



PILOTA VALENCIANA (2.5 pts)

Como sabemos que es difícil ganarnos la vida como programador probaremos en ser jugador de “pilota”. El juego consiste en tener que lanzar en un muro de un tamaño de 200px y cinco botones donde cada uno de ellos lanza la pelota dentro de un cuadrante diferente y la cantidad de pelotas a lanzar, es decir, si clico dentro del botón “AI” e introduzco el número 5, lanzará 5 pelotas dentro del recuadro Arriba Izquierda...Se debe utilizar arrays para almacenar la información.

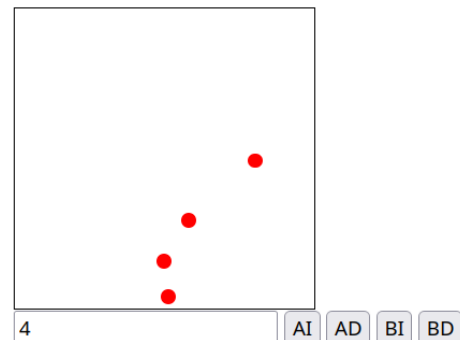
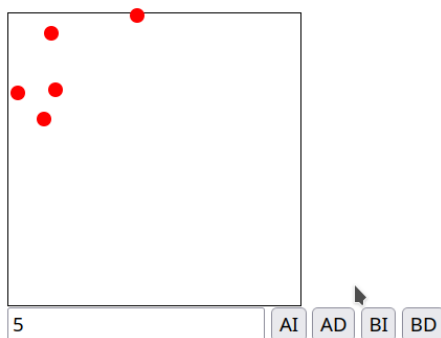
“AI” → Arriba Izquierda

“AD” → Arriba Derecha.

“BI” → Bajo Izquierda

“BD” → Bajo Derecha

“RAN” → Aleatoriamente en todo los cuadrados.



Nota I: Puede haber un pequeño margen de error a la hora de lanzar la pelota y que invada el otro recuadro..

Nota II: para pintar los cuadrados sería recomendable usar en CSS las siguientes propiedades.

```
<div style='background-color:red;position:absolute;height:10px;width:10px;top:0px;left:0px'></div>
```

```
.rojo{  
  background-color:red;  
  position:absolute;  
  height: 10px;  
  width: 10px;  
  border-radius: 50%;  
  top:0px;  
  left:0px;  
}
```



FÁBULA CONEJO Y LA LIEBRE (2.5ptos)

La tortuga y la liebre se retan a competir a ver quién llega a meta antes, cada lanzamiento se contabilizará y se mostrará sus posiciones.

Los movimientos de la tortuga son lentos, así que irá poco a poco avanzando una posición.

En cambio, la liebre como es un poco tramposa, hace una madriguera puede aparecer en cada tirada en un posición distinta porque se desubica .

Quien llegue antes a meta, es decir, la posición 0 ganará y se quedará con la casita de su oponente.

[illegible]

posición liebre: 15 -> posición tortuga: 15
posición liebre: 7 -> posición tortuga: 14
posición liebre: 14 -> posición tortuga: 13
posición liebre: 14 -> posición tortuga: 12
posición liebre: 4 -> posición tortuga: 11
posición liebre: 2 -> posición tortuga: 10
posición liebre: 11 -> posición tortuga: 9
posición liebre: 13 -> posición tortuga: 8
posición liebre: 2 -> posición tortuga: 7
posición liebre: 1 -> posición tortuga: 6
posición liebre: 0 -> posición tortuga: 5

HA GANADO LA LIEBRE

```
posición liebre: 15 -> posición tortuga: 15
posición liebre: 1 -> posición tortuga: 14
posición liebre: 3 -> posición tortuga: 13
posición liebre: 5 -> posición tortuga: 12
posición liebre: 6 -> posición tortuga: 11
posición liebre: 7 -> posición tortuga: 10
posición liebre: 12 -> posición tortuga: 9
posición liebre: 10 -> posición tortuga: 8
posición liebre: 3 -> posición tortuga: 7
posición liebre: 1 -> posición tortuga: 6
posición liebre: 9 -> posición tortuga: 5
posición liebre: 6 -> posición tortuga: 4
posición liebre: 12 -> posición tortuga: 3
posición liebre: 3 -> posición tortuga: 2
posición liebre: 10 -> posición tortuga: 1
posición liebre: 12 -> posición tortuga: 0
```

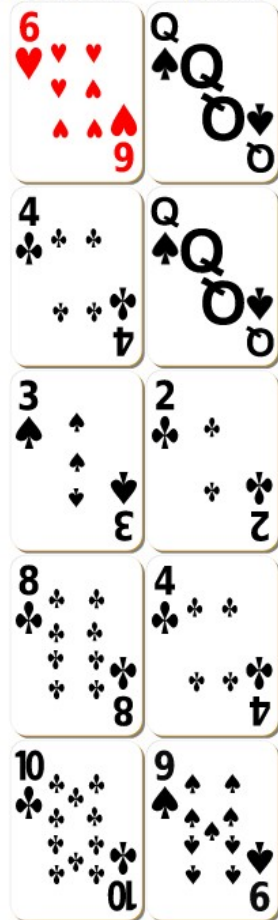
HA GANADO LA TORTUGA



POKERSTARS

Queremos hacer una competición de cartas, a los dos jugadores se le indican un número de cartas que le pasa el usuario y se almacenan en dos arrays distintos. Luego tienen que competir cada uno de ellos y mirar si en su posición cuál carta es la mayor, ya que el que tenga la carta más alta gana esa mano. Se deben contabilizar la cantidad de victorias de cada uno de ellos, en caso de empate no deben sumarse.

JUGADOR 1 ----- JUGADOR 2



Gana Jugador 2 esta mano. el jugador1 tiene 0 y el jugador2 tiene 1
Gana Jugador 2 esta mano. el jugador1 tiene 0 y el jugador2 tiene 2
Gana Jugador 1 esta mano. el jugador1 tiene 1 y el jugador2 tiene 2
Gana Jugador 1 esta mano. el jugador1 tiene 2 y el jugador2 tiene 2
Gana Jugador 1 esta mano. el jugador1 tiene 3 y el jugador2 tiene 2
GANA EL JUGADOR 1

CARTAS REPETIDAS

Tenemos una baraja de 10 cartas donde puedes salir cartas del 1 hasta el 10 de forma aleatoria, me gustaría saber cuántas cartas repetidas hay. Se debe usar arrays para resolver el ejercicio.



la carta 1 está repetida
la carta 6 está repetida
la carta 8 está repetida
la carta 9 está repetida