UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

FIB

Jocs per Computador

JC-MEI

June 2, 2025

Carles Matoses Gimenez *
Carles Matoses Gimenez †

*Email: carles.matoses@estudiantat@upc.edu
†Email: carles.matoses@estudiantat@upc.edu

Contents

1 Introducció

2 Objectius

2.1 Mapa

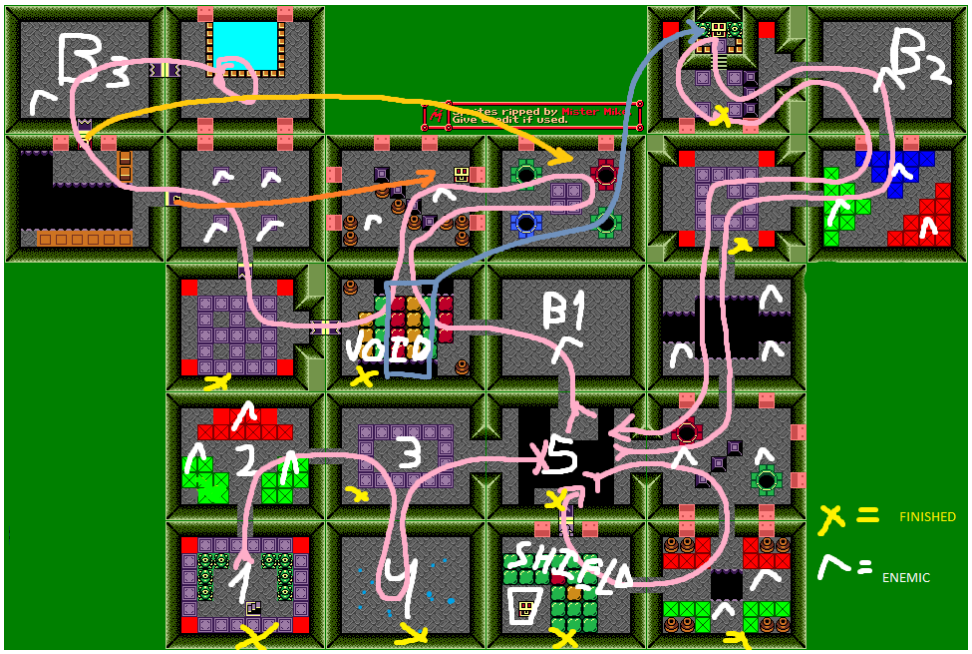

Figure 1: Mapa del joc amb el recorregut i els requerriments per a cada nivell.

3 Disseny

Per al disseny del joc no hem seguit un patró de disseny concret. El "loop" del joc incorpora una clase auxiliar que anomenada gamestatemanager. Aquesta conté una llista amb els estats necessaris. L'objectiu de gastar aquesta classe és poder anyadir i eliminar estats dinamicament. Per exemple, qual el jugador apreta "escape" i s'obre el menú, el que realment s'està fent és concatenar dos estats i sols permitir que el més alt en la jerarquia puga llegir els inputs i actualitzar-se en funció del temps.

3.1 Nivells, elements i entitats

La classe `World` guarda un diccionari amb els mapes. La classe `Mapa` conté una llista amb els nivells. La classe `Level` contente una llista amb els elements i entitats que s'han de carregar. Per al projecte s'ha creat un World que conte dos mapes, "dungeon1" i "overworld". El mapa "dungeon1" conté 29 nivells. Cada nivell es responsable de

contindre els estats per defecte de les entitats i també de customitzar events com ”entrar” i ”sortir” de les entitats.

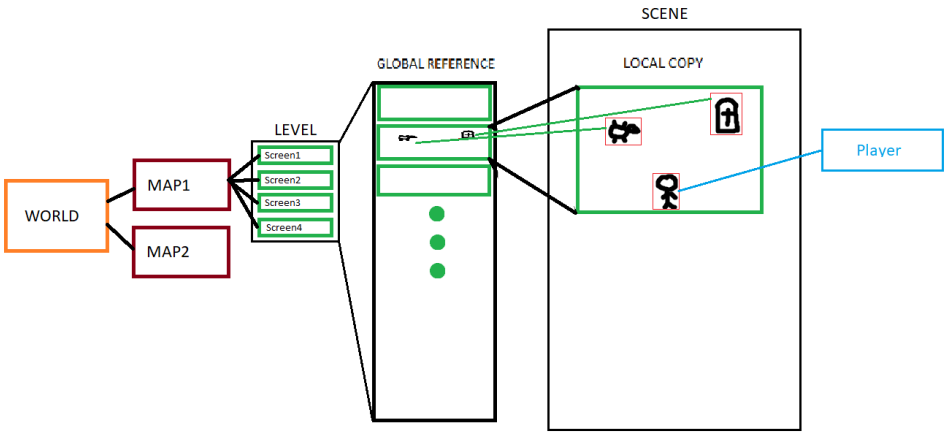

Figure 2: Jerarquia dels elements durant l’execució del joc.

3.2 Mecàniques

3.3 Triggers

4 Trenca closques

5 Enemics

6 Examples

Code examples:

6.1 Text

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec sit amet arcu aliquam, mattis sapien quis, faucibus ligula. Praesent facilisis felis libero, nec scelerisque est vestibulum a. Proin gravida tempor neque eget maximus. Aliquam vel ex arcu. Maecenas porta blandit leo, eget dapibus libero faucibus quis. Curabitur vel sapien pharetra, ultricies nulla sed, ullamcorper metus. Sed mauris sapien, dapibus et elit in, bibendum pellentesque dui. Donec consequat quam ut pulvinar sodales. Duis commodo ex pharetra mauris dictum, in ultricies purus facilisis. Donec id ornare sem. Nam diam erat, imperdiet et orci id, lacinia condimentum justo. Integer laoreet pulvinar nibh. Vivamus vel tellus lacus.

- one
- one
- one
- one
- one

Phasellus congue, massa in aliquet elementum, ipsum felis dictum enim, id maximus ante orci sit amet dui. Suspendisse blandit iaculis feugiat. Curabitur ante elit, tristique luctus hendrerit sed, posuere quis massa. Vivamus eu nunc vel neque egestas porttitor ac at nibh. Morbi odio justo, maximus sit amet elementum nec, finibus sed purus. Mauris eu posuere neque, at consequat elit. Praesent eu venenatis mauris, vel lobortis eros. Donec suscipit congue augue, ut aliquet metus volutpat ac. Vivamus mattis, nisl vel placerat euismod, tellus velit sollicitudin massa, nec sagittis nunc elit vel mauris. In id justo non nunc condimentum bibendum. Mauris scelerisque urna nisi, et lacinia justo posuere eget.

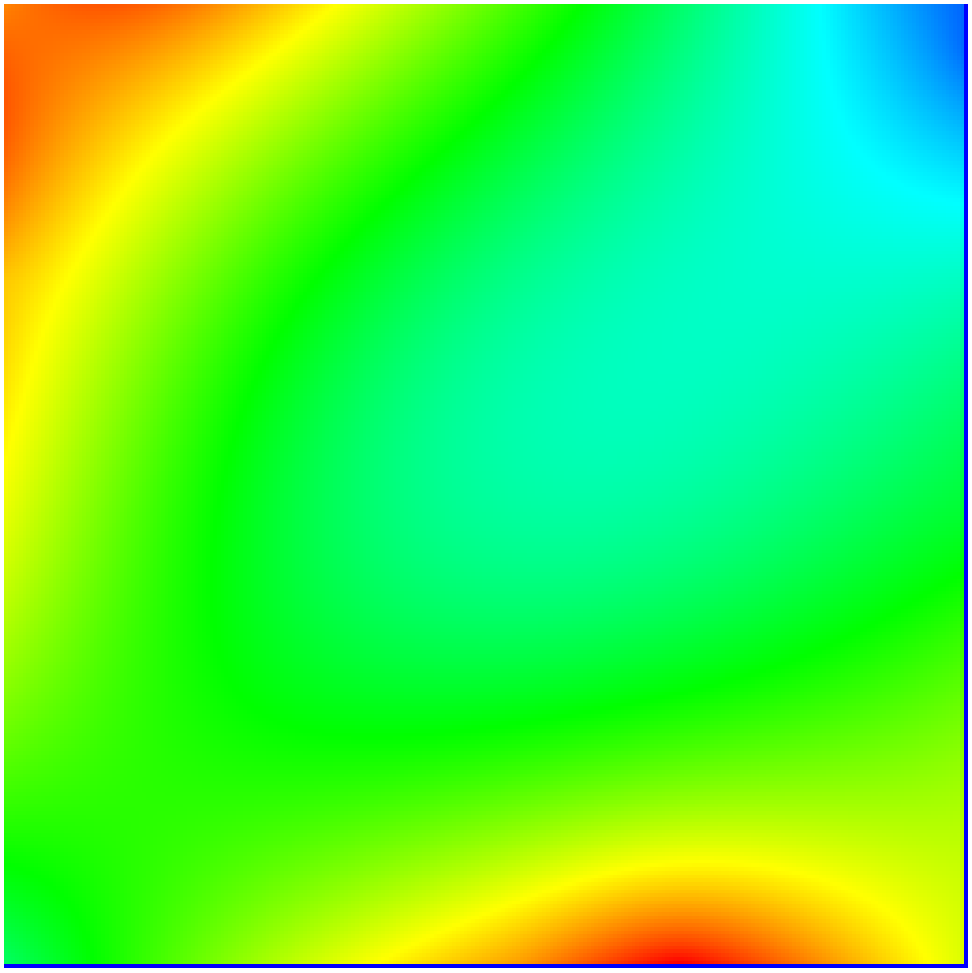

Figure 3: Test results for circuit 1

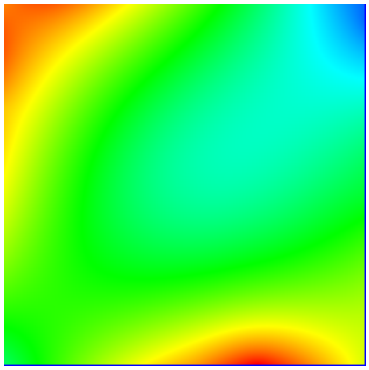

Figure 4: subimage1

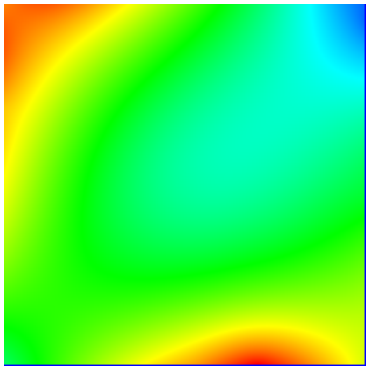

Figure 5: subimage2

6.2 Code

```
#define NB 2 // selected based on the number of threads
...
double relax_jacobi (double *u, double *utmp, unsigned sizex, unsigned sizey)
{
...
    #pragma omp parallel for reduction(+:sum) private(diff)
    for (int ii=0; ii<nbx; ii++) {
        ...
    }
    return sum;
}
```

References