



Práctica 3 2048

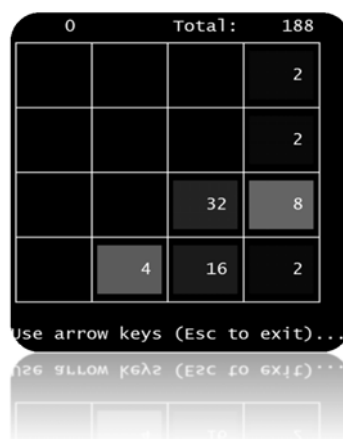
Grado en Ingeniería Informática
Grado en Ingeniería del Software
Grado en Ingeniería de Computadores

Luis Hernández Yáñez
Virginia Francisco Gilmartín
Javier Arroyo Gallardo
Raquel Hervás Ballesteros
Facultad de Informática
Universidad Complutense



Práctica 3

2048



2048

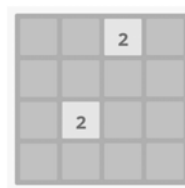
El juego

Tablero de 4 x 4 celdas (o baldosas)

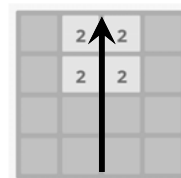
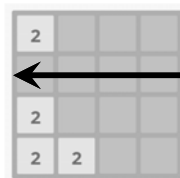
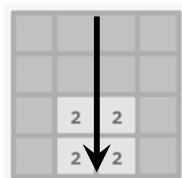
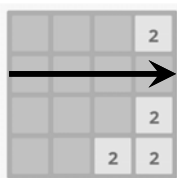
Se inicia colocando aleatoriamente dos fichas

Cada ficha que se coloca pueden valer 2 (más prob.) o 4

El jugador elige inclinar el tablero en una dirección...



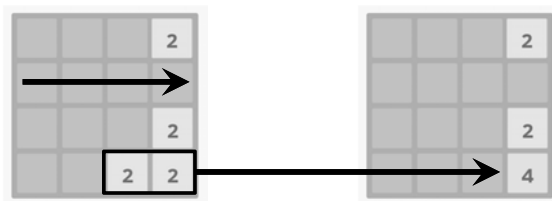
... Y todas las fichas se deslizarán en esa dirección



2048

El juego

Si una vez que las fichas se han deslizado, coinciden juntas dos fichas del mismo valor, se fusionan en una sola de valor doble



Esta nueva ficha ya no se podrá fusionar en este mismo movimiento

Una vez que todas las posibles “fusiones” se han llevado a cabo se termina el turno.

Entonces aparece una nueva ficha (que podrá valer 2 o 4) en una baldosa vacía elegida al azar



2048

El Juego

Según avanza el juego, el tablero se va llenando de fichas de diferente valor (y color):

		2	4
		4	8
	2	16	32
	2	2	16

Fin: Si se obtiene una ficha de valor 2048 😊

Si el tablero está lleno y ningún movimiento fusiona fichas ☹

Puntuación: Suma de los valores de las fichas fusionadas

2		2
	2	
	2	2
2		4

8 puntos

2		2
	2	
	2	2
2		4

12 puntos



2048

El programa

El programa reproducirá el juego de forma realista
¡En consola!

0	Total:	188
		2
		2
	32	8
4	16	2

Use arrow keys (Esc to exit)...



Cuatro etapas

1. Versión 1
 - Inicialización, visualización y cargar tablero (aprox. 06/03)
2. Versión 2
 - Deslizamiento de fichas (aprox. 13/03)
3. Versión 3
 - Juego funcionando (aprox. 20/03)
4. Versión 4
 - Mejores puntuaciones y salvar partida (opcional → 27/03)



Depuración

¡Prueba las demos!

¡Prueba tu programa con distintos casos de uso!

		2	2
4	2		8
4			8
		16	

2	4	8	16
4	8	16	32
8	16	32	64
16	32	64	128

			128
		128	256
	128	256	512
128	256	512	1024



Versión 1



Versión 1

Inicialización, visualización y carga de la partida

Tipos de datos:

- ✓ Tablero: Array bi-dimensional de $N \times N$ enteros (en principio $N=4$)

El juego debe funcionar con tableros de 6×6 , 10×10 , etc

```
const int N = 4;  
typedef int tTablero[N][N];
```

Baldosas vacías: valor 1 (2^0)

Fichas: Valores 2, 4, 8, 16, ... ($2^1, 2^2, 2^3, 2^4, \dots$)

		2	2
4	2		8
4			8
		16	

	0	1	2	3
0	1	1	2	2
1	4	2	1	8
2	4	1	1	8
3	1	1	16	1

- ✓ Enumerado tAccion:
Arriba, Abajo, Izquierda, Derecha, Salir, Nada



Version 1

Inicialización del tablero

Todas las casillas libres (es decir, a 1), excepto dos elegidas al azar

Las fichas valdrán 2 ó 4

95% la probabilidad de que valga 2

5% la de que valga 4

OJO: ¡No pueden ser las dos 4!



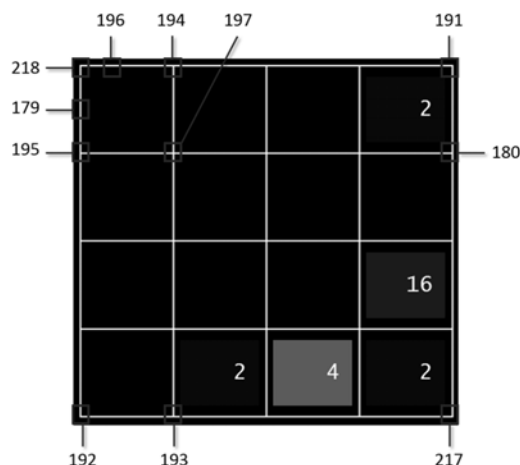
Version 1

Visualización del tablero

Limpiar la pantalla con:

`system("cls");` // Disponible en Windows.h

Los bordes son caracteres especiales:



`cout << char(codigo)`



Version 1

Color de fondo en la consola

3 líneas x 6 columnas

16

Baldosa vacía → Fondo negro

Baldosa ocupada → Color de fondo dependiente del valor + valor

Cambia el color de fondo con:

```
void setBackgroundColor(int color) {  
    HANDLE handle = GetStdHandle(STD_OUTPUT_HANDLE);  
    SetConsoleTextAttribute(handle, 15 | (color << 4));  
}
```

Usa el exponente de 2 como valor del parámetro color:

1, 2, 4, 8, 16, ... → $2^0, 2^1, 2^2, 2^3, 2^4, \dots$ (usa el logaritmo)



Version 1

Carga de la partida

4	← Dimensión
1	
1	} 1ª fila
2	
2	
4	} 2ª fila
2	
1	
8	
...	
1	} Últ. fila
1	
16	
1	
78	← Puntos



Version 1

Implementación

```
void inicializar(tTablero tablero);  
void mostrar(const tTablero tablero, int puntos, int total);  
int log2(int num);  
void cargar(tTablero tablero, int &puntos);
```



Version 1

```
void inicializar(tTablero tablero);
```

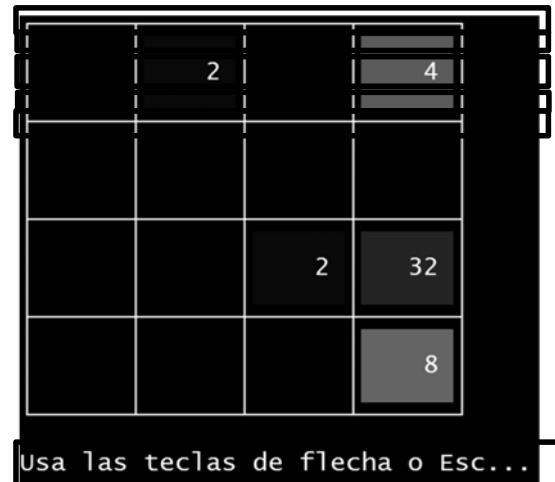
1. Inicializamos todas las celdas del tablero a vacío (1)
2. Elegimos aleatoriamente la baldosa donde colocar el primer valor
3. Elegimos el primer valor (2 o 4)
 - 95% probabilidad de 2 y 5% probabilidad de 4
4. Colocamos el primer valor en la baldosa
5. Elegimos aleatoriamente la baldosa donde colocar el segundo valor
6. Elegimos el segundo valor (2 o 4)
 - Si el anterior fue 4 → 2
 - e.o.c. → 95% probabilidad de 2 y 5% probabilidad de 4
7. Colocamos el segundo valor en la baldosa



Version 1

```
void mostrar(const tTablero tablero, int puntos, int total);
```

1. Pintamos el borde superior del tablero
2. Para cada fila del tablero:
 - a. Pintamos la primera línea
 - b. Pintamos la segunda línea (la que contiene el valor)
 - c. Pintamos la tercera línea (igual que la primera...)
 - d. Pintamos el borde que separa las filas
3. Escribimos el mensaje "Usa las teclas de..."



	2		4
		2	32
			8

Usa las teclas de flecha o Esc...



Version 1

```
int log2(int num);
```

Devuelve la potencia de 2 que da como resultado num

Ejemplo:

- ✓ Con num = 2 devuelve 1
- ✓ Con num = 4 devuelve 2
- ✓ Con num = 1024 devuelve 10



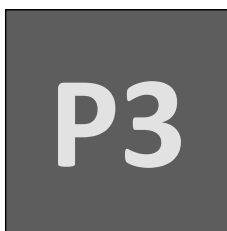
Version 1

```
void cargar(tTablero tablero, int &puntos);
```

1. Solicita el nombre del fichero con la partida a cargar
2. Abre el fichero
3. Si se ha abierto correctamente el fichero:
 - a. Se comprueba que la dimension del tablero coincide con DIM
 - b. Se cargan todas las celdas del tablero
 - c. Se cierra el fichero
4. Si no se ha abierto el fichero, lo inicializa de manera aleatoria
 - a. Se usa el subprograma `inicializa(...)`



Fundamentos de Programación



Práctica 3 2048

Grado en Ingeniería Informática
Grado en Ingeniería del Software
Grado en Ingeniería de Computadores

Luis Hernández Yáñez
Virginia Francisco Gilmartín
Javier Arroyo Gallardo
Raquel Hervás Ballesteros
Facultad de Informática
Universidad Complutense

