Maze Game

El meu primer joc en java

By Lua Trevin Hernaiz

CFGS Desenvolupament d'aplicacions multiplataforma
Institut TIC de Barcelona
M03 Programació | M05 Entorns de desnvolupament

Abril 2024

Descripció del projecte

En aquest projecte he desenvolupat un joc sense UI, s'executa i s'utilitza a un Terminal.

El joc consisteix en sortir d'un llaberint:

Al jugador se li presenta un laberint amb diverses sortides i una única entrada. La complexitat del joc neix en que el laberint també compta amb parets que no són visibles fins que el jugador hi colisiona.

El transcurs del joc es basa en un prompt que recull el moviment que vol fer el jugador i posteriorment imprimeix el resultat del mapa tras fer el moviment.
Si el jugador es xoca, es suma un intent.

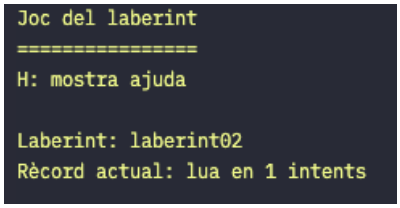
El joc acaba quan el jugador arriba a una sortida o decideix deixar de jugar.

Cada laberint compta amb un récord d'intents. Aquestos estan enmagatzemats en un fitxer csv. Quan el jugador guanya es compara la puntuació obtinguda amb el récord, si la superat s'actualitza

Funcionalitats base

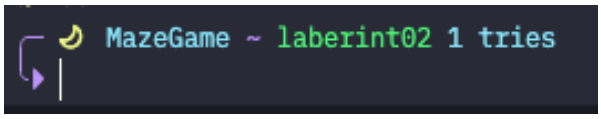
Inici de joc

Al jugador se li mostra un header amb info del joc (quin laberint s'està jugant, si existeix un récord i com es mostra el menú d'ajuda)



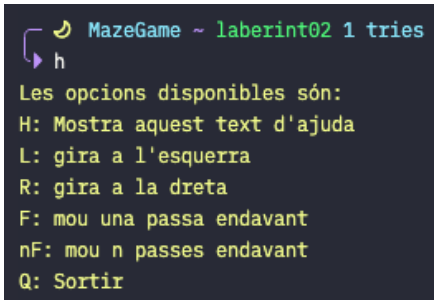
Prompt

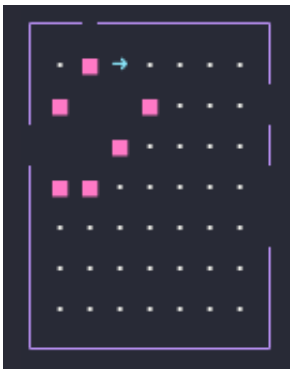
És l'encarregat de recollir el que vol fer el jugador. Compta amb info sobre el nom del joc, quin laberint s'está jugant i quants intents porta fins al moment el jugador.



Menú d'ajuda

S'ha implementat un menú d'ajuda per orientar al jugador en la partida. Aquest surt Quan el jugador introdueix caràcters invàlids o quan s'utilitza la opció h





Mapa

El mapa compta amb diverses icones escollides per a que resultin intuitives per al jugador. Compa amb parets, blocs, el jugador, caselles per a ón ha passat el jugador i caselles per descobrir.

Aquest es mostra cada cop que el jugador realitza un moviment válid.

Per cada tipus de caràcter s'han implementat color per a que resulti més senzill i agradable visualment.

Moviment Jugador

El prompt recull el que vol fer el jugador, els casos vàlids són q (sortit), h (menú d'ajuda) o un moviment. En cas de ser un moviment vàlid (més info sobre els invàlids a la seccio de control d'errors) S'actualitza la posició del jugador i es torna a imprimir el mapa.

Les quatre casólistiques després de mourse són:

- El jugador **arriba a una port**a per tant guanya
- El jugador **es xoca amb un bloc,** es suma un intent i es retorna a la posició inical
- El jugador **gira**, no s'ha de comprobar la col·lisió, es torna a mostrar el mapa i s'espera el següent moviment
- El jugador **arriba a una posició lliure**, es torna a mostrar el mapa i s'espera el següent moviment

Récord

Si el jugador guanya, es comprova si ha superat el récord d'intents. Si l'ha superat se li pregunta el nom per guardar un nou récord, si no se li dona ánims per intentar superar-lo.

Implementació

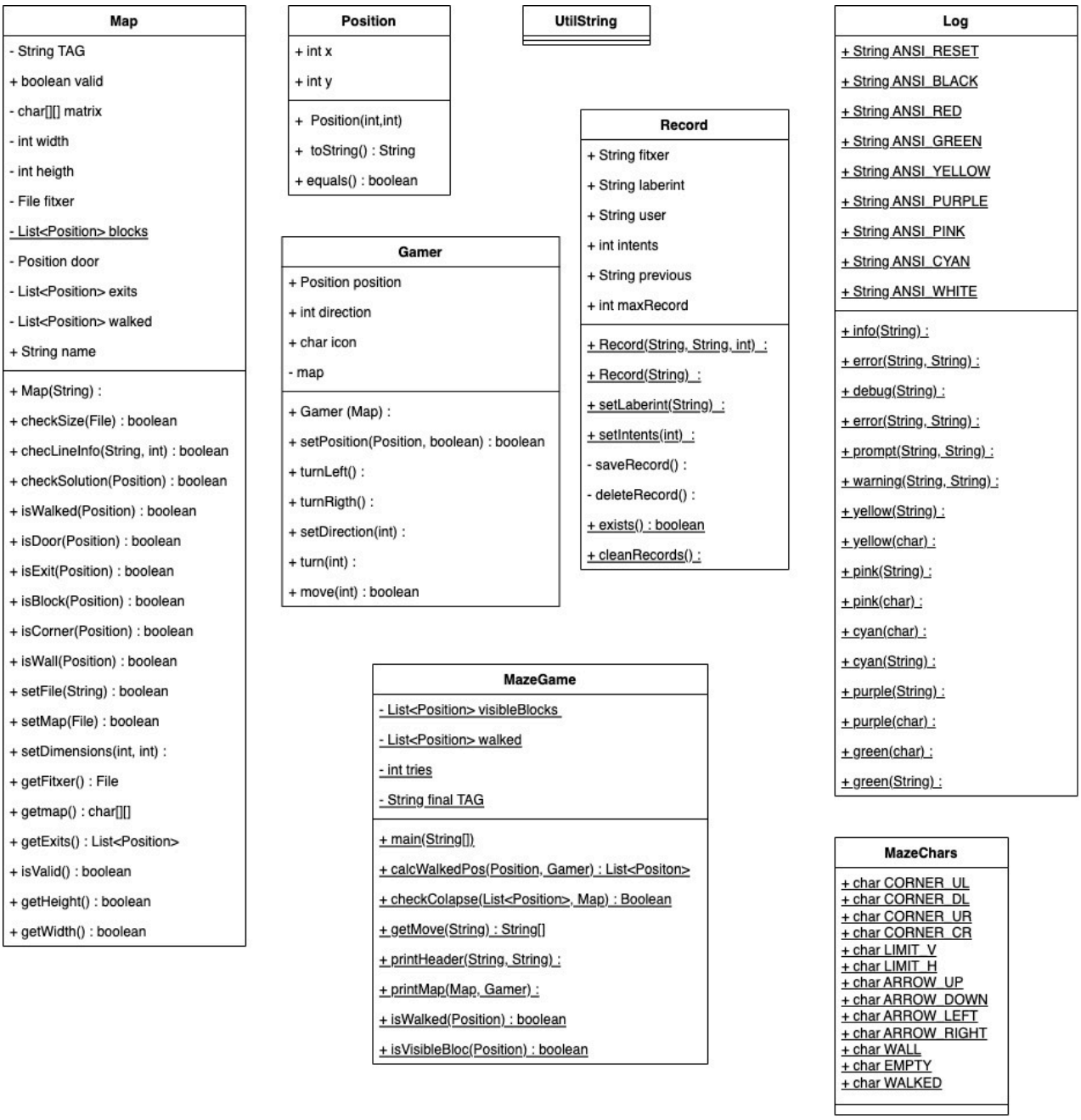


Diagrama clases

Ampliacions

Colors

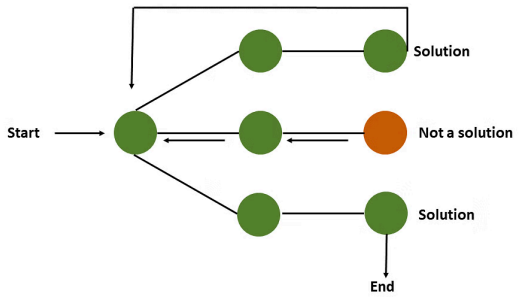
Per a facilitar la jugabilitat i fer més el joc més agradable de jugar s'han implementat Colors. Això es fa gràcies a la clase Log.

La classe log compta amb una série de métodes que simplement imprimeixen les dades rebudes amb els codis de color de terminal (guardats en variables).

Implementació detallada a l'esquema UML de classes.

Laberint solucionable

Mitjançant backtracking, s'ha implementat una fucnió recursiva que comprova si el laberint té solució. Des de l'inici comproba tots els camins possibles desde les posicions adjacents. Si a la crida inclou una trucada que acaba en una posició no vàlida (fora el mapa, parets o blocs) torna a l'ultim punt válid.



Makefile

El makefile es un Archiu que compta amb una serie de regles i comandes a executar. En aquest cas s'ha definit una recla per compilar, per borrar els archius temporals I per executar(amb argument de quin laberint es vol jugar).

He decidit implementar-ho perqué:

A l'hora de desenvolupar facilita molt la Feina ja que **només complila els archius modificats des de l'ultim cop**. Això ho fa comparant les dates de modificació dels objectes i el codi.

L'usuari no necessita saber compilar, només executar. Això es degut a que les regles tenen dependenciés. Quan s'indica que es vol executar el projecte, abans comprova si ha de compilar. En cas de necessitar-ho compila i després executa el joc.

La neteja del projecte. En aquest cas s'ha establert que els archius objecte vagin a una carpeta bin. Això es molt útil perqué aquest archius només ocupen espai inecessari un cop acabat de jugar i amb la regla de neteja simplement es borra el directori.

Múltiples girs

Al igual que quan avances, també pots girar diverses vegades indicant el numero davant de la lletra.

Control d’errors

El principal repte d’aquest projecte ha sigut controlar totes les casolístiques de fitxers o moviments invàlids. Aquí una llista del que s’ha controlat.

Laberints

Ha d’existir el fitxer

Ha de tenir una capçalera amb format NxN indicant el tamany del laberint

El laberint ha de complir les dimensions establertes a la capçalera

Les sortides han d’estar a les parets i mínim n’hi ha d’haver una.

Ha d’haber una única sortida ubicda en una de les parets

No pot haber-hi espais buits a les parets

Moviments

Ha d’estar format per carácters vàlids

En cas d’haber-hi números, han d’estar seguits d’un carácter vàlid

No pot moure’s a una paret

S’ha de comprovar colisions pas per pas. Si no, en cas de que la posició final sigui vàlida peró el camí contingui blocs, simplemet els saltaria.

Per cada cas d’error hi ha un missatge personalitzat.

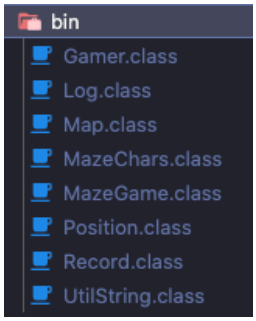
Gràcies a la classe Log ha facilitat la implementació perqué:

Depenent de la gravetat, segueixen un codi de colors

Els errors están classificats en tipus, les funcions de Log recullen un primer string per el tipus d’error i un altre del error en sí.

Estructura del projecte

A l'arrel projecte comptem amb diversos archius i directoris:



Bin

Es crea quan s'executa el joc I es borra quan s'executa la regla de neteja del makeFile. Conté tots els fitxers class del projecte.

La decisió de deixar-ho tot a una carpeta fa que l'arrel del projecte no s'ompli d'archius temporals.

Laberints

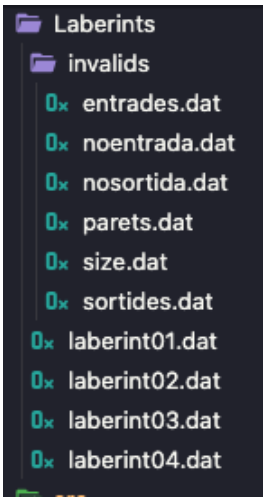
Són tots els fitxers amb les dades dels laberints. Compta amb un subdirectori a ón estan classificats els archius inváldis.

Cada laberint conté una primera línea amb el tamany i després un seguit de caràcters:

X - Blocs I parets

G - Sortides

E - Entrada



src

Carpeta amb tot el codi del projecte.

Cada classe està a un fitxer individual.

Altres fitxers

També comptem amb:

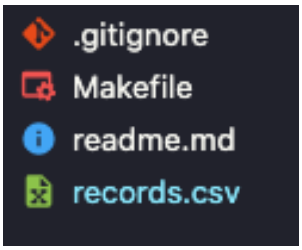
Gitignore: que s'encarrega de que no es pugin els .class al repositori

Makefile: Més info a les ampliacions

readme.md: Per donar info del projecte al repositiori

records.csv: Per enmagatzemar els records dels laberints

docs.pdf: Aquest propi document



Conclusions

En aquest projecte, he pogut aplicar els conceptes que hem après a classe, els quals ha estat fonamental tenir clarament integrats i relacionats per poder afrontar aquest repte.

Fer aquest treball de manera individual ha requerit una gran dosi de concentració i resolució de problemes. He hagut d'afrontar diverses complexitats durant el desenvolupament del projecte, des de la generació del laberint fins a la implementació de les mecàniques del joc.

No obstant això, el que realment ha enriquit la experiència és la col·laboració amb els meus companys. L'intercambi d'idees ens ha permès resoldre problemes de maneres que no hauriem pensat sols i plantejar coses de maneres diferents ha millorat significativament el joc.

A través d'aquesta experiència, he après que la combinació del treball individual i la col·laboració és fonamental per aconseguir l'èxit en projectes.

La interconnexió de coneixements de diverses àrees ha estat crucial en aquest projecte de programació del joc per terminal. He integrat principis de programació, mecàniques de joc i conceptes d'eficiència en l'algorítmia i el rendiment del codi. També he barrejat altres conceptes que no habia utilitzat a classe, com per exemple el Makefile, el que ha fet que treballi la meva adaptabilitat i creativitat per modificar coses que ja sabia en altres contextos.

Aquesta combinació ha resultat fonamental per crear un joc funcional i atractiu.

En resum, aquest projecte no només m'ha proporcionat una comprensió més profunda de la programació i el desenvolupament de jocs, sinó que també m'ha ensenyat la importància de l'intercanvi d'idees amb els meus iguals i en veure una visió global dels projectes.