

# Introducere în programare - curs și laborator - suport electronic

(C) 2020-2021 Bogdan Pătruț

## Proiecte

## Meniu

[Curs 1](#) | [Curs 2](#) | [Curs 3](#) | [Curs 4](#) | [Curs 5](#) | [Curs 6](#) | [Curs 7](#) | [Curs 8](#) | [Curs 9](#) | [Curs 10](#) | [Curs 11](#) | [Curs 12](#) | [Curs 13](#) | [Curs 14](#)  
[Laborator 1](#) | [Laborator 2](#) | [Laborator 3](#) | [Laborator 4](#) | [Laborator 5](#) | [Laborator 6](#) | [Laborator 7](#) | [Laborator 8](#) | [Laborator 9](#) | [Laborator 10](#) | [Laborator 11](#) | [Laborator 12](#) | [Laborator 13](#) | [Laborator 14](#)

[Pagina principala](#) |

### A. Lista de proiecte pentru anul universitar 2021-2022

Mai jos sunt date proiectele ce vor fi asignate echipelor de studenți din anul I, care fac pentru prima dată aceasta disciplină, nu sunt repetenți, nu repetă disciplina, nu sunt reînmatriculați sau cursanți sau studenți Erasmus sau se încadrează în vreo alfel de situație specială.

**Lista de proiecte se actualizează permanent.**

Studenții care, dintr-un motiv sau altul, repetă disciplina, precum și cursanții și studenții reinmatriculați vor face singuri un proiect la alegere sau vor prezenta proiectul realizat de ei anul trecut.

Pentru fiecare proiect sunt precizate câteva elemente simple, descrierea completă a proiectului făcându-se verbal, de către titularul de curs, către fiecare echipă în parte. De asemenea, colaboratorii pot veni cu cerințe suplimentare privind complexitatea proiectului și modul de implementare.

### 1. Gobblet Gobblers

Implementați o variantă 2D a jocului Gobblet Gobblers.

▼ Indicații

- <https://www.youtube.com/watch?v=F8F29jfZBRo>
- <https://www.raftulcujocuri.ro/strategie/583-gobblet-gobblers-cu-piese-de-plastic-blue-orange-sua-recomandat-de-mind-lab-904369g.html>

- <https://www.quora.com/How-would-you-modify-the-game-Tic-tac-toe-to-make-it-more-complex-but-without-changing-the-character-of-the-game-too-much>

## 2. Jump In

Implementați o variantă 2D / 3D a jocului Jump In

▼ Indicații

- <https://www.juland.ro/jump-in.html>
- <https://www.youtube.com/watch?v=8sEoYzcmOfc>

## 3. Pirates Hide and Seek

Implementați (2D) jocul **Pirates Hide and Seek**.

▼ Indicații

- <https://www.amazon.co.uk/Smart-Games-Pirates-Hide-Brainteaser/dp/B004YBWGC6>
- si <https://www.youtube.com/watch?v=bJ4QjQ2Rj0o>

## 4. Bridg-It

Proiectul constă în implementarea unui joc de strategie "**Bridg-It**", așa cum este descris acesta în cartea [Martin Gardner - Entertaining Mathematical Puzzles](#), începând cu pagina 59.

## 5. Împărțirea careului

Implementați un joc pentru a rezolva probleme de genul celor de la pagina 121 din cartea [Nicolae Oprișiu - Mai în glumă, mai în serios](#)

## 6. Raliu auto

Implementați un joc **Raliu auto**, așa cum este descris el în cartea „[Olimpiada jocurilor raționale](#)”.

## 7. Segmente

Implementați un joc **Segmente**, așa cum este descris el în cartea „[Olimpiada jocurilor raționale](#)”.

## 8. Război în 8

Implementați un joc **Război în 8**, așa cum este descris el în cartea „[Olimpiada jocurilor raționale](#)”.

## 9. Cinci în linie

Implementați un joc **Cinci în linie**, așa cum este descris el în cartea „[Olimpiada jocurilor raționale](#)”.

## 10. Bonol

Implementați un joc de strategie între doi jucători, ca cel descris în cartea [Gheorghe Paun - Intre matematica si jocuri \(1986\).pdf](#) la pagina 46 jos si continuand pe paginile urmatoare. Jocul se mai numeste si L-game, vedeti aici: [https://en.wikipedia.org/wiki/L\\_game](https://en.wikipedia.org/wiki/L_game)

▼ Vedeti aici câteva indicații

<http://www.chokleong.com/2015/08/03/edward-de-bono-l-game/> [https://play.google.com/store/apps/details?id=org.cocos2dx.lgame&hl=en\\_US](https://play.google.com/store/apps/details?id=org.cocos2dx.lgame&hl=en_US)

## 11. Solitarul

Implementați un joc de strategie între doi jucători, ca cel descris în cartea [Gheorghe Paun - Intre matematica si jocuri \(1986\).pdf](#) la pagina 133 si continuand pe paginile urmatoare.

▼ Informații

Următoarea lucrare de masterat se refera la jocul Solitarul si altele similare: <https://diglib.tugraz.at/download.php?id=5b6d297eea357&location=browse&fbclid=IwAR3FSuFUkcCTI5TKCvuhkxXWyfAScfW0Xt5lOExBmfBkhXUUGQ-dnihPNM0> Altă carte ce trateaza acest joc (vezi pagina 61): [http://www.milisoft.ro/doc/EDU/eScoala/zz.%20Bibliografie/A.P.%20Domoread%20-%20Jocuri%20si%20probleme%20distractive%20de%20matematica%20\(1965\).pdf](http://www.milisoft.ro/doc/EDU/eScoala/zz.%20Bibliografie/A.P.%20Domoread%20-%20Jocuri%20si%20probleme%20distractive%20de%20matematica%20(1965).pdf)

## 12. Colțul negru

Proiectul constă în implementarea unui joc de strategie **Colțul negru**, așa după cum este descris acesta în cartea „[Olimpiada jocurilor raționale](#)”, începând cu pagina 17.

## 13. Câinii și vulpea

Implementați un joc de strategie între doi jucători **Câinii și vulpea**, ca cel descris în cartea „[Olimpiada jocurilor raționale](#)”.

## 14. Smart Farmer

Proiectul constă în implementarea 2D a jocului Smart Farmer, așa cum este el descris aici:

- <https://www.youtube.com/watch?v=fqhscfrEUc>
- <https://www.smartgames.eu/uk/one-player-games/smart-farmer>

## 15. Eval - Evaluator matematic și reprezentarea grafică a arborelui asociat expresiei

Proiectul presupune implementarea în C/C++ (fără POO) a unui evaluator de expresii algebrice pentru un limbaj de programare (de exemplu Basic, care are mai puțini operatori).

Apoi, plecând de la expresia algebrică, se va crea arborele asociat acestei expresii algebrice, iar apoi acesta va fi reprezentat grafic, cu diferite opțiuni, pe ecranul calculatorului.

### ▼ Informații

Vezi pagina 103 din lucrarea „*Învățați limbajul Pascal în 12 lecții*”, [https://www.edusoft.ro/pascal12/cap5\\_pascal12.pdf](https://www.edusoft.ro/pascal12/cap5_pascal12.pdf) **Input :** un șir de caractere, care ar putea reprezenta o expresie în limbajul Basic, folosind:

- parantezele ( și );
- operatorii: +, -, /, \*, <>, =, <=, >=, or, and, xor, not, mod, \;
- funcțiile matematice sin, cos, tan, sqrt, round etc, conform <https://docs.microsoft.com/en-us/dotnet/visual-basic/language-reference/functions/math-functions>
- constante reale, întregi sau logice (true, false);
- identificatori de variabile definite prin șiruri de litere și cifre, în care prima este literă;
- valori pentru fiecare dintre variabilele date prin identificatori.

### Output :

- dacă expresia este corectă din punct de vedere sintactică, atunci va returna valoarea expresiei date;
- dacă expresia nu este corectă, atunci va returna pozițiile unde se găsesc erorile sintactice, precum și tipul erorilor

### ► Exemplu

## 16. Grafic - Graficul unei funcții

Plecând de la expresia unei funcții, dată sub formă de șir de caractere, se va face un program care să deseneze graficul funcției pe un anumit interval, punând în evidență eventualele puncte de discontinuitate, asimptotele verticale, orizontale, oblice, punctele de inflexiune, graficul funcției derivate, calcule de integrale etc.

### ▼ Indicații

Atenție! Se va folosi evaluarea funcției într-un punct, creându-se astfel un evaluator matematic (vezi și proiectul 15). Observație. Se poate pleca de la explicațiile din programul următor: <https://www.edusoft.ro/fisiere/grafic.zip>

## 17. Deriv - Derivare formală

Se dă o funcție (sub forma unui șir de caractere) și se cere să se obțină derivata sa simbolică (formală).

### ▼ Idee

Se poate transforma expresia funcției în forma postfixată, se creează arborele binar asociat expresiei, se obține din acesta arborele derivatei și apoi se generează noua expresie, după câteva simplificări ale arborelui derivat). Vezi și proiectele **Evaluator matematic**, **Vizualizator de arbore asociat unei expresii algebrice** și **Bibliotecă de operații cu arbori binari**.

▼ Informații

Vezi pagina 169 din lucrarea mea „Învățați limbajul Pascal în 12 lecții”: [https://www.edusoft.ro/pascal12/cap8\\_pascal12.pdf](https://www.edusoft.ro/pascal12/cap8_pascal12.pdf)

## 18. InterSchem - Vizualizator și interpretor de scheme logice

Să se realizeze un vizualizator și interpretor de scheme logice, precum **Interschem**: [https://www.edusoft.ro/fisiere/interschem\\_patrut.zip](https://www.edusoft.ro/fisiere/interschem_patrut.zip) Vor fi oferite detalii ulterior.

## 19. Electron - Vizualizator de scheme electronice

Să se proiecteze un vizualizator de scheme electronice, ce conține tranzistori, diode, condensatori, surse de alimentare etc. Dacă se fac și verificări și calcule pe această schemă electronică, cu atât mai bine.

▼ Un program de pornire

[https://www.edusoft.ro/fisiere/electron\\_bp.zip](https://www.edusoft.ro/fisiere/electron_bp.zip)

Prezentare video: <https://www.edusoft.ro/fisiere/video-electron.mp4> Pentru a testa programul, instalați DOS BOX și Borland Pascal. La laborator vor fi oferite detalii.

## 20. BibMat - Bibliotecă de lucru cu tablouri (vectori și matrice)

Se vor implementa funcții de operare cu tablouri uni- și bidimensionale. Atenție! Toate operațiile vor fi vizualizate grafic în timp ce se desfășoară.

▼ Exemple de operații de implementat

- citire de la tastatură sau din fișiere;
- afișare pe ecran sau în fișiere;
- adunare, scădere, înmulțire, împărțire, ridicare la putere matrice;
- obținerea transpusei și inversei unei matrice;
- obținerea determinantului unei matrice;
- operații cu vectori: suma elementelor din vector, produsul lor, sortare vector crescător sau descrescător (prin diferiți algoritmi), *shiftare* vectori la stânga sau la dreapta etc;
- înmulțirea unui vector cu un scalar;
- operații cu matrice: suma elementelor de pe diagonala principală, de sub diagonala principală, de pe diagonala secundară, de sub diagonala secundară etc.

## 21. BibLis(t). Bibliotecă de operații cu liste

Proiectul presupune programarea în C++ a unui unei biblioteci de funcții pentru operații cu liste implementate dinamic (simplu înlănțuite, dublu-înlănțuite, stive, cozi).

Atenție! Toate operatiile vor fi vizualizate grafic în timp ce se desfășoară.

Se va urmări îndeaproape teoria legată de aceste structuri de date.

**Vor fi incluse noi proiecte aici. Asocierea proiectelor către echipe va fi făcută în săptămâna a IX-a.**

## 22. CalcRO - Calculator în limba română

Programul va putea face calcule cu operatorii aritmetici sunt cei aritmetici (+, -, \*, /), plecând de la o propoziție interogativă în limba română, de genul *"Cat este suma dintre trei, saptezeci si produsul lui douăzeci cu patru sute cincisprezece?"* sau *"Care este rezultatul inmultirii lui zece mii opt sute saispzezece cu diferența dintre două milioane și paisprezece mii cincizeci și opt?"*. **Input:** un șir de caractere, conținând o întrebare în limba română despre o operație de tipul menționat.

**Output:** un șir de caractere, reprezentând rezultatul operatiei, scris cu cuvinte, in limba romana

**Restrictii:** pot fi mai mulți operanzi numere naturale intre 0 si 100.000.000 și pot fi mai mulți operatori

## 23. Math - Editor de formule matematice

Se va defini un limbaj propriu de descriere a formulelor matematice, asemănător limbajului TEX/LaTEX sau MathML și programul va putea interpreta o formulă matematică și o va afișa grafic.

Pentru a vă familiariza cu LaTeX și cum ar trebui să funcționeze programul, vedeți:

1. <https://latexeditor.lagrida.com/>
2. [https://ro.wikipedia.org/wiki/Ajutor:Formatul\\_TeX](https://ro.wikipedia.org/wiki/Ajutor:Formatul_TeX)
3. [https://en.wikipedia.org/wiki/Help:Displaying\\_a\\_formula](https://en.wikipedia.org/wiki/Help:Displaying_a_formula)
4. <https://en.wikipedia.org/wiki/MathML>
5. [https://en.wikipedia.org/wiki/Formula\\_editor](https://en.wikipedia.org/wiki/Formula_editor)

**De asemenea, programul va trebui să facă și evaluarea expresiei, pe baza atribuirii unor valori variabilelor care apar în acea expresie.**

Detalii vă poate oferi domnul Bogdan Pătruț.

## 24. Georgia - Numerale in limba georgiană

Programul va realiza transformarea numerelor in si din scrierea cu cifre (arabe) in scrierea cu cuvinte in limba georgiana. Regulile scrierii numeralelor in limba georgiana, care foloseste un alfabet propriu (dar care poate fi romanizat/latinizat) sunt descrise aici: [https://en.wikipedia.org/wiki/Georgian\\_numerals](https://en.wikipedia.org/wiki/Georgian_numerals) Pentru verificarea corectitudinii programului, vom apela la doamna profesoară Manana Chumburidze de la Akaki Tsereteli State University, din Kutaisi, Georgia.

## 25. Quoridor

Se va implementa jocul Quoridor, descris aici: [https://www.google.com/search?q=quoridor&source=lnms&tbm=vid&sa=X&sqi=2&ved=2ahUKEwj41oyF9Kj0AhXzkGoFHxWsDSEQ\\_AUoA3oECAEQBQ&biw=1920&bih=937&dpr=1](https://www.google.com/search?q=quoridor&source=lnms&tbm=vid&sa=X&sqi=2&ved=2ahUKEwj41oyF9Kj0AhXzkGoFHxWsDSEQ_AUoA3oECAEQBQ&biw=1920&bih=937&dpr=1). Pentru a obține un algoritm performant de joc împotriva calculatorului, se poate consulta această lucrare: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.100.5204&rep=rep1&type=pdf> sau asta: <https://slideplayer.com/slide/8894252/>

## 26. NS - Editor de diagrame Nassi-Schneiderman

Se va crea un desenator de diagrame Nassi-Schneiderman, plecând de la un algoritm general, descris în pseudocod. Diagramele NS sunt descrise aici:

1. <https://de.wikipedia.org/wiki/Nassi-Shneiderman-Diagramm>
2. [https://en.wikipedia.org/wiki/Nassi%E2%80%93Schneiderman\\_diagram](https://en.wikipedia.org/wiki/Nassi%E2%80%93Schneiderman_diagram)

Un exemplu de editor NS poate fi găsit aici: <https://www.edrawsoft.com/nassi-schneiderman.html>. Dar programul vostru **NU** trebuie să funcționeze astfel! El trebuie să citească dintr-un fișier text un algoritm general, descris într-un limbaj pseudocod simplu (sau descris într-un limbaj de programare simplu, de genul Python sau Basic) și să deseneze diagrama NS, interpretând acestui algoritm. Suplimentar, programul să poată executa algoritmul descris în pseudocod. Mai multe detalii vă poate oferi domnul Bogdan Pătruț.

## 27. PRINFO = Probleme

Proiectul constă în rezolvarea mai multor probleme de pe platforma [PRINFO](#), pentru a reface/consolida materia de programare esențială. Suplimentar se poate implementa un joc de Spânzurătoarea sau X și 0.

---

## B. Resurse generale pentru realizarea proiectelor

Exemple de videoclipuri pentru proiecte realizate în anii anteriori pot fi găsite aici: <https://www.edusoft.ro/videos/> și aici: <https://www.edusoft.ro/introp/video/>

În dreptul fiecărui proiect sunt/vor fi link-uri către resurse utile. Totuși, iată câteva resurse generale:

- Un mic program ce ilustrează utilizarea mouse-ului și graficii (folosind biblioteca **WinBGIm**): [click aici!](#) Acesta NU are o matrice asociată tablei jocului.

- Un alt program care ilustrează utilizarea mouse-ului și graficii (folosind biblioteca **WinBGIm**): [click aici!](#)  
Acesta ARE o matrice asociată tablei jocului.
- Un exemplu de meniu pentru jocul **Cinci cifre** (poate fi făcut ceva similar și la alte jocuri): <https://www.youtube.com/watch?v=mtW79a0sXXE&feature=share&fbclid=IwAR11IrlFoCewoiXytJhpDdufAFiNrn8g3eP95ppIIWSFQbQkDy7LxOV073U>
- Puteți folosi și grafica în mod text, așa cum a făcut un student, realizând interfața pentru un joc de șah:  
[https://www.edusoft.ro/fisiere/joc\\_sah\\_dragos.mp4](https://www.edusoft.ro/fisiere/joc_sah_dragos.mp4)
- Cei care doresc să dezvolte jocul în rețea între două calculatoare (pe unul rulând programul server, iar pe celălalt programul client):
  - pot folosi acest exemplu realizat de un student. Exemplul a fost testat pe Linux. Pentru ca exemplul să poată fi compilat, trebuie adăugat la CodeBlocks, la *Settings->Compiler->Linker settings->Other linker settings*, următoarele:

```
-lws2_32
-lmswsock
-ladvapi32
```

- Pentru a putea rula codul corect, trebuie întâi compilat și rulat programul [test\\_server.cpp](#), pentru a deschide serverul, iar apoi programul [test\\_client.cpp](#) pentru a conecta clientul la server.
- Funcțiile din [functii\\_socket.cpp](#) pot fi incluse în alte proiecte doar prin a specificarea la începutul programului a declarației

```
#include "functii_socket.cpp"
```

- Descrierea funcțiilor grafice din C: <https://www.programmingsimplified.com/c/graphics.h>
  - Pentru a desena linii de o anumită culoare: [setcolor](#) și [line](#)
  - Pentru a desena un cerc de o anumită culoare: [setcolor\(culoare\)](#) și [circle\(x,y,raza\)](#)
  - Pentru a desena un disc de o anumită culoare (cerc plin sau elipsă plină): [setcolor](#), [setfillstyle](#), [fillellypse](#)
  - Pentru linii mai groase / subțiri: [setlinestyle](#)
  - Pentru un dreptunghi umplut: [setfillstyle](#) și [bar](#)
  - Pentru a afișa texte: [settextstyle](#), [setcolor](#) și [outtextxy](#)
  - Culorile: BLACK=0, BLUE=1, GREEN=2, CYAN=3, RED=4, MAGENTA=5, BROWN=6, LIGHTGRAY=7, DARKGRAY=8, LIGHTBLUE=9, LIGHTGREEN=10, LIGHTCYAN=11, LIGHTRED=12, LIGHTMAGENTA=13, YELLOW=14, WHITE=15
- Cartea "[Olimpiada jocurilor rationale](#)" de Nicolae Oprisiu
- Cartea [Nicolae Oprisiu - Mai in gluma, mai in serios](#)
- Cartea [Gheorghe Paun - Intre matematica si jocuri \(1986\).pdf](#)
- Cartea [FLEX - 50 jocuri](#) : [http://www.boardgames-blog.ro/fisiere/FLEX-Regulament\\_scanat\\_low.pdf](http://www.boardgames-blog.ro/fisiere/FLEX-Regulament_scanat_low.pdf)
- Cartea [Martin Gardner-Alte-Amuzamente-Matematice](#)
- Cartea [Martin Gardner - The Second Scientific American Book of Mathematical Puzzles and Diversions](#)
- Cartea [536 Puzzles & Curious Problems by Henry Ernest Dudeney](#)
- Alte cărți utile pot fi găsite [aici](#).
- [Sam Loyd's Cyclopedia of 5000 Puzzles, Tricks, and Conundrums \(With Answers\)](#).
- [Challenging Mathematical Problems With Elementary Solutions, Vol. 1.pdf](#)



## C. Evaluarea proiectelor pentru anul universitar 2021-2022

Proiectele trebuie prezentate la finalul celor 14 săptămâni, în săptămâna a XIV-a.

Proiectele vor fi prezentate titularului de curs (domnului Bogdan Pătruț) și profesorului colaborator de la laborator (domnul Stelian Hadîmbu sau domnului Vasile Pojoga).

Când un proiect este prezentat, colegii de grupă pot pune întrebări. Pot participa, la cerere, și colegi de la alte grupe, care au avut același proiect de realizat. Și ei pot pune întrebări.

Fiecare student din echipă dintr-un proiect primește, astfel, două note. Notele sunt individuale (fiecare student are nota lui, în funcție de complexitatea proiectului (vezi mai jos), dar și de munca depusă la proiect, de răspunsurile la întrebări (care pot fi și întrebări din cod, cerința de a modifica bucăți din cod pentru a schimba comportamentul etc) și așa mai departe.

Fiecare echipă va pregăti și un mic film cu felul în care rulează proiectul (3-5 minute), în care unul dintre studenți sau amândoi prezintă funcționalitatea proiectului, cu voce (exemplu: [aici](#)).

Studenții care, dintr-un motiv sau altul, repetă disciplina, precum și cursanții și studenții reînmatriculați vor realiza singuri proiectul sau vor prezenta (din nou) proiectul realizat de ei anul trecut.

### Criterii generale de evaluare a proiectelor

1. Criteriile sunt orientative și punctajele sunt maxime, deoarece profesorii pot interveni cu cerințe suplimentare pe măsură ce proiectele se dezvoltă.
2. Criteriile vor fi prezentate la întâlnirea fizică de la facultate, în ziua de marți din săptămâna a IX-a.

Nota fiecărui studentul depinde și de efortul depus, de munca în echipă, de colaborarea cu colegul de echipă, de răspunsuri la întrebări din cod, de felul în care este prezentat proiectul și altele.

### ATENȚIE!

1. Dacă membrii unei echipe realizează 2-3 proiecte, adică fiecare proiectul lui, în loc de unul singur, în comun: toți studenții echipei iau nota 1.
2. Dacă nu lucrează ambii studenți (sau toți trei) la proiect: toți studenții echipei iau nota 1.
3. Dacă nu există colaborare reală între membrii unei echipe: toți studenții echipei iau nota 1.
4. Dacă se folosesc elemente suplimentare care nu fac obiectul acestui curs (STL, OOP etc.): toți studenții din respectiva echipă trebuie să cunoască aceste elemente și să fie capabili să răspundă la întrebări din domeniu, altfel primesc nota 1.
5. Dacă se lucrează în alte limbaje de programare decât C/C++, un mediu de dezvoltare care are multe elemente ”prefabricate”: nota 1
6. Dacă se lucrează un o bibliotecă grafică performantă (OpenGL, GLUT etc.): nota 1. Este permisă utilizarea bibliotecilor grafice WinBGIm, SDL, SFML, Deslin.
7. Dacă se copiază de pe Web, de la studenții din anii anteriori sau de la altă persoană, sau se copiază tutoriale de pe Youtube: nota 1
8. Dacă nu se realizează videoclipul de prezentare: se scad 2p din nota finală.
9. Dacă se răspunde greșit la întrebări, se scade câte 1p pentru fiecare greșeală făcută. Pot fi scăzute până la 6p.
10. Dacă unul dintre studenți renunță la facultate, coechipierul/coechipierii ne anunță imediat.

### Pentru jocuri de o persoană

**Elemente de bază - 5p**

1. 1p = Din oficiu
2. 1p = Definirea unor structuri de date generale, adecvate pentru tablă, piese, situație de joc etc.
3. 0,75p = Stocarea datelor în fișiere binare/text, restaurarea datelor din fișiere, posibilitatea de a lucra cu mai multe fișiere.
4. 1p = Separarea clară a datelor de algoritmi, prezența redusă a minim a elementelor „hard-codate”.
5. 0,75p = Implementarea unui algoritm general de joc, cu verificarea mutărilor corecte/greșite etc.
6. 0,5p = Programul este scris Clean code

**Elemente de algoritmică suplimentare - 2,5p**

1. 0,5p = Implementarea unei strategii de rezolvare automată a unei „provocări” din joc, bazându-se pe „mutări” aleatoare, dar corecte.
2. 1p = Implementarea unei strategii de rezolvare automată a unei „provocări” din joc, bazându-se pe un algoritm performant.
3. 1p = Elemente de generalitate, de extindere a jocului de bază (așa cum este el prezentat în mod standard).

**Elemente de interfață suplimentare - 1,5p**

1. 0,75p = Utilizarea unei biblioteci grafice simple (WinBGIm, Graphics.h, Sigil etc.), în care nu sunt multe funcții grafice predefinite, pentru interfață.
2. 0,75p = Elemente de interfață suplimentare, cum ar fi cronometru, clasament (în fișier), mai multe limbi, muzică, diferite tipuri de piese etc.

**Elemente de creativitate și competitivitate - 1p****Pentru jocuri de două persoane****Elemente de bază - 5p**

1. 1p = Din oficiu
2. 1p = Definirea unor structuri de date generale, adecvate pentru tablă, piese, situație de joc etc.
3. 0,75p = Stocarea datelor în fișiere binare/text, restaurarea datelor din fișiere, posibilitatea de a lucra cu mai multe fișiere.
4. 1p = Separarea clară a datelor de algoritmi, prezența redusă a minim a elementelor „hard-codate”.
5. 0,75p = Implementarea unui algoritm general de joc, cu verificarea mutărilor corecte/greșite etc.
6. 0,5p = Programul este scris Clean code

**Elemente de algoritmică suplimentare - 2,5p**

1. 0,5p = Implementarea unei strategii de joc pentru calculator, bazându-se pe „mutări” aleatoare, dar corecte.
2. 1p = Implementarea unei strategii de joc pentru calculator, bazându-se pe un algoritm performant, eventual cu o strategie câștigătoare.
3. 1p = Elemente de generalitate, de extindere a jocului de bază (așa cum este el prezentat în mod standard).

**Elemente de interfață suplimentare - 1,5p**

1. 0,75p = Utilizarea unei biblioteci grafice simple (WinBGIm, Graphics.h, Sigil etc.), în care nu sunt multe funcții grafice predefinite, pentru interfață.
2. 0,75p = Elemente de interfață suplimentare, cum ar fi cronometru, clasament (în fișier), mai multe limbi, muzică, diferite tipuri de piese etc.

**Elemente de creativitate și competitivitate - 1p****Pentru proiecte complexe, care nu sunt jocuri**

- ▼ Eval - evaluator și vizualizator pentru o expresie algebrică
  - Din oficiu: 1p
  - Structurile de date folosite și păstrarea lor în fișiere: 1p
  - Programul funcționează corect, fără niciun fel de greșeală, pe orice exemplu (cu oricâți operatori, funcții etc), arătând și erorile de sintaxă: 1,5p
  - Programul tratează cazurile de limită (împărțire la zero, adunare/înmulțire cu infinit, radical/logaritm din număr negativ etc): 1p
  - Programul reprezintă corect arborele asociat expresiilor, fără niciun fel de greșeală: 1,5p
  - Nodurile se redimensionează în funcție de conținutul lor: 0,5p
  - Nodurile se amplasează în funcție de structura subarborelui stâng și drept, astfel încât să nu fie înghesuite (să nu se suprapună): 0,5p
  - Programul permite mutarea nodurilor din arbore, cu mouse-ul sau cu tastele: 1p
  - Elemente suplimentare (o altă limbă, meniuri, setări de tot soiul, grafică): 1p
  - Elemente de creativitate și competitivitate: 1p
- ▼ Grafic - Graficul unei funcții
  - Din oficiu: 1p
  - Structurile de date folosite și păstrarea lor în fișiere: 1p
  - Programul reprezintă corect graficul funcțiilor, fără niciun fel de greșeală: 3p
  - Sunt reprezentate și axele de coordonate: 0,5p
  - Programul permite efectul de lupă (zoom), pentru a vedea graficul funcției mai de aproape sau de mai departe: 1,5p
  - Afisarea punctelor de minim, maxim, a asimptotelor etc: 1p
  - Elemente suplimentare (o altă limbă, meniuri, setări de tot soiul): 1p
  - Elemente de creativitate și competitivitate: 1p
- ▼ Deriv - Derivare formală
  - Din oficiu: 1p
  - Structurile de date folosite și păstrarea lor în fișiere: 1p
  - O structură de date originală, algoritm cu elemente de originalitate: 0,5
  - Programul derivează corect funcția, fără niciun fel de greșeală: 3p
  - Programul rafinează/simplifică funcția derivată (elimină înmulțirile cu 0 sau cu 1, adunările/scăderile cu 0, ridicarea la puterea 1 sau 0, împărțirea la 1, eliminarea parantezelor inutile etc.): 2p
  - Elemente suplimentare de interfață (interfață cu mai multe limbi/meniuri/setări de tot soiul): 1p
  - >
  - Elemente de creativitate și competitivitate (câteva idei: reprezentarea „matematică” a expresiilor, derivata de ordin 2, funcții cu mai multe argumente etc.) 1,5p
- ▼ BibMat - Bibliotecă de operații cu tablouri (vectori și matrice)
  - Din oficiu: 1p
  - Structurile de date folosite și păstrarea lor în fișiere: 1p
  - Implementarea funcțiilor de operare cu tablouri uni- și bidimensionale. 3p
  - Animația: 3p
  - Elemente suplimentare (alte funcții, meniu, setări, culori, sunete, etc): 1p
  - Elemente de creativitate și competitivitate: 1p
- ▼ BibLis - Bibliotecă de operații cu liste
  - Din oficiu: 1p

- Structurile de date folosite și păstrarea lor în fișiere: 1p
- Implementarea funcțiilor de operare cu liste: 3p
- Animația: 3p
- Elemente suplimentare (alte funcții, meniu, setări, culori, sunete, etc): 1p
- Elemente de creativitate și competitivitate: 1p
- ▼ InterSchem - Vizualizator și interpretor de scheme logice
  - Din oficiu: 1p
  - Structurile de date folosite și păstrarea lor în fișiere: 1p
  - Programul reprezintă corect blocurile: 1p
  - „Execută” „informația” din interiorul fiecărui bloc (chiar folosind un evaluator de expresii algebrice, dezvoltat de altă echipă (care are proiectul Eval)): 1p
  - Desenează legăturile dintre blocuri (chiar și dacă legăturile sunt directe): 1p
  - Blocurile se pot muta (cu mouse sau cu taste), legăturile mutându-se după blocuri: 1p
  - Legăturile dintre blocuri pot ocoli blocurile: 0,5p
  - Editarea conținutului blocurilor: 1p
  - Codificarea programului din schema logică în cod C++: 1p
  - Altele elemente (salvare, încărcare schema logică din fișier): 0,5p
  - Elemente de creativitate și competitivitate: 1p
- ▼ Electron - Vizualizator de scheme electronice
  - Din oficiu: 1p
  - Structurile de date folosite și păstrarea lor în fișiere, separarea clară a datelor de cod: 1,5p
  - Programul reprezintă corect piesele electronice și legăturile dintre ele (chiar și dacă legăturile sunt directe): 2,5p
  - Piesele se pot muta (cu mouse sau cu taste), legăturile mutându-se după ele: 1p
  - Piesele se pot roti, redimensiona: 1p
  - Alte elemente, cum ar fi verificarea corectitudinii circuitului, calcule fizice etc.: 0,5p
  - Editarea conținutului pieselor (adică a valorilor marimilor fizice corespunzătoare pieselor): 1p
  - Salvare, încărcare circuit electronic, alte elemente: 0,5p
  - Elemente de creativitate și competitivitate: 1p
- Details
  - Din oficiu: 1p
  - +4p pentru un program funcțional, corect
  - +4p pentru elemente suplimentare
  - Elemente de creativitate și competitivitate: 1p