

```
1  /*
2   * prueba del adt Alumno
3   */
4
5
6  #include <stdlib.h>
7  #include <stdio.h>
8  #include <string.h>
9  #include <assert.h>
10
11
12  struct Alumno_Type
13  {
14      char nombre[32];
15      int semestre;
16      float promedio;
17  };
18
19  typedef struct Alumno_Type Alumno;
20
21
22  Alumno* Alumno_Create ()
23  {
24      Alumno* n = (Alumno*) malloc (sizeof (Alumno));
25      if (n) {
26          strcpy (n->nombre, "N/A");
27          n->semestre = 1;
28          n->promedio = 0.0;
29      }
30      return n;
31  }
32
33  Alumno* Alumno_CreateWithData (char* nombre, int semestre, float promedio)
34  {
35      Alumno* n = (Alumno*) malloc (sizeof (Alumno));
36      if (n) {
37          strcpy (n->nombre, nombre);
38          n->semestre = semestre;
39          n->promedio = promedio;
40      }
41      return n;
42  }
43
44  void Alumno_Destroy (Alumno* this)
45  {
46      if (this) { free (this); }
47  }
48
49  void Alumno_Print (Alumno* this)
50  {
51      printf ("%s, %d, %.2f\n", this->nombre, this->semestre, this->promedio);
52  }
53
54  void Alumno_SetNombre (Alumno* this, char* nombre)
55  {
56      strcpy (this->nombre, nombre);
57  }
58
59  void Alumno_SetPromedio (Alumno* this, float promedio)
60  {
61      this->promedio = promedio;
62  }
63
64  void Alumno_SetSemestre (Alumno* this, int semestre)
65  {
66      this->semestre = semestre;
67  }
68
69  char* Alumno_GetNombre (Alumno* this)
70  {
```

```
71     return this->nombre;
72 }
73
74 float Alumno_GetPromedio (Alumno* this)
75 {
76     return this->promedio;
77 }
78
79 int Alumno_GetSemestre (Alumno* this)
80 {
81     return this->semestre;
82 }
83
84
85
86 //-----
87 //  Driver program
88 //-----
89 int main(void)
90 {
91     Alumno* unAlumno = Alumno_Create ();
92     assert (unAlumno);
93
94     Alumno* otroAlumno = Alumno_CreateWithData ("Javier", 10, 10.0);
95     assert (otroAlumno);
96
97     Alumno_Print (unAlumno);
98
99     Alumno_Print (otroAlumno);
100
101     Alumno_Destroy (otroAlumno);
102     Alumno_Destroy (unAlumno);
103
104     return 0;
105 }
106
107
```