

Colegiul Național „Bogdan Petriceicu Hasdeu” Buzău

Atestat la informatică

Avioane - joc C++

Îndrumător:

Prof. Alin Burța

Autor:

Silaghi Mara

CUPRINS

1.Introducere

2. Regulile jocului

3.Utilizarea aplicației

4.Limbajul de programare utilizat

5.Componentele codului

6.Concluzii

7.Bibliografie

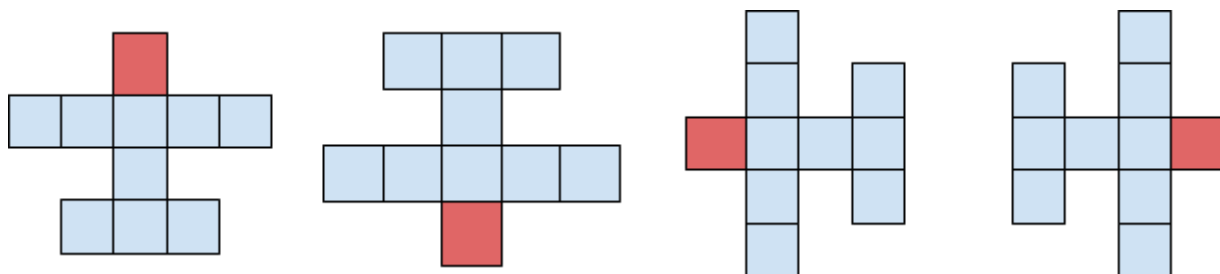
Introducere

Recent mi-am amintit de jocul „Avioane”, pe care obișnuiam să îl joc împreună cu părinții și cu colegii mei pe foi de matematică. Odată cu căutarea ideilor pentru atestat, m-am gândit la acest joc și la similitudinea formatului său cu cel al matricelor pe care le folosesc des în programele realizate în cadrul orelor de informatică de la liceu. Am considerat realizarea acestui joc în programul c++ o idee ingenioasă și folositoare atât pentru mine, cât și pentru alți oameni dornici de a-l revedea și juca.

Regulile jocului

Pentru a înțelege pe deplin programul, este necesar să cunoaștem regulile acestui joc.

Avioane este un joc de strategie între doi jucători. Se joacă pe hârtie cu pătrățele unde se trag două chenare identice, de obicei de 10x10 căsuțe. Fiecare căsuță se identifică prin două numere, reprezentând numărul liniei și al coloanei pe care se află. Primul chenar este propriu, unde se așează trei avioane, fără ca adversarul să le vadă poziția. Avioanele nu au voie să se suprapună și nici să iasă în afara chenarului. Fiecare jucător va încerca să ghicească unde sunt avioanele adversarului în al doilea chenar.



Aceasta este forma unui avion. Pătratul roșu reprezintă cabina acestuia. Avionul poate fi așezat în 4 direcții: cu cabina în sus, în jos, spre stânga sau spre dreapta.

0	1	2	3	4	5	6	7	8	9

9

Când ambii jucători au desenat cele trei avioane în chenarul propriu, jocul poate începe. Se poate trage la sorți cine este primul la rând. Fiecare mutare are două părți:

- Primul jucător anunță o anumită casuță a adversarului, spre exemplu 6 3 (numărul liniei, respectiv numărul coloanei).
- Adversarul se uită la chenarul propriu și răspunde cu una din următoarele posibilități:
 - aer - casuța este în afara oricărui avion;
 - lovit - casuța face parte dintr-un avion, dar nu s-a găsit cabina;
 - mort/doborât - casuța conține cabina unui avion.

Ambii jucători notează rezultatul pe casuțele lor, de obicei un X pentru lovit și 0 pentru aer.

Utilizarea aplicației

```

C:\Users\maras\OneDrive\Documente\codeblocks\1.exe
-----
TABLA JUCATORULUI 1
-----
 0 1 2 3 4 5 6 7 8 9
0 . . . . . . . . .
1 . . . . . . . . .
2 . . . . . . . . .
3 . . . . . . . . .
4 . . . . . . . . .
5 . . . . . . . . .
6 . . . . . . . . .
7 . . . . . . . . .
8 . . . . . . . . .
9 . . . . . . . . .

INSTRUCTIUNI (Jucatorul 1)

Acum iti vei plasa avioanele. Coordonatele neecesare sunt:
Directie (1:stanga, 2:dreapta, 3:sus, 4:jos), X (linia pe care se afla cabinaul), Y (coloana pe care se afla cabinaul)
Exemplu: 1 7 2 plaseaza un avion orizontal, cu punctul din extrema stanga pe linia 7 si coloana 2

Unde doresti sa plasezi avionul?

```

După ce codul este rulat, pe ecran apare o tablă de joc de 10x10. Primul jucător își plasează cele 3 avioane prin introducerea următoarelor date pentru fiecare avion:

- direcția: 1(cu cabina spre stânga), 2(cu cabina spre dreapta), 3(cu cabina în sus), 4(cu cabina în jos)
- coordonatele cabinei avionului (numărul liniei și coloanei pe care se află)

După introducerea datelor, apare tabla celui de-al doilea jucător, care va proceda la fel.

```
C:\Users\maras\OneDrive\Documente\codeblocks\1.exe
TABLA JUCATORULUI 1
-----
 0  1  2  3  4  5  6  7  8  9
0  .  .  .  .  .  .  .  .  .  .
1  .  S  .  .  .  .  .  .  S  .
2  .  S  .  S  .  .  S  .  S  .
3  0  S  S  S  .  .  S  S  S  0
4  .  S  .  S  .  .  S  .  S  .
5  .  S  .  .  .  .  .  .  S  .
6  .  .  .  S  S  S  .  .  .  .
7  .  .  .  S  .  .  .  .  .  .
8  .  .  S  S  S  S  S  .  .  .
9  .  .  .  .  0  .  .  .  .  .
Press any key to continue . . .
```

Cele două table de joc sunt finalizate. Pe ecran apare o tablă de joc goală, iar primul jucător trebuie să introducă coordonatele punctului pe care dorește să îl atace. După ce cel de-al doilea face același lucru, primul poate vedea dacă punctul lovit de el este aer(0), lovit(X) sau cabină(A).

```
C:\Users\maras\OneDrive\Documente\codeblocks\1.exe
TABLA JUCATORULUI 1
-----
 0  1  2  3  4  5  6  7  8  9
0  .  .  .  .  .  .  .  .  .  .
1  .  .  .  .  .  .  .  .  .  .
2  .  .  .  .  .  .  .  .  .  .
3  .  .  .  A  .  .  .  .  .  .
4  .  .  .  .  .  .  .  .  .  .
5  .  .  .  .  .  .  X  .  .  .
6  .  .  .  .  .  .  .  .  .  .
7  .  .  .  .  .  .  .  .  .  .
8  .  0  .  .  .  .  .  .  .  .
9  .  .  .  .  .  .  .  .  .  .
JUCATOR 2, SCRIE COORDONATELE PUNCTULUI PE CARE IL ATACI: 
```

Jocul continuă până când unul dintre jucători a lovit toate avioanele adversarului său, iar pe ecran se afișează câștigătorul.

Limbajul de programare utilizat

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**. Traducerea este realizată de un program specializat numit **compilator** sau **interpretor**.

Limbajul C++ este un limbaj compilat. 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).

Componentele codului

Codul se bazează pe mai multe funcții folosite în cadrul programului propriu-zis. Prin intermediul mai multor structuri repetitive, fiecare jucător este capabil să introducă cele 3x3 valori necesare pentru plasarea avioanelor sale. Coordonatele introduse sunt verificate prin intermediul funcției `Plasare_avion`, care asigură faptul că datele introduse sunt corecte și pot forma un avion. Cunoscând doar direcția și coordonatele cabinei, programul deduce toate

punctele din care este compus avionul și le marchează pe tabla jucătorului, notând cabina cu O și restul punctelor cu S. Tabla curentă este afișată prin intermediul funcției Desen_tabla, care ține cont de faptul că jocul nu a început și afișează statutul tuturor căsuțelor de pe tabla de joc. Funcția „system(“cls”)” este folosită pentru a șterge ecranul.

După ce datele necesare sunt introduse, funcția de tip bool joc_inceput devine adevărată, iar funcția Desen_tabla afișează tabla oponentului, ascunzând avioanele sale. Jucătorul introduce coordonatele punctului pe care vrea să îl atace, verificate de funcția Atac. La următoarea tură a jucătorului, se afișează tabla curentă, unde acesta poate vedea dacă punctul atacat a fost aer(0), lovit(X)sau cabină(A). Programul verifică dacă jocul a luat sfârșit după fiecare atac. Dacă toate punctele tuturor avioanelor unui jucător au fost lovite, atunci se afișează un mesaj pentru câștigătorul jocului.

Un aspect pe care îl consider interesant în program este funcția Plasare_avion, care verifică respectarea tuturor cerințelor necesare construirii unui avion. Aceasta asigură faptul că direcția este un număr natural între 1 și 4, că atât cabina, cât și toate punctele care alcătuiesc avionul nu se află în afara tablei de joc și că punctele unui avion nou introdus nu sunt deja ocupate de alte avioane create anterior.

Concluzii

Programul realizat este unul destul de complex, însă care ar putea fi îmbunătățit. În jocul original, jucătorii sunt capabili să deducă toate punctele unui avion pe baza a câtorva deja aflate. Aici, însă, aceștia sunt nevoiți să atace fiecare punct al fiecărui avion în parte, ceea ce face jocul să dureze mult mai mult timp. În această privință, în cod s-ar putea adăuga o modalitate prin care jucătorul poate lovi mai multe puncte atunci când este rândul său.

Bibliografie

- <https://www.pbinfo.ro/articole/59/introducere-in-cpp>
- <https://ro.wikipedia.org/wiki/C%2B%2B>
- [https://ro.wikipedia.org/wiki/Avioane_\(joc\)](https://ro.wikipedia.org/wiki/Avioane_(joc))
- <https://github.com/>