



UNIVERSIDAD DE LAS FUERZAS ARMADAS-ESPE SEDE SANTO DOMINGO DE LOS TSÁCHILAS

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN - DCCO-SS

CARRERA DE INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN



ESTUDIANTE: Mendoza Justin

FECHA: 23/02/2024

Taller 2 TDA-ARCHIVOS

TERCER PARCIAL

PRIMER NIVEL - FUNDAMENTOS DE PROGRAMACIÓN

Capturas de Código y Ejecución:

1. Realizar TDA Simple-Anidado.

```
main.cpp x L1ArchivoMendoza.h x L2StructMendoza.h x
1  #ifndef L2STRUCTMENDOZA_H_INCLUDED
2  #define L2STRUCTMENDOZA_H_INCLUDED
3  #include <iostream>
4  #include <fstream>
5  #include "L1ArchivoMendoza.h"
6
7  using namespace std;
8
9  void imprimirDatosPersonas();
10 void CrearArchivosMendoza(int numEstudiantes);
11 void LeerArchivosMendoza(int numEstudiantes);
12 void menuMendoza();
13 recor
14 int numEstudiantes;
15 struct Persona{
16     char nombre[20], apellido[20];
17     int edad;
18     float peso;
19     string direccion;
20     } persona1("Justin", "Mendoza", 19, 1.6, "Nueva Ixixas1"), persona2("Andres", "duenas", 18, 1.4, "El carmen"), persona3, persona4; //para un idá siempre va un ':'
21 struct Estudiante{
22     string materia, idMateria;
23     int horasMateria;
24     float promedio;
25     struct Persona infoPersona;
26     } personas[100];
```

2. Escritura de Archivo

```

main.cpp x L1ArchivoMendoza.h x *L2StructMendoza.h x
25     struct Persona infoPersona;
26 }personas[100];
27 void imprimirDatosPersonas(){
28     cout << "PERSONAS DINAMICAS" << endl;
29     cin.ignore();
30     cout << "Nombre de la persona 3: " << endl;
31     cin.getline(persona3.nombre,20,'\n');
32     cout << "Apellido de la persona 3: " << endl;
33     cin.getline(persona3.apellido,20,'\n');
34     cout << "Edad de la persona 3: " << endl;
35     cin >> persona3.edad;
36     cout << "Peso de la persona 3: " << endl;
37     cin >> persona3.peso;
38     cin.ignore();
39     cout << "Direccion de la persona 3: " << endl;
40     getline(cin,persona3.direccion);
41     cout << "Nombre de la persona 4: " << endl;
42     cin.getline(persona4.nombre,20,'\n');
43     cout << "Apellido de la persona 4: " << endl;
44     cin.getline(persona4.apellido,20,'\n');
45     cout << "Edad de la persona 4: " << endl;
46     cin >> persona4.edad;
47     cout << "Peso de la persona 4: " << endl;
48     cin >> persona4.peso;
49     cin.ignore();
50     cout << "Direccion de la persona 4: " << endl;
51     getline(cin,persona4.direccion);
52     cout << "\nLa persona 1" << endl;
53     cout << "Nombres: " << persona1.nombre << " " << persona1.apellido << endl;
54     cout << "Edad: " << persona1.edad << endl;
55     cout << "Peso: " << persona1.peso << endl;
56     cout << "Direccion: " << persona1.direccion << endl;
57     cout << "\nLa persona 2" << endl;
58     cout << "Nombres: " << persona2.nombre << " " << persona2.apellido << endl;
59     cout << "Edad: " << persona2.edad << endl;
60     cout << "Peso: " << persona2.peso << endl;
61     cout << "Direccion: " << persona2.direccion << endl;
62     cout << "\nLa persona 3" << endl;
63     cout << "Nombres: " << persona3.nombre << " " << persona3.apellido << endl;
64     cout << "Edad: " << persona3.edad << endl;
65     cout << "Peso: " << persona3.peso << endl;
66     cout << "Edad: " << persona3.edad << endl;
67     cout << "Peso: " << persona3.peso << endl;
68     cout << "Direccion: " << persona3.direccion << endl;
69     cout << "\nLa persona 4" << endl;
70     cout << "Nombres: " << persona4.nombre << " " << persona4.apellido << endl;
71     cout << "Edad: " << persona4.edad << endl;
72     cout << "Peso: " << persona4.peso << endl;
73     cout << "Direccion: " << persona4.direccion << endl;
74     cout << "\nEstructura Anidada" << endl;
75     cout << "INGRESAR EL NUMERO DE ESTUDIANTES: [1-4]" << endl;
76     cin >> numEstudiantes;
77     while(numEstudiantes<0 || numEstudiantes>4){
78         cout << "VUELVE A INGRESAR EL NUMERO DE ESTUDIANTES: [1-4]" << endl;
79         cin >> numEstudiantes;
80     }
81     for(int i=0; i<numEstudiantes; i++){
82         fflush(stdin);
83         cout << "Nombre de la persona: " << i+1 << endl;
84         cin.getline(personas[i].infoPersona.nombre,20,'\n');
85         cout << "Apellido de la persona: " << i+1 << endl;
86         cin.getline(personas[i].infoPersona.apellido,20,'\n');
87         cout << "Edad de la persona: " << i+1 << endl;
88         cin >> personas[i].infoPersona.edad;
89         cout << "Peso de la persona: " << i+1 << endl;
90         cin >> personas[i].infoPersona.peso;
91         cin.ignore();
92         cout << "Direccion de la persona: " << i+1 << endl;
93         getline(cin,personas[i].infoPersona.direccion);
94         cout << "Materia: " << i+1 << endl;
95         getline(cin,personas[i].materia);
96         cout << "id de la materia: " << i+1 << endl;
97         getline(cin,personas[i].idMateria);
98         cout << "Horas de la materia: " << i+1 << endl;
99         cin >> personas[i].horasMateria;
100        cout << "Promedio de la materia: " << i+1 << endl;
101        cin >> personas[i].promedio;
102    }
103    for(int i=0; i<numEstudiantes; i++){
104        cout << "\nEl Estudiante" << i+1 << endl;
105        cout << "Nombres: " << personas[i].infoPersona.nombre << " " << personas[i].infoPersona.apellido << endl;
106        cout << "Edad: " << personas[i].infoPersona.edad << endl;

```

Fundamentos de la Programación

```
main.cpp x L1ArchivoMendoza.h x *L2StructMendoza.h x
88     cin >> personas[i].infoPersona.peso;
89     cin.ignore();
90     cout << "Direccion de la persona: " << i+1 << endl;
91     getline(cin, personas[i].infoPersona.direccion);
92     cout << "Materia: " << i+1 << endl;
93     getline(cin, personas[i].materia);
94     cout << "id de la materia: " << i+1 << endl;
95     getline(cin, personas[i].idMateria);
96     cout << "Horas de la materia: " << i+1 << endl;
97     cin >> personas[i].horasMateria;
98     cout << "Promedio de la materia: " << i+1 << endl;
99     cin >> personas[i].promedio;
100 }
101 for(int i=0; i<numEstudiantes; i++){
102     cout << "\nEl Estudiante" << i+1 << endl;
103     cout << "Nombres: " << personas[i].infoPersona.nombre << " " << personas[i].infoPersona.apellido << endl;
104     cout << "Edad: " << personas[i].infoPersona.edad << endl;
105     cout << "Peso: " << personas[i].infoPersona.peso << endl;
106     cout << "Direccion: " << personas[i].infoPersona.direccion << endl;
107     cout << "Materia: " << personas[i].materia << endl;
108     cout << "ID Materia: " << personas[i].idMateria << endl;
109     cout << "Horas Materia: " << personas[i].horasMateria << endl;
110     cout << "Promedio Materia: " << personas[i].promedio << endl;
111 }
112 CrearArchivosMendoza(numEstudiantes);
113
114 }
115
116 void CrearArchivosMendoza(int numEstudiantes){
117     cout << "-----" << endl;
118     cout << "|| CREAR UN ARCHIVO.TXT ||" << endl;
119     cout << "-----" << endl;
120     ofstream ArchivoRegistroEstudiantes;
121     ArchivoRegistroEstudiantes.open("RegistroEstudiantes.txt", ios::app);
122     if(ArchivoRegistroEstudiantes.is_open()){
123         ArchivoRegistroEstudiantes << "\n\tSTRUCT SIMPLE" << endl;
124         ArchivoRegistroEstudiantes << "\n\t2 ESTUDIANTES ESTATICOS" << endl;
125         ArchivoRegistroEstudiantes << "\nLa persona 1" << endl;
126         ArchivoRegistroEstudiantes << "Nombres: " << persona1.nombre << " " << persona1.apellido << endl;
127         ArchivoRegistroEstudiantes << "Edad: " << persona1.edad << endl;
128         ArchivoRegistroEstudiantes << "Peso: " << persona1.peso << endl;
129         ArchivoRegistroEstudiantes << "Direccion: " << persona1.direccion << endl;
130         ArchivoRegistroEstudiantes << "\nLa persona 2" << endl;
131         ArchivoRegistroEstudiantes << "Nombres: " << persona2.nombre << " " << persona2.apellido << endl;
132         ArchivoRegistroEstudiantes << "Edad: " << persona2.edad << endl;
133         ArchivoRegistroEstudiantes << "Peso: " << persona2.peso << endl;
134         ArchivoRegistroEstudiantes << "Direccion: " << persona2.direccion << endl;
135         ArchivoRegistroEstudiantes << "\n\t2 ESTUDIANTES DINAMICOS" << endl;
136         ArchivoRegistroEstudiantes << "\nLa persona 3" << endl;
137         ArchivoRegistroEstudiantes << "Nombres: " << persona3.nombre << " " << persona3.apellido << endl;
138         ArchivoRegistroEstudiantes << "Edad: " << persona3.edad << endl;
139         ArchivoRegistroEstudiantes << "Peso: " << persona3.peso << endl;
140         ArchivoRegistroEstudiantes << "Direccion: " << persona3.direccion << endl;
141         ArchivoRegistroEstudiantes << "\nLa persona 4" << endl;
142         ArchivoRegistroEstudiantes << "Nombres: " << persona4.nombre << " " << persona4.apellido << endl;
143         ArchivoRegistroEstudiantes << "Edad: " << persona4.edad << endl;
144         ArchivoRegistroEstudiantes << "Peso: " << persona4.peso << endl;
145         ArchivoRegistroEstudiantes << "Direccion: " << persona4.direccion << endl;
146         ArchivoRegistroEstudiantes << "\n\tSTRUCT ANIDADO" << endl;
147         for(int i=0; i<numEstudiantes; i++){
148             ArchivoRegistroEstudiantes << "\nEl Estudiante" << i+1 << endl;
149             ArchivoRegistroEstudiantes << "Nombres: " << personas[i].infoPersona.nombre << " " << personas[i].infoPersona.apellido << endl;
150             ArchivoRegistroEstudiantes << "Edad: " << personas[i].infoPersona.edad << endl;
151             ArchivoRegistroEstudiantes << "Peso: " << personas[i].infoPersona.peso << endl;
152             ArchivoRegistroEstudiantes << "Direccion: " << personas[i].infoPersona.direccion << endl;
153             ArchivoRegistroEstudiantes << "Materia: " << personas[i].materia << endl;
154         }
155     }
156 }
```

Fundamentos de la Programación

```
main.cpp x L1ArchivoMendoza.h x "L2StructMendoza.h x
22 ArchivoRegistroEstudiantes << "\nLa persona 1" << endl;
23 ArchivoRegistroEstudiantes << "Nombre: " << persona1.nombre << " " << persona1.apellido << endl;
24 ArchivoRegistroEstudiantes << "Edad: " << persona1.edad << endl;
25 ArchivoRegistroEstudiantes << "Peso: " << persona1.peso << endl;
26 ArchivoRegistroEstudiantes << "Direccion: " << persona1.direccion << endl;
27 ArchivoRegistroEstudiantes << "\nLa persona 2" << endl;
28 ArchivoRegistroEstudiantes << "Nombre: " << persona2.nombre << " " << persona2.apellido << endl;
29 ArchivoRegistroEstudiantes << "Edad: " << persona2.edad << endl;
30 ArchivoRegistroEstudiantes << "Peso: " << persona2.peso << endl;
31 ArchivoRegistroEstudiantes << "Direccion: " << persona2.direccion << endl;
32 ArchivoRegistroEstudiantes << "\n2 ESTUDIANTES DINAMICOS" << endl;
33 ArchivoRegistroEstudiantes << "\nLa persona 3" << endl;
34 ArchivoRegistroEstudiantes << "Nombre: " << persona3.nombre << " " << persona3.apellido << endl;
35 ArchivoRegistroEstudiantes << "Edad: " << persona3.edad << endl;
36 ArchivoRegistroEstudiantes << "Peso: " << persona3.peso << endl;
37 ArchivoRegistroEstudiantes << "Direccion: " << persona3.direccion << endl;
38 ArchivoRegistroEstudiantes << "\nLa persona 4" << endl;
39 ArchivoRegistroEstudiantes << "Nombre: " << persona4.nombre << " " << persona4.apellido << endl;
40 ArchivoRegistroEstudiantes << "Edad: " << persona4.edad << endl;
41 ArchivoRegistroEstudiantes << "Peso: " << persona4.peso << endl;
42 ArchivoRegistroEstudiantes << "Direccion: " << persona4.direccion << endl;
43 ArchivoRegistroEstudiantes << "\n\tSTRUCT ANIDADO" << endl;
44 for(int i=0; i<numEstudiantes; i++){
45     ArchivoRegistroEstudiantes << "\nEl Estudiante" << i+1 << endl;
46     ArchivoRegistroEstudiantes << "Nombre: " << personas[i].infoPersona.nombre << " " << personas[i].infoPersona.apellido << endl;
47     ArchivoRegistroEstudiantes << "Edad: " << personas[i].infoPersona.edad << endl;
48     ArchivoRegistroEstudiantes << "Peso: " << personas[i].infoPersona.peso << endl;
49     ArchivoRegistroEstudiantes << "Direccion: " << personas[i].infoPersona.direccion << endl;
50     ArchivoRegistroEstudiantes << "Materia: " << personas[i].materia << endl;
51     ArchivoRegistroEstudiantes << "ID Materia: " << personas[i].idMateria << endl;
52     ArchivoRegistroEstudiantes << "Horas Materia: " << personas[i].horasMateria << endl;
53     ArchivoRegistroEstudiantes << "Promedio Materia: " << personas[i].promedio << endl;
54 }
55 ArchivoRegistroEstudiantes.close();
56 cout << "LOS DATOS SE HAN GUARDADO EN EL ARCHIVO 'RegistroEstudiantes.txt' GRACIASAMENTE." << endl;
57 }else{
58     cerr << "ERROR AL ABRIR EL ARCHIVO" << endl;
59 }
60 }
```

3. Lectura de Archivo

```
<global> LecturaArchivosMendoza(int numEstudiantes): void
main.cpp x L1ArchivoMendoza.h x "L2StructMendoza.h x
63 void LecturaArchivosMendoza(int numEstudiantes){
64     ifstream archivoRegistroEstudiantesMendoza;
65     string textoArchivoMendoza, archivoLecturaMendoza, siNoMendoza;
66     bool band=false, valida=false;
67     cout << "===== " << endl;
68     cout << "|| LEER UN ARCHIVO.TXT ||" << endl;
69     cout << "===== " << endl;
70     do{
71         cout << "INGRESA EL NOMBRE DEL ARCHIVO QUE DESEA ABRIR (SIN EL .TXT): " << endl;
72         cin >> archivoLecturaMendoza;
73         archivoLecturaMendoza.append(".txt");
74         archivoRegistroEstudiantesMendoza.open(archivoLecturaMendoza, ios::in);
75         if(archivoRegistroEstudiantesMendoza.fail()){
76             cout << "ERROR, NO SE PUEDE ABRIR EL ARCHIVO SOLO LECTURA.." << endl;
77         }else{
78             cout << "ARCHIVO ABIERTO EN EL MODO LECTURA.." << endl;
79             while(getline(archivoRegistroEstudiantesMendoza, textoArchivoMendoza)){
80                 cout << textoArchivoMendoza << endl;
81             }
82         }
83         archivoRegistroEstudiantesMendoza.close();
84         cout << "¿DESEAR VOLVER A INGRESAR EL NOMBRE EL ARCHIVO? [Si o No]" << endl;
85         cin >> siNoMendoza;
86         for(int i=0; siNoMendoza[i]; i++){
87             siNoMendoza[i] = tolower(siNoMendoza[i]);
88         }
89         cout << siNoMendoza << endl;
90         while(siNoMendoza!="s" && siNoMendoza!="no"){
91             cout << "VUELVA A INGRESAR [Si o No]" << endl;
92             cin >> siNoMendoza;
93             for(int i=0; siNoMendoza[i]; i++){
94                 siNoMendoza[i] = tolower(siNoMendoza[i]);
95             }
96         }
97         if(siNoMendoza=="s"){
98             band = false;
99         }else{
100             band = true;
101         }
102     }while(band!=true);
103 }
104
105 #endif // L1ARCHIVOMENDOZA H INCLUDED
```

Fundamentos de la Programación

The screenshot shows a Windows desktop with two windows. The left window is a debugger showing the execution of a C++ program. The right window is a Notepad application showing a text file named 'RegistroEstudiantes.txt'.

Debugger Window (C:\Users\ESPE\Desktop\P3Archivos\MendozaJustin\bin\Debug\P3Archivos.exe):

```
[1] INGRESAR DATOS AL TODA - CREAR EL ARCHIVO.TXT
[2] LEER EL ARCHIVO.TXT
[3] SALIR
SELECCIONA UNA OPCION DEL MENU:
1
PERSONAS DINAMICAS
Nombre de la persona 3:
Carlos
Apellido de la persona 3:
guerrero
Edad de la persona 3:
48
Peso de la persona 3:
3.6
Direccion de la persona 3:
sadsadsads
Nombre de la persona 4:
Andres
Apellido de la persona 4:
Duenas
Edad de la persona 4:
19
Peso de la persona 4:
2.3
Direccion de la persona 4:
asdsadsads
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Diereccion: Nuevo Isrrael
La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Diereccion: El carmen
La persona 3
Nombres: Carlos guerrero
Edad: 48
Peso: 3.6
Diereccion: sadsadsads
La persona 4
Nombres: Andres Duenas
Edad: 19
Peso: 2.3
Diereccion: asdsadsads
```

Notepad Window (RegistroEstudiantes.txt: Bloc de notas):

```
STRUCT SIMPLE
2 ESTUDIANTES ESTATICOS
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Diereccion: Nuevo Isrrael
La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Diereccion: El carmen
2 ESTUDIANTES DINAMICOS
La persona 3
Nombres: asdf
Edad: 12
Peso: 1.2
Diereccion: afd
La persona 4
Nombres: adsfg 23.3
Edad: 12
Peso: 12.3
Diereccion: afsgdhj
STRUCT ANIDADO
STRUCT SIMPLE
2 ESTUDIANTES ESTATICOS
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Diereccion: Nuevo Isrrael
```

The screenshot shows a Windows desktop with two windows. The left window is a debugger showing the execution of a C++ program. The right window is a Notepad application showing a text file named 'RegistroEstudiantes.txt'.

Debugger Window (C:\Users\ESPE\Desktop\P3Archivos\MendozaJustin\bin\Debug\P3Archivos.exe):

```
La persona 4
Nombres: Andres Duenas
Edad: 19
Peso: 2.3
Diereccion: asdsadsads
Estructura Anidada
INGRESAR EL NUMERO DE ESTUDIANTES: [1-4]
-1
VUELVE A INGRESAR EL NUMERO DE ESTUDIANTES: [1-4]
5
VUELVE A INGRESAR EL NUMERO DE ESTUDIANTES: [1-4]
3
Nombre de la persona: 1
Camila
Apellido de la persona: 1
Vega
Edad de la persona: 1
20
Peso de la persona: 1
3.6
Direccion de la persona: 1
sadsads
Materia: 1
dasdad
id de la materia: 1
8012
Horas de la materia: 1
12
Promedio de la materia: 1
15
Nombre de la persona: 2
Alexis
Apellido de la persona: 2
guerrero
Edad de la persona: 2
13
Peso de la persona: 2
2.36
Direccion de la persona: 2
daasafafa
Materia: 2
adsadsa
id de la materia: 2
123
Horas de la materia: 2
12
Promedio de la materia: 2
49
Nombre de la persona: 3
```

Notepad Window (RegistroEstudiantes.txt: Bloc de notas):

```
STRUCT SIMPLE
2 ESTUDIANTES ESTATICOS
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Diereccion: Nuevo Isrrael
La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Diereccion: El carmen
2 ESTUDIANTES DINAMICOS
La persona 3
Nombres: asdf
Edad: 12
Peso: 1.2
Diereccion: afd
La persona 4
Nombres: adsfg 23.3
Edad: 12
Peso: 12.3
Diereccion: afsgdhj
STRUCT ANIDADO
STRUCT SIMPLE
2 ESTUDIANTES ESTATICOS
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Diereccion: Nuevo Isrrael
```

Fundamentos de la Programación

The screenshot shows a Windows desktop with two windows. The left window is a debugger showing the execution of a C++ program. The right window is a text editor displaying the content of 'RegistroEstudiantes.txt'.

Debugger Output:

```
Promedio de la materia: 2
49
Nombre de la persona: 3
Arianna
Apellido de la persona: 3
Vergara
Edad de la persona: 3
1
Peso de la persona: 3
1
Diereccion de la persona: 3
daadsa
Materia: 3
adafaf
id de la materia: 3
123a
Horas de la materia: 3
23
Promedio de la materia: 3
18

c1 Estudiante1
Nombres: Camila Vega
Edad: 20
Peso: 3.6
Diereccion: adadads
Materia: dasdad
ID Materia: asi2
Horas Materia: 15
Promedio Materia: 15

El Estudiante2
Nombres: Alexis Guerrero
Edad: 13
Peso: 2.36
Diereccion: daasafafa
Materia: adsadsa
ID Materia: 123
Horas Materia: 12
Promedio Materia: 49

c1 Estudiante3
Nombres: Arianna Vergara
Edad: 1
Peso: 1
Diereccion: daadsa
Materia: adafaf
ID Materia: 123a
Horas Materia: 23
Promedio Materia: 18
=====
```

RegistroEstudiantes.txt:

```
STRUCT SIMPLE

2 ESTUDIANTES ESTATICOS

La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Diereccion: Nuevo Isrrael

La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Diereccion: El carmen

2 ESTUDIANTES DINAMICOS

La persona 3
Nombres: asdf
Edad: 12
Peso: 1.2
Diereccion: afd

La persona 4
Nombres: adsfg 23.3
Edad: 12
Peso: 12.3
Diereccion: afsgdhj

STRUCT ANIDADO

STRUCT SIMPLE

2 ESTUDIANTES ESTATICOS

La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Diereccion: Nuevo Isrrael
```

The screenshot shows a Windows desktop with two windows. The left window is a debugger showing the execution of a C++ program. The right window is a text editor displaying the content of 'RegistroEstudiantes.txt'.

Debugger Output:

```
Horas Materia: 23
Promedio Materia: 18
=====
[1] CREAR UN ARCHIVO.TXT ||
=====
LOS DATOS SE HAN GUARDADO EN EL ARCHIVO 'RegistroEstudiantes.txt' correctamente.
[1] INGRESAR DATOS AL TDA - CREAR EL ARCHIVO.TXT
[2] LEER EL ARCHIVO.TXT
[3] SALIR
SELECCIONA UNA OPCION DEL MENU:
2
=====
[1] CREAR UN ARCHIVO.TXT ||
=====
INGRESA EL NOMBRE DEL ARCHIVO QUE DESEA ABRIR (SIN EL .TXT):
RegistroEstudiantes
ARCHIVO ABIERTO EN EL MOD0 LECTURA..

STRUCT SIMPLE

2 ESTUDIANTES ESTATICOS

La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Diereccion: Nuevo Isrrael

La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Diereccion: El carmen

2 ESTUDIANTES DINAMICOS

La persona 3
Nombres: asdf
Edad: 12
Peso: 1.2
Diereccion: afd

La persona 4
Nombres: adsfg 23.3
Edad: 12
Peso: 12.3
Diereccion: afsgdhj

STRUCT ANIDADO

STRUCT SIMPLE
```

RegistroEstudiantes.txt:

```
STRUCT SIMPLE

2 ESTUDIANTES ESTATICOS

La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Diereccion: Nuevo Isrrael

La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Diereccion: El carmen

2 ESTUDIANTES DINAMICOS

La persona 3
Nombres: asdf
Edad: 12
Peso: 1.2
Diereccion: afd

La persona 4
Nombres: adsfg 23.3
Edad: 12
Peso: 12.3
Diereccion: afsgdhj

STRUCT ANIDADO

STRUCT SIMPLE

2 ESTUDIANTES ESTATICOS

La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Diereccion: Nuevo Isrrael
```

Fundamentos de la Programación

The screenshot shows a Windows desktop with two windows. The top window is a debugger (Visual Studio) running a C++ program. The bottom window is a Notepad application displaying the output of the program.

Debugger Window (C:\Users\ESPE\Desktop\P3Archivos\MendozaJustin\bin\Debug\P3Archivos.exe):

```
STRUCT ANIDADO
STRUCT SIMPLE
2 ESTUDIANTES ESTATICOS
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Direccion: Nuevo Isrrael
La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Direccion: El carmen
2 ESTUDIANTES DINAMICOS
La persona 3
Nombres: Carlos guerrero
Edad: 48
Peso: 3.6
Direccion: sadadsads
La persona 4
Nombres: Andres Duenas
Edad: 19
Peso: 2.3
Direccion: asdadsads
STRUCT ANIDADO
El Estudiante1
Nombres: Camila Vega
Edad: 20
Peso: 3.6
Direccion: adadads
Materia: as12
ID Materia: 12
Horas Materia: 15
Promedio Materia: 15
El Estudiante2
Nombres: Alexis Guerrero
Edad: 13
Peso: 2.36
Direccion: daasafdafa
Materia: adsadsa
ID Materia: 123
Horas Materia: 12
Promedio Materia: 49
El Estudiante3
Nombres: Arianna Vergara
Edad: 1
Peso: 1
Direccion: daadsa
Materia: adafaf
ID Materia: 123a
Horas Materia: 23
Promedio Materia: 18
¿DESEAR VOLVER A INGRESAR EL NOMBRE EL ARCHIVO? [Si o No]
no
[1] INGRESAR DATOS AL TDA - CREAR EL ARCHIVO.TXT
[2] LEER EL ARCHIVO.TXT
[3] SALIR
SELECCIONA UNA OPCION DEL MENU:
1
PERSONAS DINAMICAS
Nombre de la persona 3:
Karina
Apellido de la persona 3:
rodriguez
Edad de la persona 3:
32
Peso de la persona 3:
1.2
Dirección de la persona 3:
asad
Nombre de la persona 4:
Danny
Apellido de la persona 4:
Farias
Edad de la persona 4:
49
Peso de la persona 4:
2.3
Dirección de la persona 4:
asadadsad
```

Notepad Window (RegistroEstudiantes.txt: Bloc de notas):

```
STRUCT SIMPLE
2 ESTUDIANTES ESTATICOS
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Direccion: Nuevo Isrrael
La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Direccion: El carmen
2 ESTUDIANTES DINAMICOS
La persona 3
Nombres: asdf
Edad: 12
Peso: 1.2
Direccion: afd
La persona 4
Nombres: adsfg 23.3
Edad: 12
Peso: 12.3
Direccion: afsgdhj
STRUCT ANIDADO
STRUCT SIMPLE
2 ESTUDIANTES ESTATICOS
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Direccion: Nuevo Isrrael
El Estudiante2
Nombres: Alexis Guerrero
Edad: 13
Peso: 2.36
Direccion: daasafdafa
Materia: adsadsa
ID Materia: 123
Horas Materia: 12
Promedio Materia: 49
El Estudiante3
Nombres: Arianna Vergara
Edad: 1
Peso: 1
Direccion: daadsa
Materia: adafaf
ID Materia: 123a
Horas Materia: 23
Promedio Materia: 18
STRUCT SIMPLE
2 ESTUDIANTES ESTATICOS
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Direccion: Nuevo Isrrael
La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Direccion: El carmen
2 ESTUDIANTES DINAMICOS
La persona 3
Nombres: Karina rodriguez
Edad: 32
Peso: 1.2
Direccion: asdad
```

Fundamentos de la Programación

The image displays two screenshots of a Windows desktop environment. The top screenshot shows a console window running a C++ program and a text editor named 'RegistroEstudiantes.txt'. The console output shows a menu with options to create, read, or exit the file. The user has chosen to create the file, and the program prompts for dynamic student data (Name, Last Name, Age, Weight, Address, ID, Hours, Average). Four students are entered: Karina, Danny, Karina, and Danny. The text editor shows the data for three students: Alexis Guerrero, Arianna Vergara, and Andres duenas. The bottom screenshot shows the same console window and text editor. The console output shows the user has chosen to save the data to a file, and the program prompts for static student data. The text editor shows the data for three students: Justin Mendoza, Andres duenas, and Karina Rodriguez. The console output also shows the user has chosen to read the file, and the program displays the static student data for three students: Justin Mendoza, Andres duenas, and Karina Rodriguez.

```
C:\Users\ESPE\Desktop\P3Archivos\MendozaJustin\bin\Debug\P3Archivos.exe
¿DESEAR VOLVER A INGRESAR EL NOMBRE EL ARCHIVO? [Si o No]
no
no
[1] INGRESAR DATOS AL TODA - CREAR EL ARCHIVO.TXT
[2] LEER EL ARCHIVO.TXT
[3] SALIR
SELECCIONA UNA OPCION DEL MENU:
1
PERSONAS DINAMICAS
Nombre de la persona 3:
Karina
Apellido de la persona 3:
rodriguez
Edad de la persona 3:
32
Peso de la persona 3:
1.2
Direccion de la persona 3:
asadad
Nombre de la persona 4:
Danny
Apellido de la persona 4:
Farias
Edad de la persona 4:
49
Peso de la persona 4:
2.3
Direccion de la persona 4:
asdad
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Direccion: Nuevo Isrrael
La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Direccion: El carmen
La persona 3
Nombres: Karina rodriguez
Edad: 32
Peso: 1.2
Direccion: asdad
La persona 4
Nombres: Danny Farias
Edad: 49
Peso: 2.3
Direccion: asdad
Estructura Anidada
INGRESAR EL NUMERO DE ESTUDIANTES: [1-4]
1
Nombre de la persona: 1
Emely
Apellido de la persona: 1
aviles
Edad de la persona: 1
19
Peso de la persona: 1
2.36
Direccion de la persona: 1
adsada
Materia: 1
adafda
Id de la materia: 1
12a
Horas de la materia: 1
23
Promedio de la materia: 1
18
El Estudiante1
Nombres: Emely aviles
Edad: 19
Peso: 2.36
Direccion: adsada
Materia: adafda
ID Materia: 12a
Horas Materia: 23
Promedio Materia: 18
=====
[1] CREAR UN ARCHIVO.TXT [1]
=====
LOS DATOS SE HAN GUARDADO EN EL ARCHIVO 'RegistroEstudiantes.txt' correctamente.
[1] INGRESAR DATOS AL TODA - CREAR EL ARCHIVO.TXT
[2] LEER EL ARCHIVO.TXT
[3] SALIR
SELECCIONA UNA OPCION DEL MENU:
2
=====
[1] CREAR UN ARCHIVO.TXT [1]
=====
INGRESA EL NOMBRE DEL ARCHIVO QUE DESEA ABRIR (SIN EL .TXT):
RegistroEstudiantes
ARCHIVO ABIERTO EN EL MODO LECTURA..
```

RegistroEstudiantes.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda
El Estudiante2
Nombres: Alexis Guerrero
Edad: 13
Peso: 2.36
Direccion: daasafda
Materia: adsada
ID Materia: 123
Horas Materia: 12
Promedio Materia: 49
El Estudiante3
Nombres: Arianna Vergara
Edad: 1
Peso: 1
Direccion: daadsa
Materia: adafaf
ID Materia: 123a
Horas Materia: 23
Promedio Materia: 18
STRUCT SIMPLE
2 ESTUDIANTES ESTATICOS
La persona 1
Nombres: Justin Mendoza
Edad: 19
Peso: 1.6
Direccion: Nuevo Isrrael
La persona 2
Nombres: Andres duenas
Edad: 18
Peso: 1.4
Direccion: El carmen
2 ESTUDIANTES DINAMICOS
La persona 3
Nombres: Karina rodriguez
Edad: 32
Peso: 1.2
Direccion: asdad

Fundamentos de la Programación

The image displays two screenshots of a Windows desktop environment. The top screenshot shows a console window titled 'C:\Users\ESPE\Desktop\P3Archivos\MendozaJustin\bin\Debug\P3Archivos.exe' and a text editor window titled 'RegistroEstudiantes.txt: Bloc de notas'. The console window shows the execution of a C++ program that creates a text file named 'RegistroEstudiantes.txt'. The program prompts the user to enter student data, including name, age, weight, and address, and then saves this data to the file. The text editor window shows the contents of the file, which are organized into sections for static and dynamic student data, each followed by a 'STRUCT SIMPLE' and 'STRUCT ANIDADO' declaration.

The bottom screenshot shows the same console window and text editor window. The console window now displays the output of the program, which includes the names and details of the students entered. The text editor window shows the updated contents of the file, which now include the data for the new students entered.

Fundamentos de la Programación

```
C:\Users\ESPE\Desktop\P3ArchivosMendozaJustin\bin\Debug\P3Archivos.exe

STRUCT ANIDADO
El Estudiante1
Nombres: Emely aviles
Edad: 19
Peso: 2.36
Diereccion: adsada
Materia: adafdsa
ID Materia: 12a
Horas Materia: 23
Promedio Materia: 18
¿DESEAR VOLVER A INGRESAR EL NOMBRE EL ARCHIVO? [Sí o No]
no
no
[1] INGRESAR DATOS AL TDA - CREAR EL ARCHIVO.TXT
[2] LEER EL ARCHIVO.TXT
[3] SALIR
SELECCIONA UNA OPCION DEL MENU:
2
=====
|| CREAR UN ARCHIVO.TXT ||
=====
INGRESA EL NOMBRE DEL ARCHIVO QUE DESEA ABRIR (SIN EL .TXT):
dsadsad
ERROR, NO SE PUEDE ABRIR EL ARCHIVO SOLO LECTURA..
¿DESEAR VOLVER A INGRESAR EL NOMBRE EL ARCHIVO? [Sí o No]
te
te
VUELVA A INGRESAR [Sí o No]
no
[1] INGRESAR DATOS AL TDA - CREAR EL ARCHIVO.TXT
[2] LEER EL ARCHIVO.TXT
[3] SALIR
SELECCIONA UNA OPCION DEL MENU:
-
```