

Colegiul National "Gheorghe Sincai"

Sesiunea: mai 2021

# LUCRARE PENTRU ATESTAREA COMPETENȚELOR PROFESIONALE

---

## SNAKE

---



Elev: Stefan Diana

Profesor indrumator: Antici Mirela

# CUPRINS:

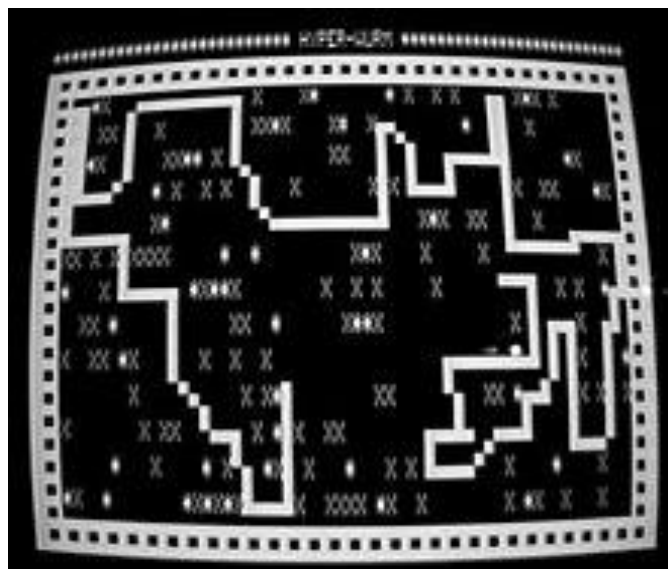
1. Introducere ..... pag. 3
2. Limbajul C++ ..... pag. 4
3. Modalitati de utilizare ..... pag. 5
4. Structura programului ..... pag. 9
5. Concluzii si imbunatatiri ..... pag. 12
6. Bibliografie ..... pag. 13

# INTRODUCERE

Proiectul meu consta în programarea unui joc (limbajul C++), afișat pe consola, ce porneste de la jocul clasic “Snake”, caruia i-am adăugat diferite elemente, cu scopul de a stârni interes în rândul jucătorilor și de a îl personaliza. Totodată, constituie rezultatul curiozității mele legate de o zonă de programare, diferită de cea experimentată până în acest moment.

Motivul alegerii acestei teme derivă atât din dorința de documentare în legătură cu acest proiect, cât și din cea de încercare a realizării sale.

“Snake” este numele comun pentru un concept de joc video în care jucătorul manevrează o linie care crește în lungime, linia în sine fiind un obstacol principal. Conceptul a luat naștere în jocul arcade Blockade din 1976, iar ușurința implementării Snake a dus la sute de versiuni (dintre care unele au cuvântul șarpe sau vierme în titlu) pentru multe platforme. După ce o variantă a fost preîncărcată pe telefoanele mobile Nokia în 1998, a existat o reapariție a interesului pentru conceptul de șarpe, deoarece a găsit un public mai mare. Există peste 420 de jocuri de tip Snake doar pentru iOS.



Jocul clasic “Snake” este primul joc video cu care am avut contact și va rămâne întotdeauna o sursă de nostalgie, motiv pentru care mi-am dorit ca acesta să reprezinte punctul de plecare al proiectului meu, pe care, mai apoi, l-am îmbunătățit.

Acest joc poate fi recunoscut de orice om, indiferent de generația din care face parte, reprezentând, astfel, o oportunitate a retrairii unor momente pline de bucurie.

# LIMBAJUL C++

Limbajul C++ a fost inventat de către [Bjarne Stroustrup](#) în 1979, ca o extindere a limbajului C. Limbajul C a fost inventat în 1969-1973 de către [Dennis Ritchie](#) pentru a realiza sistemul de operare Unix. Astfel, aproape toate programele scrise în C pot fi compilate în C++, eventual cu foarte puține modificări.

## Limbaje de programare

---

**Limbajele de programare** sunt limbaje asemănătoare cu limbajul uman. Conțin cuvinte (destul de puține), semne de punctuație, operații matematice și au reguli de scriere. Programele care rulează pe orice calculator au fost scrise într-un limbaj de programare. Există numeroase limbaje programare, precum C, C++, Pascal, Java, Python, PHP, Javascript, etc.

Programul scris într-un limbaj de programare se numește **program sursă** și trebuie traduse într-un limbaj pe care îl înțelege procesorul, numit **cod mașină**, sau **program executabil**. Pentru anumite limbaje de programare operația de traducere se numește **compilare** (cazul lui C, C++, Pascal, etc.), pentru alte limbaje (PHP, Python, Javascript, etc.) operația de traducere se numește **interpretare**.

Limbajul C++ este un limbaj compilat pentru că este o combinație între un limbaj procedural și un limbaj de programare orientat pe obiecte. De asemenea, este un limbaj scalabil, cu ajutorul căruia se pot crea aplicații ce consumă multe resurse, precum aplicații pentru desktop sau jocuri complexe la nivel grafic. Etapele scrierii unui program în C++ sunt:

- editarea programului C++; se obține fișierul sursă, cu extensia **.cpp**
- compilarea fișierului sursă; aici se verifică corectitudinea sintactică a programului (corectitudinea cuvintelor folosite, prezența semnelor de punctuație, etc.); dacă programul este corect sintactic, se va obține fișierul obiect, cu extensia **.o** sau **.obj**
- editarea de legături; se stabilesc legături între fișierul obiect curent și alte fișiere obiect, ale programatorului sau incluse în compilator; în urma acestei etape se obține programul executabil. În Windows, fișierele executabile au extensia **.exe**;
- programul executabil poate fi lansat în execuție (rulat).

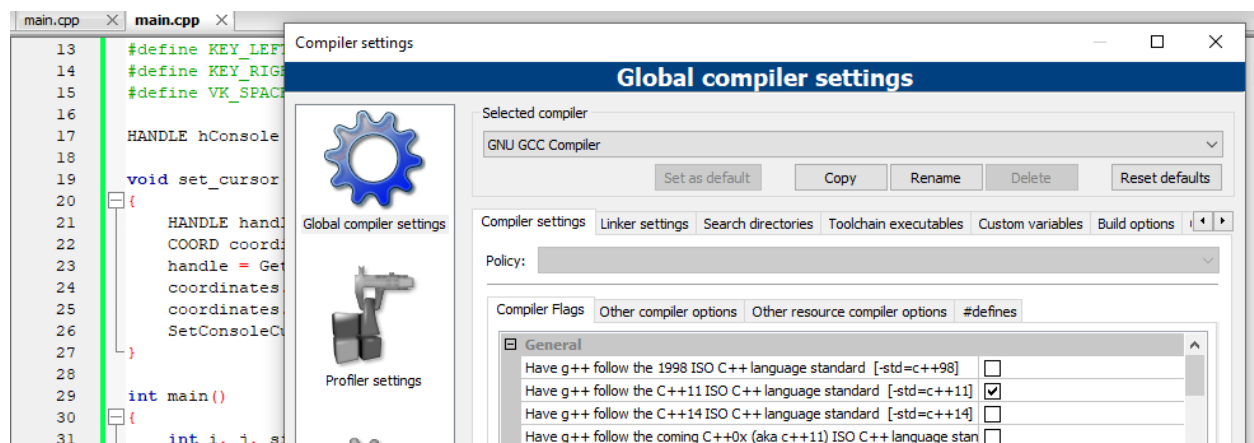
Pentru a crea programe în C++ este nevoie de un mediu de dezvoltare integrat (integrated development environment –IDE) care va ușura procesul de programare, va compila codul și îl va rula. Cel mai utilizat IDE pentru programarea C++ este Code::Blocks.

De asemenea, este un limbaj foarte rapid, care permite programarea procedurală pentru funcții intensive ale CPU-ului. Tocmai de aceea este foarte des folosit pentru engine-urile jocurilor. Ubisoft și Blizzard sunt doi dintre marii producători de jocuri care folosesc limbajul C++ în game development, unul dintre jocurile faimoase dezvoltate cu acest limbaj fiind Assassin's Creed.

Majoritatea sistemelor de operare de la Microsoft (Microsoft Windows 95, 98, 2000, XP, office, Internet Explorer) sunt dezvoltate cu ajutorul limbajului C++ și, de asemenea, câteva părți din sistemul de operare de la Apple.

## MODALITATI DE UTILIZARE

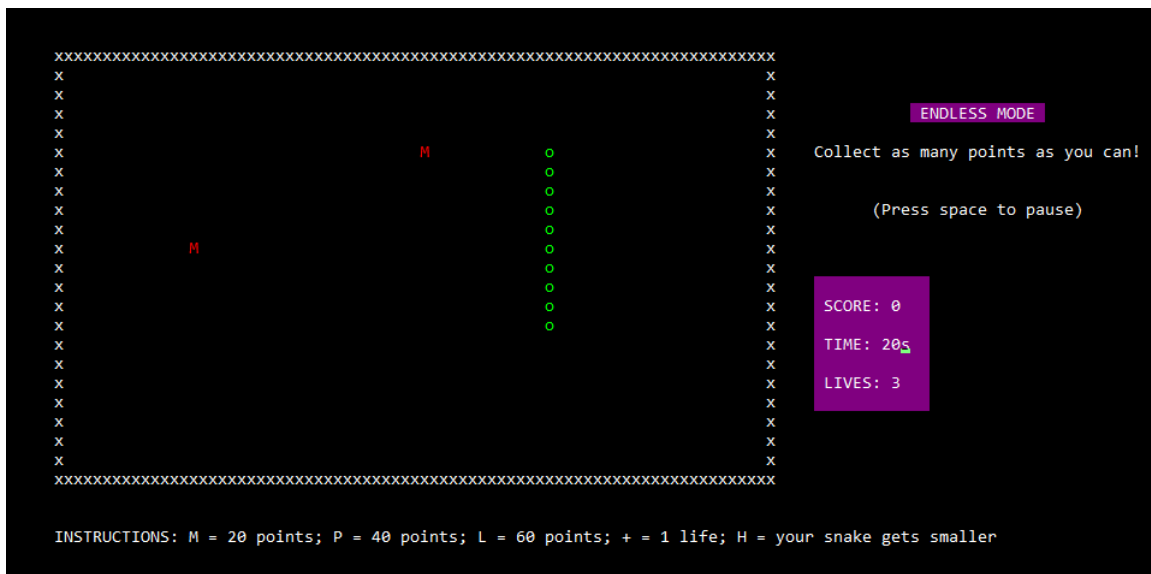
În cazul în care, la compilarea programului, apare mesajul *“#error This file requires compiler and library support for the ISO C++ 2011 standard. This support is currently experimental, and must be enabled with the -std=c++11 or -std=gnu++11 compiler options.”*, este necesară următoarea setare:



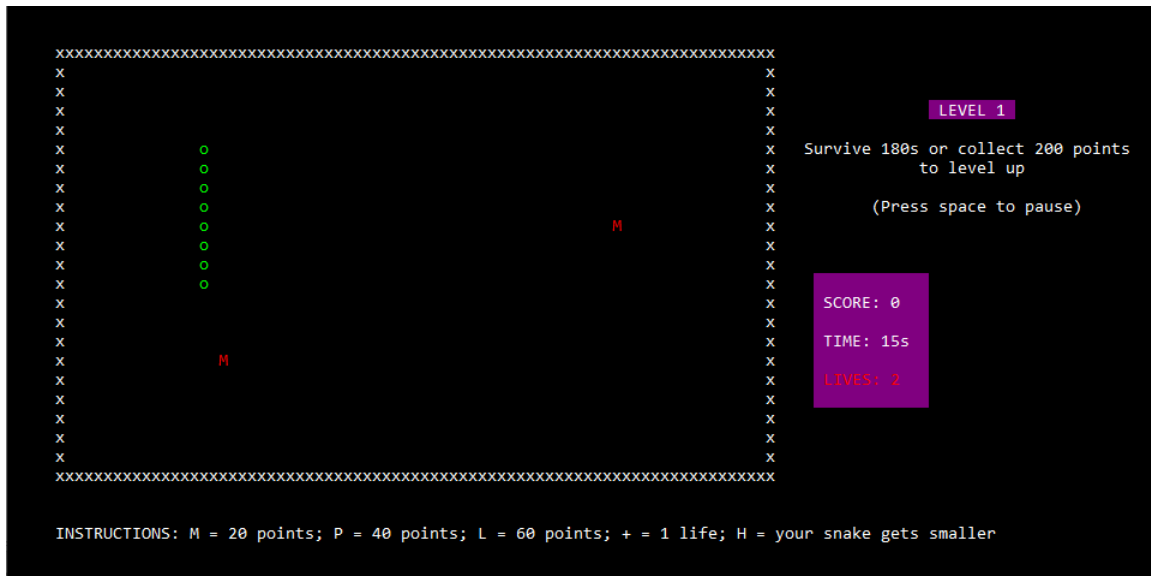
La inceputul jocului, utilizatorului i se ofera doua posibilitati (una ce constituie jocul clasic, fara final, iar cealalta, o varianta a jocului ce poate fi castigata prin parcurgerea integrala a 7 nivele distincte), care pot fi accesate conform instructiunilor afisate (sagata stanga corespunde primei variante, iar cea dreapta, celei de-a doua variante).



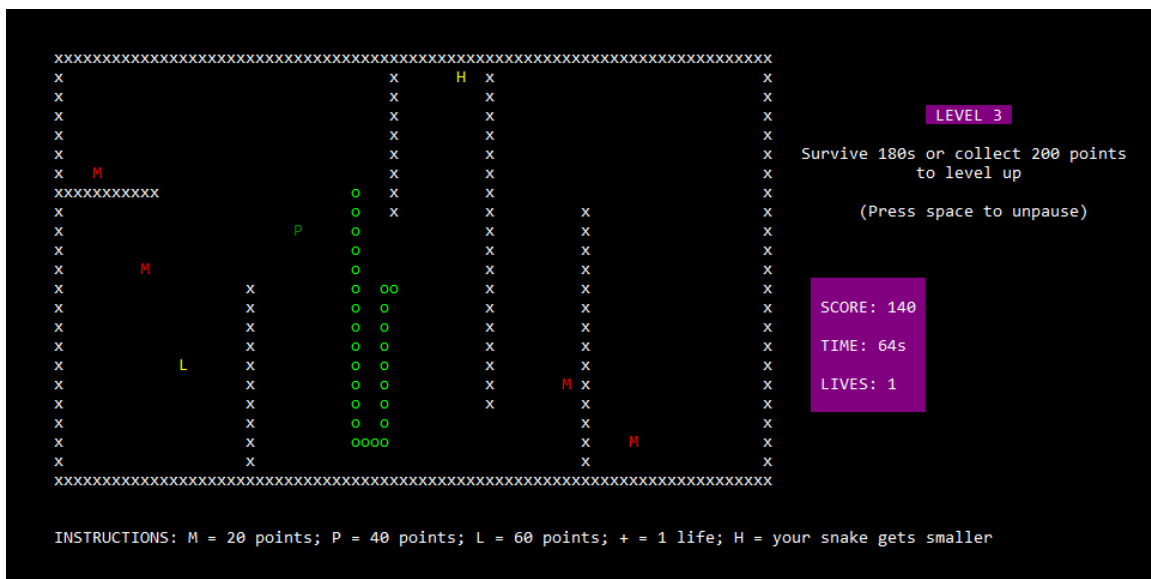
In varianta “endless”, jucatorului i se afiseaza scorul curent, timpul si numarul vietilor.



Similar se intampla in cazul celei de-a doua variante, cu mentiunea faptului ca acumularea a 180 de puncte sau atingerea unui timp de 3 minute au ca rezultat castigarea nivelului curent.



Pe parcursul jocului, in orice moment, pot fi intalnite: fructe (M, P, L), care au ca scop acumularea punctelor, conform instructiunilor; vieti (+), care dispar dupa un anumit interval de timp; litera “H” care, intalnita, are ca efect injumatatirea sarpelui, facandu-l mai usor de manevrat. De asemenea, apasarea tastei “Space” conduce la intreruperea temporara a jocului, reluarea acestuia fiind generata de o actiune similara.



Ca recompensa pentru atingerea nivelului 4, jucatorul primește posibilitatea alegerii culorii sarpelui sau, conform listei afisate:

```
CONGRATS! :) You have reached level 4. Now you can choose your snake's colour:
(1) DARK BLUE
(2) DARK GREEN
(3) DARK CYAN
(4) DARK RED
(5) DARK MAGENTA
(6) DARK YELLOW
(7) DARK GRAY
(8) GRAY
(9) BLUE
(10) GREEN
(11) CYAN
(12) RED
(13) MAGENTA
(14) YELLOW
(15) WHITE
(16) BLACK
```

Nivelul cel mai ridicat de dificultate se inregistreaza la nivelul 7 (nivelul final), in care, spre deosebire de nivelele anterioare, apare un perete miscator, care duce la pierderea unei vieti in momentul in care este intersectat.

[illegible]

Finalul jocului corespunde afisarii unuia dintre mesajele urmatoare (ale caror culoare alterneaza intre rosu si alb), in functie de rezultatul obtinut:

**YOU WON!!!**

YOU LOST!!!



# STRUCTURA PROGRAMULUI

Dintre secvențele programului, pot fi amintite următoarele:

```

while(1)
{
    switch((c=getch()))
    {
        case KEY_RIGHT:
        {
            case KEY_LEFT:
            {
                return 0;
            }
        }
    }
}

```

//Programul este impartit in 2 cazuri, corespunzatoare optiunilor initiale afisate.

```

if(nivel>=4)
{
    switch (colour)
    {
        case 1:
            SetConsoleTextAttribute(hConsole, FOREGROUND_BLUE);
            break;
        case 2:
            SetConsoleTextAttribute(hConsole, FOREGROUND_GREEN);
            break;
        case 3:
            SetConsoleTextAttribute(hConsole, FOREGROUND_GREEN | FOREGROUND_BLUE);
            break;
        case 4:
            SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
            break;
        case 5:
            SetConsoleTextAttribute(hConsole, FOREGROUND_RED | FOREGROUND_BLUE);
            break;
        case 6:
            SetConsoleTextAttribute(hConsole, FOREGROUND_RED | FOREGROUND_GREEN);
            break;
        case 7:
            SetConsoleTextAttribute(hConsole, FOREGROUND_RED | FOREGROUND_GREEN | FOREGROUND_BLUE);
            break;
        case 8:
            SetConsoleTextAttribute(hConsole, FOREGROUND_INTENSITY);
            break;
        case 9:
            SetConsoleTextAttribute(hConsole, FOREGROUND_BLUE | FOREGROUND_INTENSITY);
            break;
        case 10:
            SetConsoleTextAttribute(hConsole, FOREGROUND_GREEN | FOREGROUND_INTENSITY);
            break;
    }
}

```

// Secvența corespunderii optiunilor de culori pe care jucatorul le primește la atingerea nivelului 4.

```

if(pause)
{
    set_cursor(89,10); //move text cursor to particular screen coordinate
    cout<<"(Press space to unpause)";
    pause=0;
    contor=0;
    while(pause==0)
    {
        if(contor>100000000)
            pause=GetAsyncKeyState(VK_SPACE);
        contor++;
    }
    set_cursor(89,10); //move text cursor to particular screen coordinate
    cout<<" ";
    set_cursor(90,10); //move text cursor to particular screen coordinate
    cout<<"(Press space to pause)";
}

```

// Secventa corespunde optiunii intreruperii temporare a jocului, precum si a celei de reluare a acestuia.

```

if(runda%60==10)
{
    fructi=rand()%21+3; //generate a random number
    fructj=rand()%73+6; //generate a random number

    while(a[fructi][fructj]!='o' || a[fructi][fructj]!='x')
    {
        fructi=rand()%21+3; //generate a random number
        fructj=rand()%73+6; //generate a random number
    }

    a[fructi][fructj]='M';
    SetConsoleTextAttribute(hConsole, FOREGROUND_RED | FOREGROUND_INTENSITY); //change text color to red
    set_cursor(fructj,fructi); //move text cursor to particular screen coordinate
    cout<<a[fructi][fructj];
}

if(runda>300 && runda%150==30)

```

// Coordonatele fructelor se stabilesc prin generarea aleatorie a unor valori cuprinse in matricea corespunzatoare jocului si se asigura inexistentia peretilor sau a corpului sarpelui pe pozitia respectiva, dupa care fructul este afisat.

```

if(a[si][sj]=='H')
{
    for(i=poz-1; i<=poz-1/2; i++)
    {
        a[vi[i]][vj[i]]=' ';
        set_cursor(vj[i],vi[i]);
        cout<<' ';
    }
    l=1/2;
}

```

// Secventa are rolul de a injumatati corpul sarpelui la intalnirea literei “H”.

```

if(nivel==7 && contor%2==0)
{
    SetConsoleTextAttribute(hConsole, FOREGROUND_BLUE | FOREGROUND_GREEN | FOREGROUND_RED | FOREGROUND_INTENSITY);

    if(directie==1)
    {
        if(a[cs-1][20]!='x')
        {
            set_cursor(20,cs-1);
            a[cs-1][20]='x';
            cout<<a[cs-1][20];
            set_cursor(20,cj);
            a[cj][20]=' ';
            cout<<a[cj][20];
            cj--;
            cs--;
        }
        else
            directie=-1;
    }
    else
    {
        if(a[cj+1][20]!='x')
        {
            set_cursor(20,cj+1);
            a[cj+1][20]='x';
            cout<<a[cj+1][20];
            set_cursor(20,cs);
            a[cs][20]=' ';
            cout<<a[cs][20];
            cj++;
            cs++;
        }
        else
            directie=1;
    }
}
}

```

// Secventa stabileste directia miscarii peretelui din nivelul 7 si realizeaza modificarile corespunzatoare.

```

if(ok==1)
    Sleep(200); //sleeps for 0.2 seconds

```

// Secventa verifica daca jocul nu s-a terminat si, in caz afirmativ, incetinesc afisarea modificarilor in consola, pentru a oferi senzatia de miscare si timp de reactie jucatorului.

```

while(1)
{
    set_cursor(50,10);
    SetConsoleTextAttribute(hConsole, FOREGROUND_RED | FOREGROUND_INTENSITY); //change text color to red
    cout<<"YOU WON!!!";
    Sleep(150); //sleeps for 0.15 seconds
    set_cursor(50,10);
    SetConsoleTextAttribute(hConsole, FOREGROUND_BLUE | FOREGROUND_GREEN | FOREGROUND_RED); //change text color to white
    cout<<"YOU WON!!!";
    Sleep(150); //sleeps for 0.15 seconds
}

```

// Secventa afiseaza mesajul “YOU WON!!!”, in culori alternative, in cazul parcurgerii cu succes a tuturor nivelelor. Similiar se intampla si in cazul nereusitei, cand mesajul afisat este “YOU LOST!!!”.

# CONCLUZII SI IMBUNATATIRI

**Pekka Rantala, Chief Marketing Officer HMD Global**, a spus:

*“Jocul Snake a fost cel mai popular joc din istoria telefoanelor Nokia; de fapt, adevărata origine a jocurilor mobile poate fi descoperită la un terminal Nokia lansat în 1997. Jocul a rămas extrem de popular până în zilele noastre, nu numai pentru că generează un puternic sentiment de nostalgie, ci pentru că el s-a constituit în prima experiență de gaming mobil pentru mulți consumatori.”*

“Snake” ramane un joc de referinta in istoria telefoanelor mobile, asa ca nu e de mirare faptul ca exista aplicatii “Snake” chiar si pentru smartphone-urile care domina astazi piata.

Tocmai de aceea, consider ca imbunatatirile ce pot fi aduce acestui joc clasic sunt nelimitate, fara a se opri din a oferi satisfactie si entuziasm jucatorilor, pornind de la adaugarea unor noi optiuni, prin intalnirea carora apar diferite avantaje, si pana la impunerea unor noi reguli, cu scopul cresterii nivelului de dificultate.

In acest sens, o idee pe care urmeaza sa o implementez pentru a imi imbunatati programul este posibilitatea de a acumula fructe, ce provoaca aparitia pe consola a unor intrebari de cultura generala, generate in mod aleatoriu, alaturi de variantele de raspuns corespunzatoare. In cazul in care jucatorul ofera raspunsul corect, acesta va beneficia de diferite avantaje (acumularea unor vieti, a unor puncte sau castigarea nivelului curent, in functie de dificultatea intrebarii), iar, in caz contrar, sarpele se va mari, scazandu-i considerabil sansele unei victorii.

Exista o multime de variante similare de imbunatatire a jocului, prin adaugarea unor astfel de elemente ce captiveaza consumatorii.

Totodata, programul poate fi transformat intr-o aplicatie mobila, pentru a facilita accesul doritorilor.

# BIBLIOGRAFIE

<https://codecool.com/ro/blog/ghid-c-incepatori/>

<https://www.pbinfo.ro/articole/59/introducere-in-cpp>

<https://gist.github.com/bmatusiak/5489222>

<https://stackoverflow.com/questions/10401724/move-text-cursor-to-particular-screen-coordinate>

<https://stackoverflow.com/questions/20222277/delete-key-press-historygetkeystate-c>

<https://stackoverflow.com/questions/44481821/is-it-possible-to-change-text-color-and-background-color-in-the-console-to-diffe>

<https://www.softwaretestinghelp.com/cpp-sleep/>

<https://stackoverflow.com/questions/1658386/sleep-function-in-c>

<https://www.cplusplus.com/reference/cstdlib/rand/>

[https://www.w3schools.com/cpp/cpp\\_switch.asp](https://www.w3schools.com/cpp/cpp_switch.asp)

<https://stackoverflow.com/questions/15737495/c-execute-a-while-loop-until-a-key-is-pressed-e-g-esc>

<https://forums.fedoraforum.org/archive/index.php/t-147415.html>

Cartea:

*“Invata singur programarea jocurilor in 24 de lectii”,*

Michael Morrison, editura Corint (2005)