



UNIVERSIDAD DE GUADALAJARA

CUCEI

DIVISIÓN DE TECNOLOGÍAS PARA LA INTEGRACIÓN CIBER-HUMANA

DEPARTAMENTO DE CIENCIAS COMPUTACIONALES

PRÁCTICA No. 4

TEMA: Archivos de Acceso Aleatorio (Sistema completo)

EQUIPO No. 4

INTEGRANTES:

Munguía Guízar Marlon Uriel

Padilla Perez Jorge Daray

Pérez Palacios Perla Michelle

Pulido Tobías Rafael Agustín

SEMINARIO DE SOLUCIÓN DE PROBLEMAS DE ESTRUCTURAS DE DATOS

II | SECCIÓN D19 | 2022B

PROF. Mariscal Lugo Luis Felipe

MARCO TEÓRICO

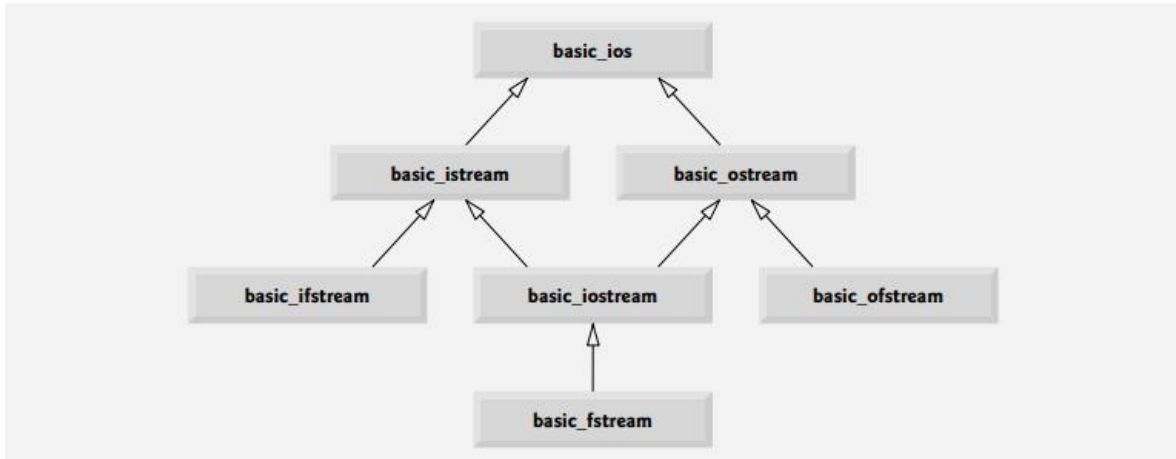


Figura 17.3 | Porción de la jerarquía de plantillas de E/S de flujos.

Bibliografía.

Deitel., H. M. & Deitel., P. J. (2008). *C++: cómo programar* (6.^a ed.). Pearson.

Código:

Main:

```
1  /*Equipo 4:
2     Munguia Guizar Marlon Uriel
3     Padilla Perez Jorge Daray
4     Perez Palacios Perla Michelle
5     Pulido Tobias Rafael Agustin
6     Seccion: D19
7     Calendario: 2022B
8  */
9  #include <iostream>
10 #include<iomanip>
11 #include<fstream>
12 #include<cstdlib>
13 #include "deportista.h"
14
15 using std::fstream;
16
17 using namespace std;
18 void menu();
19 void altas(fstream&);
20 void bajas(fstream&, fstream&);
21 void cambios(fstream&, fstream&);
22 void consultas(fstream&,int);
23 void imprimirLinea( ostream &salida, const Deportista &registro );
24 Deportista d;//un objeto o registro Deportista
25 Deportista blanco;//un objeto o registro Deportista
26
27 int main()
28 {
29     menu();
30     return 0;
31 }
32
33 void altas(fstream&archDeportSalida )
34 {
35     int numeroDeSocio = 1;
36     char nombre[15];
37     int edad;
38     char deporte[15];
39     while(numeroDeSocio != 0)
40     {
41         archDeportSalida.clear();
42         archDeportSalida.seekg(0, archDeportSalida.beg);
43         system("cls");
44         cout<<"Escriba el numero de socio(De 1 a 100, 0 para terminar la
entrada)\n?";
45         cin>>numeroDeSocio;
46         d.establecerNumeroDeSocio(numeroDeSocio);
47         //buscar la posicion en el archivo de registro especificado por el
usuario
48         if(d.obtenerNumeroDeSocio()>=1 && d.obtenerNumeroDeSocio()<=100)
49         {
50             archDeportSalida.seekp((d.obtenerNumeroDeSocio()-
1)*sizeof(Deportista),ios::beg); //seekp(n, ios::beg);
51             // lee el siguiente registro del archivo
52             archDeportSalida.read( reinterpret_cast< char * >( &d ), sizeof(
Deportista ) );
53             if ( d.obtenerNumeroDeSocio() != 0)
54             {
55                 cout << left << setw( 10 ) << "Num socio" << setw( 16 )
<< "Nombre" << setw( 11 ) << "Edad" << left
56                 << setw( 10 ) << right << "Deporte" << endl;
57                 imprimirLinea( cout, d);
58                 cout << "\n\n Numero de socio existente pruebe con otro"<< endl;
59                 system("pause");
60             }
61         }
62         else
63         {
```

Equipo 4 | Practica 4: Archivos de Acceso Aleatorio (Sistema completo)

```
64         d.establecerNumeroDeSocio(numeroDeSocio);
65     cout << "EL NUMERO DE SOCIO NO EXISTE"<<endl;
66     cout<<"Teclea nombre, edad y deporte\n?";
67     cin>>setw(15)>>nombre;
68     cin>>edad;
69     cin>>setw(15)>>deporte;
70     //establecer los valores nombre, edad y deporte del registro
71     d.establecerNombre(nombre);
72     d.establecerEdad(edad);
73     d.establecerDeporte(deporte);
74     //buscar la posicion en el archivo de registro especificado por
el usuario
75     archDeportSalida.seekp((d.obtenerNumeroDeSocio()-
1)*sizeof(Deportista)); //seekp(n, ios::beg);
76     //escribir la informaci3n especificada por el usuario en el
archivo
77     archDeportSalida.write(reinterpret_cast<const char
*>(&d),sizeof(Deportista));
78     cout << "EL NUMERO DE SOCIO SE AGREGO CORRECTAMENTE"<<endl;
79     system("pause");
80 } //while
81 }
82 }
83 //El usuario proporciona la informaci3n que se guarda en el archivo
84 archDeportSalida.clear();
85 } //main
86
87 void bajas(fstream&archDeportSalida, fstream&archdeportEntrada)
88 {
89     archdeportEntrada.clear();
90     archdeportEntrada.seekg(0, archdeportEntrada.beg);
91     int numero;
92     int elim;
93     cout << "Dame el numero de socio: "<<endl;
94     cin >> numero;
95     d.establecerNumeroDeSocio(numero);
96     archdeportEntrada.seekg((d.obtenerNumeroDeSocio()-
1)*sizeof(Deportista),ios::beg); //seekp(n, ios::beg);
97     // lee el siguiente registro del archivo
98     archdeportEntrada.read( reinterpret_cast< char * >( &d ), sizeof(
Deportista ) );
99     if ( d.obtenerNumeroDeSocio() != 0)
100     {
101         cout << left << setw( 10 ) << "Num socio" << setw( 16 )
102         << "Nombre" << setw( 11 ) << "Edad" << left
103         << setw( 10 ) << right << "Deporte" << endl;
104         imprimirLinea( cout, d);
105         cout << "Desea eliminar el socio ? '1' si, '0' no" << endl;
106         cin >> elim;
107         if (elim == 1)
108         {
109             archDeportSalida.clear();
110             archDeportSalida.seekp(0, archDeportSalida.beg);
111             d.establecerNumeroDeSocio(numero);
112             //buscar la posicion en el archivo de registro especificado por
el usuario
113             archDeportSalida.seekp((d.obtenerNumeroDeSocio()-
1)*sizeof(Deportista)); //seekp(n, ios::beg);
114             //escribir la informaci3n especificada por el usuario en el
archivo
115             archDeportSalida.write(reinterpret_cast<const char
*>(&blanco),sizeof(Deportista));
116             cout << "Borrado corectamente" << endl;
117             system("pause");
118         }
119     else
120     {
121         cout << "Regresando al menu..." << endl;
122         system("pause");
123     }
124 }
```

```

125         else
126         {
127             cout << "EL NUMERO DE SOCIO NO EXISTE"<<endl;
128             system("pause");
129         }
130     archdeportEntrada.clear();
131 }
132
133 void cambios(fstream&archDeportSalida, fstream&archdeportEntrada)
134 {
135     archdeportEntrada.clear();
136     archdeportEntrada.seekg(0, archdeportEntrada.beg);
137     int numero;
138     char nombre[15];
139     int edad;
140     char deporte[15];
141     int elim;
142     cout << "Dame el numero de socio: "<<endl;
143     cin >> numero;
144     d.establecerNumeroDeSocio(numero);
145     archdeportEntrada.seekg((d.obtenerNumeroDeSocio()-
146 1)*sizeof(Deportista),ios::beg); //seekp(n, ios::beg);
147     // lee el siguiente registro del archivo
148     archdeportEntrada.read( reinterpret_cast< char * >(&d), sizeof(
149 Deportista ) );
150     if ( d.obtenerNumeroDeSocio() != 0)
151     {
152         cout << left << setw( 10 ) << "Num socio" << setw( 16 )
153         << "Nombre" << setw( 11 ) << "Edad" << left
154         << setw( 10 ) << right << "Deporte" << endl;
155         imprimirLinea( cout, d);
156         cout << "Desea cambiar el socio ? '1' si, '0' no" << endl;
157         cin >> elim;
158         if (elim == 1)
159         {
160             archDeportSalida.clear();
161             archDeportSalida.seekp(0, archDeportSalida.beg);
162             d.establecerNumeroDeSocio(numero);
163             if (d.obtenerNumeroDeSocio()>=1 &&
164 d.obtenerNumeroDeSocio()<=100){
165                 cout<<"Teclea nombre, edad y deporte\n?";
166                 cin>>setw(15)>>nombre;
167                 cin>>edad;
168                 cin>>setw(15)>>deporte;
169                 //establecer los valores nombre, edad y deporte del registro
170                 d.establecerNombre(nombre);
171                 d.establecerEdad(edad);
172                 d.establecerDeporte(deporte);
173                 //buscar la posicion en el archivo de registro especificado
174                 archDeportSalida.seekp((d.obtenerNumeroDeSocio()-
175 1)*sizeof(Deportista)); //seekp(n, ios::beg);
176                 //escribir la informaci3n especificada por el usuario en el
177                 archivo
178                 archDeportSalida.write(reinterpret_cast<const char
179 *>(&d),sizeof(Deportista));
180                 cout << "Cambiado correctamente" << endl;
181                 system("pause");
182                 archDeportSalida.clear();
183             }
184             else
185             {
186                 cout << "Numero de socio invalido..."<< endl;
187             }
188         }
189     }
190     else
191     {
192         cout << "Regresando al menu..." << endl;
193         system("pause");
194     }
195 }

```

```

189         else
190         {
191             cout << "EL NUMERO DE SOCIO NO EXISTE"<<endl;
192             system("pause");
193         }
194     archdeportEntrada.clear();
195 }
196
197 void consultas(fstream&archdeportEntrada, int opc)
198 {
199     archdeportEntrada.clear();
200     archdeportEntrada.seekg(0, archdeportEntrada.beg);
201     switch(opc)
202     {
203     case 4:
204     {
205         system("cls");
206         cout << "Consulta individual" << endl;
207         int numero;
208         cout << "Dame el numero de socio: "<<endl;
209         cin >> numero;
210         d.establecerNumeroDeSocio(numero);
211         archdeportEntrada.seekg((d.obtenerNumeroDeSocio()-
1)*sizeof(Deportista), ios::beg); //seekp(n, ios::beg);
212         // lee el siguiente registro del archivo
213         archdeportEntrada.read( reinterpret_cast< char * >( &d ), sizeof(
Deportista ) );
214         if ( d.obtenerNumeroDeSocio() != 0)
215         {
216             cout << left << setw( 10 ) << "Num socio" << setw( 16 )
<< "Nombre" << setw( 11 ) << "Edad" << left
217             << setw( 10 ) << right << "Deporte" << endl;
218             imprimirLinea( cout, d);
219         }
220     }
221     else
222     {
223         cout << "EL NUMERO DE SOCIO NO EXISTE"<<endl;
224     }
225     system("pause");
226     break;
227 }
228 case 5:
229 {
230     system("cls");
231     cout << "Consulta general" << endl;
232     cout << left << setw( 10 ) << "Num socio" << setw( 16 )
<< "Nombre" << setw( 11 ) << "Edad" << left
233     << setw( 10 ) << right << "Deporte" << endl;
234     // lee el primer registro del archivo
235     archdeportEntrada.read( reinterpret_cast< char * >( &d ), sizeof(
Deportista ) );
236     // lee todos los registros del archivo
237     while ( archdeportEntrada && !archdeportEntrada.eof() )
238     {
239         // muestra un registro
240         if ( d.obtenerNumeroDeSocio() != 0)
241         {
242             imprimirLinea( cout, d);
243         }
244         // lee el siguiente registro del archivo
245         archdeportEntrada.read( reinterpret_cast< char * >( &d ), sizeof(
Deportista ) );
246     } // fin de while
247     system("pause");
248 }
249 break;
250 }
251 archdeportEntrada.clear();
252 } // fin de consultas
253
254 // muestra un solo registro

```

```

256     void imprimirLinea( ostream &salida, const Deportista &registro )
257     {
258         salida << left << setw( 10 ) << registro.obtenerNumeroDeSocio()
259         << setw( 16 ) << registro.obtenerNombre()
260         << setw( 11 ) << registro.obtenerEdad()
261         << setw( 10 ) << setprecision( 2 ) << right << fixed
262         << showpoint << registro.obtenerDeporte() << endl;
263     } // fin de la función imprimirLinea
264
265     void menu()
266     {
267         fstream archDeportENSA("deportistas.dat", ios::out | ios::binary | ios::in );
268         if ( !archDeportENSA)
269         {
270             cerr << "No se pudo abrir el archivo ensa." << endl;
271             exit( 1 );
272         } // fin de if
273
274         int opc;
275         do
276         {
277             system("cls");
278             cout << "1) altas" << endl;
279             cout << "2) bajas" << endl;
280             cout << "3) cambios" << endl;
281             cout << "4) Consulta individual (numero de socio)" << endl;
282             cout << "5) Consultas generales" << endl;
283             cout << "6) Salir" << endl;
284             cout << "Ingresar opcion: " << endl;
285             cin >> opc;
286             switch(opc)
287             {
288                 case 1:
289                 {
290                     altas(archDeportENSA);
291                     break;
292                 }
293                 case 2:
294                 {
295                     bajas(archDeportENSA, archDeportENSA);
296                     break;
297                 }
298                 case 3:
299                 {
300                     cambios(archDeportENSA, archDeportENSA);
301                     break;
302                 }
303                 case 4:
304                 {
305                     consultas(archDeportENSA, opc);
306                     break;
307                 }
308                 case 5:
309                 {
310                     consultas(archDeportENSA, opc);
311                     break;
312                 }
313                 case 6:
314                 {
315                     cout << "saliendo..." << endl;
316                     archDeportENSA.close();
317                     exit(1);
318                     break;
319                 }
320             }
321         } while (opc!=6);
322     }
323 }

```

Deportista.h:

```
1  /*Equipo 4:
2  Munguãa Guãzar Marlon Uriel
3  Padilla Perez Jorge Daray
4  Pãrez Palacios Perla Michelle
5  Pulido Tobãas Rafael Agustãn
6  Secciã³n: D19
7  Calendario: 2022B
8  */
9  #ifndef DEPORTISTA_H
10 #define DEPORTISTA_H
11 #include <cstring>
12 #include <string>
13
14 using namespace std;
15 using std::string;
16
17
18 class Deportista
19 {
20     public:
21         //constructor predeterminado
22         Deportista(int = 0, string = "", int = 0, string = "");
23
24         //funciones de acceso para NumeroDeSocio
25         void establecerNumeroDeSocio(int);
26         int obtenerNumeroDeSocio() const;
27
28         //funciones de acceso para nombre
29         void establecerNombre(string);
30         string obtenerNombre() const;
31
32         //funciones de acceso para edad
33         void establecerEdad(int);
34         int obtenerEdad() const;
35
36         //funciones de acceso para deporte
37         void establecerDeporte(string);
38         En obtenerDeporte() const;
39
40     protected:
41
42     private:
43         int numeroDeSocio;
44         char nombre[15];
45         int edad;
46         char deporte[15];
47
48 }; //Fin de la clase Deportista
49
50
51 #endif // DEPORTISTA_H
```


DeportistaCPP:

```
1  /*Equipo 4:
2  Munguãa Guãzar Marlon Uriel
3  Padilla Perez Jorge Daray
4  Pãrez Palacios Perla Michelle
5  Pulido Tobãas Rafael Agustãn
6  Secciã'n: D19
7  Calendario: 2022B
8  */
9  #include "Deportista.h"
10 #include <cstring>
11
12 using namespace std;
13
14 //Constructor
15 Deportista::Deportista(int valorNumeroDeSocio, string valorNombre, int valorEdad, string
valorDeporte)
16 {
17     //ctor
18     establecerNumeroDeSocio(valorNumeroDeSocio);
19     establecerNombre(valorNombre);
20     establecerEdad(valorEdad);
21     establecerDeporte(valorDeporte);
22 } //fin constructor
23
24 //Obtiene el valor del nãmero de socio
25 int Deportista::obtenerNumeroDeSocio() const{
26     return numeroDeSocio;
27 } //fin int obtenerNumeroDeSocio
28
29 //Establece el valor del nãmero de Socio
30 void Deportista::establecerNumeroDeSocio(int valorNumeroDeSocio){
31     numeroDeSocio=valorNumeroDeSocio;
32 } //fin void establecerNumeroDeSocio
33
34 //Obtiene el valor del nombre
35 string Deportista::obtenerNombre() const{
36     return nombre;
37 } //fin string obtenerNombre
38
39 //Establece el valor del nombre
40 void Deportista::establecerNombre(string cadenaNombre){
41     const char*valorNombre = cadenaNombre.data();
42     int longitud = cadenaNombre.size();
43     longitud = (longitud < 15 ? longitud : 14);
44     strncpy(nombre, valorNombre, longitud);
45     nombre[longitud] = '\\0';
46 } //fin void establecerNombre
47
48 //Obtiene el valor de la edad
49 int Deportista::obtenerEdad() const{
50     return edad;
51 } //fin int obtenerEdad
52
53 //Establece el valor de la edad
54 void Deportista::establecerEdad(int valorEdad){
55     edad=valorEdad;
56 } //fin void establecerEdad
57
58
59 //Obtiene el valor del deporte
60 string Deportista::obtenerDeporte() const{
61     return deporte;
62 } //fin string obtenerDeporte
63
64 //Establece el valor del deporte
65 void Deportista::establecerDeporte(string cadenaDeporte){
66     const char*valorDeporte = cadenaDeporte.data();
67     int longitud = cadenaDeporte.size();
```

```
68     longitud = (longitud < 15 ? longitud : 14);
69     strncpy(deporte, valorDeporte, longitud);
70     deporte[longitud] = '\\0';
71 } //fin void establecerDeporte
```

imágenes del archivo físico:



deportistas.dat: Bloc de notas

Archivo	Edición	Formato	Ver	Ayuda
1	Sayde		1	Natación
	Rafael co		1	Volibol



deportistas.dat: Bloc de notas

Archivo	Edición	Formato	Ver	Ayuda
	1	Antonio o	,	Basquetbol
			1	Alfredo o
			8	KickBoxing



deportistas.dat: Bloc de notas

Archivo	Edición	Formato	Ver	Ayuda
---------	---------	---------	-----	-------

L Luis - Futbol



deportistas.dat: Bloc de notas

Archivo	Edición	Formato	Ver	Ayuda
---------	---------	---------	-----	-------

P Josue o o & Atletismo

Capturas:

PASO 1: Elegir la opción 5 (consultas generales)

```
"D:\SEDA 2\practicas\Practica4\bin\Debug\Practica4.exe"
Consulta general
Num socio Nombre      Edad      Deporte
1          Sayde       16        Natación
5          Francisco   42        Fútbol
10         Rafael      17        Vólibol
15         Alfredo     56        KickBoxing
80         Josue       38        Atletismo
90         Arturo      67        Beisbol
Presione una tecla para continuar . . .
```

PASO 2: Elegir la opción 4 (consultas individuales) (número de socio = 1)

```
"D:\SEDA 2\practicas\Practica4\bin\Debug\Practica4.exe"
Consulta individual
Dame el numero de socio:
1
Num socio Nombre      Edad      Deporte
1          Sayde       16        Natación
Presione una tecla para continuar . . .
```

PASO 3: Elegir la opción 4 (consultas individuales) (número de socio = 2)

```
"D:\SEDA 2\practicas\Practica4\bin\Debug\Practica4.exe"
Consulta individual
Dame el numero de socio:
2
EL NUMERO DE SOCIO NO EXISTE
Presione una tecla para continuar . . .
```

PASO 4: Elegir la opción 3 (cambios) (número de socio =5) (nombre="Antonio", edad=44, deporte="Basquetbol")

```
"D:\SEDA 2\practicass\Practica4\bin\Debug\Practica4.exe"
1) altas
2) bajas
3) cambios
4) Consulta individual (numero de socio)
5) Consultas generales
6) Salir
Ingresar opcion:
3
Dame el numero de socio:
5
Num socio Nombre      Edad      Deporte
5      Francisco      42      Futbol
Desea cambiar el socio ? '1' si, '0' no
1
Teclea nombre, edad y deporte
?Antonio 44 Basquetbol
Cambiado correctamente
Presione una tecla para continuar . . .
```

PASO 5: Elegir la opción 3 (cambios) (número de socio =6)

```
"D:\SEDA 2\practicass\Practica4\bin\Debug\Practica4.exe"
1) altas
2) bajas
3) cambios
4) Consulta individual (numero de socio)
5) Consultas generales
6) Salir
Ingresar opcion:
3
Dame el numero de socio:
6
EL NUMERO DE SOCIO NO EXISTE
Presione una tecla para continuar . . .
```

PASO 6: Elegir la opción 2 (bajas) (número de socio =90) (respuesta a la pregunta de confirmación = SI)

```
"D:\SEDA 2\practicass\Practica4\bin\Debug\Practica4.exe"
1) altas
2) bajas
3) cambios
4) Consulta individual (numero de socio)
5) Consultas generales
6) Salir
Ingresar opcion:
2
Dame el numero de socio:
90
Num socio Nombre      Edad      Deporte
90      Arturo      67      Beisbol
Desea eliminar el socio ? '1' si, '0' no
1
Borrado correctamente
Presione una tecla para continuar . . .
```

PASO 7: Elegir la opción 2 (bajas) (número de socio =90)

```
"D:\SEDA 2\practicass\Practica4\bin\Debug\Practica4.exe"
1) altas
2) bajas
3) cambios
4) Consulta individual (numero de socio)
5) Consultas generales
6) Salir
Ingresar opcion:
2
Dame el numero de socio:
90
EL NUMERO DE SOCIO NO EXISTE
Presione una tecla para continuar . . .
```

Equipo 4 | Practica 4: Archivos de Acceso Aleatorio (Sistema completo)

PASO 8: Elegir la opción 1 (altas) (número de socio =76, nombre ="Luis", edad =45, deporte ="Futbol")

```
"D:\SEDA 2\practicas\Practica4\bin\Debug\Practica4.exe"
Escriba el numero de socio(De 1 a 100, 0 para terminar la entrada)
?76
EL NUMERO DE SOCIO NO EXISTE
Teclea nombre, edad y deporte
?Luis 45 Futbol
EL NUMERO DE SOCIO SE AGREGO CORRECTAMENTE
Presione una tecla para continuar . . .
```

PASO 9: Elegir la opción 1 (altas) (número de socio =76)

```
"D:\SEDA 2\practicas\Practica4\bin\Debug\Practica4.exe"
Escriba el numero de socio(De 1 a 100, 0 para terminar la entrada)
?76
Num socio Nombre      Edad      Deporte
76      Luis      45      Futbol

Numero de socio existente pruebe con otro
Presione una tecla para continuar . . .
```

PASO 10: Elegir la opción 5 (consultas generales)

```
"D:\SEDA 2\practicas\Practica4\bin\Debug\Practica4.exe"
Consulta general
Num socio Nombre      Edad      Deporte
1      Sayde      16      Natación
5      Antonio      44      Basquetbol
10     Rafael      17      Volibol
15     Alfredo      56      KickBoxing
76     Luis      45      Futbol
80     Josue      38      Atletismo
Presione una tecla para continuar . . .
```

Conclusiones:

- **Munguía Guízar Marlon Uriel.**
 - ✓ Para esta práctica se utilizaron 2 funciones mas a parte de las adquiridas por los programas anteriores a este, haciendo que el trabajo se vaya completando más fácil y adquiriendo mejores habilidades en la practica con los nuevos métodos agregados
- **Padilla Perez Jorge Daray.**
 - ✓ En esta practica se vuelve a utilizar algunas de las practicas anteriores lo cual esta muy bien, con el hecho de agregar 2 funciones más, las cuales son cambios y eliminar con numero de socio, muchísimo mas interesante que la anterior practica al igual que difícil al menos para mi a la hora de insertar con condiciones al igual que tener mas controlado al usuario final de la aplicación.
- **Pérez Palacios Perla Michelle.**
 - ✓ En conclusión, esta practica estuvo sencilla ya que solo se agrego elimina y cambios lo cual no estuvo tan pesado, además de reutilizar algunas funciones que ya teníamos
- **Pulido Tobías Rafael Agustín.**
 - ✓ En esta ocasión la practica fue relativamente sencilla gracias a que fue la culminación d las practicas anteriores por lo que a la hora de desarrollar facilito el trabajo, en mi caso la implementación de la función elimina fue un poco complicada, pero en general con la utilización de las funciones vistas en las practicas anteriores la logre entender.