**1º Kaprekar.**

El matemático indio Dattaraya Ramchandra Kaprekar trabajó en teoría de números, realizando varios descubrimientos a lo largo de su vida. Uno de ellos fue el conjunto de los que, desde entonces, se conocen como números de Kaprekar, que son aquellos números enteros positivos que, al ser elevados al cuadrado, pueden descomponerse (para una base dada, que asumiremos ser base 10) en dos enteros positivos cuya suma es igual al número original.

Por ejemplo, el número 703 es un número de Kaprekar, dado que 703 elevado al cuadrado es 494209 que puede descomponerse en 494 y 209 cuya suma da, de nuevo, 703. Otro ejemplo es el 9 (9^2 = 81 y 8 + 1 = 9).

Hay que tener presente que ambos números en la descomposición no tienen por qué tener el mismo número de dígitos. Por ejemplo en el caso del número 2728 tenemos que 2728^2 = 7441984 que es número de Kaprekar porque 744 + 1984 = 2728.

Realiza **modularmente** un programa que muestre los números de Kaprekar que hay en los primeros 10000 números. ***(3 puntos)***

**2º Voyyy caaminooo Mooooria...**

Frodo y Sam, tras las largas caminatas que hacían, se sentaban a descansar y para despejarse solían jugar al siguiente juego. Visto desde fuera el juego tampoco es demasiado divertido, pero a ellos parecía encantarles y distraerles de su pesada carga.

En cada partida, ambos eligen dos números positivos, por ejemplo:

Frodo: 7 12

Sam: 3 8

Después los ponen todos seguidos, 71238 y empieza la partida. En cada turno, uno de ellos suma 1 a cada número de forma que el 7 se convierte en 8, el 12 en 13, etc. Al ponerlos de nuevo todos juntos queda: 81349. En el siguiente turno le toca al siguiente, que vuelve a sumar uno a cada número, después al siguiente, y así hasta llegar a sumar 10. Cuando, al colocar todos los números seguidos, la cadena incrementa su longitud, el jugador que ha sumado gana un punto. Tras los 10 turnos, gana el que tiene más puntos.

La partida ejemplo es:

Inicio: 71238

Frodo: 81349

Sam: 914510 => Al sumar, Sam gana un punto.

Frodo: 1015611 => Ahora gana un punto Frodo.

Y así sucesivamente. Al final se indicará el resultado del juego.+

Realiza un **programa modular** para la resolución del mismo. La elección correcta de los módulos, además de facilitarte el trabajo, es un aspecto que se tendrá muy en cuenta en la corrección de este ejercicio. ***(3,25 puntos)***

**3º F-1000.**

Trabajamos para una empresa de tratamiento profesional de productos generados naturalmente; ricos en vitaminas, minerales, antioxidantes y fibra y que aportan pocas calorías y un alto porcentaje de agua… vamos, en una frutería.

Esta frutería se ha modernizado y tiene un brazo robótico, el F-1000, que coge las cajas de fruta de una estantería y las apila en un palé.

Los módulos que me permiten manejar el brazo son los siguientes:

Procedimiento: **Recibir\_pedido((ref) cantidad)** 🡪 nos devolverá, por referencia, la cantidad de cajas que tenemos que coger de la estantería y apilar en el palé. En *cantidad* se devuelve un 0 si no hay ningún pedido.

Función: **Detectar\_caja(fila, columna)** 🡪 mueve el brazo a esa posición y devuelve ***verdadero*** si en esa posición existe una caja y ***falso*** en caso contrario.

Procedimiento: **Coger(fila, columna)** 🡪 Coge la caja que haya en esa posición de la estantería entre sus pinzas.

Procedimiento: **Apilar(posición)** 🡪 Mueve el brazo y apila la caja que tiene en las pinzas en el palé en la posición que le indiquemos.

Procedimiento: **Posicion\_Inicial()** 🡪 Coloca el brazo robótico en su posición de reposo inicial.

En la estantería existen 150 cajas de frutas distribuidas en 10 filas por 15 columnas.

Cuando **iniciamos el sistema** la estantería está llena de cajas, por lo que el brazo podrá acceder a la caja que está en la fila 1, columna 1 (que será la que está la primera de la primera columna); la forma de ir cogiendo cajas será: **agotar las cajas de una columna antes de pasar a la siguiente columna**. Antes de coger la caja se debe comprobar si la caja existe en esa posición. Caso de no existir el programa acabará, se mostrará un mensaje indicando que el brazo pasa a estar inoperativo y que se ha producido un error.

Si **no hay errores**, y una vez servido el pedido que hayamos recibido, el robot volverá a la **posición inicial** y procederá a comprobar si tiene más pedidos. Cuando reciba otro pedido se procede a repetir la operación de apilarlo cómo se ha explicado anteriormente, con las mismas condiciones en caso de error.

El **programa acabará** cuando no queden cajas en la estantería. Antes de acabar, el programa colocará el brazo en su posición inicial, se mostrará un mensaje indicando que el brazo pasa a estar inoperativo y se mostrará, también, si el último pedido se pudo satisfacer o no. ***(3,75 puntos)***