

```
1
2  #ifndef  ALUMNO_INC
3  #define  ALUMNO_INC
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* Alumno_CreateWithData (char* nombre, int semestre, float promedio);
25
26  void Alumno_Destroy (Alumno* this);
27
28  void Alumno_Print (Alumno* this);
29
30  void Alumno_SetNombre (Alumno* this, char* nombre);
31
32  void Alumno_SetPromedio (Alumno* this, float promedio);
33
34  void Alumno_SetSemestre (Alumno* this, int semestre);
35
36  char* Alumno_GetNombre (Alumno* this);
37
38  float Alumno_GetPromedio (Alumno* this);
39
40  int Alumno_GetSemestre (Alumno* this);
41
42  /*-----*/
43  /**
44   * @brief Asigna un alumno en otro. Su equivalente de alto nivel es:
45   *
46   * this := other
47   *
48   * @param this Es el destino de los datos
49   * @param other Es el origen de los datos
50   *
51   * @pre Ambos objetos existen.
52   */
53  /*-----*/
54  void Alumno_Assign (Alumno* this, Alumno* other);
55
56
57  #endif  /* ----- #ifndef ALUMNO_INC ----- */
58
```