

MULTITHREAD KULLANARAK SAMURAI SUDOKU ÇÖZME

Oğuzhan Koç-Yusuf Arslan
Bilgisayar Mühendisliği Bölümü
Kocaeli Üniversitesi
190201184190201183

oguzhankoc9855@hotmail.com ysf.arslan2017@gmail.com

Özet

Proje multithread yapısını kullanarak verilecek samurai sudoku üzerinden çözüm bulan bir masaüstü uygulamasıdır. Uygulama C# kullanılarak geliştirilmiştir. Dokümanda projenin tanımı, çözüme yönelik yapılan araştırmalar, kullanılan fonksiyonlar, proje bitimindeki deneysel sonuçlar ve kod bilgisi gibi programın oluşumunu açıklayan başlıklara yer verilmiştir. Doküman sonunda projeyi hazırlarken kullandığımız kaynaklar bulunmaktadır.

• Giriş

Projede amaç multithread kullanılarak verilen samurai sudokuyu dinamik olarak çözdürmektir. Sudoku başlangıç değerlerini .txt uzantılı dosyadan almaktadır. Dosyadan alınan bilgiler 9x9 luk 5 adet sudokuya gönderilir. Bu 9x9 luk sudokuların ortadaki sudokuyla geri kalan sudokular arasındaki ortak alanlar çözdürülmeye çalışılır. Çözümü form üzerinden gösterilir. Çıkan işlemlerin sonuçları .txt uzantılı dosyalara aktarılır.

Problem Tanımı

Bu projede bizden multithread kullanılarak samurai sudoku çözdürememiz beklenmektedir. Bu sudokuda başlangıçta .txt uzantılı bir dosyadan okuma yapmamız bekleniyor. Çözümlerden çıkan tüm işlemlerin .txt uzantılı dosyalara kaydedilir. Son olarak çıkan çözümleri kaç adet adımda yapıldığı gösterilir.

Genel Yapı

Fonksiyonlar

karsilastir: Bu fonksiyonda ortadaki sudoku ve köşelerdeki sudokuların ortak bölgelerindeki boş kutularda işlem yapılır. Sonra ihtimalleri_yaz_2 fonksiyonunda tutulan ihtimaller karşılaştırılıp yeni bir listede tutulur.

karsilastir_new: ihtimallerin tutulduğu liste de ikili kalanlara göre doğru olan sonucu bulmaya çalışır.

ihimalleri_yaz_2: Sudokuların ortak alanlarındaki bölgelerde boş olan noktaların olma ihtimallerini bulur ve listede tutar.

tekrar: karsilastir, ihtimalleri_yaz_2 ve karsilastir_new fonksiyonlarını kullanarak ortak alanlarda bulunan noktaları çözer.

Dosya_yukle:Dosyadan veri alıp bu verileri ayrıştırarak sudokulara atılır.Böylece bu veriler diğer fonksiyonlar için hazır hale gelir.
add_btun_Click:Samuray sudokunun çözülmemiş hali gösterilir.
sonuc_btn_Click:Bu butonla threadler çalıştırılarak sudokunun çözümü dinamik olarak gösterilir
islemler_kaydet:Sudoku çözümü yaptırılırken yapılan adımların tümü bu fonksiyon sayesinde .txt uzantılı dosyalara göndermeye hazır hale getirilir
txt_yaz:islemler_kaydetde tutulan işlem adımları bu fonksiyon sayesinde dosyalara yazılır.
solvesudoku:Bu fonksiyon sayesinde backtracking yöntemi kullanılarak sudoku çözümü yapılır.
uygunluk:Gönderilen sayının gönderilen sudoku hücresine uygun olup olmadığına bakılır.
Sınıflar
durum:Bu sınıfta sudoku çözerken yapılan değişiklikler tutulur
ihtimal: Bu sınıfta olması muhtemel ihtimaller tutulur.
form2: Yapılan işlemleri grafik olarak gösteririz.

Temel Bilgiler

Proje geliştirmede:

Programlama dili olarak "C#" kullanılmıştır.

Program geliştirme ortamı olarak "Visual Studio Code" kullanılmıştır.

Yapılan Araştırmalar:

Başlangıçta Sudokuyu çözme konusunda zorlandık Sudokuyu çözebilmek için yöntemler araştırma yoluna gittik.Bunun için en uygun yöntemin backtracking olduğunu düşündük.

<https://www.geeksforgeeks.org/> Sitedeki çözümleri uygulamaya çalıştık.

Bundan sonra sudokuyu threade uygulamaya çalıştık.Bunun için thread hakkında araştırma yaptık.

Deneyisel Sonuçlar ve Ekran Görüntüleri

Form2

ekle

sonuc

thread 1

thread 2

thread 3

thread 4

thread 5

savisi

Form2

sonuc

0	0	5	7	0	0	0	2	0
4	9	0	0	6	0	0	1	0
0	0	7	0	0	4	9	0	6
0	0	6	0	0	0	0	0	8
0	7	0	0	0	0	0	9	0
2	0	0	0	0	0	3	0	0
5	0	8	9	0	0	7	0	0
0	1	0	0	3	0	0	8	5
0	2	0	0	0	5	6	0	0
0	0	0	4	0	6	0	0	0
0	0	6	0	5	0	2	0	0
0	0	0	2	0	8	0	0	0
0	0	8	5	0	0	0	2	0
6	2	0	0	4	0	0	5	0
0	0	7	0	0	8	9	0	3
0	0	6	0	0	0	0	0	2
0	5	0	0	0	0	0	4	0
8	0	0	0	0	0	3	0	0
1	0	5	9	0	0	2	0	0
0	3	0	0	6	0	0	7	1
0	6	0	0	0	3	5	0	0

0	0	9	6	0	0	0	2	0
1	4	0	0	5	0	0	3	0
0	0	2	0	0	1	7	0	8
0	0	3	0	0	0	0	0	2
0	5	0	0	0	0	0	6	0
4	0	0	0	0	0	5	0	0
6	0	5	8	0	0	2	0	0
0	1	0	0	7	0	0	8	6
0	2	0	0	0	4	3	0	0
0	0	8	9	0	0	0	6	0
9	6	0	0	2	0	0	5	0
0	0	2	0	0	8	1	0	9
0	0	1	0	0	0	0	0	6
0	8	0	0	0	0	0	2	0
7	0	0	0	0	0	5	0	0
2	0	6	7	0	0	4	0	0
0	3	0	0	9	0	0	7	8
0	9	0	0	0	4	2	0	0

thread 1

thread 2

thread 3

thread 4

thread 5

savisi

[illegible]

The screenshot shows a Java Swing window titled "Form2". On the left, there is a button labeled "sonuc". The main area contains a 15x15 grid of numbers. The grid is composed of several blocks of numbers, some of which are highlighted in light blue. The numbers are arranged in a pattern that suggests a 15x15 grid, with some cells missing or highlighted. On the right side, there is a list of threads and their IDs:

Thread Name	Thread ID
thread 1	1349
thread 2	825
thread 3	1160
thread 4	727
thread 5	942

Below the thread list, there is a button labeled "savisi".

Kaynakça

<https://www.geeksforgeeks.org>

<https://docs.microsoft.com/tr-tr/dotnet/api/system.threading.thread?view=net-6.0>

<https://stackoverflow.com>

<https://github.com>

<https://tr.wikipedia.org/wiki/Sudoku>

<https://www.c-sharpcorner.com/article/c-sharp-write-to-file/>