Ministerul Educatiei Nationale

Colegiul National Elena Cuza

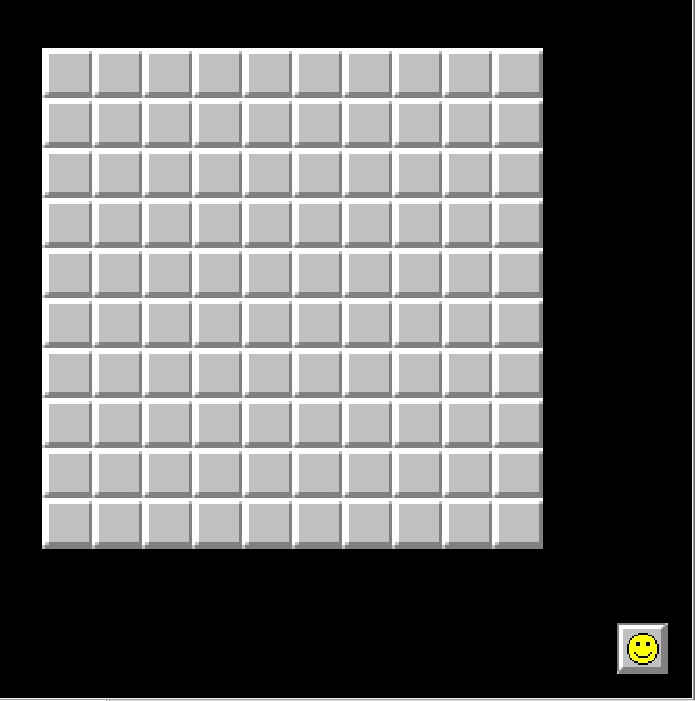
**ATESTAT DE COMPETENTE PROFESIONALE LA INFORMATICA**

Profesor Absolvent

Petre Carmen Ganescu Robert-Florinel

Craiova 2018

Overview

**Jucătorul este prezentat iniţial cu o grilă de pătrate nediferenţiate. Unele pătrate selectate aleatoriu, necunoscute jucătorului, sunt desemnate să conţină mine. De obicei, dimensiunea grilei şi numărul de mine sunt stabilite în avans de către utilizator, fie prin introducerea numerelor sau selectarea de la nivelurile de calificare definite, în funcţie de implementări. (în varianta Microsoft, aceasta este limitată la 30 de ori cu 24, cu 667 de mine.)**

**Jocul are ca scop dezvaluirea fiecarui patrat, făcând clic sau accesand in alt mod patratul indicat. Dacă un pătrat care conţine o mină este dezvăluita, jucătorul pierde jocul. Dacă este dezvăluită, o cifră este afişată în loc de pătrat, indicând cât de multe pătrate adiacente conţin mine; în cazul în care nu sunt mine adiacente, pătratul devine gol, şi toate pătratele adiacente vor fi recursiv dezvăluite. Jucătorul utilizează aceste informaţii pentru a deduce conţinutul altor pătrate, şi poate astfel dezvaluii in conditii de siguranţă fiecare pătrat sau marca pătratul ca conţinând o mină.**

**În unele versiuni, un semn de întrebare poate fi plasat într-un pătrat nedezvăluit pentru a servi ca un ajutor pentru deducere logică. Anumite implementări pot permite jucătorilor un joc mai rapid "clear around " un pătrat dezvaluit o dată cu numărul corect de mine care au fost semnalizate în jurul acestuia. Jocul este câştigat atunci când toate patrate sunt dezvăluite, pentru că toate minele au fost localizate.**

History

**Minesweeper are originile în cele mai vechi jocuri mainframe din anii 1960 și 1970. Cel mai vechi strămoş al Minesweeper a fost Jerimac Ratliff ' s Cube. Stilul gameplay-ul de bază a devenit un segment popular de gen de joc de puzzle în anii 1980, cu astfel de titluri ca minat-out (Quicksilva, 1983), Yomp (virgin interactive, 1983), şi cub. Cubul a fost succedat de logica neobosită (sau RLogic pe scurt), de Conway, Hong și Smith, disponibil pentru MS-DOS încă din 1985; jucătorul a luat rolul de un privat în Statele Unite Marine Corps, oferind un mesaj important la centrul de comandă din SUA. RLogic a avut o mai mare similitudine la Minesweeper decât la cub în concept, dar o serie de diferenţe există.**

## Library used

#include <graphics.h>

S-a utilizat pentru functii diverse cum ar fi:

-Crearea unei ferestre:

initwindow(700,700,"",-3,-3);

-Setarea fontului a culorii si a stilului:

settextstyle(0,HORIZ\_DIR,4);

setcolor(4);

outtextxy(50,600,"AI PIERDUT");

outtextxy(50,600,"AI CASTIGAT");

-Citirea de imagini pentru crearea graficii jocului:

for(int i=1;i<=10;i++)

for(int j=1;j<=10;j++){

y=i\*50;x=j\*50;

if(flags[i][j]==a[i][j]-10)

readimagefile("10.jpg",x,y,x+50,y+50);

else

readimagefile(word[a[y/50][x/50]],x,y,x+50,y+5}

#include <windows.h>

S-a utilizat pentru functii diverse cum ar fi:

-Accesarea coordonatelor mouse-ului:

POINT currentPosition;

GetCursorPos(&currentPosition);

x=currentPosition.x;

y=currentPosition.y;

-Apasarea butoanelor de pe mouse:

GetAsyncKeyState(VK\_LBUTTON)

GetAsyncKeyState(VK\_RBUTTON)

-Pentru crearea unei pauze:

Sleep(70);

delay(80);

#include <ctime>

S-a utilizat pentru functii diverse cum ar fi:

-Prelucrarea in timp real a anumitor date:

srand(time(0));

#include <cstring>

S-a utilizat pentru functii diverse cum ar fi:

-Setarea matricei cu valoarea 0:

memset(a,0,sizeof(a));

memset(matriceFill,0,sizeof(matriceFill));

memset(flags,0,sizeof(flags));

#include <fstream>

S-a utilizat pentru citirea din fisier:

Pe langa acestea,s-a mai utilizat:

Un algoritm de umplere:

void algFill(int x,int y)

{

matriceFill[x][y]=-1;

flags[x][y]=1;

for(int i=0;i<4;i++){

int nou\_i=x+di[i];

int nou\_j=y+dj[i];

if(ok(nou\_i,nou\_j)&&a[x][y]==a[nou\_i][nou\_j] )

algFill(nou\_i,nou\_j);

Algoritm de numerotare a bombelor:

bool check(int i,int j){

if(a[i][j]==9)

return 1;

return 0;

}

int checkAll(int i,int j){

int k=0;

k+=check(i+1,j-1)+check(i+1,j)+check(i+1,j+1)+check(i,j-1)+check(i,j+1)+check(i-1,j)+check(i-1,j-1)+check(i-1,j+1);

return k;

}

int bombe(int b){

int x,y;

srand(time(0));

while(b!=0){

x=rand()%n+1;

y=rand()%n+1;

if(a[x][y]!=9){

a[x][y]=9;

}

else

b++;

b--;

}

}

Algoritm de inserare “flag”:

if(GetAsyncKeyState(VK\_RBUTTON)){

delay(80);

round(x,y,50);

if(flags[y/50][x/50]==0){

delay(30);

readimagefile("flag.jpg",x,y,x+50,y+50);

flags[y/50][x/50]=-1;

}

}

Algoritm de stergere “flag”:

if(GetAsyncKeyState(VK\_RBUTTON)){

delay(30);

if(GetAsyncKeyState(VK\_RBUTTON)){

delay(80);

round(x,y,50);

if(flags[y/50][x/50]==-1){

readimagefile("button.jpg",x,y,x+50,y+50);

flags[y/50][x/50]=0;

}

}

}

Algoritm ce constata conditia de “defeat”:

if(a[y/50][x/50]==9){

setcolor(4);

readimagefile("redbomb.jpg",x,y,x+50,y+50);

lose();

readimagefile("redbomb.jpg",x,y,x+50,y+50);

outtextxy(50,600,"AI PIERDUT"); DEFEAT=1;

}

else

readimagefile(word[a[y/50][x/50]],x,y,x+50,y+50);

flags[y/50][x/50]=1;

Algoritm ce constata conditia de “Win”:

if(bomba==b){

for(int i=1;i<=n;i++)

for(int j=1;j<=n;j++){

int y=i\*50,x=j\*50;

if(flags[i][j]==-1)

readimagefile("10.jpg",x,y,x+50,y+50);

else

readimagefile(word[a[y/50][x/50]],x,y,x+50,y+50);

}

while(Ai\_Castigat){

setcolor(rand()%10+1);Sleep(70);

outtextxy(50,600,"AI CASTIGAT");

if(GetAsyncKeyState(VK\_LBUTTON)){

GetCursorPos(&currentPosition);

x=currentPosition.x;

y=currentPosition.y;

round(x,y,25);

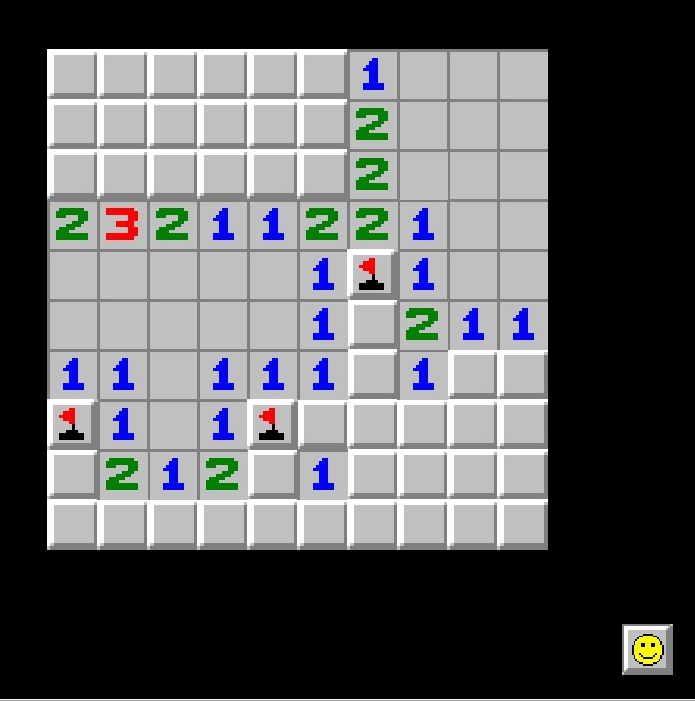
if(x>624&&x<676&&y>624&&y<676)

Ai\_Castigat=0;

}

## Gameplay

**Primul pas: Elemente de baza**

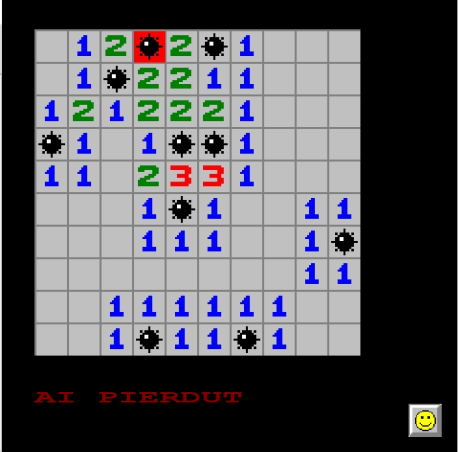
Inainte de a incepe jocul, trebuie sa fie cunoscut cum functioneaza jocul de fapt.

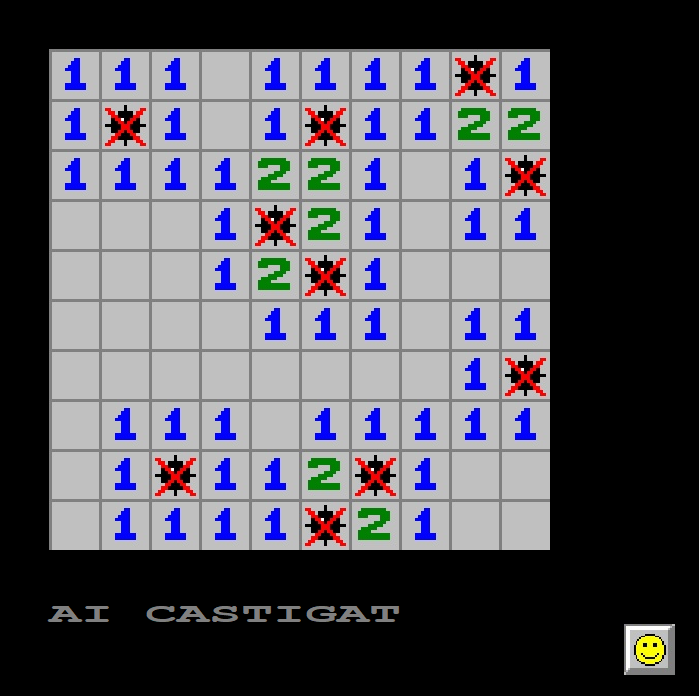
Jocul este foarte simplu, chiar daca la prima vedere pare ceva greu si imposibil. Se face click pe un patrat si drept urmare se va returna un numar. Acest numar reprezinta cat de multe mine sunt in jurul acesteia. Daca gasiti mina, se poate pune un steag care nu poate fi deschis.

**Al doilea pas: Strategia**

Strategia este simpla:

Cand jocul incepe, deschide patratele din jurul colturilor, de obicei sunt blocurile cu rata de peste 95% sa fie bune.

Marcheaza toate minele cu click dreapta care sunt clar vizibile.In jurul unui patrat sunt 8 casute, daca nimeresti numarul 1 si este singura care se deschide, nu te ingrijora sansele sa nimeresti bomba din prima sunt mici.

**Al treilea pas: sfaturi +** **indicii**

Gasirea minelor in zonele apropiate de blocurile cu 1 este de foarte mare ajutor, pentru ca deschide multe patrate si ofera indicii spre patrate diferite de bombe

Ai grije. Doar patratele deschise pe care le stii sunt sigure .

O sa fie nevoie de cel putin 10 jocuri pana cand o sa fi in stare sa poti completa unul.

Multa distractie iti urez.

**Concluzie:**

Minesweeper este un joc video de tip puzzle single-player. Obiectivul jocului este de a goli o tablă dreptunghiulară care conţine ascunse "minele" sau bombe fără a detona oricare dintre ele, cu ajutorul indicilor depe patrate cu numere, ce indica cate mine sunt învecinate în fiecare domeniu. Jocul provine din anii 1960 şi a fost scris pentru multe platforme de calcul în uz astăzi. Are multe variaţii şi ramificaţii.

Un joc ce merita jucat, inteles si castigat.

Multa bafta in rezolvarea acestuia.

**Bibliografie:**

-Manualul de informatica pentru clasa a IX-a, c++(profil real intesiv, specializare matematica-infromatica.

-Manualul de informatica pentru clasa a X-a, c++(profil real intesiv, specializare matematica-infromatica.

-Manualul de informatica pentru clasa a XI-a, c++(profil real intesiv, specializare matematica-infromatica.

-Manualul de informatica pentru clasa a XII-a, c++(profil real intesiv, specializare matematica-infromatica.

**Webografie:**

<https://www.youtube.com/watch?v=cK9pmJdTK9Y&list=PL5UFsTza4wWSNhe0xuO6ELw7ORU-UHNDO>

<https://msdn.microsoft.com/en-us/library/windows/desktop/bg126469(v=vs.85).aspx>