

```

1  /*****
2  Nombre:      p3f11.c
3  Autor:      Programación I
4  Fecha:      31-10-19
5  ACCIONES A REALIZAR:
6
7  Codificar las funciones que se indican a continuación correspondientes a la lectura y
8  escritura de las notas de los 4 parciales e invocarlas desde el programa principal,
9  sustituyendo la parte de código relacionada con estas acciones, codificada en la fase 1.
10
11  *****/
12  #include<stdio.h>
13  #define NOTAS 4
14  #define NOTAS_inf 0.0
15  #define NOTAS_sup 10.0
16
17  /*****
18  FASE 2
19  *****/
20
21  /*
22  * Función: leerRealEnRango
23  * Lee de la entrada estándar un número real en el rango [rangoInf, rangoSup].
24  * Si se teclea un número fuera de rango lo indica y vuelve a solicitar un nuevo valor.
25  * Parámetros de entrada:
26  * rangoInf: real, rango inferior del número a leer
27  * rangoSup: real, rango superior del número a leer
28  * Precondiciones: ninguna
29  * Parámetros de salida pasados por referencia: ninguno
30  * Valor devuelto por la función: número real leído por la entrada estándar (teclado)
31  */
32  double leerRealEnRango (double rangoInf, double rangoSup);
33
34  /*
35  * Función: leerNotasParciales
36  * Lee de la entrada estándar las notas de los 4 parciales de la asignatura,
37  * indicando el número de cada parcial (del 1 al 4)
38  * Parámetros de entrada: ninguno
39  * Parámetros de salida pasados por referencia:
40  * notas: array de reales con las notas de los 4 parciales
41  * Valor devuelto por la función: ninguno
42  */
43  void leerNotasParciales (double notas[]);
44
45  /*
46  * Función: escribirNotasParciales
47  * Escribe en la salida estándar las notas de los 4 parciales de la asignatura,
48  * indicando el número de cada parcial (del 1 al 4)
49  * Parámetros de entrada:
50  * notas: array de reales con las notas de los 4 parciales
51  * Precondiciones: notas tiene que estar inicializado
52  * Parámetros de salida pasados por referencia: ninguno
53  * Valor devuelto por la función: ninguno
54  */
55  void escribirNotasParciales (const double notas[]);
56
57  /*
58  * Función: calcularNotaFinal
59  * Calcula la nota final de la asignatura haciendo la media ponderada
60  * de los 4 parciales (15%, 15%, 25% y 45%), respectivamente
61  * Parámetros de entrada:
62  * notas: array de reales con las notas de los 4 parciales
63  * ponderación: array de enteros con la ponderación de cada parcial
64  * Precondiciones: notas y ponderación tienen que estar inicializados
65  * Parámetros de salida pasados por referencia: ninguno
66  * Valor devuelto por la función: valor real con la nota final de la asignatura
67  */
68  double calcularNotaFinal (const double notas[], const int ponderacion[]);
69
70  int main(void) {
71
72      double nota[NOTAS], notaFin;
73      int parcial;
74      int i;
75      int opc;
76      leerNotasParciales(nota);
77      escribirNotasParciales(nota);
78      const int porcentajeParcial[4]={15,15,25,45};
79
80      do{
81          printf("\n*****Opciones*****");
82          printf("\n*      1. Calcular nota final      *");
83          printf("\n*      2. Modificar nota parcial   *");
84          printf("\n*      3. Calcular rango nota final *");

```

```

85     printf("\n* 4. Salir *");
86     printf("\n*****\n");
87     printf("\nTeclee opcion: ");
88     scanf("%d", &opc);
89     fflush(stdin);
90
91     switch(opc){
92     case 1:
93         printf("Esta usted en la OPCIÓN 1: Calcular nota final.\n");
94         notaFin = calcularNotaFinal (nota, porcentajeParcial);
95         printf("La nota final es: %.2f", notaFin);
96
97         break;
98
99     case 2:
100         printf("Esta usted en la OPCIÓN 2: Modificar nota parcial.\n");
101
102         break;
103
104     case 3:
105         printf("Esta usted en la OPCIÓN 3: Calcular rango nota final.\n");
106
107         break;
108
109     case 4:
110         printf("Gracias por utilizar este programa");
111         break;
112     default:
113         printf("No es una OPCIÓN!\n");
114     }
115
116     }while (opc!=4);
117
118     return 0;
119 }
120
121 /*
122 * Función: leerRealEnRango
123 * Lee de la entrada estándar un número real en el rango [rangoInf, rangoSup].
124 * Si se teclea un número fuera de rango lo indica y vuelve a solicitar un nuevo valor.
125 * Parámetros de entrada:
126 *   rangoInf: real, rango inferior del número a leer
127 *   rangoSup: real, rango superior del número a leer
128 * Precondiciones: ninguna
129 * Parámetros de salida pasados por referencia: ninguno
130 * Valor devuelto por la función: número real leído por la entrada estándar (teclado)
131 */
132 double leerRealEnRango(double rangoInf, double rangoSup){
133
134     double num;
135
136     scanf("%lf", &num);
137     fflush(stdin);
138
139     while ((num<rangoInf) || (num>rangoSup)){
140         printf("Valor fuera de rango. Teclee nuevo valor: ");
141         scanf("%lf", &num);
142         fflush(stdin);
143     }
144     return num;
145 }
146
147 /*
148 * Función: leerNotasParciales
149 * Lee de la entrada estándar las notas de los 4 parciales de la asignatura,
150 * indicando el número de cada parcial (del 1 al 4)
151 * Parámetros de entrada:
152 *   notas: array de reales con las notas de los 4 parciales
153 * Precondiciones: notas tiene que estar inicializado
154 */
155 void leerNotasParciales (double nota[]){
156     int i;
157
158     for(i=0; i<NOTAS; i++){
159         printf("Teclee nota del parcial %d: ", i+1);
160         nota[i]=leerRealEnRango(NOTAS_inf, NOTAS_sup);
161     }
162 }
163
164 /*
165 * Función: escribirNotasParciales
166 * Escribe en la salida estándar las notas de los 4 parciales de la asignatura,
167 * indicando el número de cada parcial (del 1 al 4)
168 * Parámetros de entrada:
169 *   notas: array de reales con las notas de los 4 parciales
170 * Precondiciones: notas tiene que estar inicializado

```

```

169  * Parámetros de salida pasados por referencia: ninguno
170  * Valor devuelto por la función: ninguno
171  */
172  void escribirNotasParciales (const double notas[]){
173      int i;
174      printf("\nLas notas obtenidas en los parciales son:\n");
175
176      for(i=0;i<NOTAS;i++){
177          printf("Nota del parcial %d: %g\n",i+1,notas[i]);
178      }
179  }
180  /*****
181      Nombre:      p3f11.c
182      Autor:       Programación I
183      Fecha:       31-10-19
184      ACCIONES A REALIZAR:
185
186      Codificar las acciones correspondientes a la opción 1 del menú, Calcular nota final.
187      Para calcular la nota final será necesario declarar en main() una array constante con 4
188      valores de
189      tipo entero que se correspondan con el porcentaje que pondera cada parcial. Estos
190      valores son: 15, 15, 25 y 45.
191
192      Se deberá codificar e invocar la función que se describe a continuación, y una vez
193      obtenido el resultado
194      de la nota final, este se debe escribir por la salida estándar.
195
196      *****/
197  /*
198  * Función: calcularNotaFinal
199  * Calcula la nota final de la asignatura haciendo la media ponderada
200  * de los 4 parciales (15%, 15%, 25% y 45%), respectivamente
201  * Parámetros de entrada:
202  *   notas: array de reales con las notas de los 4 parciales
203  *   ponderación: array de enteros con la ponderación de cada parcial
204  *   Precondiciones: notas y ponderación tienen que estar inicializados
205  * Parámetros de salida pasados por referencia: ninguno
206  * Valor devuelto por la función: valor real con la nota final de la asignatura
207  */
208  double calcularNotaFinal (const double nota[], const int ponderacion[]){
209      int i;
210      double array1[NOTAS], suma=0, notaFinal;
211
212      for(i=0;i<NOTAS;i++){
213          array1[i]=nota[i]*ponderacion[i];
214      }
215      for(i=0;i<NOTAS;i++){
216          suma=suma+array1[i];
217      }
218      notaFinal=(suma/100);
219      return notaFinal;
220  }
221
222

```