PROBLEMA REGINELOR

Grigore Lucian-Florin

Stoica Serban

~clasa a XII-a A~

~profesor coordonator : Trocaru Dana~

~Colegiul National “B. P. Hasdeu”~

Cuprins

1. Enuntarea problemei abordate
2. Descrierea limbajului de programare utilizat
3. Cerinte de sistem
4. Descrierea programului
5. Utilitatea practica a lucrarii
6. Bibliografie

*A. Enuntarea problemei abordate*

**Problema damelor** (sau **problema reginelor**) tratează plasarea a opt regine de șah pe o tablă de șah astfel încât să nu existe două regine care se amenință reciproc. Astfel, se caută o soluție astfel încât nicio pereche de două regine să nu fie pe același rând, pe aceeași coloană, sau pe aceeași diagonală. Problema cu opt regine este doar un caz particular pentru problema generală, care presupune plasarea a *n* regine pe o tablă de șah *n*×*n* în aceleași condiții. Pentru această problemă, există soluții pentru toate numerele naturale *n* cu excepția lui *n*=2 și *n*=3.

**Istorie**

Compozitorul de șah Max Bezzel a publicat problema celor opt regine în 1848. Franz Nauck a publicat primele soluții în 1850. Nauck a și extins problema la *n* regine, cu *n* regine pe o tablă de șah de *n* × *n* pătrate.

De atunci, mulți matematicieni, inclusiv Carl Friedrich Gauss, au lucrat atât la problema inițială cu opt regine cât și la varianta generalizată cu *n*regine. În 1874, S. Gunther a propus o metodă care utilizează determinanții pentru a găsi soluții. J. W. L. Glaisher a rafinat abordarea lui Gunther.

În 1972, Edsger Dijkstra a folosit această problemă pentru a ilustra puterea a ceea ce el numește programare structurată. El a publicat o descriere foarte detaliată a unui algoritm backtracking de parcurgere în adâncime. **Backtracking** este numele unui algoritm general de descoperire a tuturor soluțiilor unei probleme de calcul, algoritm ce se bazează pe construirea incrementală de soluții-candidat, abandonând fiecare candidat parțial imediat ce devine clar că acesta nu are șanse să devină o soluție validă.

*B. Descrierea limbajului de programare utilizat*

Numele limbajului C# a fost inspirat din notaţia ♯ (diez) din muzică, care indică faptul că nota muzicală urmată de ♯ este mai înaltă cu un semiton. Este o similitudine cu numele limbajului C++, unde ++ reprezintă atât incrementarea unei variabile cu valoarea 1, dar şi faptul că C++ este mai mult decât limbajul C. Limbajul C# a fost dezvoltat în cadrul Microsoft. Principalii creatori ai limbajului sunt Anders Hejlsberg, Scott Wiltamuth şi Peter Golde. Prima implementare C# larg distribuită a fost lansată de către Microsoft ca parte a iniţiativei .NET în iulie 2000. Din acel moment, se poate vorbi despre o evoluţie spectaculoasă. Mii de programatori de C, C++ şi Java, au migrat cu uşurinţă spre C#, graţie asemănării acestor limbaje, dar mai ales calităţilor noului limbaj. La acest moment, C# şi-a câştigat şi atrage în continuare numeroşi adepţi, devenind unul dintre cele mai utilizate limbaje din lume.

Creatorii C# au intenţionat să înzestreze limbajul cu mai multe facilităţi. Succesul de care se bucură în prezent, confirmă calităţile sale:

* Este un limbaj de programare simplu, modern, de utilitate generală, cu productivitate mare în programare.
* Este un limbaj orientat pe obiecte.
* Permite dezvoltarea de aplicaţii industriale robuste, durabile.
* Oferă suport complet pentru dezvoltarea de componente software, foarte necesare de pildă în medii distribuite. De altfel, se poate caracteriza C# ca fiind nu numai orientat obiect, ci şi orientat spre componente.

La aceste caracteristici generale se adaugă şi alte trăsături, cum este de pildă suportul pentru internaţionalizare, adică posibilitatea de a scrie aplicaţii care pot fi adaptate cu uşurinţă pentru a fi utilizate în diferite regiuni ale lumii unde se vorbesc limbi diferite, fără să fie necesare pentru aceasta schimbări în arhitectura software. În strânsă legatură cu Arhitectura .NET (.NET Framework) pe care funcţionează, C# gestionează în mod automat memoria utilizată. Eliberarea memoriei ocupate (garbage collection) de către obiectele care nu mai sunt necesare aplicaţiei, este o facilitate importantă a limbajului. Programatorii nu mai trebuie să decidă singuri, aşa cum o fac de pildă în C++, care este locul şi momentul în care obiectele trebuie distruse. În C# se scriu de asemenea aplicaţii pentru sisteme complexe care funcţionează sub o mare varietate de sisteme de operare, cât şi pentru sisteme dedicate (embeded systems). Acestea din urmă se întind pe o arie largă, de la dispozitive portabile cum ar fi ceasuri digitale, telefoane mobile, MP3 playere, până la dispozitive staţionare ca semafoare de trafic, sau controlere pentru automatizarea producţiei. Din punct de vedere sintactic C# derivă din limbajul C++, dar include şi influenţe din alte limbaje, mai ales Java.

**Despre .NET Framework**

Arhitectura .NET este o componentă software care oferă un mediu de programare şi de execuţie a aplicaţiilor pentru sistemele de operare Microsoft. NET Framework este un mediu care permite dezvoltarea şi rularea aplicaţiilor şi a serviciilor Web, independente de platformă. Limbajul C# se află într-o strânsă legatură cu arhitectura .NET. Iniţial, C# a fost dezvoltat de către Microsoft pentru crearea codului platformei .Net, la fel cum destinaţia iniţială a limbajului C a fost aceea de a implementa sistemul de operare UNIX. .NET pune la dispoziţie o colecţie impresionantă de clase organizate în biblioteci, pe care C# le utilizează. Este momentul să precizăm că C# funcţionează având .NET ca infrastructură, dar .NET suportă şi alte limbaje, cum este C++, Visual Basic sau Java. În oricare dintre aceste limbaje programaţi, aveţi la dispoziţie aceleaşi biblioteci de clase. .NET se realizează în acest fel interoperabilitatea limbajelor.

**Despre Microsoft Visual Studio**

**Microsoft Visual Studio** este un mediu de dezvoltare integrat (integrated development environment - IDE) de la Microsoft. Acesta poate fi folosit pentru a dezvolta aplicaţii consolă şi aplicații cu interfaţă grafică pentru toate platformele suportate de Microsoft Windows (ex. .NET Framework, Windows Mobile etc).

*C. Cerinte de sistem*

Hardware :

* Procesor 1.8 GHz sau mai rapid . Recomandat : Dual-core sau mai bun
* 2 GB de RAM . Recomandat : 4 GB minim
* Spatiu Hard Disk : Pana la 130 GB spatiu disponibil , in functie de caracteristicile instalate . Instalarea normal are nevoie de 20-50 GB spatiu liber . Recomandat : instalarea Windows si Visual Studio pe un Solid State Drive ( SSD ) pentru o viteza imbunatatita.
* Placa video care suporta cel putin o rezolutie 720p ( 1280 x 720 ) . Recomandat : o rezolutie WXGA ( 1366 x 768 ) sau mai buna .

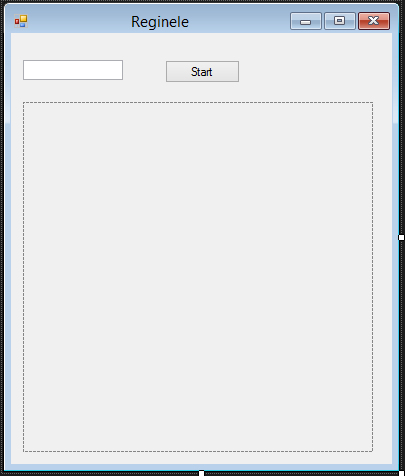
Software :

* Windows 10 ( versiunea 1507 sau mai mare ) , Windows Server 2016 , Windows 8.1 , Windows Server 2012 R2 , Windows 7 SP1 ( cu update-urile la zi ) .
* .NET Framerowk 4.5 pentru instalare . Programul va instala ulterior .NET Framework 4.6.1 , pentru rulare .
* Internet Explorer 11 sau Edge pentru scenariile care au nevoie de acces la internet .

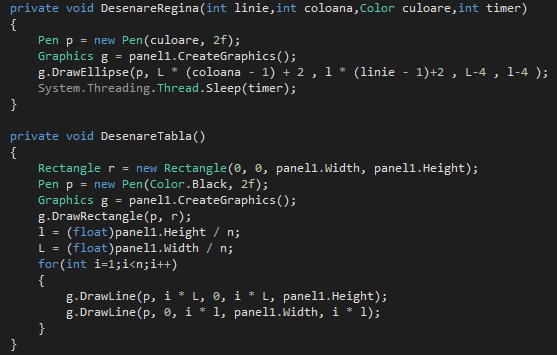
*D. Descrierea programului*

Intrucat mediul de programare oferit de Visual Studio uneste intr-un maniera usor de inteles idei originate din limbaje de programare diferite ( C , C++ , Java ) , dificultatea realizarii programului este reprezentata de imbinarea intr-un mod efficient a acestor multiple concept .

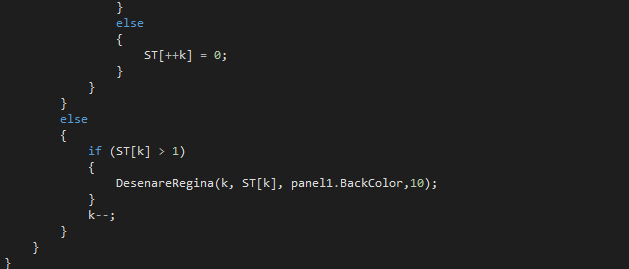
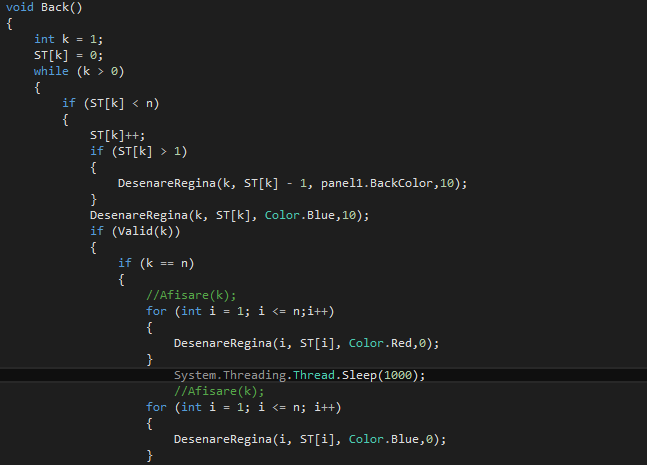
Fereastra programului este una simpla , formata dintr-o caseta de text in care se pot introduce valori pentru dimensiunea tablei de sah si , implicit , a numarului de regine folosit , un buton “Start” pentru demararea programului si un panou pe care va fi ilustrat modul de rulare al algoritmului.



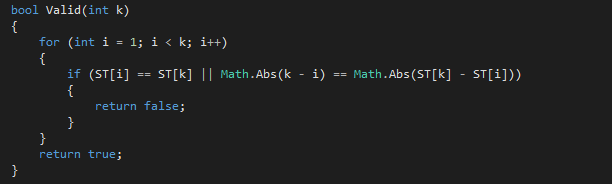
Algoritmului de Backtracking folosit i-a fost asociat o serie de functii grafice predefinite si nu numai care sa afiseze pe panoul folosit o ilustratie vizuala in timp real a iteratiilor algoritmului backtracking . Aceste functii predefinite sunt continute in biblioteca Windows.Drawing. Pe langa acestea , au fost create doua functii denumite , intuitive , “DesenareRegina” si “DesenareTabla”.



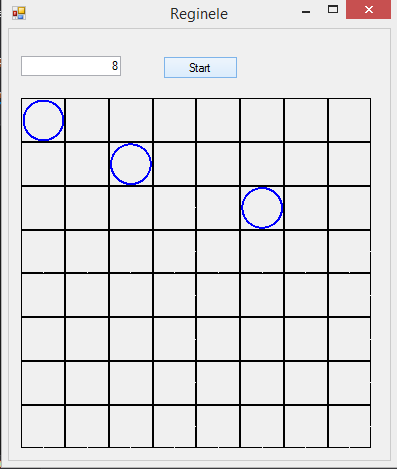
Cu acestea fiind zise , algoritmul backtacking imbina programarea clasica cu cea orientata pe obiecte reprezentata de utilizarea capacitatilor specifice Drawing.



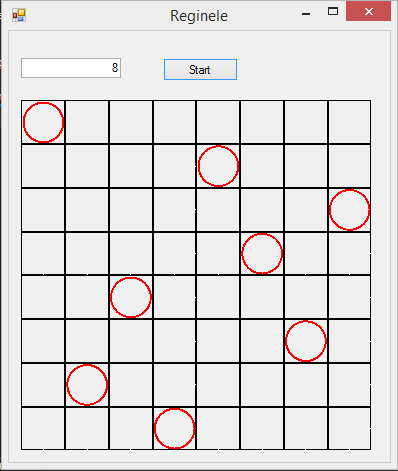
Unde functia Valid este urmatoarea:



In timpul rularii algoritmului , pe tabla de sah desenata vor fi desenate cercuri simbolizand pozitia unei regine .



Atunci cand este gasita o solutie la problema, programul se opreste din rulare pentru o perioada determinata , iar pozitiile corespondente fiecarei regine sunt indicate de cate un cerc rosu , de aceasta data.



Programul va continua rularea pana la afisarea tuturor solutiilor , dupa care va elibera tabla de sah de orice pozitie ocupata anterior.

*E. Utilitatea practica a lucrarii*

Proiectul poate fi utilizat in scop educativ in vederea predarii , intelegerii si aprofundarii problemei reginelor si a metodei de programare **backtracking**. De asemenea , poate fi folosit ca mijloc de ilustrare a programarii utilizand limbajul C# , a folosirii mediului Visual Studio si a imbinarii de concepte din limbaje diferite.

*F. Bibliografie*

* Wikipedia
* MSDN Microsoft
* Microsoft Docs
* Infoarena
* www.math.uaic.ro
* C# pentru liceu – editura Infomat