Pemecah Sudoku Interaktif Berbasis Python

Proposal Tugas Akhir

Kelas TA 1

Muhammad Zakaria Musa NIM: 1103130047



Program Studi Sarjana Teknik Informatika
Fakultas Informatika
Universitas Telkom
Bandung
2017

Lembar Persetujuan

Pemecah Sudoku Interaktif Berbasis Python

A Python-Based Interactive Sudoku Solver.

Muhammad Zakaria Musa NIM: 1103130047

Proposal ini diajukan sebagai usulan pembuatan tugas akhir pada Program Studi Sarjana Teknik Informatika Fakultas Informatika Universitas Telkom

> Bandung, 1 November 2017 Menyetujui

Calon Pembimbing 1

Calon Pembimbing 2

Yanti Rusmawati, Ph.D.

NIP: 15711785-1

Muhammad Arzaki, M.Kom.

NIP: 15871701-2

Abstrak

Sudoku adalah sebuah permainan teka-teki yang biasanya dimainkan oleh satu orang. Dalam pengerjaannya banyak pemain sudoku yang terjebak dan tidak dapat menyelesaikan teka-teki. Hal ini disebabkan oleh pemain salah memasukkan angka atau sudoku tidak memiliki solusi. Oleh karena itu perlu dibuat sebuah pemecah sudoku yang dapat memberikan keterpenuhan dari sebuah sudoku. Sehingga pemain dapat mengetahui sebuah sudoku memiliki solusi atau tidak. Dalam pengerjaannya metode yang digunakan adalah SAT solver yang dibuat dalam bahasa python. Setelah pembuatan aplikasi selesai akan dilakukan pengujian untuk mengontrol kualitas aplikasi. Pengujian akan menggunakan metode pengujian kotak hitam

Kata Kunci: metode formal, SAT solver, Sudoku, pengujian kotak hitam

Daftar Isi

Al	ostra	k	i									
Daftar Isi												
Ι	Pen	dahuluan	1									
	1.1	Latar Belakang	1									
	1.2	Perumusan Masalah	2									
	1.3	Batasan Masalah	2									
	1.4	Tujuan	2									
	1.5	Rencana Kegiatan	2									
	1.6	Jadwal Kegiatan	3									
II	Kaj	ian Pustaka	4									
	2.1	Sudoku	4									
	2.2	Satisfiability $Problem(SAT)$	5									
		2.2.1 Sudoku Sebagai Masalah SAT	5									
III	Met	odologi	7									
	3.1	Analisa spesifikasi dan Kebutuhan sistem	7									
	3.2	Translasi spesifikasi sistem	7									
	3.3	Pembuatan aplikasi	8									
		3.3.1 Spesifikasi Aplikasi	8									
		3.3.2 <i>Analysis</i>	8									
		3.3.3 Design	8									
		3.3.4 <i>Coding</i>	9									
	3.4	Pengujian aplikasi	9									
	3.5	Analisis sistem	9									
Da	aftar	Pustaka	10									
Lampiran												

Bab I

Pendahuluan

1.1 Latar Belakang

Pada Tahun 2015 The Organization for Economic Co-operation and Development (OECD) merilis nilai dari Programme for International Student Assessment (PISA) [1]. Yang merupakan nilai kemampuan siswa pada matematika, membaca, dan membaca. Pada kategori matematika Indosnesia menempati peringkat 65 dari 71 negara dengan nilai 386. Sedangkan nilai rata-rata OECD untuk matematika adalah 470 [2]. Hal ini sungguh memprihatinkan. Oleh karena itu diperlukan katalis yang dapat mengembangkan kemampuan logical thinking serta problem solving anak, salah satunya adalah dengan sudoku.

Sudoku berasal dari kata $S\bar{u}ji$ wa dokushin ni kagiru yang berarti angkanya harus tunggal [3] adalah suatu puzzle (teka-teki) yang direpresentasikan oleh sebuah matriks (array dua dimensi) berukuran $n^2 \times n^2$ yang dibangun dari n^2 dengan submatriks (atau blok) yang berukuran $n \times n$. Sudoku merupakan puzzle yang biasanya dimainkan oleh satu orang. Pada awal permainan, terdapat beberapa sel yang telah terisi yang disebut dengan pemberian (givens). Untuk menyelesaikan permainan ini, seorang pemain harus mengisi setiap sel yang belum terisi dengan angka di antara 1 sampai n^2 sedemikan sehingga setiap baris, setiap kolom, dan setiap blok (submatriks berukuran $n \times n$) memuat tepat satu bilangan di antara 1 sampai n^2 . Biasanya suatu sudoku didesain agar tepat memiliki satu kemungkinan solusi. Hal ini juga mengakibatkan sudoku dapat diselesaikan hanya dengan mengandalkan penalaran yang sederhana. Pengisian suatu sel dapat dilakukan dengan meninjau kemungkinan dari isi sebuah sel.

Banyak pemain yang terjebak pada teka-teki sudoku dan tidak dapat melanjutkan permainan. Hal ini terjadi karena pemain mengisi nilai yang salah atau sudoku tidak memiliki solusi. Oleh karena itu banyak pemain yang membutuhkan pemecah sudoku yang dapat memberikan keterpenuhan dari sebuah sudoku. Sehingga pemain dapat mengetahui sebuah sudoku memiliki solusi atau tidak.

Hingga saat ini, sudah banyak penelitian yang membahas penyelesaian sudoku secara matematis maupun komputasional. Salah satu metode yang cu-

kup dikenal adalah penyelesaian sudoku dengan memanfaatkan masalah keterpenuhan formula proposisional (propositional satisfiability problem). Dengan pendekataan ini, syarat-syarat yang harus dipenuhi oleh suatu sudoku dimodelkan dengan satu atau lebih formula logika proposisi [4, 5]. Keterpenuhan (satisfiability) dari himpunan formula yang memodelkan syarat-syarat ini akan menjamin bahwa suatu sudoku memiliki suatu solusi.

Pada tugas akhir ini, penulis akan membuat pemecah sudoku interaktif berbasis python. Bahasa python dipilih karena implementasi SAT solver dalam bahasa python masih masih sedikit dibandingkan pada bahasa C dan C++ [6]. Aplikasi akan menggunakan pengujian kotak hitam untuk menilai kualitas aplikasi [7].

1.2 Perumusan Masalah

Berdasarkan latar belakang tersebut, rumusan masalah pada tugas akhir ini adalah bagaimana cara membuat SAT solver bebasis python untuk pemecah sudoku yang interaktif.

1.3 Batasan Masalah

Batasan pada tugas akhir ini terbatas pada:

- 1. Penulis hanya membangun aplikasi sudoku solver berukuran (4×4) , (9×9) , (16×16) .
- 2. Untuk setiap ukuran sudoku memiliki tiga tingkat kesulitan. Yaitu mudah, menengah, sulit.
- 3. Program mengeluarkan keterpenuhan dari sudoku.
- 4. Program mengeluarkan solusi dari sudoku jika pengguna mengiginkan.
- 5. Aplikasi akan diuji dengan metode pengujian kotak hitam (black box testing).

1.4 Tujuan

Tujuan yang ingin dicapai pada tugas akhir ini adalah untuk membuat pemecah sudoku yang interaktif.

1.5 Rencana Kegiatan

Pada pengerjaan tugas akhir ini beberapa hal yang akan saya lakukan adalah sebagai berikut:

- 1. Analisa spesifikasi dan Kebutuhan sistem.
- 2. Translasi spesifikasi sistem.

- 3. Pembuatan aplikasi.
- 4. Pengujian aplikasi.
- 5. Analisis sistem.
- 6. Penulisan laporan.

1.6 Jadwal Kegiatan

Jadwal pengerjaan tugas akhir sesuai dengan alur yang telah dibuat.

Tabel 1.1: Jadwal Kegiatan.

No	Jenis Kegiatan	Bulan						
	Jenis Regiatan	November	Desember	Januari	Februari	Maret	April 2018	Mei 2018
		2017	2017	2018	2018	2018		
1	Analisa spesifikasi dan ke-							
	butuhan sistem							
2	Translasi spesifikasi sistem							
3	Pembuatan aplikasi							
4	Pengujian aplikasi							
5	Analisis sistem							
6	Penulisan Laporan							

Bab II

Kajian Pustaka

Pada bab ini penulis akan menjelaskan teori yang digunakan selama pengerjaan tugas akhir.

2.1 Sudoku

Sudoku berasal dari kata $S\bar{u}ji$ wa dokushin ni kagiru yang berarti angkanya harus tunggal [3]. Sudoku dipopulerkan di jepang oleh Nikoli in the paper Monthly Nikolist pada April 1984. Sudoku adalah suatu puzzle (teka-teki) yang direpresentasikan oleh sebuah matriks (array dua dimensi) berukuran $n^2 \times n^2$ yang dibangun dari n^2 dengan submatriks (atau blok) yang berukuran $n \times n$. Sudoku merupakan puzzle yang biasanya dimainkan oleh satu orang. Pada awal permainan, terdapat beberapa sel yang telah terisi yang disebut dengan pemberian (givens). Untuk menyelesaikan permainan ini, seorang pemain harus mengisi setiap sel yang belum terisi dengan angka di antara 1 sampai n^2 sedemikan sehingga setiap baris, setiap kolom, dan setiap blok (submatriks berukuran $n \times n$) memuat tepat satu bilangan di antara 1 sampai n^2 . Biasanya suatu sudoku didesain agar tepat memiliki satu kemungkinan solusi. Hal ini juga mengakibatkan sudoku dapat diselesaikan hanya dengan mengandalkan penalaran yang sederhana. Pengisian suatu sel dapat dilakukan dengan meninjau kemungkinan dari isi sebuah sel.

Sebelum sudoku modern berkembang. Puzzle latin square terlebih dahulu berkembang yang di repretasikan oleh sebuah matriks berukuran $n \times n$ yang memiliki angka 1hinggan yang harus diisi oleh pemain. untuk menyelesakain puzzle setiap baris dan kolom pada latin square harus memiliki nilai angka yang berbeda. Latin square pertama kali diciptakan oleh Euler pada tahun 1783 [8]. Pada 19 November 1892 Le Siècle sebuah surat kabar di perancis mempublikasikan magic square yaitu sebuah teka-teki yang di repretasikan oleh sebuah matriks berukuran 9×9 yang memiliki dua buah submatriks berukuran 3×3 . Tujuan dari magic square adalah pemain mengisi baris dan kolom sehingga nilai penjumlahan setiap baris dan kolom itu sama. Sudoku modern sendiri lahir dari Howard Garns yang dipublikasikan di Dell Magazines pada 1979 [10].

2.2 Satisfiability Problem(SAT)

Masalah SAT (SAT problem) adalah salah satu masalah penting dalam logika komputasional [11]. Dalam pengerjaannya SAT berfokus pada membuktikan sebuah klausa satisfiable sehingga SAT berfokus pada menemukan sebuah model yang membuat klausa tersebut satisfiable. Jika tidak ditemukan sebuah model yang membuat klausa tersebut satisfiable maka klausa tersebut unsatisfiable. Masalah SAT merupakan masalah NP-complete, hingga saat ini tidak terdapat algoritma yang efisien untuk memecahkan masalah tersebut.

2.2.1 Sudoku Sebagai Masalah SAT

Sudoku memiliki beberapa aturan contohnya untuk sudoku berukuran 9×9 yaitu :

- 1. Setiap baris memuat bilangan antara 1 hingga 9.
- 2. Setiap kolom memuat bilangan antara 1 hingga 9.
- 3. Setiap submatriks atau blok 3×3 memuat bilangan antara 1 hingga 9.
- 4. Setiap sel memuat tepat satu bilangan antara 1 hingga 9.

Dari aturan-aturan tersebut akan ditranslasikan menjadi bentuk CNF lalu digunakan pada SAT *solver*. Dengan aturan yang telah ditranslasikan dalam CNF sebagai berikut:

1. Setiap baris memuat bilangan antara 1 hingga 9:

$$\textstyle \bigwedge_{r=1}^{9} \textstyle \bigwedge_{n=1}^{9} \textstyle \bigvee_{c=1}^{9} p\left(r,c,n\right)$$

2. Setiap kolom memuat bilangan antara 1 hingga 9:

$$\bigwedge_{c=1}^{9} \bigwedge_{n=1}^{9} \bigvee_{r=1}^{9} p(r, c, n)$$

3. Setiap submatriks atau blok 3×3 memuat bilangan antara 1 hingga 9 :

$$\bigwedge_{i=0}^{2} \bigwedge_{j=0}^{2} \bigwedge_{n=1}^{9} \bigvee_{r=3i+1}^{3i+3} \bigvee_{c=3j+1}^{3j+3} p(r,c,n)$$

4. Setiap sel memuat tepat satu bilangan antara 1 hingga 9:

$$\textstyle \bigwedge_{r=1}^{9} \textstyle \bigwedge_{c=1}^{9} \textstyle \bigwedge_{n=1}^{8} \textstyle \bigwedge_{i=n+1}^{9} \, \left(\neg p \left(r, c, n \right) \vee \neg p \left(r, c, i \right) \right)$$

Definisi dari p(r, c, n) adalah sel yang memiliki perpotongan baris r dan kolom c dengan nilai n, dengan $1 \le r, c, n \le 9$. Sebagai contoh :

- p(1,3,2) berarti sel pada baris 1 dan kolom 3 memuat nilai 2
- p(4,2,9) berarti sel pada baris 4 dan kolom 2 memuat nilai 9

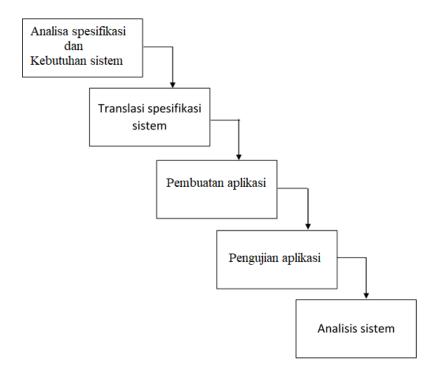
Pada aturan 1 untuk menyatakan baris r memuat bilangan antara 1 hingga 9, maka dapat ditulis dengan $\bigvee_{c=1}^{9} p(r,c,n)$. Untuk menyatakan bahwa baris r memuat semua bilangan pada $\{1,...,9\}$ maka dapat ditulis dengan

kolom c memuat semua bilangan pada $\{1,...,9\}$ maka dapat ditulis dengan $\bigwedge_{n=1}^{9}\bigvee_{r=1}^{9}p\left(r,c,n\right)$. Untuk menyatakan bahwa setiap kolom memuat semua bilangan pada $\{1,...,9\}$ dapat ditulis dengan $\bigwedge_{c=1}^{9}\bigwedge_{n=1}^{9}\bigvee_{r=1}^{9}p\left(r,c,n\right)$

Bab III

Metodologi

Pada bab ini penulis menjelaskan mengenai metodologi yang digunakan selama pengerjaan tugas akhir.



Gambar 3.1: Metodologi.

3.1 Analisa spesifikasi dan Kebutuhan sistem

Menganalisa spesifikasi-spesifikasi pada teka-teki sudoku, lalu membuat kebutuhan dari aplikasi sistem yang akan mengeluarkan *use case* aplikasi.

3.2 Translasi spesifikasi sistem

Spesifikasi yang sudah di dapat diubah dalam bentuk logika proposisi lalu dijadikan dalam bentuk CNF.

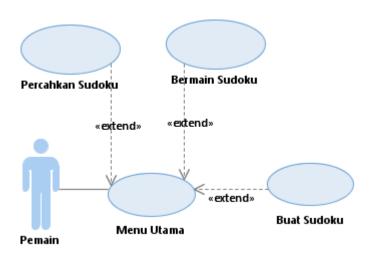
3.3 Pembuatan aplikasi

Spesifikasi yang sudah dalam bentuk CNF akan digunakan pada SAT Solver. Lalu use case yang didapat akan dibuat fungsionalitasnya. Pembuatan aplikasi menggunakan waterfall design tahapan-nya berupa:

3.3.1 Spesifikasi Aplikasi

Spesifikai aplikasi adalah sebagai berikut:

- 1. Bermain sudoku.
- 2. Membuat sudoku.
- 3. Memecahkan sudoku.



Gambar 3.2: Use Case Diagram.

3.3.2 Analysis

Dalam *analysis*, penulis menjabarkan mulai dari hal mendasar yaitu kendala global dalam pengembangan perangkat lunak yang sudah ada. Adapun masalahnya adalah minimnya aplikasi pemecah sudoku yang interaktif.

$3.3.3 \quad Design$

Fitur yang akan ada pada aplikasi tersebut adalah sebagai berikut

- 1. Bermain sudoku.
- 2. Membuat sudoku.
- 3. Pecahkan sudoku.

3.3.4 Coding

Lalu dalam tahap *coding* penulis akan menggunakan bahasa pemograman python dengan SAT *solver* [13] sebagai dasarnya. Setelah itu akan ditambahkan GUI dan *use case* yang telah di buat.

3.4 Pengujian aplikasi

Pada tahap ini aplikasi akan diujikan kebeberapa penguji dengan diberikan lembat *test case* yang meliputi beberapa aspek kualitas pada aplikasi. Teknik yang digunakan pada pengujian adalah pengujian kotak hitam [7].

3.5 Analisis sistem

Menganalisa kelayakan kualitas dari aplikasi berdasarkan hasil dari pengujian.

Daftar Pustaka

- [1] "Cara memecahkan teka teki sudoku," https://id.wikihow.com/Memecahkan-Teka-teki-Sudoku.
- [2] "The latest ranking of top countries in math, reading, and science is out," http://www.businessinsider.sg/pisa-worldwide-ranking-of-math-science-reading-skills-2016-12/?r=US&IR=T.
- [3] "Ed pegg jr.'s math games: Sudoku variations," http://www.mathpuzzle.com/MAA/41-Sudoku
- [4] G. Kwon and H. Jain, "Optimized CNF encoding for sudoku puzzles," in *Proc. 13th International Conference on Logic for Programming Artificial Intelligence and Reasoning (LPAR2006)*, 2006, pp. 1–5.
- [5] I. Lynce and J. Ouaknine, "Sudoku as a SAT Problem." in ISAIM, 2006.
- [6] "Understanding SAT by Implementing a Simple SAT Solver in Python," http://sahandsaba.com/understanding-sat-by-implementing-a-simple-sat-solver-in-python.html.
- [7] B. Beizer, Black-box testing: techniques for functional testing of software and systems. John Wiley & Sons, Inc., 1995.
- [8] S. Jones, P. Roach, and S. Perkins, "Properties of sudoku puzzles," in *Proceedings of the 9th international conference on Software Engineering*. IEEE Computer Society Press, 11 2007, pp. 7–11.
- [9] "Supplément de l'article: Les ancêtres français du sudoku," https://web.archive.org/web/20061210103525/http://cboyer.club.fr/multimagie/SupplAnce
- [10] "The answer men," http://content.time.com/time/magazine/article/0,9171,2137423,00.htm
- [11] M. Huth and M. Ryan, Logic in Computer Science: Modelling and reasoning about systems. Cambridge University Press, 2004.
- [12] W. W. Royce, "Managing the development of large software systems: concepts and techniques," in *Proceedings of the 9th international conference*

on Software Engineering. IEEE Computer Society Press, 1987, pp. 328–338.

[13] "A simple SAT solver implemented in Python (with some non-SAT abilities)." https://github.com/stephenroller/satsolver.

Lampiran

Pada bab ini akan berisi lampiran dari pengerjaan tugas akhir, contohnya skrip ${\rm NuSMV}.$