

Candy Crush

LUCRARE PENTRU ATESTAREA COMPETENȚELOR PROFESIONALE

Elevi: Axinte Cosmin-Andrei

Profesor îndrumător: Neagu Lucian

Liceu: Colegiul Național "Costache Negruzzi" Iași

Sesiune: Mai 2022

Cuprins:

1. Motivul alegerii temei	pag.4
2. Aplicația “Candy Crush”	pag.5
3. Explicarea codului	pag.9
4. Posibilități de îmbunătățire	pag.14
5. Resurse	pag.14
6. Bibliografie	pag.14
7. Codul sursă	pag.15

Motivul alegerii temei

Motivul alegerii temei este popularitatea jocului Candy Crush, simplitatea înțelegerii conceptului, dar și complexitatea problemelor oferite de acest proiect.

Șablonul oferit de Candy Crush ne-a oferit o imagine de ansamblu a proiectului și ocazia de a lucra modular cu elementele matricii.

La început acest proiect a fost o teza la informatică în urma căreia am fost captivați de concept și am dorit să ducem aplicația mai departe. Proiectul pune în valoare lucrul cu matricile, având ca problemă principală determinarea eficienței a elementelor 3-in-a-row, adică cel puțin trei elemente de același tip aflate pe poziții adiacente vertical sau orizontal.

De asemenea, prezența unor piese speciale care au rol de joker (se pot asocia cu orice alte elemente) și îndeplinesc funcții deosebite constituie o provocare în plus, întrucât aceste cazuri trebuie testate separat.

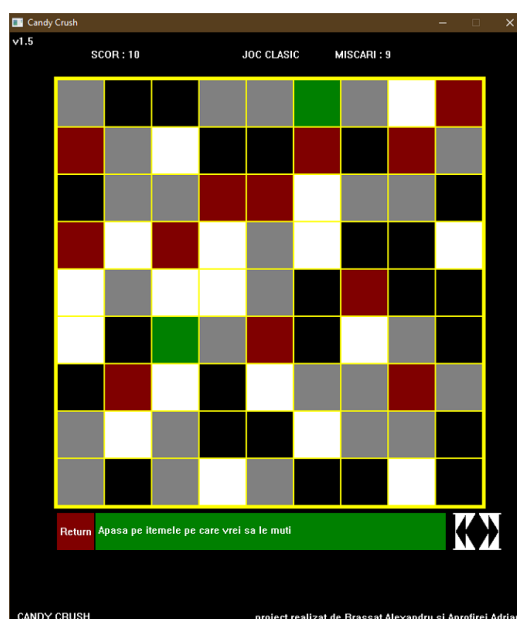
APLICATIA "CANDY CRUSH"

Descriere

Candy Crush este construită pe modelul clasic match 3, oferind jucătorului posibilitatea de a juca un joc cu 10 mutări și de a ajunge în topul scorurilor obținute de alții. Acesta se deschide într-o fereastră de 700x800.

Abilități

- Joc cu 10 mutari
- Joc infinit
- Reținerea unui high score și ordonarea lui crescătoare
- Schimbarea vitezei de derulare
- Reținerea texturilor în memorie pentru eficiența de timp
- Texturi de debug, în cazul în care nu sunt găsite imaginile în folder



Mod de utilizare

În momentul inițializării programului, se deschide o fereastră grafică. În cadrul acesteia, utilizatorul poate apăsa pe mai multe butoane cu mouse-ul:

- Începe un joc cu 10 mutări (clasic)
- Începe un joc cu mutări infinite
- Tutorial
- Ieșire

La începerea jocului, se afișează o matrice de 10 x 10 cu cel mult 5 elemente distincte, scorul (inițial 0) și mutările rămase (inițial 10). Viteza de desfășurare a jocului poate fi schimbată prin apăsarea celor două săgeți din dreapta jos a ecranului.

Jucătorul poate selecta două elemente alăturate pentru a le interschimba, cu scopul de a uni cel puțin 3 de același fel pe linie sau coloană. În cazul în care elementele schimbate nu îndeplinesc condiția de 3 pe linie sau coloana, matricea revine la forma inițială și se afișează mesajul "MISCARE INVALIDĂ".

În caz contrar, se șterg șirurile obținute și se reumple matricea până când nu mai sunt elemente 3-in-a-row, iar scorul crește în funcție de numărul elementelor șterse.

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

Pot apărea și elemente speciale, pe lângă fructe, ce au abilitați particulare:

- Coroanele – apar atunci când sunt șterse elemente de același tip pe linie și pe coloană, iar acestea se comportă ca un joker. Atunci când este distrusă o coroană, aceasta șterge întreaga linie și coloană pe care se află;
- Bombele – apar aleatoriu cu o raritate mai mare decât fructele; atunci când se face dublu click pe bombă toate elementele din jurul ei dispar într-un patrat de 3x3 (fără a se adăuga la scor), în urmă rămânând o piatră;
- Pietrele – apar în urma exploziei unei bombe; nu pot fi mutate și sunt distruse doar de o bombă;
- Atunci când scorul obținut se va încadra pe leaderboard-ul memorat în fișier, scorul va căpăta o culoare galbenă.

După 10 mutari, jocul se termină și programul se întoarce la secvența de titlu. În cazul în care jucătorul a obținut un high score, acesta va fi rugat să introducă un nume de maxim 11 caractere (în cazul în care nu se introduce niciun nume jucătorul va fi numit din oficiu “guest”), după care tabela de scoruri se va actualiza.

Atunci când utilizatorul dorește să închidă aplicația acesta trebuie să apese butonul “IESIRE” din partea de jos a ecranului.

Explicarea codului

La inițializarea programului se citește din fișierul "highscore.sav" și se reține în memorie, după care începe secvența de titlu.

```
46  int main ()
47  {
48      //read highscore.sav
49      ifstream fin("highscore.sav");
50      for(int i=1; i<=5; i++)
51      {
52          fin>>li[i].scor;
53          fin.getline(li[i].nume,15);
54          strcpy(li[i].nume,li[i].nume+1);
55      }
56      fin.close();
57      initwindow(700,800,"Candy Crush");
58      //begin
59      loadimages();
60      title();
61      closegraph();
62      return 0;
63  }
```

Loop-ul funcției title() desenează meniul și verifică dacă userul apasă un buton. Dacă se apasă butonul "Tutorial" se va deschide o interfață nouă care explică regulile și proprietățile elementelor, cu posibilitatea întoarcerii la ecranul principal.

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
190 void title()
191 {
192     int x1,y1;
193     drawtitle(); //desenez titlul
194     clearmouseclick(WM_LBUTTONDOWN);
195     while(1)
196     {
197         if(ismouseclick(WM_LBUTTONDOWN)) //daca dau click
198         {
199             x1 = mousex();
200             y1 = mousey(); //pozitia mouse
201             clearmouseclick(WM_LBUTTONDOWN);
202             if(buton(250,450,210,250,x1,y1))
203             {
204                 joc10(); //buton start joc clasic
205                 clearmouseclick(WM_LBUTTONDOWN);
206                 drawtitle();
207                 highscore();
208             }
209             else if(buton(250,450,260,300,x1,y1))
210             {
211                 jocinf(); //buton start joc infinit
212                 clearmouseclick(WM_LBUTTONDOWN);
213                 drawtitle();
214             }
215             else if(buton(250,450,310,350,x1,y1))
216             {
217                 tutorial(); //buton tutorial
218                 clearmouseclick(WM_LBUTTONDOWN);
219                 drawtitle();
220             }
221             else if(buton(250,450,690,730,x1,y1)) return; //buton exit
222         }
223     }
```

În cazul în care userul începe un joc, se apelează una din funcțiile `joc10()` sau `jocinf()`, iar după finalizare se testează obținerea unui high score, caz în care se adaugă în listă în așa fel încât lista să fie crescătoare.

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
66 void jocl0()  
67 {  
68     cleardevice();  
69     miscari=10;  
70     genereaza();  
71     t[0][0]=0;  
72     speed=0;  
73     afisare();  
74     strcpy(x,"JOC CLASIC");  
75     outtextxy(315,22,x);  
76     speed=75;  
77     do  
78     {  
79         getInput();  
80         if(miscari==1) break;  
81         afisare();  
82         setfillstyle(SOLID_FILL,RED);  
83         bar(116,650,590,700);  
84         setcolor(WHITE);  
85         setbkcolor(RED);  
86         strcpy(x,"Se rezolva matricea");  
87         outtextxy(120,665,x);  
88         setbkcolor(BLACK);  
89         do refill();  
90         while (primitivecheck());  
91         miscari--;  
92         updatemiscari();  
93     }  
94     while(miscari>0);  
95     if (miscari<0) return;  
96     setcolor(WHITE);  
97     strcpy(x,"MISCARI EPUIZATE. APASA CLICK PENTRU A CONTINUA");  
98     setfillstyle(0,BLACK);  
99     bar(150,310,550,370);  
100    outtextxy(170,330,x);  
101    while (1) if(ismouseclick(WM_LBUTTONDOWN)) break;  
102    clearmouseclick(WM_LBUTTONDOWN);  
103 }
```

```
105 void jocinf()  
106 {  
107     cleardevice();  
108     miscari=0;  
109     genereaza();  
110     t[0][0]=0;  
111     speed=0;  
112     afisare();  
113     strcpy(x,"JOC INFINIT");  
114     outtextxy(315,22,x);  
115     speed=75;  
116     do  
117     {  
118         getInput();  
119         if(miscari==1) break;  
120         afisare();  
121         setfillstyle(SOLID_FILL,RED);  
122         bar(116,650,590,700);  
123         setcolor(WHITE);  
124         setbkcolor(RED);  
125         strcpy(x,"Se rezolva matricea");  
126         outtextxy(120,665,x);  
127         setbkcolor(BLACK);  
128         do refill();  
129         while (primitivecheck());  
130         miscari++;  
131         updatemiscari();  
132     }  
133     while(miscari>0);  
134     clearmouseclick(WM_LBUTTONDOWN);  
135 }
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

În cadrul acestor funcții, genereaza() calculează un seed pentru funcția rand() utilizând ora curentă și poziția mouse-ului, se generează o matrice, se rezolvă și apoi se initializează scor = 0 și miscari = 10. Se initializează matricea cu elemente aleatorii, iar apoi cât timp matricea nu este rezolvată (lucru testat de funcția primitivecheck()) se vor șterge secvențele 3-in-a-row și se vor deplasa elementele în jos, generându-se elemente noi care cad de deasupra matricii, fără a afișa întreg procesul.

```
749 void genereaza()
750 {
751     int i,j;
752     t[0][0]=-1;
753     speed=0;
754     //random seed
755     time_t time0;
756     time(&time0);
757     srand(mousex()+mousey()+ (int)time0);
758     for(i=1; i<=n; i++) for(j=1; j<=n; j++) t[i][j]=trurand();
759     while (primitivecheck()) refill();
760     S=0;
761     boardinit();
762     t[0][0]=0;
763 }
```

Funcția getInput() înregistrează poziția a două click-uri consecutive și testează mai multe situații:

- Dacă click-ul 1 se află pe unul din butoane se execută funcția butonului
- Dacă click-ul 1 se află în cadrul matricei se selectează elementul apăsat
- Dacă click-ul 2 este în afara tăblii de joc sau nu este adiacent cu elementul inițial se anulează selecția
- Dacă ambele selecții sunt valide se interschimbă elementele și se testează prezența unei secvențe 3-in-a-row în ambele poziții prin funcția check().

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

Cât timp mai sunt mutări disponibile se așteapta selecția jucătorului și se rezolvă matricea rezultată(funcția refill()), lucruri realizate în funcția getInput().

```
422 void refill()
423 {
424     int i,j,k;
425     bool ok=1;
426     while(ok)
427     {
428         ok=0;
429         for(j=1; j<=n; j++)
430         {
431             for(i=n; i>=1; i--)
432             {
433                 k=i;
434                 if(t[i][j]==0)
435                 {
436                     ok=1;
437                     while(k>0)
438                     {
439                         swap(t[k][j],t[k-1][j]);
440                         k--;
441                     }
442                     if(k!=i)
443                     {
444                         t[1][j]=trurand();
445                         break;
446                     }
447                 }
448             }
449         }
450         if(buton(600,630,650,700,mousex(),mousey()) && ismouseclick(WM_LBUTTONDOWN))
451         {
452             speed+=25;
453             itoa(speed,x,10);
454             outtextxy(650,700,x);
455         }
456         else if(ismouseclick(WM_LBUTTONDOWN) && buton(635,665,650,700,mousex(),mousey()) && speed>0)
457         {
458             speed-=25;
459             strcpy(x,"");
460             outtextxy(650,700,x);
461             itoa(speed,x,10);
462             outtextxy(650,700,x);
463         }
464         else if(ismouseclick(WM_LBUTTONDOWN) && buton(64,114,650,700,mousex(),mousey()) )
465         {
466             miscari=-1;
467             return;
468         }
469         clearmouseclick(WM_LBUTTONDOWN);
470         afisare();
471     }
472 }
```

Posibilități de îmbunătățire

- Eficientizarea algoritmilor de găsire a secvențelor 3-in-a-row;
- Identificare secvențelor 3-in-a-row ideale în funcția bestcheck();

Resurse

HARDWARE

Procesor: Intel(R) Core(TM) i5-4460 CPU @ 3.20GHz 3.20 GHz

Memorie RAM: 8.00 GB

Sistem de operare pe 64 de biti

SOFTWARE

Windows 10

PENTRU SCRIEREA PROGRAMULUI

Programul Code Blocks 13.12

Limbajul: C++

Compiler: MinGW (+ librăria libbgi.a)

Alte librării utilizate: graphics2, winbgim

Bibliografie

Borland Graphics Interface (BGI) pentru Windows:

<https://www.cs.colorado.edu/~main/bgi/doc/>

<https://stackoverflow.com/>

Codul sursă

```
#include <fstream>

#include <iostream>

#include <time.h>

#include "graphics.h"

using namespace std;

//n = dimensiunea matricii; t = matrice; S = scor total; stemp = scor pe miscarea curenta; x = string
pentru afisare; mult = multiplier pt score;

int n=9,t[15][15],S,stemp,miscari,mult=2,poly[10],speed=150; char x[300];

void *el[15];

struct hs{int scor; char nume[15];}li[15];

///scene

void joc10(); // incepe un joc cu 10 mutari

void jocinf(); // incepe un joc cu mutari infinite

void tutorial(); // tutorial

void title();

///editare matrice

void testspecial(int i,int j); // testeaza legaturile cu elementele speciale

void process(int x,int y,bool dir); // sterge liniile care sunt minim 3 in a row

int check(int i,int j,bool fa); // verifica daca t[i][j] face parte dintr-o secventa de minim 3 si o scoate
daca fa==1 AICI SE ADAUGA SCORUL

void refill(); // reumple matricea

bool primitivecheck(); // vede daca in toata matricea exista un sir de minim 3

///meta

void loadimages();

bool buton(int X1,int X2,int Y1,int Y2,int x,int y);
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
void genereaza(); // genereaza matricea
void getInput(); // Input
int trurand(); // genereaza chestii random
void updatescor();
void updatemiscari();
void showsc(); // afiseaza leaderbordul
void highscore(); // pune high score nou in leader board
///desene
void boardinit(); // deseneaza bg o data la inceput
void afisare(); // arat elementele matricii
void drawtitle(); // secventa de titlu
void watermark();
///fructe
void sel(int a,int b);
void ori(int a,int b);
void ver(int a,int b);

int main (){
    //read highscore.sav
    ifstream fin("highscore.sav");
    for(int i=1;i<=5;i++) {fin>>li[i].scor; fin.getline(li[i].nume,15); strcpy(li[i].nume,li[i].nume+1);}
    fin.close();
    initwindow(700,800,"Candy Crush");
    //begin
    loadimages();
    title();
    closegraph();
    return 0;
}

///scene
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
void joc10() {
    cleardevice(); miscari=10;
    genereaza();
    t[0][0]=0;
    speed=0;
    afisare();
    strcpy(x,"JOC CLASIC"); outtextxy(315,22,x);
    speed=75;
    do {
        getInput();
        if(miscari== -1) break;
        afisare();
        setfillstyle(SOLID_FILL,RED);bar(116,650,590,700);
        setcolor(WHITE); setbkcolor(RED); strcpy(x,"Se rezolva matricea"); outtextxy(120,665,x);
        setbkcolor(BLACK);
        do refill(); while (primitivecheck()) ;
        miscari--; updatemiscari();
    }while(miscari>0);
    if (miscari<0) return;
    setcolor(WHITE);
    strcpy(x,"MISCARI EPUIZATE. APASA CLICK PENTRU A CONTINUA");
    setfillstyle(0,BLACK);
    bar(150,310,550,370);outtextxy(170,330,x);
    while (1) if(ismouseclick(WM_LBUTTONDOWN)) break;
    clearmouseclick(WM_LBUTTONDOWN);
}

void jocinf() {
    cleardevice(); miscari=0;
    genereaza();
    t[0][0]=0;
```


LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
speed=0;
afisare();
strcpy(x,"JOC INFINIT"); outtextxy(315,22,x);
speed=75;
do {
    getInput();
    if(miscari== -1) break;
    afisare();
    setfillstyle(SOLID_FILL,RED);bar(116,650,590,700);
    setcolor(WHITE); setbkcolor(RED); strcpy(x,"Se rezolva matricea"); outtextxy(120,665,x);
    setbkcolor(BLACK);
    do refill(); while (primitivecheck()) ;
    miscari++; updatemiscari();
}while(miscari>0);
clearmouseclick(WM_LBUTTONDOWN);
}

void tutorial() {
    cleardevice();clearmouseclick(WM_LBUTTONDOWN);
    setcolor(WHITE);
    setfillstyle(1,RED);
    bar(64,650,114,700);
    setcolor(WHITE); setbkcolor(RED); strcpy(x,"Return"); outtextxy(68,667,x); setbkcolor(BLACK);
    strcpy(x,"REGULIE JOCULUI"); outtextxy(290,102,x);
    strcpy(x,"Scopul jocului este de a uni cate trei sau mai multe fructe de acelasi fel pe linie sau coloana."); outtextxy(60,142,x);
    strcpy(x,"Exista cateva elemente speciale care indeplinesc alte functii:"); outtextxy(60,162,x);
    putimage(2*64+1,3*64+1,el[5],COPY_PUT);
    strcpy(x,"Se cupleaza cu orice fruct si atunci cand este distrusa sterge toate");
    outtextxy(200,202,x);
    strcpy(x,"elementele de pe linia si coloana pe care se afla"); outtextxy(200,222,x);
    putimage(2*64+1,5*64+1,el[6],COPY_PUT);
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
    strcpy(x,"Se activeaza atunci cand este apasat de doua ori si distruge un patrat de");
outtextxy(200,322,x);

    strcpy(x,"3x3 in jurul sau, lasand in urma un element destructibil doar de o alta");
outtextxy(200,342,x);

    strcpy(x,"bomba. Elementele sterse astfel nu cresc scorul"); outtextxy(200,362,x);

    putimage(2*64+1,7*64+1,el[7],COPY_PUT);

    strcpy(x,"Acest element nu poate fi mutat si poate fi distrus doar de o bomba");
outtextxy(200,468,x);

    //

    strcpy(x,"Jocul clasic este un joc cu 10 mutari; este singurul al carui scor este salvat in lista high
score-uri."); outtextxy(60,530,x);

    strcpy(x,"Jocul infinit este un joc cu mutari infinite"); outtextxy(60,550,x);

    strcpy(x,"Pentru a selecta elemente ele trebuiesc apasate cu mouse-ul doar cand apare mesajul in
partea"); outtextxy(60,570,x);

    strcpy(x,"de jos a ecranului. Pentru a anula selectia dati click inafara tablei de joc.");
outtextxy(60,590,x);

    strcpy(x,"Pentru a mari sau a micsora viteza programului apasati una dintre cele doua sageti din
dreapta jos"); outtextxy(60,610,x);

    //

    watermark();

    while(1) if(ismouseclick(WM_LBUTTONDOWN)) if(buton(64,114,650,700,mousex(),mousey()))
{clearmouseclick(WM_LBUTTONDOWN);return;}

}

void title() {

    int x1,y1;

    drawtitle(); //desenez titlul

    clearmouseclick(WM_LBUTTONDOWN);

    while(1)

        if(ismouseclick(WM_LBUTTONDOWN)) { //daca dau click

            x1 = mousex(); y1 = mousey(); //pozitia mouse

            clearmouseclick(WM_LBUTTONDOWN);

            if(buton(250,450,210,250,x1,y1)) {joc10(); clearmouseclick(WM_LBUTTONDOWN);
drawtitle(); highscore();} //buton start joc clasic
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
else if(buton(250,450,260,300,x1,y1)) {jocinf(); clearmouseclick(WM_LBUTTONDOWN);
drawtitle();} //buton start joc infinit

else if(buton(250,450,310,350,x1,y1)) {tutorial(); clearmouseclick(WM_LBUTTONDOWN);
drawtitle();} //buton tutorial

else if(buton(250,450,690,730,x1,y1)) return; //buton exit

}

}
```

///editare matrice

```
void destroy(int i,int j) {
    switch (t[i][j]) {
        case 5: {
            //plusbum
            int k;
            S+=5; t[i][j]=0;
            for(k=1;k<=n;k++) {
                S+=t[i][k]+t[j][k];
                destroy(i,k); destroy(k,j);
                if(t[0][0]!=-1) {ori(k,j); ver(i,k);}
            }
            return;
        }
        case 6: {
            //explode
            t[i][j]=0;
            for(int a=-1;a<2;a++)
                for(int b=-1;b<2;b++) {
                    if(t[i+a][j+b]==6) destroy(i+a,j+b);
                    else t[i+a][j+b]=0;
                }
            t[i][j]=7;
            return;
        }
    }
}
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
    }  
    case 7:{return;}  
    default: {S+=1; t[i][j]=0; return;}  
    }  
}
```

```
void testspecial(int i,int j,int &Sl,int &Sv) {  
    int L,V,k; Sl=Sv=0;  
    //afisare();  
    for(int l=1;l<5;l++) {  
        L=V=0;  
        k=1;  
        while(l==t[i+k][j] && i+k<=n){V++;k++;}k=1;  
        while(l==t[i-k][j] && i-k>=1){V++;k++;}  
        k=1;  
        while(l==t[i][j+k] && j+k<=n){L++;k++;}k=1;  
        while(l==t[i][j-k] && j-k>=1){L++;k++;}  
        if(L>=2 || V>=2) {Sv=V; Sl=L; return;}  
    }  
}
```

```
void process(int x,int y,bool dir){  
    //0 = vertical; 1 = orizontal;  
    int k,s=0;  
    if(dir==0){k=1;if(t[0][0]!=-1) ori(x,y);  
        while ((t[x][y]==t[x+k][y] || t[x+k][y]==5) && x+k<=n)  
            {if(t[0][0]!=-1) ori(x+k,y); destroy(x+k,y);s++;k++;}k=1;  
        while ((t[x][y]==t[x-k][y] || t[x-k][y]==5) && x-k>=1)  
            {if(t[0][0]!=-1) ori(x-k,y); destroy(x-k,y);s++;k++;}  
        }  
    else {if(t[0][0]!=-1)ver(x,y);
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
k=1;
while ((t[x][y]==t[x][y+k] || t[x][y+k]==5) && y+k<=n)
    {if(t[0][0]!=-1) ver(x,y+k); destroy(x,y+k);s++;k++;}k=1;
while ((t[x][y]==t[x][y-k] || t[x][y-k]==5) && y-k>=1)
    {if(t[0][0]!=-1) ver(x,y-k); destroy(x,y-k);s++;k++;}
}
//if(t[0][0]!=-1) afisare();
//stemp+=(s+1);
}

int check(int i,int j,bool fa){
    int Sv=0,Sl=0,k=1;
    if(t[i][j]<=0) return 0;
    stemp=0;
    if(t[i][j]<5) {
        //vertical
        k=1;
        while((t[i][j]==t[i+k][j] || t[i+k][j]==5) && i+k<=n){Sv++;k++;}k=1;
        while((t[i][j]==t[i-k][j] || t[i-k][j]==5) && i-k>=1){Sv++;k++;}
        k=1;//orizontal
        while((t[i][j]==t[i][j+k] || t[i][j+k]==5) && j+k<=n){Sl++;k++;}k=1;
        while((t[i][j]==t[i][j-k] || t[i][j-k]==5) && j-k>=1){Sl++;k++;}
    }
    else if(t[i][j]==5) testspecial(i,j,Sl,Sv);
    if(fa) {
        k=t[i][j];
        if(Sv>=2) {process(i,j,0); t[i][j]=k;}
        if(Sl>=2) {process(i,j,1); t[i][j]=k;}
        if(Sv>=2 || Sl>=2) {destroy(i,j); if(t[0][0]!=1) S+=Sl+Sv;}
        if(Sl>=2 && Sv>=2) t[i][j]=5;
        if(t[0][0]==0) updatescor();
    }
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
}  
  
if(Sv>=2 || Sl>=2) return Sv+Sl;  
  
return 0;  
  
}  
  
void refill() {  
    int i,j,k; bool ok=1;  
    while(ok) {ok=0;  
        for(j=1;j<=n;j++) {  
            for(i=n;i>=1;i--) {k=i;  
                if(t[i][j]==0) {ok=1;  
                    while(k>0){swap(t[k][j],t[k-1][j]);k--;}  
                    if(k!=i){t[1][j]=trurand();break;}  
                }  
            }  
        }  
    }  
  
    if(buton(600,630,650,700,mousex(),mousey()) && ismouseclick(WM_LBUTTONDOWN))  
{speed+=25; itoa(speed,x,10); outtextxy(650,700,x);}  
  
    else if(ismouseclick(WM_LBUTTONDOWN) && buton(635,665,650,700,mousex(),mousey()) &&  
speed>0) {speed-=25; strcpy(x," "); outtextxy(650,700,x); itoa(speed,x,10); outtextxy(650,700,x);}  
  
    else if(ismouseclick(WM_LBUTTONDOWN) && buton(64,114,650,700,mousex(),mousey()) )  
{miscari=-1; return;}  
  
    clearmouseclick(WM_LBUTTONDOWN);  
    afisare();  
}  
}  
  
bool primitivecheck(){  
    int i,j,k,l,vertica[10][10],orizont[10][10];bool nr=0;  
    for(i=1; i<=n; i++) for(j=1; j<=n; j++) vertica[i][j]=orizont[i][j]=0;  
    for(i=1; i<=n; i++) for(j=1; j<=n; j++) {  
        if(t[i][j]==0) nr=1;
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
if(t[i][j]!=0 && t[i][j]!=7) {
    if(t[i][j]==5) testspecial(i,j,k,l);
    else {
        k=t[i][j];
        if(!orizont[i][j]){
            l=0; do {l++;}while((t[i+l][j]==k || t[i+l][j]==5) && i+l<10); l--;
            if(l>=2) {nr=1; for(l;l>=0;l--) {orizont[i+l][j]=1; ori(i+l,j);}}
        }
        if(!vertica[i][j]) {
            l=0; do {l++;}while((t[i][j+l]==k || t[i][j+l]==5) && j+l<10); l--;
            if(l>=2) {nr=1; for(l;l>=0;l--) {vertica[i][j+l]=1; ver(i,j+l);}}
        }
    }
}

for(i=1; i<=n; i++) for(j=1; j<=n; j++) if(vertica[i][j] || orizont[i][j]) {destroy(i,j); if(vertica[i][j] &&
orizont[i][j]) t[i][j]=5;}

if(t[0][0]!=-1) {Sleep(speed); afisare();}

return nr;
}

///meta
void loadimages() {
    setactivepage(2);

    //0 - nimic
    setfillstyle(1,YELLOW); bar(0,0,700,800);
    readimagefile("./skins/nimic.jpg",1,1,62,62);
    el[0]=malloc(imagesize(1,1,62,62));
    getimage(1,1,62,62,el[0]);

    //1 - pepene
    setfillstyle(1,BLACK); bar(0,0,64,64);
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
readimagefile("./skins/pepene.jpg",1,1,62,62);  
el[1]=malloc(imagesize(1,1,62,62));  
getimage(1,1,62,62,el[1]);  
//2 - struguri  
setfillstyle(1,DARKGRAY); bar(0,0,64,64);  
readimagefile("./skins/strugure.jpg",1,1,62,62);  
el[2]=malloc(imagesize(1,1,62,62));  
getimage(1,1,62,62,el[2]);  
//3 - pruna  
setfillstyle(1,RED); bar(0,0,64,64);  
readimagefile("./skins/pruna.jpg",1,1,62,62);  
el[3]=malloc(imagesize(1,1,62,62));  
getimage(1,1,62,62,el[3]);  
//4 - portocala  
setfillstyle(1,WHITE); bar(0,0,64,64);  
readimagefile("./skins/portocala.jpg",1,1,62,62);  
el[4]=malloc(imagesize(1,1,62,62));  
getimage(1,1,62,62,el[4]);  
//5 - coroana  
setfillstyle(1,MAGENTA); bar(0,0,64,64);  
readimagefile("./skins/coroana.jpg",1,1,62,62);  
el[5]=malloc(imagesize(1,1,62,62));  
getimage(1,1,62,62,el[5]);  
//6 - bomba  
setfillstyle(1,GREEN); bar(0,0,64,64);  
readimagefile("./skins/bomba.jpg",1,1,62,62);  
el[6]=malloc(imagesize(1,1,62,62));  
getimage(1,1,62,62,el[6]);  
//7 - piatra  
setfillstyle(1,BLUE); bar(0,0,64,64);  
readimagefile("./skins/piatra.jpg",1,1,62,62);
```


LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
el[7]=malloc(imagesize(1,1,62,62));
getimage(1,1,62,62,el[7]);

//8 - sageata stanga
setfillstyle(1,WHITE); bar(0,0,700,800);
readimagefile("./skins/sageata st.jpg",1,1,30,50);
el[8]=malloc(imagesize(1,1,30,50));
getimage(1,1,30,50,el[8]);

//9 - sageata
setfillstyle(1,WHITE); bar(0,0,700,800);
readimagefile("./skins/sageata.jpg",1,1,30,50);
el[9]=malloc(imagesize(1,1,30,50));
getimage(1,1,30,50,el[9]);

//10 - mainbg
setfillstyle(1,BLACK); bar(0,0,700,800);
readimagefile("./skins/title.jpg",1,1,700,800);
el[10]=malloc(imagesize(1,1,700,800));
getimage(1,1,700,800,el[10]);

//11 - jocbg
setfillstyle(1,BLACK); bar(0,0,700,800);
readimagefile("./skins/bgjoc.jpg",1,1,700,800);
el[11]=malloc(imagesize(1,1,700,800));
getimage(1,1,700,800,el[11]);

//
setactivepage(1);
setvisualpage(1);
}

bool buton(int X1,int X2,int Y1,int Y2,int x,int y) {
    //setfillstyle(1,BLUE); bar(X1,Y1,X2,Y2);
    if(x>X1 && x<X2 && y>Y1 && y<Y2) {return 1;}
    return 0;
}
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
}
```

```
void getInput(){
    bool f=1;
    int x1,x2,y1,y2;
    clearmouseclick(WM_LBUTTONDOWN);
    setfillstyle(SOLID_FILL, GREEN); bar(116, 650, 590, 700);
    setcolor(WHITE); setbkcolor(GREEN); strcpy(x, "Apasa pe itemele pe care vrei sa le muti");
    outtextxy(120, 665, x); setbkcolor(BLACK);
    while (f){
        if(ismouseclick(WM_LBUTTONDOWN)) {
            x1 = mousex();
            y1 = mousey();
            clearmouseclick(WM_LBUTTONDOWN);
            if(buton(64, 114, 650, 700, x1, y1)) {miscari=-1; return;}
            else if(buton(600, 630, 650, 700, x1, y1) && speed<900) {speed+=25; itoa(speed, x, 10);
            outtextxy(650, 700, x);}
            else if(buton(635, 665, 650, 700, x1, y1) && speed>0) {speed-=25; strcpy(x, " ");
            outtextxy(650, 700, x); itoa(speed, x, 10); outtextxy(650, 700, x);}
            else {
                x1/=64; y1/=64;
                if (y1>=1 && y1<=n && x1>=1 && x1<=n && t[y1][x1]!=7){
                    sel(x1, y1);
                    while (1) if (ismouseclick(WM_LBUTTONDOWN)) {
                        x2 = mousex();
                        y2 = mousey();
                        clearmouseclick(WM_LBUTTONDOWN);
                        if(buton(64, 114, 650, 700, x2, y2)) {miscari=-1; return;}
                        x2/=64; y2/=64;
                        if (x1==x2 && y1==y2 && t[y2][x2]==6)
                            {destroy(y1, x1); return;}
                    }
                }
            }
        }
    }
}
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
else if (y2>=1 && y2<=n && x2>=1 && x2<=n && abs(x1-x2)+abs(y1-y2)==1 &&
t[y2][x2]!=7){
    if(f==1) {
        f=0;
        swap(t[y1][x1],t[y2][x2]);
        afisare();
        Sleep(150);
        if(!f && !check(y2,x2,0) && !check(y1,x1,0)) {
            swap(t[y1][x1],t[y2][x2]);
            afisare();
            setcolor(WHITE); setbkcolor(GREEN);strcpy(x,"MISCARE
INVALIDA");outtextxy(120,650,x); setbkcolor(BLACK);
            f=1;
            break;
        }
    }
    else {
        mult=2;
        check(y1,x1,1);
        check(y2,x2,1);
        afisare();
        mult=1;
        setfillstyle(1,GREEN);
        bar(120,650,590,667);
        return;
    }
}
else break;
}
else {afisare();
//setfillstyle(SOLID_FILL,BLACK);
setcolor(WHITE); setbkcolor(GREEN);strcpy(x,"MISCARE
INVALIDA");outtextxy(120,650,x); setbkcolor(BLACK);f=1; break;}
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
    }
    }
    }
    }
}

}

void genereaza(){
    int i,j;t[0][0]=-1;
    speed=0;
    //random seed
    time_t time0;
    time(&time0);
    srand(mousex()+mousey()+(int)time0);
    for(i=1; i<=n; i++) for(j=1; j<=n; j++) t[i][j]=trurand();
    while (primitivecheck()) refill();
    S=0;
    boardinit();
    t[0][0]=0;
}

int trurand() {
    time_t time0;
    time(&time0);
    int po=(mousex()+mousey()+rand()+(int)time0);
    if(po%120==0) return 6;
    return po%4+1;
}

void updatescor() {
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
setcolor(WHITE);
strcpy(x,"SCOR :");
outtextxy(110,22,x);
strcpy(x,"");
outtextxy(160,22,x);
itoa(S, x, 10);
if(S>li[5].scor) {setcolor(YELLOW); strcat (x," HS");}
outtextxy(160,22,x);
}
```

```
void updatemiscari() {
    if(miscari>-1) {
        setcolor(WHITE);
        strcpy(x,"MISCARI :"); outtextxy(440,22,x);
        strcpy(x,""); outtextxy(508,22,x);
        itoa(miscari, x, 10); outtextxy(508,22,x);
    }
}
```

```
void showsc() {
    int i;
    system("CLS");
    setfillstyle(1,BLACK);
    bar(235,390,465,620);
    strcpy(x,"----- HIGH SCORES -----");
    outtextxy(250,400,x);
    for(i=1;i<=5;i++) {
        itoa(i,x,10);outtextxy(250,400+i*30,x);
        strcpy(x,".");outtextxy(260,400+i*30,x);
        strcpy(x,li[i].nume);outtextxy(270,400+i*30,x);
        itoa(li[i].scor,x,10);outtextxy(400,400+i*30,x);
    }
}
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
}  
}  
  
void highscore() {  
    int i,j;  
    drawtitle();  
    for(i=1;i<=5;i++){  
        if(li[i].scor<S) {  
            for (j=6;j>i;j--) li[j]=li[j-1];  
            li[i].scor=S; strcpy(li[i].nume,"guest");  
            setcolor(WHITE);  
            strcpy(x,"NOU HIGH SCORE! Scrie-ti numele:");  
            outtextxy(236,580,x); x[0]='\0'; j=-1;  
            while(kbhit()) getch(); //clear buffer  
            while(1) {  
                x[200]=getch();  
                if (x[200]==0) getch(); // caractere speciale( delete, sageti, pgup etc.)  
                else if(x[200]=='\r') break; // enter  
                else if(x[200]==8) { // backspace  
                    x[j--]='\0';  
                    outtextxy(310,600,"");  
                }  
                else if(j<10 && x[200]>31 && x[200]<127) {x[++j]=x[200]; x[j+1]='\0';}  
                if(j<-1) j=-1;  
                outtextxy(310,600,x);  
            }  
            if(strlen(x)>0) strcpy(li[i].nume,x);  
            drawtitle();  
            strcpy(x,"BRAVO!!!");  
            outtextxy(310,600,x);  
        }  
    }  
}
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
        ofstream fout("highscore.sav");for(i=1;i<=5;i++) fout<<li[i].scor<<"
"<<li[i].nume<<"\n";fout.close();

        break;

    }

}

}

///desene

void boardinit () {

    putimage(1,1,el[11],COPY_PUT);

    putimage(600,650,el[8],COPY_PUT);

    putimage(635,650,el[9],COPY_PUT);

    strcpy(x,"Delay : 75 "); outtextxy(597,700,x);

    setfillstyle(SOLID_FILL,YELLOW);bar(60,60,644,644);

    setfillstyle(SOLID_FILL,RED);bar(64,650,114,700);

    setfillstyle(SOLID_FILL,RED);bar(116,650,590,700);

    setcolor(WHITE); setbkcolor(RED); strcpy(x,"Return"); outtextxy(68,667,x); setbkcolor(BLACK);

    watermark();

    updatemiscari(); updatescor();

}

void afisare (){

    int i,j;

    if(t[0][0]==-1) return;

    Sleep(speed);

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

    {

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

            putimage(i*64+1,j*64+1,el[t[j]][i],COPY_PUT);

    }

    Sleep(speed);
```

LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
updatescor(); updatemiscari();
}

void drawtitle() {
    putimage(1,1,el[10],COPY_PUT);
    showsc(); // scor
    setcolor(WHITE);
    //
    setfillstyle(1,BLACK);
    bar(250,690,450,730);
    strcpy(x,"EXIT"); outtextxy(330,702,x);
    //
    setfillstyle(1,BLACK);
    bar(250,250,450,210);
    strcpy(x,"JOC CLASIC"); outtextxy(310,222,x);
    //
    setfillstyle(1,BLACK);
    bar(250,300,450,260);
    strcpy(x,"JOC INFINIT"); outtextxy(310,272,x);
    //
    setfillstyle(1,BLACK);
    bar(250,350,450,310);
    strcpy(x,"TUTORIAL"); outtextxy(315,322,x);
    //
    watermark();
}

void watermark() {
    setbkcolor(BLACK);
    strcpy(x,"CANDY CRUSH"); outtextxy(10,782,x);
}
```


LUCRARE DE ATESTAT (INFORMATICA) : CANDY CRUSH

```
strcpy(x,"proiect realizat de Axinte Cosmin");outtextxy(332,783,x);

strcpy(x,"v1.5"); outtextxy(4,4,x);

}

///fructe

void sel(int a,int b) {

    a=a*64; b=b*64;

    setcolor(BLUE);

    rectangle(a+1,b+1,a+62,b+62); rectangle(a+2,b+2,a+61,b+61); rectangle(a+3,b+3,a+60,b+60);
rectangle(a+4,b+4,a+59,b+59);

}

void ori(int b,int a) {

    a=a*64; b=b*64;

    setfillstyle(1,GREEN);

    bar(a+20,b+2,a+44,b+62);

}

void ver(int b,int a) {

    a=a*64; b=b*64;

    setfillstyle(1,GREEN);

    bar(a+2,b+20,a+62,b+44);

}
```