

JUAN HERRUZO HERRERO

y

MARIO LLORENTE PÉREZ

1. Enunciado

El objetivo de la práctica es implementar un programa Java que permita al usuario jugar a una versión fácil del célebre juego candy crash, en un entorno de texto (no gráfico).

Pertenece a un grupo de juegos basados en la idea de alinear al menos 3 piezas (“match 3”) que son eliminadas de un tablero(Para más información ver enunciado en la plataforma)

2. Principal

Algoritmo: Versión simple del conocido juego Candy Crush

Entrada: Un número entero para seleccionar el modo de juego

Cuatro números enteros para seleccionar los distintos elementos del juego

Salida: Verificación del modo de juego elegido

Puntuación obtenida

Tablas de elementos obtenidas durante el juego

Objetivo: Realizar el juego candy crush en distintas dificultades

Observaciones:

- La entrada 0 0 0 0 durante el transcurso del juego acaba el juego en curso y muestra la puntuación final
- Siempre habrá un movimiento válido en la tabla
- Tanto el número de movimientos como la altura y la anchura de la tabla son constantes globales y se pueden modificar según especificaciones
- Se establece por defecto 10 movimientos y una altura y anchura de 9
- Se establece una tabla predeterminada para realizar pruebas

Pruebas

Caso	Entradas	Salida esperada	Salida obtenida
1	0	"Gracias por jugar!"	
2	5	"Modo incorrecto, elija un modo valido"	
3	-1	"Modo incorrecto, elija un modo valido"	
4	1	"Modo facil" (matriz)	
Suponiendo que el primer dato se ha introducido bien.			
4	0 0 0 0	"Puntuación final : x Gracias por jugar!"	
5	0 0 0 -1	"Coordenadas no adyacentes o fuera del limite, escoja otras:X=[1,columnas] Y=[1,filas]"	
6	1 1 1 3	"Coordenadas no adyacentes o fuera del limite, escoja otras:X=[1,columnas] Y=[1,filas]"	
7	(movimiento valido)	(matriz modificada)	
8	(movimiento no valido)	"Movimiento nulo, repita otro:"	
9	150 150 150 151	"Coordenadas no adyacentes o fuera del limite, escoja otras:X=[1,columnas] Y=[1,filas]"	
10	-1 -1 -1 -2	"Coordenadas no adyacentes o fuera del limite, escoja otras:X=[1,columnas] Y=[1,filas]"	

Entorno

Entrada: Variables: modo,coordenadas(x1,y1),(x2,y2): enteros

Constantes: Movimientos, X,Y, CARAMELOS

Local: Variables Point

Usa:

mostrarTabla : procedimiento

actualizarTabla: procedimiento

moverCaramelos: procedimiento

crearTabla: procedimiento

initJuego:procedimiento

datosValidos: función

tablaCorrecta función

juego: Procedimiento

sonCero: función

Algoritmo

Inicio

Haz

Imprimir petición de modo

Leer y verificar modo

Mientras modo incorrecto

Imprimir "Modo incorrecto, elija un modo valido"

Imprimir petición de modo

Leer y verificar modo

Juego

Inicia juego

Crea tabla

Comprueba modo

Crea tabla

Verifica tabla

Si tabla no correcta

Crea tabla

Verifica tabla

Si tabla correcta

Da tabla

Crea contador

Comprueba contador

Mientras contador < movimientos y no son Cero

Imprime movimientos restantes

Imprime tabla

Imprimir petición de coordenadas

Leer y verificar coordenadas

Si coordenadas incorrectas

Imprimir "Coordenadas no adyacentes o fuera del limite, escoja otras"

Leer y verificar coordenadas

Si coordenadas correctas

Mover caramelos

Verificar movimiento

Si movimiento incorrecto

Imprimir "Movimiento nulo, repita otro:"

Recoloca caramelos

Si movimiento correcto

Comprueba tabla

Actualiza puntuación

Contador ++

Imprime puntuacion parcial

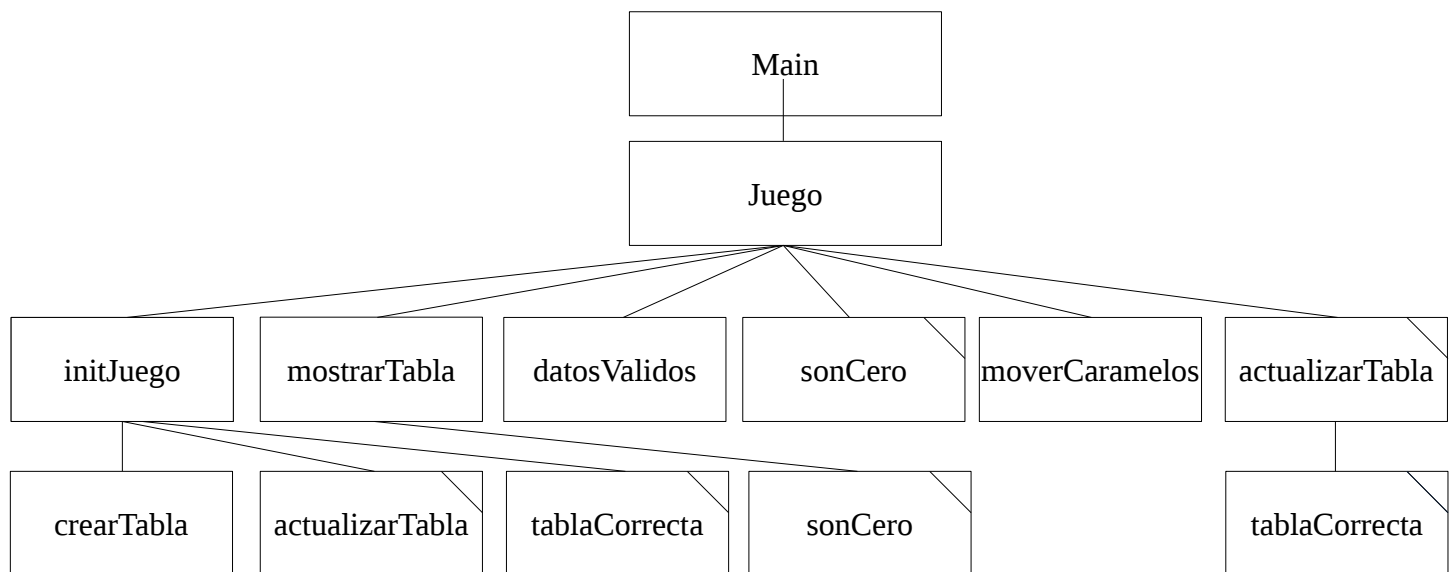
Imprime "Puntuación final" + puntuación

Mientras modo != salir

Imprime "gracias por jugar"

Fin

3. Módulos



3.1. Módulo

Funcion: Juego

Entrada: entero: nivel scanner: in

Devuelve: -

Objetivo: Realizar el proceso del juego.

Usa:

FILAS : Constantes

COLUMNAS : Constantes

initJuego : Procedimiento

mostrarTabla : Procedimiento

datosValidos : Funcion

sonCero : Fucion

moverCaramelos : Procedimiento

actualizarTabla : Procedimiento

3.2. Módulo

Funcion: InitJuego

Entrada: entero: modo char[][]: tabla point: puntuacion

Devuelve: -

Objetivo: Crear una tabla jugable.

Usa:

crearTabla : procedimiento

actualizarTabla : procedimiento

tablaCorrecta : función

3.3. Módulo

Funcion: moverCaramelos

Entrada: entero: x1 y1 x2 y2 char[][]: tabla

Devuelve: -

Objetivo: Mover los caramelos en la tabla

Usa: -

3.4. Módulo

Funcion: mostrarTabla

Entrada: char[][]: tabla

Devuelve: -

Objetivo: mostrar la tabla en pantalla

Usa: -

3.5. Módulo

Funcion: actualizarTabla

Entrada: char[][]: tabla entero: modo point: puntuacion

Devuelve: -

Objetivo: Quitar bloques de tres o mas de la tabla y actualizar la puntuacion.

Usa:

CARAMELOS : Constante

tablaCorrecta : funcion

3.6. Módulo

Funcion: tablaCorrecta

Entrada: char[][] : tabla

Devuelve: boolean, cierto si la tabla tiene movimeintos.

Objetivo: Ver si la tabla tiene movimientos posibles.

Usa: -

3.7. Módulo

Funcion: crearTabla

Entrada: char[][]: tabla entero: modo

Devuelve: -

Objetivo: Crear una tabla.

Usa:

CARAMELOS : Constante

3.8. Módulo

Funcion: datosValidos

Entrada: enteros : x1 y1 x2 y2

Devuelve: boolean, cierto si los datos son adyacentes y estan dentro del rango o todos son igual a 0.

Objetivo: Comprueba si los datos son adyacentes y estan dentro del rango o todos son igual a 0

Usa: sonCero : funcion

3.9. Módulo

Funcion: sonCero

Entrada: enteros : x1 y1 x2 y2

Devuelve: boolean, cierto si todos los datos son igual a 0.

Objetivo: Comprueba si todos los datos son igual a 0.

Usa: -

Subalgoritmo	Entrada	Salida	Devuelve	Descripción
juego	Nivel, in	Imprime en pantalla	-	El juego
initJuego	tabla modo puntuacion	-	-	Inicia el juego para que pueda ser jugado
moverCaramelo	x1 y1 x2 y2 tabla	-	-	Cambia los caramelos de sitio
mostrarTabla	tabla	Imprime en pantalla	-	Muestra la tabla por pantalla
actualizarTabla	tabla modo Puntuacion	-	-	Aactualiza los bloques
tablaCorrecta	tabla	-	boolean	Cierto si la tabla tiene algun movimiento
crearTabla	tabla modo	-	-	Crea una tabla al azar
datosValidos	x1 y1 x1 y2	-	boolean	Cierto si los datos son válidos
sonCero	x1 y1 x2 y2	-	boolean	Cierto si son todos los valores 0