

**GRADO EN INGENIERÍA QUÍMICA INDUSTRIAL**  
**INFORMÁTICA - CURSO 1º - 2019-2020**  
**SEGUNDA CONVOCATORIA**

Nombre:	Apellidos:	DNI:	
---------	------------	------	--

**Problema 3 (4 puntos):** Se desea informatizar el almacén de una empresa con el fin de organizar el material de una forma más eficiente.

El almacén está dividido en estanterías y cada estantería está formada por un conjunto de baldas (estantes). De esta forma, la dupla estantería-balda localiza el paquete de forma única puesto que sólo puede haber un paquete en cada balda.

Para organizar la información referente a los paquetes almacenados se emplearán las siguientes estructuras:

```
struct strBalda
{
    int altura;           // Altura de la balda
    int codPaq;           // Código que identifica al paquete ubicado en la balda
    float pesoPaq;        // Peso del paquete (si codPaq = -1, el peso se ignora)
};
struct strEstanteria
{
    int anchura;          // Ancho de la estantería (todas las baldas mismo ancho)
    float pesoMax;        // Peso máximo que soporta la estantería.
    struct strBalda balda [10];
    int numBaldas;        // Número de baldas que posee la estantería
};
struct strEstanteria almacen[100];
int numEstanterias;      // Número de estanterías que posee el almacén.
```

Cada paquete tiene un código de identificación único. En el momento en el que se guarda un paquete en el almacén, el código del paquete y su peso quedan registrados en los datos de la balda que lo guarda. Cuando una balda está vacía cuando su código de paquete (codPaq) está a -1 y, en ese caso, deberá ignorarse el peso del paquete (pesoPaq) pues puede que nuestro programa no lo haya establecido a 0.

**a) (0.75 punto)** Implemente una función que reciba por parámetros el código del paquete y la estantería en la que se encuentra o -1 si no se encuentra. Además, deberá mostrar por pantalla la estantería la balda en la que se encuentra.

**(En sucio)**

**b) (0.75 punto)** Puesto que cada estantería tiene un peso máximo propio, la suma del peso de todos los paquetes almacenados en la estantería no puede exceder de ese peso máximo. En base a ello, realice una función que reciba por parámetros la posición de la estantería en el vector *almacen* y que devuelva el peso que aún es capaz de soportar la estantería teniendo en cuenta si cada balda almacena o no un paquete.

**Por ejemplo:** Una estantería con un peso máximo de 100.0 Kg, un paquete de 10.0 Kg y otro de 30.0 Kg, devolverá 60.

**(En sucio)**

c) (1 punto) *Realice una función que muestre las estadísticas de ocupación del almacén.* La función deberá mostrar por pantalla:

- El número de estanterías totalmente ocupadas (todas sus baldas ocupadas).
- El número de estanterías totalmente vacías (todas sus baldas disponibles).

**(En sucio)**

d) **(1.5 punto)** Realice una función para registrar un nuevo paquete en el almacén. Para ello deberá buscar un hueco que cumpla con las exigencias de peso, altura y anchura del paquete, pues el paquete no deberá provocar que se rebase el peso máximo de la estantería, ni su anchura podrá ser mayor que la anchura de la estantería y ni su altura superior a la altura de la balda. Una vez encontrado el hueco, deberá actualizar la información de la balda con el código del paquete y su peso.

La función devolverá el número de la estantería donde se ha ubicado, en caso de no haber hueco, devolverá -1

```
int registrarPaquete(int Código, int alto, int ancho, float peso)
```

Informática 2019-2020. 1º Curso. Grado de QUIMICA. Examen 2º convocatoria. Septiembre 2020.			A
Apellidos:	Nombre:	DNI:	
Aclaraciones: a) conteste este ejercicio en el mismo folio; b) se permite entregarlo a lápiz siempre que sea legible; c) copiar de cualquier forma implica un cero en esta prueba y por tanto, suspenso en la convocatoria correspondiente.			

NOTA: imprimir este documento a dos caras

**NOTA ACLARATORIA EXTRA:** en los ejercicios en los que se pide escribir el cuerpo de una función, salvo que en el enunciado se especifique claramente que dicha función debe usar el teclado y/o la pantalla, su código no debe contener llamadas a `scanf()` o `printf()` o funciones análogas, mucho menos para pedir parámetros por teclado o devolver valores por pantalla. **Cualquier uso indebido** de las funciones de manejo de teclado y pantalla en este sentido **será penalizado con la suspensión del ejercicio completo**, independientemente de que el resto de él sea correcto.

**EJERCICIO 1. (1 pto.)** Escriba la cabecera y el cuerpo de una función en C que tome como parámetros un momento concreto del día, expresado como **valores enteros de hora y minuto**. La función devolverá un **valor numérico** indicando qué actividad es la que se está realizando en función de en qué intervalo se encuentre ese momento del día.

#### ACTIVIDADES

- De 03:59 a 13:41 inclusive: Actividad 1
- De 13:42 a 21:55 inclusive: Actividad 2
- De 21:56 a 03:58 inclusive: Actividad 3



**(En sucio)**

**EJERCICIO 2** (1 pto.) Escriba la cabecera y el cuerpo de una función en C que tome como parámetro una **cadena de caracteres**, cuyo contenido representa un código de activación para una aplicación (puede tener dígitos y/o letras). La función deberá devolver 1 si el código de activación es válido, y 0 si no lo es.

Los requisitos para que un código de activación sea válido son los siguientes (**deben cumplirse todos**):

- El código de activación debe tener al menos 15 caracteres de longitud.
- El código de activación debe contener al menos una vez la letra B.
- El código debe contener exactamente 6 letras minúsculas dentro de él (no necesariamente seguidas).