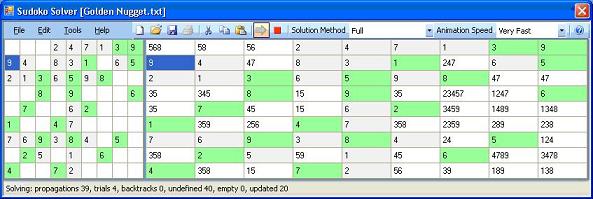
**ANALISIS SUDOKU SOLVER**

**MENGGUNAKAN ALGORITMA GENERATE AND TEST**

Sudoku adalah adalah sejenis [teka-teki](http://id.wikipedia.org/wiki/Teka-teki) logika yang tujuannya adalah untuk mengisikan angka-angka dari 1 sampai 9 ke dalam kotak 9×9 yang terdiri dari 9 sub kotak 3×3 tanpa ada angka yang berulang di satu baris, kolom atau sub kotak.



Salah satu algoritma yang dapat digunakan untuk menyelesaikan permainan Sudoku adalah algoritma ‘Generate and Test’. Algoritma ini akan menghasilkan solusi yang mungkin dan mengujinya untuk melihat apakah solusi itu benar atau tidak. Jika benar, maka proses selesai, jika tidak maka akan dihasilkan solusi lain dan ulangi hal tersebut sampai didapatkan solusi yang benar. Untuk menyelesaikan permainan ini digunakan algoritma generate and test dengan versi rekursif, sebagai berikut:

Function TrialAndError

Test whether the current grid is valid

If it is not, return an error code.

Pick an unallocated cell

For each possible value for that cell.

Remember the current state of the grid

Set the value of the cell to the next possible value

Call TrialAndError recursively

If on return we have successfully found a solution

Return ‘success’.

Otherwise reset the grid to its previous state

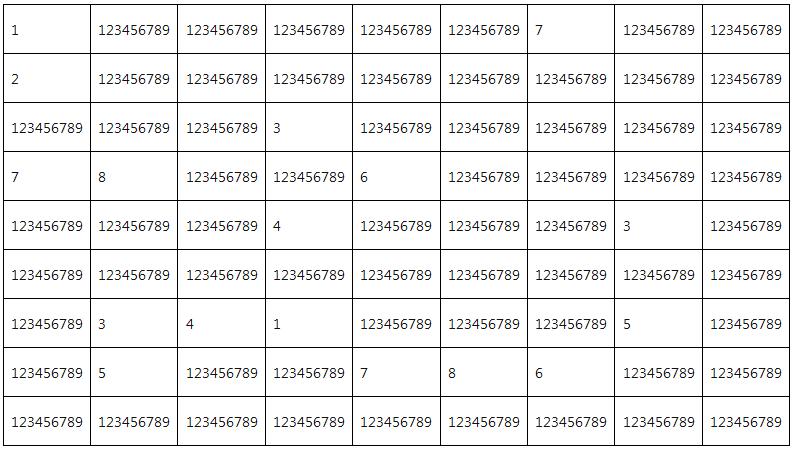
Repeat for next possible value

Proses diatas menjamin ditemukannya suatu solusi, akan tetapi memakan waktu yang cukup besar. Dalam kasus terburuk kita harus menguji setiap sel terisi dengan nilai 1 sampai 9 secara rekursif. Jadi dengan 64 sel yang tidak dialokasikan kita bisa mencoba 9 dengan 64 kemungkinan yang sebanding dengan 1063 kemungkinan.

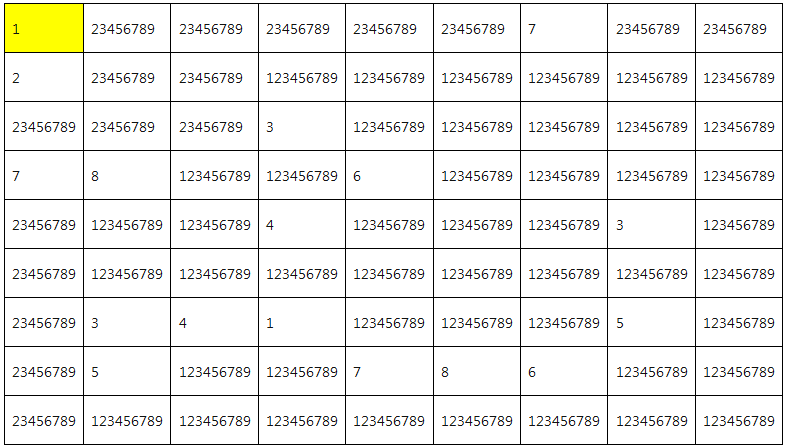
**Propagasi kendala - Pemangkasan kemungkinan solusi**

Masalah yang terdapat pada algoritma generate and test diatas adalah bahwa kita pada akhirnya harus menguji banyak konfigurasi dari grid yang tidak valid. Propagasi kendala adalah teknik untuk mengurangi nilai kemungkinan yang harus kita uji, sehingga dapat mempercepat pencarian solusi. Ini merupakan suatu teknik yang canggih untuk memecahkan masalah yang besar dan kompleks. Di dalam kendala pemrograman, kendala hanyalah sebuah hubungan antara dua variable atau lebih yang membatasi nilai mereka dalam beberapa cara. Setiap variable termasuk ke dalam suatu domain (set nilai yang mungkin). Jadi dalam kasus Sudoku variable adalah grid , domain dari variable adalah bilangan bulat dari 1 sampai 9, dan kendala adalah aturan bahwa pada setiap sel-sel baris, kolom, dan sub kotak harus unik.

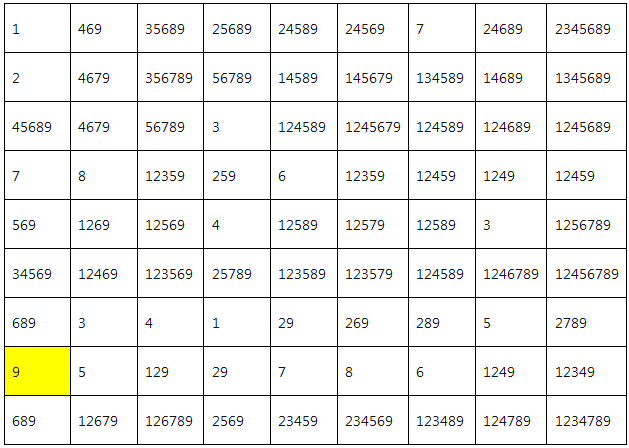
Propagasi kendala berarti mengurangi nilai yang mungkin dari variable. Misalnya dibawah ini adalah grid kemungkinan pemecahan teka-teki yang ditunjukkan diatas.



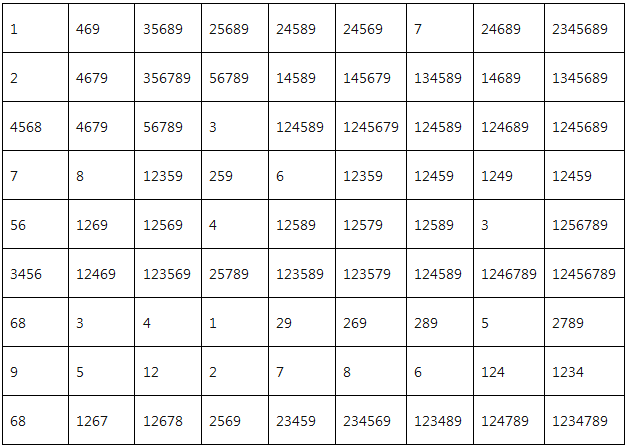
Pertimbangan sel pertama berada di (0,0) dengan nilai 1. Aturan unik berarti bahwa tidak ada sel lain pada baris, kolom, atau sub kotak yang sama, sehingga kita dapat menghapus nilai 1dari kemungkinan yang kotak. Di bawah ini merupakan grid yang telah dilakukan propagasi:



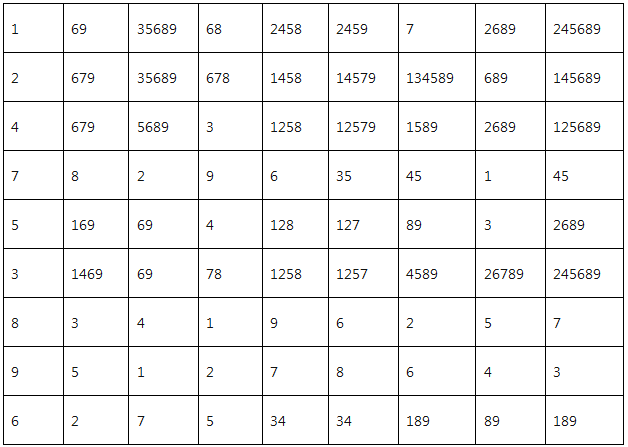
Kita dapat mengulangi proses ini untuk setiap sel dalam grid nilai (tunggal) yang pasti. Pada akhirnya kita menghapus begitu banyak nilai yang mungkin bahwa beberapa sel-sel baru juga mengandung nila tunggal. Hal ini berlaku untuk sel (7,0), dalam grid ini dapat mengurangi bisa mengurangi nilai 9.



Kemudian kita dapat melanjutkan proses dan menyebarkan nilai pada sel-sel itu juga.



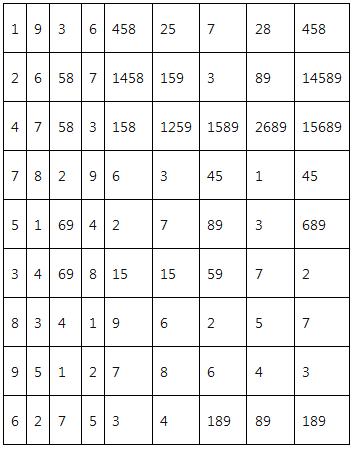
Berikut adalah contoh pada penyelesaian proses propagasi:



Terdapat beberapa cara tambahan pengaplikasian kenala unik yang disebut dengan aturan ‘unique value’. Pertimbangkan baris pertma dari nilai yang mungkin:



Sel ketiga adalah satu-satunya yang mengandung nilai 3. Ini berarti bahwa sel harus menjadi satu-satunya sel yang dapat memiliki nilai 3. Jadi jika mendeteksi situasi ini, baik untuk baris, kolom maupun sub kotak, kita dapat memaksa sel untuk memiliki nilai itu, dan sekali lagi menjutukan propagasi. Dengan menggunakan metode ini kita dapat lebih mengurangi contoh grid untuk kemungkinan berikut:



Hal ini dimungkinkan untuk memecahkan beberapa Sudoku murni menggunakan propagasi kendala secara mudah. Bahkan jika kita tidak mencapai solusi dengan propagasi kendala, hal utama adalah bahwa kita telah secara signifikan mengurangi kemungkinan nilai dari sel. Beberapa sel hanya memiliki dua kemungkinan nilai, jadi jika sekarang kita memilih sel-sel sebagai titik awal untuk metode jejak dan kesalahan kita hanya harus menghasilkan dan menguji dua kemungkinan.