

**APLIKASI *GRAPH COLORING* DENGAN ALGORITMA *TABU SEARCH*
DALAM PENYELESAIAN MASALAH PENJADWALAN KERETA API**

Skripsi

untuk memenuhi sebagian persyaratan

mencapai derajat Sarjana S-1

Program Studi Matematika



diajukan oleh

Rifka Wulan Permatasari

11610039

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UIN SUNAN KALIJAGA
YOGYAKARTA

2015



PENGESAHAN SKRIPSI/TUGAS AKHIR

Nomor : UIN.02/D.ST/PP.01.1/015/2016

Skripsi/Tugas Akhir dengan judul : Aplikasi *Graph Coloring* dengan Algoritma *Tabu Search* dalam Penyelesaian Masalah Penjadwalan Kereta Api

Yang dipersiapkan dan disusun oleh :
Nama : Rifka Wulan Permatasari
NIM : 11610039
Telah dimunaqasyahkan pada : 28 Desember 2015
Nilai Munaqasyah : A

Dan dinyatakan telah diterima oleh Fakultas Sains dan Teknologi UIN Sunan Kalijaga

TIM MUNAQASYAH :

Ketua Sidang

Noor Saif Muli, M.Sc
NIP. 19820617 200912 1 005

Penguji I

Much. Abrori, S.Si, M.Kom
NIP.19720423 199903 1 003

Penguji II

Dr. Muhammad Wakhid Musthofa, M.Si
NIP.19800402 200501 1 003

Yogyakarta, 4 Januari 2016

UIN Sunan Kalijaga
Fakultas Sains dan Teknologi



Dr. Maizer Said Nahdi, M.Si
NIP. 19550427 198403 2 001

HALAMAN PERSETUJUAN SKRIPSI

Hal : Persetujuan Skripsi

Lamp : 3 Eksemplar Skripsi

Kepada

Yth. Dekan Fakultas Sains dan Teknologi

UIN Sunan Yogyakarta di Yogyakarta

Assalamu'alaikum wr. wb.

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengadakan perbaikan seperlunya, maka kami selaku pembimbing berpendapat bahwa skripsi Saudara:

Nama : Rifka Wulan Permatasari

NIM : 11610039

Judul Skripsi : *Aplikasi Graph Coloring* dengan Algoritma *Tabu Search* dalam Penyelesaian Masalah Penjadwalan Kereta Api (Studi Kasus: Kereta Api Yogyakarta-Solo)

Sudah dapat diajukan kembali kepada Program Studi Matematika Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu dalam Matematika.

Dengan ini kami berharap agar skripsi/tugas akhir Saudara tersebut diatas dapat segera dimunaqisyahkan. Atas perhatiannya kami ucapkan terimakasih

Wassalamu'alaikum wr. wb.

Yogyakarta, 14 Desember 2015
Pembimbing



Noor Saif Muhammad Mussafi, M. Sc.

NIP: 19820617 200912 1 005

HALAMAN PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini:

Nama : Rifka Wulan Permatasari

NIM : 11610039

Program Studi : Matematika

Fakultas : Sains dan Teknologi

Dengan ini saya menyatakan bahwa skripsi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 14 Desember 2015

Yang menyatakan



Rifka Wulan Permatasari

NIM.11610039

HALAMAN PERSEMBAHAN

Atas karunia Allah Subhanahu Wata'ala

Karya sederhana ini saya persembahkan untuk :

Sayekti, mamaku tersayang yang selalu memberi motivasi, memberi kasih sayangnya, memberikan seluruh energi dalam mengasuh, mendidik serta memanjatkan doa untuk kelancaran putrimu ini.

Heryanto, papaku tercinta, papa yang selalu memperhatikan setiap langkah yang saya ambil, selalu memberikan semua pengorbanan yang tulus dan dukungan dalam bentuk apapun yang tak ternilai harganya serta doa untuk putrimu ini.

Andryan Muhammad Ikram, adikku yang selalu bersikap dewasa, terima kasih atas dukungan, bantuan, dan semangat untuk setiap hal yang saya lakukan.

Muhammad Rangga Al-Fathir, adikku tersayang yang paling kecil dan lucu, yang selalu membuat wajah ini tersenyum.

Muktashim Amrillah, terima kasih untuk semuanya. Semua motivasi, bantuan, arahan, semangat dan semuanya. Thanks for everything ^^

Almamater tercinta Universitas Islam Negeri Sunan Kalijaga Yogyakarta dan Fakultas Sains dan Teknologi serta Progam Studi Matematika.

MOTTO

“Hiduplah seperti pohon kayu yang lebat buahnya; hidup di tepi jalan dan dilempari orang dengan batu, tetapi dibalas dengan buah”

(Abu Bakar Sibli)

The best revenge for the people who have insulted you is the success that you can show them later.

Happiness is not how much money we have, but how much time we can be thankful.

KATA PENGANTAR

Puji syukur kehadirat Allah SWT yang telah melimpahkan segala rahmat dan hidayah-Nya, sehingga skripsi yang berjudul “Aplikasi *Graph Coloring* dengan Algoritma *Tabu Search* dalam Penyelesaian Masalah Penjadwalan Kereta Api” dapat terselesaikan guna memenuhi syarat memperoleh gelar kesarjanaan di Program Studi Matematika Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta.

Shalawat serta salam senantiasa tercurahkan kepada Nabi besar Muhammad SAW, yang membawa umat manusia dari zaman kegelapan menuju zaman yang terang seperti saat ini. Penulis menyadari skripsi ini tidak akan selesai tanpa motivasi, bantuan, bimbingan dan arahan dari berbagai pihak. Oleh karena itu, dengan kerendahan hati penulis mengucapkan rasa terimakasih kepada:

1. Ibu Dr. Maizer Said Nahdi, M.Si, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sunan Kalijaga Yogyakarta.
2. Bapak Dr. M. Wakhid Musthofa, M.Si, selaku Ketua Program Studi Matematika Fakultas Sains dan Teknologi Universitas Islam Negeri Sunan Kalijaga Yogyakarta.
3. Bapak Noor Saif Muhammad Mussafi, M.Sc, selaku dosen pembimbing skripsi, yang selalu meluangkan waktunya dalam membimbing, memotivasi, serta mengarahkan sehingga skripsi ini dapat terselesaikan.

4. Bapak/Ibu Dosen dan Staf Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta atas ilmu, bimbingan dan pelayanan selama perkuliahan dan penyusunan skripsi.
5. Mama, papa dan adik-adikku tersayang dan seluruh keluarga besar di Yogyakarta yang selalu memberikan dukungan baik moril maupun materiil.
6. Teman hidupku Muktashim Amrillah atas semua bantuan, dorongan dan semangatnya untukku.
7. Sahabat-sahabat terbaikku Silke, Irma, Insan, Esti, Tina, Fergie, Riska, dan Vao yang selalu memberikan semangat, bantuan dan dukungannya kepadaku.
8. Temanku Muhammad Zakuan dan Sulistiono serta teman-teman matematika 2011 yang selalu memberikan bantuan dan motivasi hingga terselesaikannya skripsi.
9. PT. KAI DAOP 6 atas semua bantuannya dalam penelitian.
10. Kepada semua pihak yang tidak dapat penulis sebutkan satu per satu, atas doa dan motivasinya yang telah membantu dalam penyusunan skripsi.

Penulis menyadari masih terdapat kesalahan dan kekurangan dalam penulisan skripsi ini, oleh karena itu penulis mengharapkan adanya kritik dan saran yang membangun dari semua pihak. Akhir kata, penulis berharap semoga skripsi ini bisa bermanfaat dan membantu bagi berbagai pihak.

Yogyakarta, 12 Desember 2015

Penulis

Rifka Wulan Permatasari

NIM. 11610039

DAFTAR ISI

HALAMAN JUDUL.....	i
PENGESAHAN SKRIPSI	ii
PERSETUJUAN SKRIPSI	iii
PERNYATAAN KEASLIAN SKRIPSI.....	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTTO	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiv
DAFTAR LAMPIRAN.....	xvi
ABSTRAK	xvii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah.....	3
1.5 Manfaat Penelitian	4
1.6 Tinjauan Pustaka	5

1.7 Metode Penelitian.....	7
1.8 Sistematika penulisan.....	9
BAB II DASAR TEORI	11
2.1 Teori Graf.....	11
2.1.1 Definisi Graf.....	12
2.1.2 Graf Sederhana	12
2.1.3 Graf Lengkap (<i>Complete Graph</i>)	13
2.1.4 Graf Ganda (<i>Multigraph</i>).....	14
2.1.5 Istilah dalam Graf	15
2.2 Pewarnaan Graf.....	17
2.3 Penjadwalan	20
2.4 Aplikasi Pewarnaan Graf dalam Penjadwalan	20
2.5 Algoritma <i>Tabu Search</i>	22
2.6 Mekanisme <i>Tabu Search</i>	23
2.7 HTML (<i>Hyper Text Markup Language</i>)	26
2.8 CSS <i>Bootstraps</i>	27
2.9 PHP (<i>Personal Home Page Tools</i>).....	28
2.9.1 Script PHP	29
2.9.2 Operator Aritmatika, <i>Relational</i> , Logika dan <i>Assignment</i>	30
2.9.3 <i>Statement Control</i> pada PHP.....	32
2.10 DBMS (<i>Database Management Systems</i>).....	36

BAB III PEMBAHASAN.....	37
3.1 Transformasi Penjadwalan ke dalam Graf	37
3.1.1 Cara Kerja Transformasi Data ke Graf	38
3.1.2 Studi Kasus	39
3.2 Pewarnaan Graf pada Penjadwalan menggunakan Algoritma	
<i>Tabu Search</i>	41
3.2.1 Cara Kerja Pewarnaan Graf menggunakan	
Algoritma <i>Tabu Search</i>	43
3.2.2 Penentuan Waktu Kereta Api Yogyakarta-Solo	44
3.2.3 Penerapan Algoritma <i>Tabu Search</i> dalam	
menentukan Penjadwalan Kereta Api	48
3.3 Rancang Bangun Penjadwalan menggunakan Algoritma	
<i>Tabu Search</i>	62
3.3.1 Spesifikasi	63
3.3.2 Membuat Aplikasi Bahasa Pemrograman PHP	63
3.3.3 Uji Coba Program	75
BAB VI PENUTUP	86
4.1 Kesimpulan	86
4.2 Saran.....	87
DAFTAR PUSTAKA	89
LAMPIRAN-LAMPIRAN.....	91

DAFTAR TABEL

Tabel 1.1 Perbedaan Penelitian	6
Table 2.1 Operator Aritmatika	30
Tabel 2.2 <i>Relational Operator</i>	31
Tabel 2.3 Operator Logika	31
Tabel 2.4 <i>Assignment Operator</i>	32
Tabel 3.1 Pengambilan Mata Kuliah.....	40
Tabel 3.2 Kereta Api Yogyakarta-Solo.....	45
Tabel 3.3 Waktu Tunggu di Stasiun dalam Satuan Menit	47
Tabel 3.4 Rute Kereta Api	48
Tabel 3.5 Representasi Kereta.....	49
Tabel 3.6 Status Ketetangaan.....	50
Tabel 3.7 Hasil Pewarnaan Acak Kereta.....	54
Tabel 3.8 Konflik Pewarnaan.....	55
Tabel 3.9 Iterasi Pertama <i>Tabu Search</i>	57
Tabel 3.10 Tabu List	57
Tabel 3.11 Solusi Optimal.....	58
Tabel 3.12 Pengelompokkan <i>Nodes</i> dan Kereta Berdasarkan Warna.....	59
Tabel 3.13 Pemetaan Kereta ke Jadwal Pemberangkatan Kereta	62
Tabel 3.14 Spesifikasi Perangkat Keras.....	63
Tabel 3.15 Spesifikasi Perangkat Lunak.....	63

Tabel 3.16 Solusi Optimal Pewarnaan Graf dengan *Tabu Search*
menggunakan Rancang Bangun Program85

Tabel 4.1 Solusi Optimum Perhitungan Manual dan Rancang Bangun..87



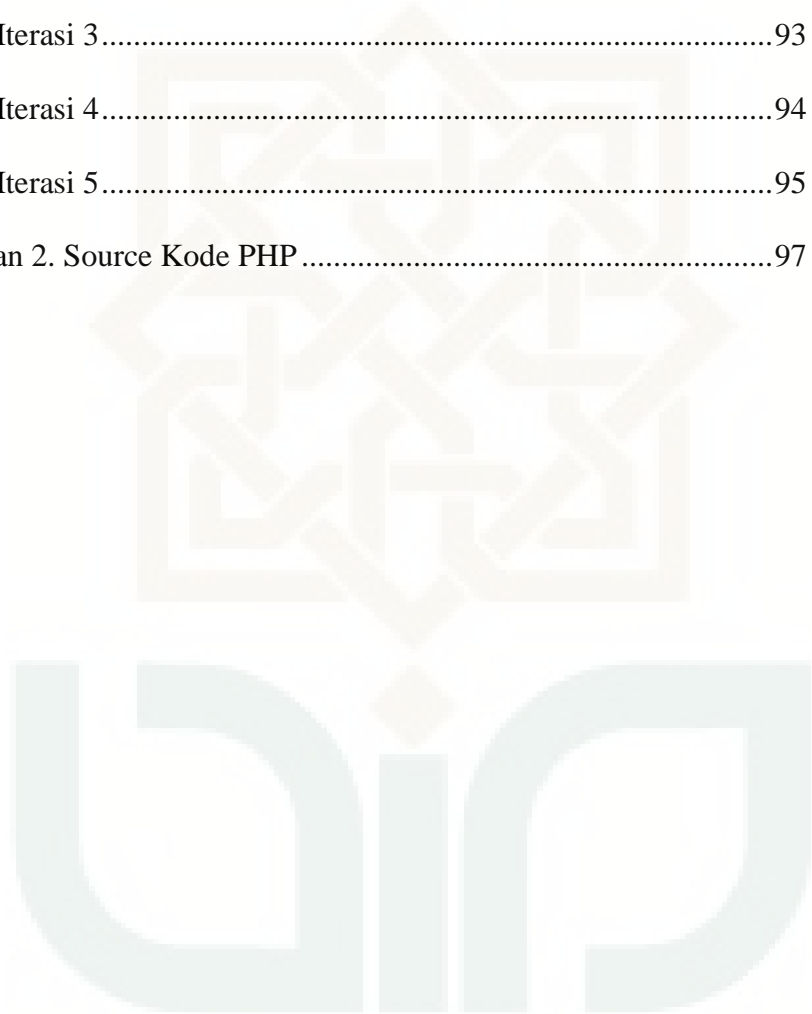
DAFTAR GAMBAR

Gambar 1.1 Skema Langkah Penelitian	9
Gambar 2.1 Contoh Graf.....	12
Gambar 2.2 Graf Lengkap K_n $1 \leq n \leq 6$	13
Gambar 2.3 Contoh Graf Ganda (<i>Multigraph</i>)	15
Gambar 2.4 Keterhubungan Graf G	15
Gambar 2.5 Keterkaitan Graf G	16
Gambar 2.6 Derajat Titik Graf G	16
Gambar 2.7 Graf G	18
Gambar 2.8 Ilustrasi <i>Insertion Move</i>	24
Gambar 2.9 Ilustrasi <i>Swap Move</i>	25
Gambar 2.10 Mekanisme Umum <i>Tabu Search</i>	26
Gambar 2.11 (a) Contoh Program PHP (b) Tampilan Program.....	29
Gambar 3.1 <i>Flowchart</i> Algoritma <i>Tabu Search</i>	38
Gambar 3.2 Representasi Data ke dalam Graf	40
Gambar 3.3 <i>Flowchart</i> Pewarnaan Graf menggunakan Algoritma <i>Tabu Search</i>	44
Gambar 3.4 Transformasi Data ke dalam Graf	51
Gambar 3.5 Graf Hasil Pewarnaan.....	58
Gambar 3.6 Tampilan Awal Program	75
Gambar 3.7 Form Awal Kota.....	77

Gambar 3.8 Form Input Data Kota	77
Gambar 3.9 Form Awal Jadwal	78
Gambar 3.10 Form Input Jadwal.....	79
Gambar 3.11 Form Status Ketetangaan	80
Gambar 3.12 Graf Awal.....	81
Gambar 3.13 Tampilan Proses Perhitungan.....	82
Gambar 3.14 Graf Hasil Pewarnaan menggunakan Program	83
Gambar 3.15 Jadwal Kereta setelah dilakukan Pewarnaan.....	84

DAFTAR LAMPIRAN

Lampiran 1. Iterasi	91
1. Iterasi 1	91
2. Iterasi 2	92
3. Iterasi 3	93
4. Iterasi 4	94
5. Iterasi 5	95
Lampiran 2. Source Kode PHP	97



ABSTRAK

APLIKASI *GRAPH COLORING* DENGAN ALGORITMA *TABU SEARCH* DALAM PENYELESAIAN MASALAH PENJADWALAN KERETA API

Oleh:

Rifka Wulan Permatasari

NIM. 11610039

Penjadwalan berperan penting dalam proses kegiatan yang berlangsung dalam kehidupan sehari-hari. Penjadwalan kereta khususnya sangat penting dalam dunia transportasi. Pembuatan jadwal ini memiliki berbagai kendala, seperti keterbatasan jumlah dan kapasitas kereta, permintaan konsumen, dan lokasi tujuan kereta. Salah satu usaha yang dapat dilakukan yaitu mengoptimalkan waktu penjadwalan setiap harinya.

Permasalahan penjadwalan kereta dapat diselesaikan dan direpresentasikan menggunakan graf sederhana. Nomor kereta dinyatakan sebagai *node* dan rute dinyatakan sebagai sisi. Kasus penjadwalan tersebut dapat diselesaikan dengan penerapan *graph coloring* menggunakan Algoritma *Tabu Search*. Cara kerja Algoritma *Tabu Search* dimulai dengan penentuan *initial solution* menggunakan pewarnaan *random*, evaluasi *move* menggunakan metode *swap*, *update Tabu List*, kemudian apabila kriteria pemberhentian terpenuhi maka proses Algoritma *Tabu Search* berhenti, jika tidak maka kembali pada evaluasi *move*. Proses perhitungan Algoritma *Tabu Search* dilakukan secara manual dan rancang bangun menggunakan bahasa pemrograman *PHP* pada PT. KAI DAOP 6.

Berdasarkan proses perhitungan yang dilakukan manual dan menggunakan rancang bangun diperoleh dua solusi alternatif yang optimal dengan pewarnaan minimum sebanyak 12 warna. Solusi optimal pertama diperoleh menggunakan perhitungan manual terdiri dari 12 jadwal pemberangkatan kereta dimulai dengan nomor kereta 256-275, 255-276, 277-289, 273-288, 253-287, 280-285, 281-272, 254-286, 278-291, 281-290, 284-279, 274-282. Solusi optimal kedua menggunakan perhitungan rancang bangun diperoleh jadwal pemberangkatan kereta dimulai dengan nomor kereta 283-276, 290-280, 255-279, 256-272, 278-282, 275-284, 274-289, 273-285, 253-291, 286-254, 287-277, 288-281 dengan jumlah jadwal pemberangkatan yang sama, yaitu 12 jadwal.

Kata Kunci: Penjadwalan kereta, *Graph Coloring*, Algoritma *Tabu Search*

BAB I

PENDAHULUAN

1.1 Latar Belakang

Teori graf merupakan salah satu ilmu yang membantu dalam mengatasi berbagai permasalahan dalam berbagai disiplin ilmu. Salah satu cabang matematika ini mempunyai banyak manfaat dalam mengatasi permasalahan sosial maupun permasalahan dalam kehidupan sehari-hari. Teori graf mulai dikenalkan oleh salah satu matematikawan yang bernama *Leonhard Euler* pada tahun 1736. Ide tersebut muncul pada saat penyelesaian masalah jembatan *Königsberg*. Masalah tersebut kemudian dimodelkan Euler dalam bentuk graf dengan memisalkan daratan sebagai sebuah simpul (*node*) dan jembatan yang menghubungkan antara sebarang dua daratan sebagai sebuah sisi.

Salah satu permasalahan yang banyak memanfaatkan teori graf adalah masalah penjadwalan. Beker (1974) mengemukakan penjadwalan merupakan kegiatan untuk mengalokasikan sejumlah sumber daya yang tersedia untuk memastikan bahwa perencanaan dapat berjalan dengan baik dengan waktu dan tenaga yang digunakan secara efisien. Salah satu bidang yang memanfaatkan teori graf dalam penjadwalan adalah dunia perkeretaapian.

Kereta Api merupakan salah satu alat transportasi yang banyak digunakan oleh masyarakat, selain murah serta memiliki kenyamanan keselamatan perjalanan, mereka memiliki ketepatan dalam waktu, sehingga menjadi alasan mengapa

transportasi ini banyak digunakan. Dengan semakin tingginya arus kepadatan penumpang yang menggunakan kereta api, khususnya pada rute Yogyakarta-Solo dan Solo-Yogyakarta, hal ini berakibat Perseroan Terbatas Kereta Api Indonesia (PT. KAI) harus mampu mengoptimalkan penjadwalan keberangkatan kereta yang disesuaikan dengan tingkat kebutuhan angkutan kereta api. Setiap kereta api yang berangkat dari suatu stasiun dipastikan selalu singgah di beberapa stasiun sebelum mencapai stasiun tujuan. Hal ini mengakibatkan resiko yang besar yang harus diambil PT. KAI dalam menyusun penjadwalan operasional keberangkatan kereta api, yang tentu membutuhkan banyak informasi tentang keadaan lintasan sepanjang ditempuhnya perjalanan kereta api.

Permasalahan tersebut dapat diatasi dengan suatu sistem yang dapat membantu dalam mengatasi dan mengoptimalkan dalam pembuatan penjadwalan kereta api. Salah satu algoritma yang menangani masalah penjadwalan adalah Algoritma *Tabu Search*. Menurut Aladag dan Hocaoglu (2007: 56) *Tabu Search* adalah sebuah metode optimasi yang berbasis pada *local search*. Proses pencarian bergerak dari satu solusi ke solusi berikutnya, dengan cara memilih solusi tetangga (*neighborhood*) terbaik dari solusi sekarang (*current*) yang tidak tergolong solusi terlarang (*tabu*). Ide dasar dari Algoritma *Tabu Search* adalah mencegah proses pencarian dari *local search* agar tidak melakukan pencarian ulang pada ruang solusi yang sudah pernah ditelusuri, dengan memanfaatkan suatu struktur memori yang mencatat sebagian jejak proses pencarian yang telah dilakukan. Oleh karena itu, dalam penelitian ini akan dibahas suatu sistem yang dibangun menggunakan *Tabu*

Search dan diharapkan dapat membantu petugas PT. KAI dalam membuat penjadwalan yang optimal.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijabarkan, maka dapat dirumuskan masalah-masalah sebagai berikut:

1. Bagaimana merancang dan merepresentasikan permasalahan penjadwalan sistem kereta api rute Yogyakarta-Solo dan Solo-Yogyakarta ke dalam graf?
2. Bagaimana mengimplementasikan pewarnaan graf dengan menggunakan Algoritma *Tabu Search* dalam sebuah sistem penjadwalan kereta api rute Yogyakarta-Solo dan Solo-Yogyakarta?

1.3 Tujuan Penelitian

Tujuan dari penelitian ini ditetapkan berdasarkan rumusan masalah yang telah dijabarkan. Tujuan penelitian ini antara lain:

1. Merancang dan merepresentasikan permasalahan penjadwalan sistem kereta api rute Yogyakarta-Solo dan Solo-Yogyakarta ke dalam graf.
2. Mengimplementasikan pewarnaan graf dengan menggunakan Algoritma *Tabu Search* dalam sebuah sistem penjadwalan kereta api rute Yogyakarta-Solo dan Solo-Yogyakarta.

1.4 Batasan Masalah

Batasan-batasan dari masalah yang dihadapi perlu ditetapkan, dengan maksud agar pembahasan dan penyusunan sistem dapat dilakukan secara terarah dan sesuai

dengan yang diharapkan. Adapun batasan-batasan masalah pada penelitian ini antara lain:

1. Studi kasus yang ditelaah dalam masalah penjadwalan kereta api ini adalah penjadwalan kereta dengan rute Yogyakarta-Solo dan Solo-Yogyakarta dengan mengkondisikan jalur kereta api adalah pada rute ganda. Hal ini dikarenakan jalur kereta Yogyakarta-Solo menggunakan rute ganda.
2. Penyusunan jadwal kereta jalur Yogyakarta-Solo dibuat dengan mengabaikan kereta lain yang melewati jalur tersebut.
3. Variabel yang digunakan pada skripsi ini adalah nomor kereta api, kota awal dan kota tujuan.
4. Stasiun yang digunakan hanya Stasiun Yogyakarta dan Stasiun Solo Balapan.
5. Program (rancang bangun) menggunakan Bahasa *PHP*.
6. Program tidak menghitung waktu karena waktu sudah ditetapkan (*default*).
7. Program dapat menghitung secara optimum dengan jumlah minimum 24 kereta.
8. Program hanya dapat menghitung kereta yang memiliki jalur sama dengan rute tujuan berbeda, dengan jumlah kereta yang seimbang.

1.5 Manfaat Penelitian

Manfaat dari penelitian yang dapat diperoleh diantaranya:

1. Manfaat bagi akademisi, menambah wawasan tentang pewarnaan graf, khususnya dalam Algoritma *Tabu Search* dan pemanfaatannya untuk pelayanan publik seperti penjadwalan kereta api rute Yogyakarta-Solo.

2. Manfaat bagi praktisi, memberi rekomendasi model yang baru untuk pembuatan jadwal sehingga dapat menjadi salah satu acuan untuk dapat mengembangkan sistem baru yang lebih efektif dan efisien.
3. Manfaat bagi PT. KAI, memberikan masukan dalam menentukan penyusunan jadwal kereta api rute Yogyakarta-Solo dan Solo-Yogyakarta.

1.6 Tinjauan Pustaka

Ada beberapa penelitian terdahulu yang membahas mengenai aplikasi pewarnaan graf yang juga menjadi acuan dalam penelitian ini, diantaranya:

1. Jurnal yang berjudul "*Implementasi Masalah Pewarnaan Graf dengan Algoritma Tabu Search pada Penjadwalan Kuliah*" yang ditulis oleh Ida Suryani, Purwanto, dan Mohamad Yasin. (Universitas Negeri Malang, 2013). Jurnal tersebut membahas mengenai pembuatan jadwal kuliah, khususnya jadwal UTS semester ganjil dengan menggunakan program *Borland Delphi 7*. Pada penelitian ini juga dijelaskan bagaimana membuat sebuah jadwal kuliah dengan Algoritma *Tabu Search* dengan menggunakan 3 variabel, yaitu jenis mata kuliah, ruang kelas, dan dosen pengampu mata kuliah tersebut.
2. Skripsi yang berjudul "*Aplikasi Metode Pewarnaan Graf pada Penjadwalan Kegiatan Perkuliahan di Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta*" yang ditulis oleh Muhammad Mahrus. (Universitas Islam Negeri Sunan Kalijaga, 2011). Skripsi tersebut memaparkan mengenai masalah pembuatan jadwal perkuliahan di Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta. Pada penelitian ini dibahas mengenai penerapan

pewarnaan graf terhadap penjadwalan teratur dan penjadwalan acak. Algoritma yang digunakan pada skripsi tersebut menggunakan Algoritma *Welch-Powell*.

3. Skripsi yang berjudul “*Penerapan Pewarnaan Graf Menggunakan Algoritma Welch-Powell pada Radio Frequency Assignment Problem di Provinsi Daerah Istimewa Yogyakarta (DIY)*” yang ditulis oleh Nurkhasanah. (Universitas Islam Negeri Sunan Kalijaga, 2013). Skripsi tersebut memaparkan mengenai pembuatan model distribusi frekuensi gelombang radio di Provinsi DIY agar tidak terjadi interferensi. Pada penelitian ini juga dijelaskan bagaimana Algoritma *Welch-Powell* diaplikasikan pada *Radio Frequency Assignment Problem*.

Penelitian yang menjadi tinjauan pustaka penulis, disajikan dalam bentuk sebagai berikut :

Tabel 1.1 Perbedaan Penelitian

No.	Nama Peneliti	Judul Penelitian	Perbedaan
1	Ida Suryani, Purwanto, dan Mohamad Mahrus (2013)	Implementasi Masalah Pewarnaan Graf dengan Algoritma <i>Tabu Search</i> pada Penjadwalan Kuliah	Jurnal tersebut membahas tentang pembuatan jadwal UTS dengan program <i>Borland Delphi 7</i> , sedangkan skripsi ini membahas pembuatan jadwal kereta api dengan program berbasis PHP.
2	Muhammad Mahrus (2011)	Aplikasi Metode Pewarnaan Graf pada Penjadwalan Kegiatan Perkuliahan di Fakultas Sains dan Teknologi	Skripsi tersebut membahas tentang pembuatan jadwal kuliah dengan Algoritma <i>Welch-Powell</i> , sedangkan skripsi

		UIN Sunan Kalijaga Yogyakarta	ini membahas pembuatan jadwal kereta api dengan Algoritma <i>Tabu Search</i> .
3	Nurkhasanah (2013)	Penerapan Pewarnaan Graf Menggunakan Algoritma <i>Welch-Powell</i> pada <i>Radio Frequency Assignment Problem</i> di Provinsi Daerah Istimewa Yogyakarta (DIY)	Skripsi tersebut membahas tentang pembuatan model distribusi frekuensi gelombang radio dengan Algoritma <i>Welch-Powell</i> , sedangkan skripsi ini membahas pembuatan jadwal kereta api dengan Algoritma <i>Tabu Search</i> .

Skripsi dengan judul “Aplikasi *Graph Coloring* dengan Algoritma *Tabu Search* dalam Penyelesaian Masalah Penjadwalan Kereta Api” ini, menggunakan ketiga penelitian tersebut sebagai acuan. Penelitian ini akan menyelesaikan *graph coloring* menggunakan Algoritma *Tabu Search* dengan metode perbaikan *swap* kemudian dibuat program (rancang bangun) menggunakan bahasa pemrograman PHP dengan harapan dapat menghasilkan solusi yang optimal dan mempermudah penggunaannya ataupun pengembangannya.

1.7 Metode Penelitian

Berdasarkan tujuannya, penelitian ini termasuk dalam jenis penelitian kualitatif. Penelitian ini bersifat deskriptif, yaitu mencari fakta dengan interpretasi yang tepat dengan tujuan untuk membuat secara sistematis dan akurat mengenai fakta-fakta. Objek penelitian ini adalah jadwal kereta api rute Yogyakarta-Solo dan Solo-Yogyakarta dengan menggunakan algoritma berbasis *local search*, yaitu Algoritma *Tabu Search*.

Metode pengumpulan data dan analisis data dilakukan dengan dua cara, yaitu:

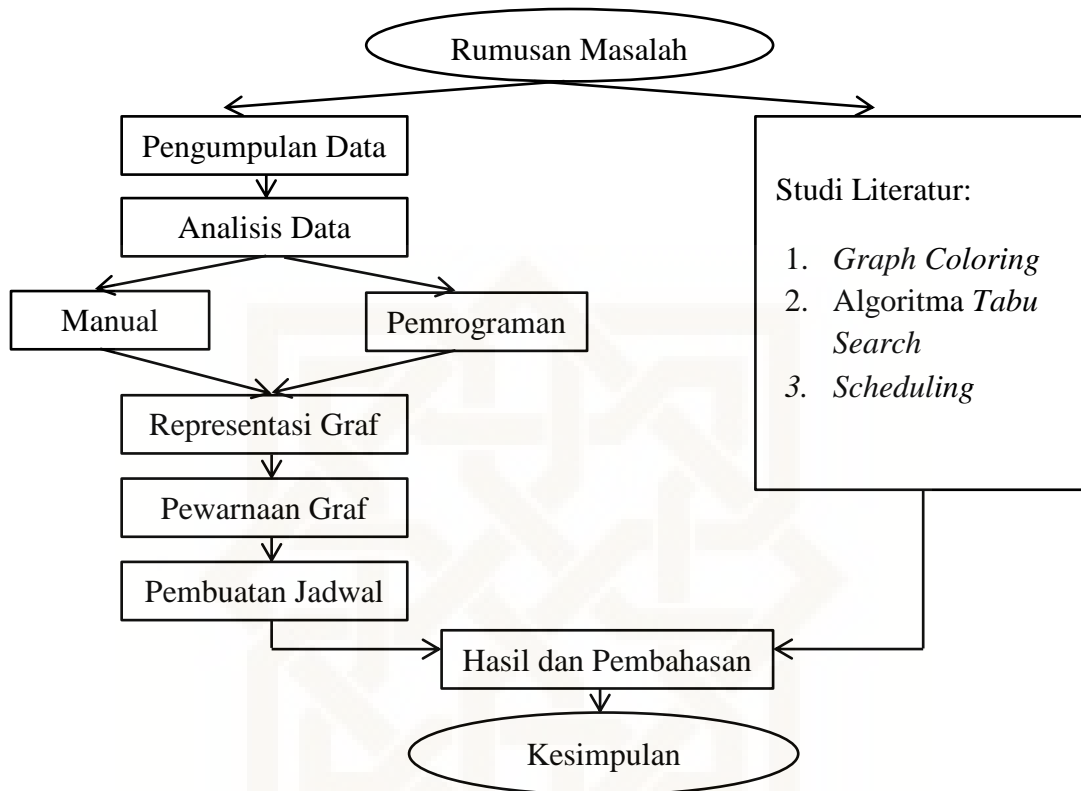
1. Penelitian lapangan

Pada penelitian ini memerlukan data primer dari PT. Kereta Api Indonesia DAOP 6 melalui wawancara. Adapun data sekunder diperoleh dengan cara mencari data-data dari internet.

2. Studi Literatur

Untuk menganalisis model dengan Algoritma *Tabu Search*, penyelesaian, hingga kesimpulan. Diperlukan buku-buku, makalah, ataupun jurnal yang dapat digunakan sebagai referensi.

Selanjutnya diberikan skema penelitian yang meliputi langkah penelitian, pengumpulan data, studi literatur, hasil solusi dan pembahasan serta kesimpulan (Perhatikan Gambar 1.1).



Gambar 1.1 Skema Langkah Penelitian

1.8 Sistematika Penulisan

Sistematika yang digunakan dalam penelitian ini adalah sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini membahas tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian, tinjauan pustaka, dan sistematika penelitian. Bab ini menjadi kerangka yang digunakan dalam penelitian ini.

BAB II LANDASAN TEORI

Pada bab ini membahas tentang berbagai pengertian dasar yang akan menjadi landasan pembahasan dalam bab berikutnya. Materi pokok dalam bab ini yaitu teori-

teori yang berkaitan tentang graf, Algoritma *Tabu Search*, penjadwalan kereta api dan PHP.

BAB III PEMBAHASAN

Bab ini merupakan pembahasan dari hasil penelitian yang berupa penyelesaian persoalan penjadwalan kereta api rute Yogyakarta-Solo maupun Solo-Yogyakarta dengan Algoritma *Tabu Search*.

BAB IV PENUTUP

Bab ini berisi kesimpulan dari hasil penelitian yang dilakukan dan saran-saran yang membangun.

BAB IV

PENUTUP

4.1 Kesimpulan

Berdasarkan hasil pembahasan tentang penerapan Algoritma *Tabu Search* pada *Graph Coloring* dapat ditarik kesimpulan sebagai berikut:

1. Untuk membuat graf dari data yang telah diperoleh, data tersebut dikelompokkan menjadi 3 bagian, yaitu nomor kereta, kota awal pemberangkatan dan kota tujuan. Setelah data dikelompokkan, data kereta yang memiliki *adjacent* dengan kereta lainnya dihubungkan dengan sebuah sisi dengan melihat kota awal maupun kota tujuan yang sama. Setelah semua node dihubungkan terbentuklah representasi graf dari data kereta yang telah diperoleh.
2. Pada kasus penjadwalan kereta PT KAI diperoleh solusi optimum dengan menggunakan Algoritma *Tabu Search*. Pewarnaan minimal yang didapat menggunakan perwarnaan graf melalui perhitungan manual dan rancang bangun adalah 12 warna. Ke-12 warna tersebut diartikan sebagai 12 waktu pemberangkatan kereta. Berdasarkan hasil perhitungan dihasilkan dua jadwal optimum alternatif menggunakan perhitungan manual dan rancang bangun sebagai berikut (lihat Tabel 4.1):

Tabel 4.1 Solusi Optimum Perhitungan Manual dan Rancang Bangun

Jam	Manual		Rancang Bangun	
	No. Kereta	Tujuan	No. Kereta	Tujuan
5.00 – 6.09	256	Solo-Yogyakarta	283	Solo-Yogyakarta
	275	Yogyakarta-Solo	276	Yogyakarta-Solo
6.24 – 7.33	255	Solo-Yogyakarta	290	Solo-Yogyakarta
	276	Yogyakarta-Solo	280	Yogyakarta-Solo
7.48 – 8.57	277	Yogyakarta-Solo	255	Solo-Yogyakarta
	289	Solo-Yogyakarta	279	Yogyakarta-Solo
9.12 – 10.21	273	Yogyakarta-Solo	256	Solo-Yogyakarta
	288	Solo-Yogyakarta	272	Yogyakarta-Solo
10.36 – 11.45	253	Yogyakarta-Solo	278	Yogyakarta-Solo
	287	Solo-Yogyakarta	282	Solo-Yogyakarta
12.00 – 13.09	280	Yogyakarta-Solo	275	Yogyakarta-Solo
	285	Solo-Yogyakarta	284	Solo-Yogyakarta
13.24 – 14.33	281	Yogyakarta-Solo	274	Yogyakarta-Solo
	272	Solo-Yogyakarta	289	Solo-Yogyakarta
14.48 – 15.57	254	Yogyakarta-Solo	273	Yogyakarta-Solo
	286	Solo-Yogyakarta	285	Solo-Yogyakarta
16.12 – 17.21	278	Yogyakarta-Solo	253	Yogyakarta-Solo
	291	Solo-Yogyakarta	291	Solo-Yogyakarta
17.36 – 18.45	281	Yogyakarta-Solo	286	Solo-Yogyakarta
	290	Solo-Yogyakarta	254	Yogyakarta-Solo
19.00 – 20.09	284	Solo-Yogyakarta	287	Solo-Yogyakarta
	279	Yogyakarta-Solo	277	Yogyakarta-Solo
20.24 – 21.33	274	Yogyakarta-Solo	288	Solo-Yogyakarta
	282	Solo-Yogyakarta	281	Yogyakarta-Solo

4.2 Saran

Berdasarkan penelitian yang telah dilakukan, maka terdapat beberapa saran sebagai berikut:

1. Kemampuan program yang telah dibuat masih terbatas pada data yang sederhana. Sehingga diharapkan oleh peneliti selanjutnya dapat mengembangkan program pada data yang lebih kompleks.
2. Bagi Peneliti selanjutnya dapat membandingkan Algoritma *Tabu Search* dengan Algoritma-Algoritma lain seperti Algoritma Genetik dan Algoritma *Simulated Annealing* untuk menyelesaikan penjadwalan dengan pewarnaan graf. Sehingga

dapat diperoleh kelebihan dan kekurangan Algoritma *Tabu Search* dibandingkan Algoritma-Algoritma tersebut.

3. Implementasi program yang telah dibuat menggunakan bahasa pemrograman PHP juga dapat dikembangkan menggunakan *software* lain misalnya *software* Matlab karena terdapat fitur GUI yang dapat menampilkan graf atau juga dapat menggunakan *software java* karena stuktur penulisan *code* hampir sama dengan bahasa PHP.

DAFTAR PUSTAKA

- Aladag, C.H. and Hocaoglu, G. 2007. *A Tabu Search Algorithm to Solve Course Timetabling Problem*, Hacettepe Journal of Mathematics and Statistics 36(1), 53-64.
- Baker, Kenneth R. 1974. *Introduction to Sequencing and Scheduling*. New York: John Wiley & Sons, Inc. Beasley, J.E.
- Berlianty, I dan Arifin, M. 2010. *Teknik-teknik Optimasi Heuristik*. Yogyakarta: Graha Ilmu.
- Bondy, J.A. & Murty, U.S.R. 1976. *Graph Theory With Applications*. London: The Macmillan Press LTD.
- Diestel, Reinhard. 2005. *Graph Theory*. United States of America: Springer-Verlag New York, Inc.
- Glover, F & Kochenberger, G.A. (Eds). (2003). *Handbook of Metaheuristics*. Dordrecht: Kluwer Academic Publisher.
- Glover, F & Laguna, M. 1997. *Tabu Search*. Massachusetts: Kluwer Academic Publisher.
- Glover, F & Marti, R. 2006. *Metaheuristic Procedures for Training Neural Networks*. Alba and Marti (Eds.), Springer: 53-70.
- Goodairrie, Edgar G. & Parmenter, Michael M. 2002. *Discrete Mathematics with Graph Theory Second Edition*. United States of America: Prentice-Hall, Inc.
- Jensen, Tommy R. and Bjarne Toft. 1995. *Graph Coloring Problem*. John Wiley & Sons, Inc., New York.
- Kadir, Abdul. 2008. *Dasar Pemrograman WEB Dinamis Menggunakan PHP*. Yogyakarta: Penerbit Andi.
- Katuyoso, Martin, Wiwin Sulisty, S.T, M.Kom., Dian W.Chandra, S.Kom. M.Cs. 2012. *Sistem Penjadwalan Kereta Api Pada Jadwal Kereta Api Semarang-Jakarta*. Salatiga: Universitas Kristen Satya Wacana.
- Mahrus, Muhammad. (2011). *Aplikasi Metode Pewarnaan Graf pada Penjadwalan Kegiatan Perkuliahan di Fakultas Sains dan Teknologi UIN Sunan Kalijaga*

Yogyakarta. Skripsi, tidak diterbitkan, Universitas Islam Negeri Sunan Kalijaga: Yogyakarta

- Munir, Rinaldi. 2007. *Matematika Diskrit*. Bandung: Informatika.
- Nurkhasanah. (2013). *Penerapan Pewarnaan Graf Menggunakan Algoritma Welch-Powell pada Radio Frequency Assignment Problem di Provinsi Daerah Istimewa Yogyakarta (DIY)*. Skripsi, tidak diterbitkan, Universitas Islam Negeri Sunan Kalijaga: Yogyakarta
- Prasetyo, Eko. 2008. *Pemrograman Web PHP dan MySql untuk Sistem Informasi Perpustakaan*. Yogyakarta: Graha Ilmu.
- Rosen, Kenneth H. 2012. *Discrete Mathematics and Its Application Seventh Edition*. New York: Mc-Graw-Hill.
- Russel, Stuart dan Peter Norvig. 2010. *Artificial Intelligence: A Modern Approach*, Pearson, Boston.
- Suarga, Dr. 2012. *Algoritma dan Pemrograman*. Yogyakarta: Andi.
- Suryani, Ida, Purwanto, M. Yasin. 2013. *Implementasi Masalah Pewarnaan Graph Dengan Algoritma Tabu Search Pada Penjadwalan Kuliah*. Malang: FMIPA Universitas Negeri Malang
- Suyanto. 2010. *Algoritma Optimasi: Deterministik atau Probabilitik*. Yogyakarta: Graha Ilmu.
- Taha, H. A. 2003. *Operations Research: An Introduction seventh Edition*. Prentice Hall, Inc.
- Wilson, R.J and Watkins, J.J. 1990. *Graphs an Introductory Approach a First Discrete Mathematics*. John Willy And Sons, Inc., Canada.
- Wilson, Robert J. 1990. *Graphs and Their Uses*. The Mathematical Association of America, Washington.

LAMPIRAN 1

Solusi Awal

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	4	8	12	6	7	2	6

ITERASI 1

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik
Warna (13,17)	5	8	2	1	4	12	3	9	6	10	7	11	9	5	3	10	11	1	4	8	12	6	7	2	4
Warna (13,19)	5	8	2	1	4	12	3	9	6	10	7	11	4	5	3	10	11	1	9	8	12	6	7	2	4
Warna (13,20)	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	11	1	9	4	12	6	7	2	4
Warna (13,23)	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	11	1	9	4	12	6	8	2	4
Warna (13,24)	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	9	4	12	6	8	7	4

Tabu List

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	4	8	12	6	7	2	6
Warna	5	8	2	1	4	12	3	9	6	10	7	11	9	5	3	10	11	1	4	8	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	4	5	3	10	11	1	9	8	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	11	1	9	4	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	11	1	9	4	12	6	8	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	9	4	12	6	8	7	4

ITERASI 2

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik
Warna (17,13)	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	2	1	9	4	12	6	8	7	4
Warna (17,19)	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	2	4	12	6	8	7	4
Warna (17,20)	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	4	1	2	9	12	6	8	7	4
Warna (17,23)	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	8	1	2	9	12	6	4	7	4
Warna (17,24)	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	7	1	2	9	12	6	4	8	2

Tabu List

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	4	8	12	6	7	2	6
Warna	5	8	2	1	4	12	3	9	6	10	7	11	9	5	3	10	11	1	4	8	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	4	5	3	10	11	1	9	8	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	11	1	9	4	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	11	1	9	4	12	6	8	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	9	4	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	2	1	9	4	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	2	4	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	4	1	2	9	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	8	1	2	9	12	6	4	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	7	1	2	9	12	6	4	8	2

ITERASI 3

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik
Warna (19,13)	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	7	1	11	9	12	6	4	8	2
Warna (19,17)	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	7	9	12	6	4	8	2
Warna (19,20)	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	9	7	12	6	4	8	2
Warna (19,23)	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	4	7	12	6	9	8	2
Warna (19,24)	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	8	7	12	6	9	4	2

Tabu List

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	4	8	12	6	7	2	6
Warna	5	8	2	1	4	12	3	9	6	10	7	11	9	5	3	10	11	1	4	8	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	4	5	3	10	11	1	9	8	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	11	1	9	4	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	11	1	9	4	12	6	8	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	9	4	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	2	1	9	4	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	2	4	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	8	1	2	9	12	6	4	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	7	1	2	9	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	7	1	11	9	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	7	9	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	9	7	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	4	7	12	6	9	8	2

Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	8	7	12	6	9	4	2
--------------	---	---	---	---	---	----	---	---	---	----	---	----	---	---	---	----	----	---	---	---	----	---	---	---	---

ITERASI 4

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik
Warna (20,13)	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	11	1	8	2	12	6	9	4	2
Warna (20,17)	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	8	11	12	6	9	4	2
Warna (20,19)	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	11	8	12	6	9	4	2
Warna (20,23)	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	11	9	12	6	8	4	2
Warna (20,24)	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	11	4	12	6	8	9	2

Tabu List

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	4	8	12	6	7	2	6
Warna	5	8	2	1	4	12	3	9	6	10	7	11	9	5	3	10	11	1	4	8	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	4	5	3	10	11	1	9	8	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	11	1	9	4	12	6	7	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	11	1	9	4	12	6	8	2	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	9	4	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	2	1	9	4	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	2	4	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	4	1	2	9	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	8	1	2	9	12	6	4	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	7	1	2	9	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	7	1	11	9	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	7	9	12	6	4	8	2

Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	9	7	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	4	7	12	6	9	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	8	7	12	6	9	4	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	11	1	8	2	12	6	9	4	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	8	11	12	6	9	4	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	11	8	12	6	9	4	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	11	9	12	6	8	4	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	11	4	12	6	8	9	2

ITERASI 5

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik	
Warna (23,13)	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	2	1	11	4	12	6	7	9	2	2
Warna (23,17)	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	7	1	11	4	12	6	2	9	2	2
Warna (23,19)	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	7	1	2	4	12	6	11	9	2	2
Warna (23,20)	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	7	1	2	11	12	6	4	9	0	0

Tabu list

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik	
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	4	8	12	6	7	2	6	6
Warna	5	8	2	1	4	12	3	9	6	10	7	11	9	5	3	10	11	1	4	8	12	6	7	2	4	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	4	5	3	10	11	1	9	8	12	6	7	2	4	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	11	1	9	4	12	6	7	2	4	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	11	1	9	4	12	6	8	2	4	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	9	4	12	6	8	7	4	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	2	1	9	4	12	6	8	7	4	4

Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	9	1	2	4	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	4	1	2	9	12	6	8	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	8	1	2	9	12	6	4	7	4
Warna	5	8	2	1	4	12	3	9	6	10	7	11	11	5	3	10	7	1	2	9	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	7	1	11	9	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	7	9	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	9	7	12	6	4	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	4	7	12	6	9	8	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	2	5	3	10	11	1	8	7	12	6	9	4	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	11	1	8	2	12	6	9	4	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	8	11	12	6	9	4	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	11	8	12	6	9	4	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	11	9	12	6	8	4	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	7	5	3	10	2	1	11	4	12	6	8	9	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	2	1	11	4	12	6	7	9	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	7	1	11	4	12	6	2	9	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	7	1	2	4	12	6	11	9	2
Warna	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	7	1	2	11	12	6	4	9	0

Solusi Akhir Tabu Search

	V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₁₄	V ₁₅	V ₁₆	V ₁₇	V ₁₈	V ₁₉	V ₂₀	V ₂₁	V ₂₂	V ₂₃	V ₂₄	Konflik
Warna	5	8	2	1	4	12	3	9	6	10	7	11	8	5	3	10	7	1	2	11	12	6	4	9	0

LAMPIRAN 2

Source Code PHP

```
</br></br>
<?php
    include('info/graph/core.php');
?>
<!doctype html>
<html>
<head>
<title>Friend Wheel</title>
<!-- Copyright 1998-2015 by Northwoods Software Corporation. -->
    <script src="go-debug.js"></script>
    <link href="goSamples.css" rel="stylesheet" type="text/css" />
<!-- you don't need to use this -->
<script id="code">
    function WheelLayout() {
        go.CircularLayout.call(this);
    }
    go.Diagram.inherit(WheelLayout, go.CircularLayout);

    // override makeNetwork to set the diameter of each node and
    ignore the TextBlock label
    WheelLayout.prototype.makeNetwork = function(coll) {
        var net = go.CircularLayout.prototype.makeNetwork.call(this,
coll);
        net.vertexes.each(function(cv) {
            cv.diameter = 50; // because our desiredSize for nodes is
(20, 20)
        });
        return net;
    }

    // override commitNodes to rotate nodes so the text goes away from
the center,
    // and flip text if it would be upside-down
    WheelLayout.prototype.commitNodes = function() {
        go.CircularLayout.prototype.commitNodes.call(this);
        this.network.vertexes.each(function(v) {
            var node = v.node;
            if (node === null) return;
            // get the angle of the node towards the center, and rotate it
accordingly
            var a = v.actualAngle;
            if (a > 90 && a < 270) { // make sure the text isn't upside
down
                var textBlock = node.findObject("TEXTBLOCK");
                textBlock.angle = 180;
            }
            node.angle = a;
        });
    }
};
</script>
</head>
</html>
</?php>
```

```

    });
};
// end WheelLayout class

var highlightColor = "red"; // color parameterization

function init() {
    if (window.goSamples) goSamples(); // init for these samples --
    you don't need to call this
    var $ = go.GraphObject.make; // for conciseness in defining
    templates

    myDiagram =
    div
    $(go.Diagram, "myDiagram", // must be the ID or reference to
    {
        autoScale: go.Diagram.Uniform,
        padding: 10,
        contentAlignment: go.Spot.Center,
        layout:
        $(WheelLayout, // set up a custom CircularLayout
        // set some properties appropriate for this sample
        {
            arrangement: go.CircularLayout.ConstantDistance,
            nodeDiameterFormula: go.CircularLayout.Circular,
            spacing: 10,
            aspectRatio: 0.7,
            sorting: go.CircularLayout.Optimized
        },
        isReadOnly: true,
        click: function(e) { // background click clears any
        remaining highlighteds
            e.diagram.startTransaction("clear");
            e.diagram.clearHighlighted();
            e.diagram.commitTransaction("clear");
        }
    });

    // define the Node template
    myDiagram.nodeTemplate =
    $(go.Node, "Horizontal",
    {
        selectionAdorned: false,
        locationSpot: go.Spot.Center, // Node.location is the
        center of the Shape
        locationObjectName: "SHAPE",
        mouseEnter: function(e, node) {
            node.diagram.clearHighlighted();
            node.linksConnected.each(function(l) { highlightLink(l,
            true); });
            node.isHighlighted = true;

```



```

        var tb = node.findObject("TEXTBLOCK");
        if (tb !== null) tb.stroke = highlightColor;
    },
    mouseLeave: function(e, node) {
        node.diagram.clearHighlighted();
        var tb = node.findObject("TEXTBLOCK");
        if (tb !== null) tb.stroke = "black";
    }
},
new go.Binding("text", "text"), // for sorting the nodes
$(go.Shape, "Ellipse",
    {
        name: "SHAPE",
        fill: "lightgray", // default value, but also data-
bound
        stroke: "transparent", // modified by highlighting
        strokeWidth: 2,
        desiredSize: new go.Size(20, 20),
        portId: ""
    }, // so links will go to the shape, not the whole node
    new go.Binding("fill", "color"),
    new go.Binding("stroke", "isHighlighted",
        function(h) { return h ? highlightColor :
"transparent"; })
        .ofObject()),
    $(go.TextBlock,
        { name: "TEXTBLOCK" }, // for search
        new go.Binding("text", "text")
    );

function highlightLink(link, show) {
    link.isHighlighted = show;
    link.fromNode.isHighlighted = show;
    link.toNode.isHighlighted = show;
}

// define the Link template
myDiagram.linkTemplate =
    $(go.Link,
        {
            routing: go.Link.Normal,
            curve: go.Link.Bezier,
            selectionAdorned: false,
            mouseEnter: function(e, link) { highlightLink(link, true);
},
            mouseLeave: function(e, link) { highlightLink(link,
false); }
        },
        $(go.Shape,
            new go.Binding("stroke", "isHighlighted",
                function(h, shape) { return h ?
highlightColor : shape.part.data.color; })

```

```

        .ofObject(),
        new go.Binding("strokeWidth", "isHighlighted",
            function(h) { return h ? 2 : 1; })
        .ofObject()
    );

    generateGraph();
}

function generateGraph() {
    var names = <?php echo json_encode($k);?>;
    var warna = <?php echo json_encode($color);?>;

    var nodeDataArray = [];
    for (var i = 0; i < names.length; i++) {
        nodeDataArray.push({ key: i, text: names[i], color: '#008B8B'
    });
    }
    var jArray= <?php echo json_encode($neigh); ?>;
    var linkDataArray = [];
    num = names.length;
    var items = [[]];var f = 0;
    for(var i=0;i<num;i++){
        for(var j=0;j<num;j++){
            items[i,j] = jArray[f];
            f++;
            if(items[i,j] == 1){
                linkDataArray.push({ from: i, to: j, color:
'#008B8B' });
            }
        }
    }
    myDiagram.model = new go.GraphLinksModel(nodeDataArray,
linkDataArray);
}
</script>
</head>
<body onload="init()">
<div id="sample">
    <div id="myDiagram" style="border: solid 1px black; background:
white; width: 90%; height: 600px" ></div>
</div>

</body>
</html>

```