 inc 1 hacer

 si nombre[i] = nombBus entonces

 retornar i

 fin_si

fin_para

fin_busqueda



PSEUDOCODIGO

consultarPersona(cadena nombre[], cadena telefono[], cadena fechaNac[], entero n)

cadena nomBus

escribir "Nombre a buscar: "

leer nomBus

$p \leftarrow \text{búsqueda}(\text{nombre}, n, \text{nomBus})$

si $p \neq -1$ entonces

 escribir "Datos de la Persona"

 escribir "Datos de la Persona"

 escribir "Nombre: ", nombre[p]

 escribir "Teléfono: ", telefono[p]

 escribir "Fecha de Nacimiento "

 , fechaNac[p]

sino

 escribir "El nombre no se encuentra"

fin_si

fin_consultarPersona



PSEUDOCODIGO

modificarPersona(cadena nombre[], cadena telefono[], cadena fechaNac[], entero n)

```
cadena nomBus
escribir "Nombre a modificar: "
leer nomBus
p ← búsqueda(nombre,n,nomBus)
si p ≠ -1 entonces
    escribir "Datos de la Persona"
        escribir "Datos de la Persona"
        escribir "Nombre: ", nombre[p]
        escribir "Teléfono: ", telefono[p]
        escribir "Fecha de Nacimiento ", fechaNac[p]
    escribir "Nuevo nombre: "
    leer nombre[p]
    escribir "Nuevo teléfono: "
    leer telefono[p]
    escribir "Nueva fecha de Nacimiento: "
    leer fechaNac[p]
sino
    escribir "El nombre no se encuentra"
fin_si
```

fin_modificarPersona



PSEUDOCODIGO

eliminar(cadena nombre[], cadena telefono[], cadena fechaNac[], entero n(R), entero p)

entero i

para i \leftarrow p hasta n - 2 inc 1 hacer

 nombre[i] \leftarrow nombre[i+1]

 telefono[i] \leftarrow telefono[i+1]

 fechaNac[i] \leftarrow fechaNac[i+1]

fin_para

n \leftarrow n - 1

fin_eliminar



PSEUDOCODIGO

eliminarPersona(cadena nombre[], cadena telefono[], cadena fechaNac[], entero n(R))

cadena nomBus

escribir "Nombre a eliminar: "

leer nomBus

$p \leftarrow \text{búsqueda}(\text{nombre}, n, \text{nomBus})$

si $p \neq -1$ entonces

 eliminar(nombre, teléfono, fechaNac, n, p)

 escribir "Dato eliminado"

sino

 escribir "El nombre no se encuentra"

fin_si

fin_eliminarPersona



PSEUDOCODIGO

reporteDatos(cadena nombre[], cadena telefono[], cadena fechaNac[], entero n)

entero i

para i \leftarrow 0 hasta n - 1 inc 1 hacer

 escribir nombre[i], ", ", telefono[i], ", ",
 fechaNac[i]

fin_para

fin_reporteDatos



PSEUDOCODIGO

mostrarPersonas(cadena nombre[], cadena telefono[], cadena fechaNac[], entero n)

 escribir "Personas"

 reporteDatos (nombre,telefono,fechaNac,n)

fin_mostrarPersonas



PSEUDOCODIGO

Algoritmo Agenda

cadena nombre[100], teléfono[100],
fechaNac[100]

entero n, op

$n \leftarrow 0$

Hacer

 escribir "AGENDA"

 escribir "[1] Ingresar persona"

 escribir "[2] Consultar persona"

 escribir " [3] Modificar persona"

 escribir "[4] Eliminar persona"

 escribir "[5] Mostrar personas"

 escribir "[6] Salir "

 escribir "Ingrese opcion (1-6):"

 leer op

según_sea op hacer

 caso 1:

 ingresarPersona(nombre,telefono,fechaNac,n)

 caso 2:

 consultarPersona(nombre,telefono,fechaNac,n)

 caso 3:

 modificarPersona(nombre,telefono,fechaNac,n)

 caso 4:

 eliminarPersona(nombre,telefono,fechaNac,n)

 caso 5:

 mostrarPersonas(nombre,telefono,fechaNac,n)

fin_según_sea

mientras op \neq 6

fin_algoritmo



CODIFICACION

```
#include<iostream>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
using namespace std;
#define MAX 100

void ingresarPersona(char nombre[][60], char telefono[][20], char fechaNac[][20],int &n);
int busqueda(char nombre[][60], char telefono[][20], char fechaNac[][20],int n);
void consultarPersona(char nombre[][60], char telefono[][20], char fechaNac[][20],int n);
void modificarPersona(char nombre[][60], char telefono[][20], char fechaNac[][20],int n);
void eliminar(char nombre[][60], char telefono[][20], char fechaNac[][20],int &n,int p);
void eliminarPersona(char nombre[][60], char telefono[][20], char fechaNac[][20],int &n);
void reporteDatos(char nombre[][60], char telefono[][20], char fechaNac[][20],int n);
void mostrarPersonas(char nombre[][60], char telefono[][20], char fechaNac[][20],int n);

int main(int argc, char *argv[])
{
    char nombres[MAX][60],telefono[MAX][20],fechaNac[MAX][20];
    int n=0,op;
    do{
        system("cls");
        cout<<"AGENDA"<<endl;
        cout<<"[1] Ingresar persona"<<endl;
        cout<<"[2] Consultar persona"<<endl;
        cout<<"[3] Modificar persona"<<endl;
        cout<<"[4] Eliminar persona"<<endl;
        cout<<"[5] Mostrar personas"<<endl;
        cout<<"[6] Salir "<<endl;
        cout<<"Ingrese opcion (1-6):";
        cin>>op;
        cin.get();
        switch(op){
            case 1: ingresarPersona(nombres,telefono,fechaNac,n);break;
            case 2: consultarPersona(nombres,telefono,fechaNac,n);break;
            case 3: modificarPersona(nombres,telefono,fechaNac,n);break;
            case 4: eliminarPersona(nombres,telefono,fechaNac,n);break;
            case 5: mostrarPersonas(nombres,telefono,fechaNac,n);break;
```




CODIFICACION

```
    }
    }while(op!=6);
}

void ingresarPersona(char nombre[][60], char telefono[][20], char fechaNac[][20],int &n)
{
    system("cls");
    cout<<"Ingreso Nombre: ";
    cin.getline(nombre[n],60);
    cout<<"Ingreso telefono :";
    cin.getline(telefono[n],20);
    cout<<"Ingreso fecha de Nacimineto:";
    cin.getline(fechaNac[n],20);
    n++;
}

int busqueda(char nombre[][60], int n, char nomBus[])
{
    int i;
    for(i=0;i<n;i++)
    {
        if(strcmp(nombre[i],nomBus)==0)
            return i;
    }
    return -1;
}

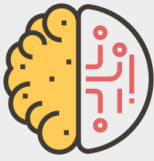
void consultarPersona(char nombre[][60], char telefono[][20], char fechaNac[][20],int n)
{
    char nomBus[60];
    int p;
    system("cls");
    cout<<"Nombre a buscar : ";
    cin>>nomBus;
    p=busqueda(nombre,n,nomBus);
```



CODIFICACION

```
if(p!=-1)
{
    cout<<"Datos de la Persona"<<endl;
    cout<<"Nombre : "<<nombre[p]<<endl;
    cout<<"Telefono : "<<telefono[p]<<endl;
    cout<<"Fecha de Nacimiento "<<fechaNac[p]<<endl;
}
else
    cout<<"El nombre no se encuentra"<<endl;
    system("pause");
}

void modificarPersona(char nombre[][60], char telefono[][20], char fechaNac[][20],int n)
{
    char nomBus[60];
    int p;
    system("cls");
    cout<<"nombre a modificar : ";
    cin.getline(nomBus,60);
    p=busqueda(nombre,n,nomBus);
    if(p!=-1)
    {
        cout<<"Datos de la Persona"<<endl;
        cout<<"Nombre : "<<nombre[p]<<endl;
        cout<<"Telefono : "<<telefono[p]<<endl;
        cout<<"Fecha de Nacimiento "<<fechaNac[p]<<endl;
        cout<<"Nuevo Nombre : ";
        cin.getline(nombre[p],60);
        cout<<"Nuevo Telefono : ";
        cin.getline(telefono[p],20);
        cout<<"Nuevo Fecha de Nacimiento : ";
        cin.getline(fechaNac[p],20);
    }
    else
        cout<<"El nombre no se encuentra"<<endl;
        system("pause");
}
```



CODIFICACION

```
void eliminar(char nombre[][60], char telefono[][20], char fechaNac[][20],int &n,int p)
{
    int i;
    for(i=p;i<n-1;i++)
    {
        strcpy(nombre[i],nombre[i+1]);
        strcpy(telefono[i],telefono[i+1]);
        strcpy(fechaNac[i],fechaNac[i+1]);
    }
    n=n-1;
}

void eliminarPersona(char nombre[][60], char telefono[][20], char fechaNac[][20],int &n)
{
    char nomBus[60];
    int p;
    system("cls");
    cout<<"nombre a eliminar : ";
    cin>>nomBus;
    p=busqueda(nombre,n,nomBus);
    if(p!=-1)
    {
        eliminar(nombre,telefono,fechaNac,n,p);
        cout<<"Dato eliminado"<<endl;
    }
    else
        cout<<"El dato no se encuentra"<<endl;
    system("pause");
}

void reporteDatos(char nombre[][60], char telefono[][20], char fechaNac[][20],int n)
{
    int i;

    for(i=0;i<n;i++)
        cout<<nombre[i]<<" , "<<telefono[i]<<" , "<<fechaNac[i]<<endl;
}

void mostrarPersonas(char nombre[][60], char telefono[][20], char fechaNac[][20],int n)
{
    system("cls");
    cout<<"Personas"<<endl;
    reporteDatos (nombre,telefono,fechaNac,n);
    system("pause");
}
```



SALIDA DE PANTALLA

AGENDA

- [1] Ingresar persona
- [2] Consultar persona
- [3] Modificar persona
- [4] Eliminar persona
- [5] Mostrar personas
- [6] Salir

Ingrese opcion (1-6):1

Ingrese Nombre: Jose

Ingrese teléfono :555

Ingrese fecha de Nacimineto:25/12/1977

Presione una tecla para continuar . . .

Ingrese opcion (1-6):2

Nombre a buscar: Jose

Nombre a buscar : Jose

Datos de la Persona

Nombre : Jose

Telefono : 555

Fecha de Nacimiento 25/12/1977

Presione una tecla para continuar . . .

Ingrese opcion (1-6):3

nombre a modificar : Jose

Datos de la Persona

Nombre : Jose

Telefono : 555

Fecha de Nacimiento 25/12/1977

Nuevo Nombre : Miguel

Nuevo Telefono : 777

Nuevo Fecha de Nacimiento : 09/09/2000

Presione una tecla para continuar . . .

Ingrese opcion (1-6):4

Nombre a eliminar: Miguel

Dato eliminado

Presione una tecla para continuar . . .

Ingrese opcion (1-6):5

Personas

Presione una tecla para continuar . . .

Ingrese opcion (1-6):6

<< El programa ha finalizado: codigo de salida:

0 >>