**TEMA 1. EJERCICIOS**

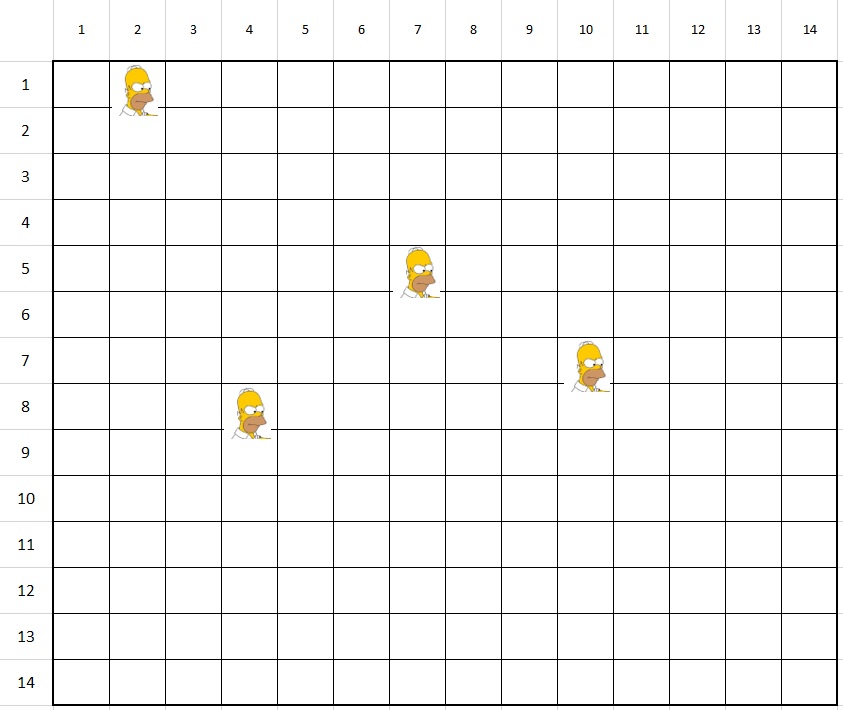
**Ej1:** Diseña el algoritmo que realice la siguiente función:

* V1.0: Generar números aleatorios (del 1 al 10) para rellenar un vector (array) de 6 posiciones.
* V2.0: Generar números aleatorios (del 1 al 10) para rellenar un vector (array) de 6 posiciones de manera que los números que se almacenen en el array no sean repetidos.

**Ej2:** Diseña el algoritmo que realice la siguiente función:

Tenemos un juego en el que colocaremos de manera aleatoria en un tablero de NxM una serie de Hommer Simpson, de manera que las posiciones en las que se colocará el Hommer correspondiente serán aleatorias. El problema es que no podré colocar un Hommer Simpson en una casilla si ésta ya está ocupada por otro Hommer Simpson que se ha colocado en ella previamente.

En la imagen vemos un ejemplo de tablero de 14x14 donde se colocan 4 Hommers. Realiza el diagrama de flujo y el código en Java correspondiente.



**Ej3:** Diseña el algoritmo que realice la siguiente función:

**v1.0** Queremos buscar un número que se lee por teclado en un vector de 100 números que previamente hemos rellenado de manera aleatoria con número de 1 a 1000 (sin importar si están o no repetidos), de manera que si existe el número me diga la primera posición en la que está, y sino existe, que el programa indique por pantalla que el número a buscar no está incluido en el vector.

**v2.0** Queremos realizar el mismo programa anterior PERO si el número está en el vector quiero saber TODAS las posiciones en que está, porque como pueden estar los números del vector repetidos puede estar en más de una posición.