

# Práctica 9 de programación en C

El paso por referencia se usa para devolver valores en los argumentos de entrada

- En la función se declara el argumento de entrada como un puntero. Para ello, se escribe el tipo de datos, el símbolo \* y el nombre de la variable puntero (en el ejemplo, las variables área y perímetro)
- Dentro de la función se modifica adecuadamente la variable a la que apunta (en el ejemplo, \*área y \*perímetro)
- En la invocación a la función se escribe en el lugar correspondiente el símbolo & y el nombre de la variable

```
#include <stdio.h>
float cuadrado(float x){
    return x*x;
}
void area_perimetro(float b, float h, float *area, float *perimetro){
    *area=b*h;
    *perimetro=2*(b+h);
}
int main(){
    float x, y, base, altura, Area, Perimetro;
    printf("x? ");
    scanf("%f",&x);
    y=cuadrado(x);
    printf("x*x=%f\n",y);
    printf("introduce la base y la altura: ");
    scanf("%f%f",&base,&altura);
    area_perimetro(base,altura,&Area,&Perimetro);
    printf("Area=%f \t Perímetro=%f\n",Area,Perimetro);
}
```

area (puntero)



\*area



El contenido de la variable area (puntero) es la dirección de la variable \*area

En la invocación se da la dirección de la variable Area, mediante el uso de & (igual que sucede con scanf)

&area



Area



## Enunciado

La práctica consistirá en completar un ejercicio con una serie de funciones.

Se desea disponer de un programa en C para la gestión de pasajeros en un avión con 20 plazas disponibles. Cada plaza del avión puede estar libre (0), ocupada por un hombre (1), una mujer (2) o un niño (3). Inicialmente todas las plazas del avión estarán libres. Para ello se definen las siguientes constantes:

```
#define LIBRE 0  
#define HOMBRE 1  
#define MUJER 2  
#define NINO 3
```

El programa mostrará un menú por pantalla con las siguientes opciones:

1. Reservar plaza
  2. Obtener asientos libres
  3. Facturación
  4. Buscar plaza doble
  5. Salir
- Seleccione una opción:

Se mostrará el menú por pantalla repetidamente hasta que el usuario elija la opción *Salir*. Para cada una de las opciones del menú se implementará una función. Algunas de éstas funciones ya se encuentran implementadas en *avion.c*.

→ Disponible en Poliformat

### Funciones

- Función **Menu**: ya implementada. Muestra el menú por pantalla. Lee la opción elegida por el usuario (asegurándose que es correcta) y retorna al programa principal la opción.
- Función **Inicializa\_Asientos**: ya implementada. Inicializa el vector de asientos del avión a LIBRE.
- Función **Reservar\_Plaza**: ya implementada. Solicita al pasajero el asiento que desea ocupar. Si dicho asiento está libre, el viajero lo ocupa y además el programa debe preguntar si el viajero es hombre, mujer o niño. La función muestra un mensaje indicando si se ha reservado la plaza o si ya está ocupada.
- Funciones **Obtener\_Asientos\_Libres**, **Calculo\_Facturacion** y **Hay\_Plaza\_Doble**: a implementar por el alumno. Se definen con más detalle posteriormente.
- **Programa principal**: parcialmente completado. Debe rellenarse con las llamadas a las funciones **Obtener\_Asientos\_libres**, **Calculo\_Facturacion** y **Hay\_Plaza\_Doble**. Además, deben escribirse los resultados de estas funciones, tal y como se especifica posteriormente.

- Función **Menu** (ya implementada): Muestra el menú por pantalla. Lee la opción elegida por el usuario (asegurándose que es correcta) y retorna a la función principal la opción.

```
26 // Menú de opciones
27 int Menu()
28 {
29     int opc;
30
31     do{
32         printf("\n");
33         printf("1. Reservar plaza\n");
34         printf("2. Obtener asientos libres\n");
35         printf("3. Facturacion\n");
36         printf("4. Buscar plaza doble\n");
37         printf("5. Salir\n");
38         printf("Seleccione una opcion: ");
39         scanf("%d",&opc);
40     }while(opc<1 || opc>5);
41
42     return opc;
43 }
```

- Función **Inicializa\_Asientos** (ya implementada): Inicializa el vector de asientos del avión a libres.

```
18 // Inicializa todos los asientos a LIBRE
19 void Inicializa_Asientos(int asientos[])
20 {
21     int i;
22     for (i=0;i<ASIENTOS;i++)
23         asientos[i]=LIBRE;
24 }
25
```

- Función **Reservar\_Plaza** (ya implementada): Solicita al pasajero el asiento que desea ocupar. Si dicho asiento está libre, el viajero lo ocupa y además el programa debe preguntar si el viajero es hombre, mujer o niño. La función muestra un mensaje indicando si se ha reservado la plaza o si ya está ocupada.

```
45 // Reservar una plaza
46 void Reservar_Plaza(int asientos[])
47 {
48     int plaza, tipo;
49     do{
50         printf("Que asiento quieres? Entre 0 y %d ",ASIENTOS-1);
51         scanf("%d",&plaza);
52     }while (plaza<0 || plaza>=ASIENTOS);
53
54     if (asientos[plaza] == LIBRE)
55     { // Plaza Libre -> Reservar
56         do{
57             printf("El viajero es hombre=%d, mujer=%d, ni~no=%d?\n",HOMBRE,MUJER,NINO);
58             scanf("%d",&tipo);
59         }while (tipo != HOMBRE && tipo != MUJER && tipo != NINO);
60
61         asientos[plaza] = tipo;
62         printf("La plaza %d ha quedado reservada\n",plaza);
63     }
64     else
65         printf("Lo sentimos. La plaza %d ya esta reservada\n",plaza);
66 }
67
```



## avion.c (disponible en Poliformat)

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 #define TRUE 1
5 #define FALSE 0
6
7 #define ASIENTOS 20
8
9 #define LIBRE 0
10 #define HOMBRE 1
11 #define MUJER 2
12 #define NINO 3
13
14 #define PRIMER_ASIENTO_PRIMCLASE 0
15 #define PRIMER_ASIENTO_BUSINESS 4
16 #define PRIMER_ASIENTO_TURISTA 11
17
18 // Inicializa todos los asientos a LIBRE
19 void Inicializa_Asientos(int asientos[])
20 {
21     int i;
22     for (i=0;i<ASIENTOS;i++)
23         asientos[i]=LIBRE;
24 }
25
26 // Menú de opciones
27 int Menu()
28 {
29     int opc;
30
31     do{
32         printf("\n");
33         printf("1. Reservar plaza\n");
34         printf("2. Obtener asientos libres\n");
35         printf("3. Facturación\n");
36         printf("4. Buscar plaza doble\n");
37         printf("5. Salir\n");
38         printf("Seleccione una opcion: ");
39         scanf("%d",&opc);
40     }while(opc<1 || opc>5);
41
42     return opc;
43 }
```

```
45 // Reservar una plaza
46 void Reservar_Plaza(int asientos[])
47 {
48     int plaza, tipo;
49     do{
50         printf("Que asiento quieres? Entre 0 y %d ",ASIENTOS-1);
51         scanf("%d",&plaza);
52     }while (plaza<0 || plaza>=ASIENTOS);
53
54     if (asientos[plaza] == LIBRE)
55     { // Plaza Libre -> Reservar
56         do{
57             printf("El viajero es hombre=%d, mujer=%d, ni~no=%d?\n",HOMBRE,MUJER,NINO);
58             scanf("%d",&tipo);
59         }while (tipo != HOMBRE && tipo != MUJER && tipo != NINO);
60
61         asientos[plaza] = tipo;
62         printf("La plaza %d ha quedado reservada\n",plaza);
63     }
64     else
65         printf("Lo sentimos. La plaza %d ya esta reservada\n",plaza);
66 }
67
68 // Obtener asientos Libres
69
70 // Cálculo de la facturación
71
72 // Buscar una plaza doble
73
```

Estas funciones deben se implementada

```
74 int main ( )
75 {
76     int opc , asientos[ASIENTOS];
77     // Nuevas variables
78
79     // Inicializamos todos los asientos a LIBRE
80     Inicializa_Asientos(asientos);
81
82     do{
83         opc = Menu();
84         switch(opc)
85         {
86             case 1:
87                 // Llamada a la función reservar plaza
88                 Reservar_Plaza(asientos);
89                 break;
90
91             case 2:
92                 // Llamada a la función Obtener_asientos_Libres
93
94                 // Escribir los asientos Libres
95
96                 // Escribir el porcentaje de ocupación del avión
97
98                 break;
99
100             case 3:
101                 // Llamada a la función Calculo_facturacion
102
103                 // Escribir las facturaciones
104
105                 break;
106
107             case 4:
108                 // Llamada a la función Hay_plaza_doble
109
110                 // Escribir las plazas dobles o mensaje de error
111
112                 break;
113         }
114     }while (opc != 5);
115
116     return 0;
117 }
118 }
```

- La función **main** debe rellenarse con las llamadas a las funciones **Obtener\_Asientos\_libres**, **Calculo\_Facturacion** y **Hay\_Plaza\_Doble**.
- Además, deben escribirse los resultados de estas funciones, tal y como se especifica posteriormente.
- La estructura del `main()` de este programa es semejante a la estructura del `main()` del programa que debéis desarrollar en la práctica 8

# Funciones a implementar

1. Escribir una función **Obtener\_Asientos\_Libres** que obtenga los asientos libres del avión. Para ello se le pasará como parámetro:
  - a) El vector de asientos.
  - b) Un nuevo vector *asientos\_libres* que se rellenará con los números de los asientos libres del avión.
  - c) Un parámetro por referencia *num\_asientos* donde se devolverá el número de asientos que hay libres.

**Ejemplo:** si los asientos ocupados del avión fuesen 1, 3, 5, 7, 9, 19, el vector de asientos libres contendría: 0, 2, 4, 6, 8, 10, 11, 12, 13, 14, 15, 16, 17, 18 y *num\_asientos*=14.

El programa principal mostrará por pantalla las plazas libres de las que dispone el avión.

```
void Obtener_Asientos_Libres(int asientos[],int aientos_libres[], int *num_libres)
```

# Funciones a implementar

2. Escribir una función **Calculo\_Facturacion** que obtenga la facturación de las plazas ocupadas del avión.

En nuestro avión hay tres tipos de plazas: primera, bussiness y turista. Los asientos 0 al 3 corresponden a primera clase. Los asientos 4 al 10 a clase bussines y el resto a clase turista.

La función solicitará al usuario el importe del billete para cada uno de los tipos de plazas. Posteriormente, calculará la facturación por cada tipo de plaza y la facturación total. La función devolverá como parámetros por referencia la facturación por cada tipo de plaza (facturación de primera, facturación de bussines y facturación de turista) y devolverá como valor de retorno de la función la facturación total.

La función utilizará las siguientes constantes:

```
#define PRIMER_ASIENTO_PRIMCLASE 0
#define PRIMER_ASIENTO_BUSSINES 4
#define PRIMER_ASIENTO_TURISTA 11
```

El programa principal mostrará por pantalla la facturación total y la facturación de cada tipo de plaza.

```
float Calculo_Facturacion(int asientos[], float *facturacion_primera,
    float *facturacion_turista, float *facturacion_bussines)
```

# Funciones a implementar

3. Escribir una función **Hay\_Plaza\_Doble** que indique si hay dos plazas libre consecutivas en el mismo lado del pasillo. Nuestro avión está distribuido en cuatro columnas, dos a la izquierda del pasillo y dos a la derecha, de la siguiente forma:

```
0  1 --- 2  3
4  5 --- 6  7
...
```

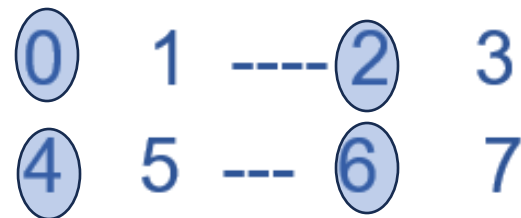
Ésta función debe indicar si se han encontrado dos asientos libres consecutivos en la misma fila y al mismo lado del pasillo. Por ejemplo, si el asiento 0 y 3 estuviesen ocupados, los dos primero asientos libres serían el 4 y 5. La función retornará **TRUE** si se han encontrado dos asientos libres consecutivos o **FALSE** en caso contrario. Además, devolverá, como parámetros por referencia, los números de los asientos.

El programa principal será el encargado de mostrar un mensaje indicando si se han encontrado o no las plazas y, si se han encontrado, los números de los asientos.

```
int Hay_Plaza_Doble (int asientos[], int *plaza_num_1, int *plaza_num_2)
```

# Funciones a implementar

- Para implementar adecuadamente la función debes tener en cuenta que para la determinación de si hay dos asientos libres en la misma parte del avión y cuales son, debes recorrer los índices de asientos pares ( $i = 0, 2, 4, \dots$ ) y comprobar que los asientos de posiciones  $i$  e  $i+1$  están libres.



Muchas gracias  
y  
¡ a programar !

