FPIC
FORMACIÓN PROFESIONAL
IES CAMPANILLAS

# PROGRAMACIÓN – 1º D.A.W.

## TRIMESTRE 2 – CONTROL 1 – MODELO B

16 de febrero de 2015

---

### INSTRUCCIONES

➔ El alumno debe entregar una carpeta con las soluciones al examen cuyo nombre debe estar formado por el número de lista seguido de las iniciales. Por ejemplo, Facundo Romuedo Piladro que es el número 8 de la lista entregaría una carpeta con nombre **Ex08frp**.
➔ Los ficheros o carpetas correspondientes a las soluciones se deben nombrar igual que la carpeta junto con el número del ejercicio, por ejemplo **Ex08frp1.java, Ex08frp2.java, etc**.
➔ En los comentarios de cada programa se debe indicar el nombre completo, la fecha y el turno.

---

### EJERCICIOS

1. Realiza una nueva versión del juego “Busca el tesoro” realizado en clase con los siguientes cambios:
   - Hay tres tesoros en lugar de uno. Para ganar, el jugador debe encontrar dos de los tres tesoros.
   - Sigue habiendo una mina. Si el jugador la pisa, pierde.
   - No pueden coincidir las coordenadas de ninguno de los elementos, ya sean tesoros o mina.
   - Para que el ejercicio se pueda corregir con facilidad, se deben mostrar por pantalla, a modo de chuleta, las coordenadas de todos los elementos (tesoros y mina).

2. Realiza una función que devuelva el número “parizado” a otro que se pasa como parámetro. Cada dígito del número parizado es el mismo dígito del número original en caso de que éste sea par, o el siguiente par en caso de que el dígito original sea impar (el siguiente par del 9 será el 0). Por ejemplo el parizado de 308566 es el 408666 y el parizado de 92491 es el 2402. A continuación se muestra la cabecera de la función:

   ```
   public static long parizado(long x)
   ```

3. Se dice que una matriz cuadrada – tiene el mismo número de filas que de columnas - es triangular inferior cuando todos los valores que están por encima de la diagonal principal son cero. Por ejemplo, la siguiente matriz es triangular inferior:
   2 0 0
   7 4 0
   5 1 8
   Realiza una función que diga si una matriz dada es o no triangular superior y pruébala en un programa. Recuerda que si la matriz no es cuadrada, se puede concluir directamente que no es triangular superior, sin hacer ningún otro tipo de comprobación. La cabecera de la función es la siguiente:

   ```
   public static boolean esTriangularInferior(int[ ][ ] a)
   ```

4. Una cadena de multicines nos ha encargado una aplicación para la venta on-line de entradas. Como parte de esa aplicación, tenemos que implementar una función que coloque a los espectadores en una fila de butacas. Una fila de butacas es un array de una dimensión. Cada celda se corresponde a una butaca que tiene el valor 0 si está libre y el valor 1 si está ocupada. La función recibe como parámetros el array con la información de la fila de butacas y un número que es la cantidad de gente que se quiere colocar en esa fila. Si se puede colocar a los espectadores con éxito, la función devolverá un 0 y si no se puede, devolverá un -1. Lo primero que tiene que hacer la función es comprobar si hay sitio, si no lo hay devolverá -1. Si hay sitio, primero intentará colocar juntos a todo el grupo y, si no puede, los irá colocando en los primeros huecos que encuentre.
   Por ejemplo, si a = { 0, 1, 0, 0, 0, 1, 0 } y se quieren colocar 3 nuevos espectadores, la función devolverá 0 y el array se habrá modificado quedando así: a = { 0, 1, 1, 1, 1, 1, 0 }.
   Prueba la función desde un programa. La cabecera de la función es la siguiente:

   ```
   public static int ocupa(int[ ] fila, int usuarios)
   ```