

# Laboratorio 13

## Estructuras

**Nombre:** Gabriel Fernando Rodriguez Cutimbo

**CUI:** 20212157

**Grupo:** B

**Repositorio GitHub:**

[https://github.com/gaco123/EPCC\\_CCII.git](https://github.com/gaco123/EPCC_CCII.git)

### 1. Ejercicios

Resolver los siguientes ejercicios planteados:

1. Implementar un programa que maneje un arreglo de estructuras para almacenar los nombres y las fechas de cumpleaños de sus n compañeros. Y debe mostrarse por pantalla quienes cumplen años en este mes.

#### Código:

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

//Estructura esencial
struct amigo{
    string nombres;
    string fecha;
};

//Función para comprobar si cumple este mes un amigo
void esteMes(amigo a, string actual){
    string m1 = a.fecha.substr(3,2);
    string m2 = actual.substr(3,2);
    if(m1==m2){
        cout<<a.nombres<<" tiene su cumpleaños este mes\n";
    }
}

int main(int argc, char *argv[]){
    //Memoria
    string actual;
```

```
vector<amigo> lista;
int n;
string tempn;
string tempf;

//Datos Iniciales
cout<<"Ingrese cuantos n amigos tiene: ";
cin>>n;
cout<<"Ingrese la fecha actual (formato DD/MM/AAAA): ";
getline(cin,actual);
getline(cin,tempf);
cout<<endl;

//Datos de cada amigo
for(int i=0; i<n; i++){
    cout<<"Ingrese el nombre completo de su amigo: ";
    getline(cin,tempn);
    cout<<"Ingrese la fecha de cumpleaños de su amigo (formato DD/MM/AAAA): ";
    getline(cin,tempf);
    amigo a={tempn,tempf};
    lista.push_back(a);
}
cout<<endl;

//Comprobación de cada amigo con la función
for(auto i=lista.begin(); i!=lista.end(); i++){
    esteMes(*i,actual);
}

return 0;
}
```

### Funcionamiento:

```
Ingrese cuantos n amigos tiene: 2
Ingrese la fecha actual (formato DD/MM/AAAA): 21/06/2022

Ingrese el nombre completo de su amigo: Pepe Miguel Quispe Casas
Ingrese la fecha de cumpleaños de su amigo (formato DD/MM/AAAA): 12/06/2000
Ingrese el nombre completo de su amigo: Marcos Fernando Mamani Torrez
Ingrese la fecha de cumpleaños de su amigo (formato DD/MM/AAAA): 26/10/1999

Pepe Miguel Quispe Casas tiene su cumpleaños este mes

<< El programa ha finalizado: codigo de salida: 0 >>
<< Presione enter para cerrar esta ventana >>_
```

2. Implementar un programa que maneje un arreglo de estructuras que calcule la nota final del Ciencia de la Computación. El programa debe permitir el ingreso de cualquier cantidad de alumnos y para cada alumno, se podrá ingresar nombre, grupo, nota de la primera fase, segunda fase, tercera fase y proyecto final. El porcentaje de cada ítem es 15%, 20%, 25% y 40% respectivamente.

### Código:

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

//Estructura esencial
struct alumno{
    string nombre;
    string grupo;
    double nota1;
    double nota2;
    double nota3;
    double notapf;
    double total;
};

int main(int argc, char *argv[]){
    //Memoria
    vector<alumno> lista;
    string si;

    //Lista de infinitos alumnos
    do{
        alumno a;
        cout<<"Ingrese el nombre del alumno: ";
        getline(cin,a.nombre);
        cout<<"Ingrese el grupo del alumno: ";
        getline(cin,a.grupo);
        cout<<"Ingrese la nota de la primera fase: ";
        cin>>a.nota1;
        cout<<"Ingrese la nota de la segunda fase: ";
        cin>>a.nota2;
        cout<<"Ingrese la nota de la tercera fase: ";
        cin>>a.nota3;
        cout<<"Ingrese la nota del proyecto final: ";
        cin>>a.notapf;
        a.total=((a.nota1*15)+(a.nota2*20)+(a.nota3*25)+(a.notapf*40))/100;
        cout<<"Desea continuar? (escriba si), caso contrario escriba cualquier cosa: ";
        getline(cin,si);
        getline(cin,si);//Por alguna razón no agarra al primer getline
        lista.push_back(a);
        cout<<endl;
    } while(si=="si");
}
```

```
//Mostrar nota final de cada alumno
for(auto i=lista.begin(); i!=lista.end(); i++){
    cout<<"El alumno "<<(*i).nombre<<" del grupo "<<(*i).grupo<<" tiene una nota final de
"<<(*i).total<<"\n";
}

return 0;
}
```

### Funcionamiento:

```
Ingrese el nombre del alumno: Pepe
Ingrese el grupo del alumno: A
Ingrese la nota de la primera fase: 12
Ingrese la nota de la segunda fase: 16
Ingrese la nota de la tercera fase: 14
Ingrese la nota del proyecto final: 20
Desea continuar? (escriba si), caso contrario escriba cualquier cosa: si

Ingrese el nombre del alumno: Marcos
Ingrese el grupo del alumno: B
Ingrese la nota de la primera fase: 16
Ingrese la nota de la segunda fase: 14
Ingrese la nota de la tercera fase: 15
Ingrese la nota del proyecto final: 19
Desea continuar? (escriba si), caso contrario escriba cualquier cosa: no

El alumno Pepe del grupo A tiene una nota final de 16.5
El alumno Marcos del grupo B tiene una nota final de 16.55
```

3. Implemente un programa que maneje un arreglo de estructuras que solicite el nombre, edad y talla de 10 jugadores, debe mostrar por pantalla los que son menores de 20 años y tienen una talla mayor a 1,70 mts de altura.

### Código:

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

//Estructura esencial
struct jugador{
    string nombre;
```

```
int edad;
double talla;
};

int main(int argc, char *argv[]){
    //Memoria
    vector<jugador> lista;
    string si;

    //Lista de 10 jugadores
    for(int i=0; i<10; i++){
        jugador a;
        cout<<"Ingrese el nombre del jugador: ";
        getline(cin,a.nombre);
        if(i>=1){
            getline(cin,a.nombre);
        }
        cout<<"Ingrese la edad del jugador: ";
        cin>>a.edad;
        cout<<"Ingrese la talla del jugador: ";
        cin>>a.talla;
        lista.push_back(a);
        cout<<endl;
    }

    //Mostrar a jugadores menores de 20 años y con una talla mayor a 1.70mts
    for(auto i=lista.begin(); i!=lista.end(); i++){
        if((*i).edad<20&&(*i).talla>1.70){
            cout<<"El jugador "<<(*i).nombre<<" es menor de 20 años y tiene una talla mayor a
1.70mts\n";
        }
    }

    return 0;
}
```

### Funcionamiento:

```
Ingrese el nombre del jugador: Marcus
Ingrese la edad del jugador: 19
Ingrese la talla del jugador: 1.85

Ingrese el nombre del jugador: Edgar
Ingrese la edad del jugador: 16
Ingrese la talla del jugador: 1.7

El jugador Marcus es menor de 20 años y tiene una talla mayor a 1.70mts
```

4. Implemente un programa que maneje un arreglo de estructuras que solicite nombre, sexo y sueldo de los empleados de una empresa y debe mostrar por pantalla el menor y mayor sueldo.

**Código:**

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

//Estructura esencial
struct empleado{
    string nombre;
    string sexo;
    double sueldo;
};

int main(int argc, char *argv[]){
    //Memoria
    vector<empleado> lista;
    string si;

    //Lista de empleados
    do{
        empleado a;
        cout<<"Ingrese el nombre del empleado(a): ";
        getline(cin,a.nombre);
        cout<<"Ingrese el sexo del empleado (Hombre o Mujer): ";
        getline(cin,a.sexo);
        cout<<"Ingrese el sueldo del empleado: ";
        cin>>a.sueldo;
        cout<<"Desea continuar? (escriba si), caso contrario escriba cualquier cosa: ";
        getline(cin,si);
        getline(cin,si); //Por alguna razón no agarra al primer getline
        lista.push_back(a);
        cout<<endl;
    } while(si=="si");

    //Encontrar el mayor y menor sueldo entre los empleados
    empleado mayor=lista.at(0);
    empleado menor=lista.at(0);
    for(auto i=lista.begin(); i!=lista.end(); i++){
        if((*i).sueldo>mayor.sueldo){
            mayor=*i;
        }
        if((*i).sueldo<menor.sueldo){
            menor=*i;
        }
    }
}
```

```
//Mostrar
if(mayor.sexo=="Hombre"){
    cout<<"El empleado "<<mayor.nombre<<" tiene el mayor sueldo, el cual es "<<mayor.sueldo<<"\n";
}
else if(mayor.sexo=="Mujer"){
    cout<<"La empleada "<<mayor.nombre<<" tiene el mayor sueldo, el cual es "<<mayor.sueldo<<"\n";
}
if(menor.sexo=="Hombre"){
    cout<<"El empleado "<<menor.nombre<<" tiene el menor sueldo, el cual es "<<menor.sueldo<<"\n";
}
else if(menor.sexo=="Mujer"){
    cout<<"La empleada "<<menor.nombre<<" tiene el menor sueldo, el cual es "<<menor.sueldo<<"\n";
}

return 0;
}
```

### Funcionamiento:

```
Ingrese el nombre del empleado(a): Mark
Ingrese el sexo del empleado (Hombre o Mujer): Hombre
Ingrese el sueldo del empleado: 1450
Desea continuar? (escriba si), caso contrario escriba cualquier cosa: si

Ingrese el nombre del empleado(a): Alicia
Ingrese el sexo del empleado (Hombre o Mujer): Mujer
Ingrese el sueldo del empleado: 1500
Desea continuar? (escriba si), caso contrario escriba cualquier cosa: si

Ingrese el nombre del empleado(a): Gert
Ingrese el sexo del empleado (Hombre o Mujer): Hombre
Ingrese el sueldo del empleado: 1400
Desea continuar? (escriba si), caso contrario escriba cualquier cosa: no

La empleada Alicia tiene el mayor sueldo, el cual es 1500
El empleado Gert tiene el menor sueldo, el cual es 1400
```

---

## 2. Entregables

Al final estudiante deberá:

1. Compactar el código elaborado y subirlo al aula virtual de trabajo. Agregue sus datos personales como comentario en cada archivo de código elaborado.
2. Elaborar un documento que incluya tanto el código como capturas de pantalla de la ejecución del programa. Este documento debe de estar en formato PDF.
3. El nombre del archivo (comprimido como el documento PDF), será su LAB13\_GRUPO\_A/B/C\_CUI\_1erNOMBRE\_1erAPELLIDO.

(Ejemplo: LAB13\_GRUPO\_A\_2022123\_PEDRO\_VASQUEZ).

4. Debe remitir el documento ejecutable con el siguiente formato:

LAB13\_GRUPO\_A/B/C\_CUI\_EJECUTABLE\_1erNOMBRE\_1erAPELLIDO

(Ejemplo: LAB13\_GRUPO\_A\_EJECUTABLE\_2022123\_PEDRO\_VASQUEZ).

En caso de encontrarse trabajos similares, los alumnos involucrados no tendrán evaluación y serán sujetos a sanción.