

**PENYELESAIAN *TRAVELLING SALESMAN PROBLEM*
DENGAN ALGORITMA *ARTIFICIAL BEE COLONY*
(STUDI KASUS : PENDISTRIBUSIAN HEWAN QURBAN PPHQ AMM)**

Skripsi

Untuk memenuhi sebagai persyaratan
Mencapai derajat sarjana S-1



disusun oleh :

Rinaldi Perdana Putra

11610034

**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA
YOGYAKARTA
2015**

PENGESAHAN SKRIPSI/TUGAS AKHIR

Nomor : UIN.02/D.ST/PP.01.1/035/2016

Skripsi/Tugas Akhir dengan judul

: Penyelesaian *Travelling Salesman Problem* dengan *Algoritma Artificial Bee Colony* (Studi Kasus : Pendistribusian Hewan Qurban PPHQ AMM)

Yang dipersiapkan dan disusun oleh

Nama : Rinaldi Perdana Putra

NIM : 11610034

Telah dimunaqasyahkan pada

: 28 Desember 2015

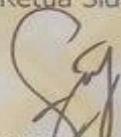
Nilai Munaqasyah

: A -

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

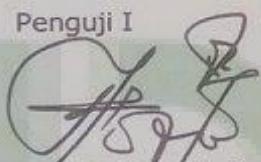
TIM MUNAQASYAH :

Ketua Sidang



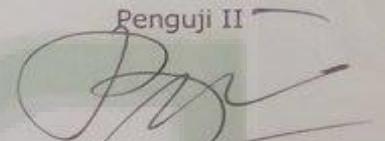
Noor Saif Mu. Mussafi, M.Sc
NIP. 19820617 200912 1 005

Penguji I



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

Penguji II



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

Yogyakarta, 5 Januari 2016
UIN Sunan Kalijaga

Fakultas Sains dan Teknologi



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 : Rinaldi Perdana Putra

NIM : 11610034

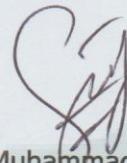
Judul Skripsi : Penyelesaian *Travelling Salesman Problem* dengan Algoritma *Artificial Bee Colony* (Studi Kasus: Pendistribusian Qurban PPHQ AMM)

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 mengharap agar skripsi/tugas akhir Saudara tersebut diatas dapat segera dimunaqisyahkan. Atas perhatiannya kami ucapan 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 : Rinaldi Perdana Putra

NIM : 11610034

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



Rinaldi Perdana P
NIM.11610034

HALAMAN PERSEMBAHAN

Alhamdulillah puji syukur kepada Allah SWT atas segala rahmat dan karunia-Nya yang telah memberikan kekuatan, keikhlasan serta kemauan sehingga

penulis dapat menyelesaikan tugas akhir ini

**Terima kasih atas segala dukungan motivasi, serta semangat yang tiada henti
dari orang tuaku tercinta**

Bapak Sudarminto, dan Ibu Indarni

Serta seluruh keluargaku yang selalu penulis sayangi

Teman teman PAL

**Lukman, Sulis, Syauqi, Fuad, Dayat, Wachid, Eruit, Taufan, Juni,
Ridwan, Dwi, Fuji dan Dina**

Almamater kebanggaanku

Program Studi Matematika

Fakultas Sains dan Teknologi

UIN Sunan Kalijaga Yogyakarta

MOTTO

**“Bermimpilah yang tinggi. Tapi jangan berusaha menggapai mimpi tersebut,
melainkan berusahalah melampauinya”**

(Anies Baswedan)

KATA PENGANTAR

Puji syukur kehadirat Allah SWT yang telah melimpahkan segala rahmat dan hidayah-Nya, sehingga skripsi yang berjudul “Penyelesaian *Travelling Salesman Problem* dengan Algoritma *Artificial Bee Colony* (Studi Kasus: Pendistribusian Hewan Qurban PPHQ AMM)” dapat terselesaikan guna memenuhi syarat memperoleh gelar kesarjanaan di Jurusan Matematika Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta.

Shalawat serta salam senantiasa tercurahkan kepada Nabi besar Muhammad SAW, yang senantiasa kita tunggu syafaatnya di hari akhir nanti. 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. Ayahanda terkasih Sudarminto dan Ibunda tersayang Indarni yang senantiasa memberikan kasih sayang, motivasi, doa, semangat dan segala pengorbanan untuk memperjuangkan penulis. Karya ini khusus penulis tujukan untuk Ayah dan Ibu tercinta.
6. Kakak-kakakku dan keluargaku yang memberikan nasehat dan dorongan semangat untuk penulis.
7. Sahabat PAL (Lukman, Sulis, Syauqi, Fuad, Bang Dayat, Wachid, Eruit, Taufan, Ridwan, Juni, Dwi (Uthe), Fuji (Fufu), dan Dina) terima kasih atas canda dan tawa yang menghiasi hari-hari indah penulis. Kenangan bersama kalian tidak akan penulis lupakan.
8. Kepada teman-teman matematika 2011 yang selalu memberikan dukungan dan motivasi hingga terselesaiannya skripsi.
9. Teman teman PPHQ AMM terima kasih atas segala bantuan 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, Desember 2015

Penulis

Rinaldi Perdana Putra
NIM. 11610034

DAFTAR ISI

HALAMAN JUDUL.....	i
PENGESAHAN SKRIPSI.....	ii
PERSETUJUAN SKRIPSI.....	iii
PERNYATAAN KEASLIAN SKRIPSI.....	iv
HALAMAN PERSEMPAHAN.....	v
HALAMAN MOTTO.....	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiii
DAFTAR LAMPIRAN.....	xiv
ABSTRAK.....	xv
BAB I PENDAHULUAN	
A. Latar Belakang.....	1
B. Rumusan Masalah.....	3
C. Tujuan Penelitian.....	3
D. Batasan Masalah.....	3
E. Manfaat Penelitian.....	4
F. Tinjauan Pustaka.....	4
G. Metode Penelitian.....	6
H. Sistematika Penulisan.....	8
BAB II DASAR TEORI	
A. Teori Graf.....	9
1. Definisi Graf.....	10
2. Jenis-jenis Graf.....	11
3. Keterhubungan.....	13
4. Graf Berbobot (<i>Weighted Graph</i>).....	17
5. Graf Berarah Berbobot (<i>Weighted Directed Graph</i>).....	18
6. Graf Hamilton.....	18
B. <i>Travelling Salesman Problem</i> (TSP).....	19
C. Algoritma <i>Artificial Bee Colony</i>	21
D. <i>Artificial Bee Colony</i> untuk Menyelesaikan <i>Travelling Salesman Problem</i>	22
E. Matlab.....	26
1. GUI (<i>Graphical User Interface</i>) pada Matlab.....	27
2. Operator Relasi dan Logika.....	29
3. Statement Control pada Matlab.....	30
BAB III PEMBAHASAN	
A. Konsep dan Langkah Algoritma <i>Artificial Bee Colony</i>	35
1. Konsep Algoritma <i>Artificial Bee Colony</i>	35

2. Langkah Algoritma <i>Artificial Bee Colony</i>	35
a. Pembentukan inisial solusi (Solusi awal).....	36
b. Fase <i>employed bee</i>	37
c. Fase <i>onlooker bee</i>	37
d. Fase <i>scout bee</i> dengan metode <i>2-Opt</i>	37
e. Kriteria pemberhentian.....	38
B. Penerapan Algoritma <i>Artificial Bee Colony</i> pada TSP.....	39
1. Penerapan Algoritma <i>Artificial Bee Colony</i> Secara Manual.....	41
a. Langkah 1.....	42
b. Langkah 2.....	43
c. Langkah 3.....	45
d. Langkah 4.....	47
e. Langkah 5.....	49
f. Langkah 6.....	51
g. Langkah 7.....	52
2. Rancang Bangun Algoritma ABC dalam Menyelesaikan TSP.....	53
BAB IV PENUTUP	
A. Kesimpulan.....	70
B. Saran.....	72
DAFTAR PUSTAKA	73
LAMPIRAN-LAMPIRAN	75

DAFTAR TABEL

Tabel 1.1 Perbedaan Penelitian.....	5
Tabel 2.1 Operator relasi.....	30
Tabel 2.2 Operator logika.....	30
Tabel 3.1 Daftar depot dan pelanggan.....	39
Tabel 3.2 Jarak depot ke pelanggan dan antar pelanggan dalam satuan kilometer.....	41
Tabel 3.3 Solusi Optimal TSP pada Kasus Pendistribusian Qurban.....	52
Tabel 3.4 Spesifikasi perangkat keras (<i>Hardware</i>).....	53
Tabel 3.5 Spesifikasi perangkat lunak (<i>Software</i>).....	53
Tabel 3.6 String <i>property static text5</i>	54
Tabel 3.7 Solusi optimal pada pengujian program.....	69
Tabel 4.1 Solusi optimum perhitungan manual dan rancangan bangun....	71

DAFTAR GAMBAR

Gambar 1.1 Skema langkah penelitian.....	7
Gambar 2.1 Jembatan <i>Könisberg</i>	9
Gambar 2.2 Model graf jembatan <i>Könisberg</i>	10
Gambar 2.3 Contoh graf.....	10
Gambar 2.4 Graf nol.....	11
Gambar 2.5 Graf lengkap.....	12
Gambar 2.6 Graf ganda (<i>Multigraph</i>).....	12
Gambar 2.7 Graf tak berarah (<i>Undirected Graph</i>).....	12
Gambar 2.8 Graf berarah (<i>Digraph</i>).....	13
Gambar 2.9 Graf G_1 terhubung dan G_2 tidak terhubung.....	14
Gambar 2.10 Graf H.....	14
Gambar 2.11 Contoh panjang (<i>length</i>).....	16
Gambar 2.12 Graf sederhana H_2	17
Gambar 2.13 Graf berbobot.....	18
Gambar 2.14 Graf berarah berbobot.....	18
Gambar 2.15 Contoh TSP.....	20
Gambar 2.16 Model Algoritma <i>Artificial Bee Colony</i>	21
Gambar 2.17 Metode <i>2-Opt</i> dalam TSP.....	25
Gambar 2.18 GUI MATLAB.....	27
Gambar 2.19 Tampilan <i>fig-file</i>	28
Gambar 2.20 Tampilan <i>m.file</i> dan GUI.....	29
Gambar 3.1 Langkah Algoritma <i>Artificial Bee Colony</i>	35
Gambar 3.2 Flow chart langkah-langkah Algoritma <i>Artificial Bee Colony</i>	38
Gambar 3.3 Peta Pendistribusian Hewan Qurban pada Wilayah 5.....	40
Gambar 3.4 GUI_ABC.fig.....	56
Gambar 3.5 Pembuatan <i>background</i> program.....	57
Gambar 3.6 Tampilan menu help.....	64
Gambar 3.7 Tampilan awal program.....	65
Gambar 3.8 Tampilan input matriks jarak.....	66
Gambar 3.9 <i>Input</i> matriks jarak.....	67
Gambar 3.10 Form <i>input</i> data.....	67
Gambar 3.11 Tampilan hasil proses perhitungan.....	68

DAFTAR LAMPIRAN

Lampiran 1. Iterasi.....	75
Iterasi 1.....	75
Iterasi 2.....	76
Iterasi 3.....	77
Iterasi 4.....	78
Iterasi 5.....	79
Iterasi 6.....	80
Iterasi 7.....	81
Iterasi 8.....	82
Iterasi 9.....	83
Iterasi 10.....	84
Lampiran 2. Source Kode M.Files.....	85

ABSTRAK

PENYELESAIAN *TRAVELLING SALESMAN PROBLEM* DENGAN ALGORITMA *ARTIFICIAL BEE COLONY* (STUDI KASUS : PENDISTRIBUSIAN QURBAN PPHQ AMM)

Oleh:

Rinaldi Perdana Putra

NIM.11610034

Matematika merupakan ilmu yang luas dan banyak berkaitan dengan kehidupan, penerapan ilmu tersebut salah satunya pada masalah pecarian rute terpendek. Sebuah perusahaan akan mengoptimalkan pendistribusian suatu produk untuk menekan biaya operasional. Dengan mencari rute pendistribusian minimum, perusahaan dapat menekan waktu dan biaya operasional yang harus dikeluarkan.

Permasalahan pencarian rute minimum dapat direpresentasikan menggunakan graf berarah dan memiliki bobot dan disebut dengan TSP (*Travelling Salesman Problem*). Depot dan pelanggan dinyatakan sebagai simpul, sedangkan jalan dinyatakan sebagai sisi. Kasus TSP tersebut dapat diselesaikan dengan menggunakan Algoritma *Artificial Bee Colony* (ABC). Cara kerja algoritma ini dimulai dengan menentukan solusi awal yaitu membuat rute pendistribusian secara acak sebanyak n kali lalu dihitung masing-masing jaraknya. Selanjutnya dilakukan inisialisasi solusi kemudian solusi tersebut diperbaiki dengan mencari solusi tetangga. Langkah berikutnya adalah menghitung nilai *fitness* masing-masing solusi yang akan digunakan untuk menghitung nilai probabilitas. Langkah terakhir yaitu perbaikan solusi dengan menggunakan metode *2-Opt*, solusi yang diperbaiki adalah solusi yang tidak mengalami peningkatan setelah dilakukan proses perhitungan. Selanjutnya proses perhitungan berulang dari langkah pertama sampai maksimum iterasi.

Berdasarkan proses perhitungan diperoleh solusi alternatif dengan dua metode. Solusi pertama diperoleh dengan menggunakan perhitungan manual dengan jarak sebesar 11,8 km dan solusi kedua menggunakan program diperoleh jarak optimal sebesar 11,8 km.

Kata kunci: *Travelling Salesman Problem* (TSP), Algoritma *Artificial Bee Colony*

BAB I

PENDAHULUAN

A. Latar Belakang

Matematika merupakan ilmu yang luas dan banyak berkaitan dengan kehidupan, penerapan ilmu tersebut salah satunya pada masalah pencarian rute terpendek. Dalam matematika permasalahan pencarian rute terpendek dijelaskan dalam teori graf. Secara umum teori graf adalah cabang ilmu matematika yang membahas tentang graf, dimana komponen utama dari graf adalah *vertex* dan *edges* yang merupakan simpul dan sisi. Simpul dalam masalah ini merupakan kota tujuan sedangkan sisi merupakan jalan. Permasalahan pencarian rute terpendek dibahas dalam masalah *Travelling Salesman Problem* (TSP). Permasalahan utama pada TSP adalah seorang sales harus mengantarkan barang melewati semua kota dan hanya singgah satu kali selama perjalanan dengan rute paling minimum (David L. Applegate, Robert E. Bixby, Vasek Chvátal & William J. Cook, 2006). Tujuan dari permasalahan TSP adalah meminimumkan waktu tempuh untuk meminimumkan biaya transportasi. TSP merupakan permasalahan *Non deterministic Polynomial-Hard (NP-Hard)* karena penyelesaian secara eksak sangat sulit dilakukan. Oleh karena itu para ahli merumuskan metode pendekatan untuk menyelesaikan masalah tersebut seperti pendekatan *Metaheuristic* antara lain Algoritma *Ant Colony*, *Genetic Algorithm*, *Tabu search*, dan *Artificial Bee colony*.

Algoritma *Artificial Bee colony* dirancang oleh Karaboga pada tahun 2005 (Karaboga, 2005) berdasarkan perilaku kecerdasan lebah dalam berkoloni untuk mencari sumber makanan. Algoritma ini memiliki 3 komponen utama dalam

mencari sumber makanan terbaik, yaitu *employed bee*, *onlooker bee*, dan *scout bee*. *Employed bee* bertugas mencari sumber makanan berdasarkan kecerdasannya dan mengevaluasi jumlah nektarnya, jumlah *employed bee* sama dengan jumlah sumber makanan. Setelah *employed bee* menemukan sumber makanan terbaik mereka kembali ke sarang dan memberikan informasi kepada *onlooker bee* yang telah menunggu untuk mengeksplorasi sumber makanan tersebut. Pada permasalahan TSP yang menjadi sumber makanan adalah kemungkinan solusi yang dibangkitkan secara acak lalu masuk ke dalam tahap algoritma, berulang sampai ditemukan solusi terbaik.

Pusat Pengadaan Hewan Qurban (PPHQ) mempunyai permasalahan mencari rute minimum untuk mengantarkan hewan qurban satu muatan penuh ke semua pelanggan tepat satu kali dan kembali ke tempat asal. Penyelesaian secara manual akan menghabiskan banyak waktu untuk menentukan arah tujuan terdekat, untuk itu Algoritma *Artificial Bee Colony* dapat digunakan untuk mencari solusi optimal dalam menyelesaikan permasalahan tersebut. Solusi optimal dalam permasalahan ini yaitu menemukan rute minimum. Pembuatan suatu program yang dapat mempercepat proses pencarian solusi optimal pada TSP. Oleh karena itu, program dengan Algoritma *Artificial Bee Colony* diharapkan dapat memudahkan pencarian solusi optimal yang lebih efektif dan efisien sehingga dirumuskan judul penelitian yaitu “**Penyelesaian Travelling Salesman Problem dengan Algoritma Artificial Bee Colony (pada studi kasus: Masalah Pendistribusian Hewan Qurban PPHQ AMM)**”.

B. Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, maka dirumuskan permasalahan sebagai berikut:

1. Bagaimana pencarian rute terpendek dalam permasalahan TSP pada studi kasus pendistribusian qurban PPHQ AMM menggunakan Algoritma *Artificial Bee Colony*?
2. Bagaimana rancang bangun (program) *Artificial Bee Colony* dalam menyelesaikan TSP berbasis Matlab 8.1 (R2013a)?

C. Tujuan Penelitian

Tujuan penelitian ini adalah :

1. Menjelaskan pencarian rute terpendek dalam permasalahan TSP pada studi kasus pendistribusian qurban PPHQ AMM dengan Algoritma *Artificial Bee Colony*.
2. Membuat program yang dapat mengaplikasikan Algoritma *Artificial Bee Colony* dalam menyelesaikan TSP berbasis Matlab 8.1 (R2013a).

D. Batasan Masalah

Pembatasan masalah pada penelitian ini yaitu:

1. Graf yang digunakan dalam permasalahan ini adalah graf berbobot yang berarah (*weighted directed graph*) dan graf Hamilton.
2. Permasalahan ini tidak mempertimbangkan waktu dan kondisi jalan (kemacetan).
3. Permasalahan ini hanya mencakup satu wilayah distribusi dan satu hari waktu pendistribusian.

4. Hewan yang didistribusikan hanya kambing karena PPHQ AMM tidak mendistribusikan sapi.
5. *Input* yang diperlukan berupa jumlah node, jarak antar node, jumlah solusi, limit, dan maksimal iterasi.
6. Bahasa pemrograman menggunakan Matlab 8.1 (R2013a).

E. Manfaat Penelitian

Hasil penelitian ini diharapkan dapat bermanfaat sebagai berikut :

1. Bagi Mahasiswa

Memberikan informasi bagaimana cara menyelesaikan TSP dengan Algoritma *Artificial Bee Colony* serta penerapannya dengan program.

2. Bagi Umum

Memberikan rekomendasi kepada lembaga masyarakat tentang pencarian rute minimum TSP pada masalah pendistribusian qurban menggunakan Algoritma *Artificial Bee Colony* dengan disertai program.

F. Tinjauan Pustaka

Penelitian ini menggunakan beberapa literatur baik berasal dari jurnal penelitian, skripsi, maupun referensi lainnya.

Beberapa sumber yang digunakan sebagai acuan pada penelitian ini diantaranya Jurnal yang berjudul “Penyelesaian Travelling Salesman Problem dengan Algoritma Heuristik” yang ditulis oleh Filman Ferdian pada tahun 2009. Jurnal tersebut menjelaskan penyelesaian TSP dengan metode Heuristik. Metode penyelesaian yang digunakan adalah *minimum spanning tree* dengan Algoritma Prim dan Kruskal. Sedangkan pada skripsi ini digunakan metode metaheuristik

karena solusi yang diperoleh lebih mendekati optimal. Hal ini karena metode metaheuristik melakukan pencarian yang lebih teliti dengan iterasi.

Referensi selanjutnya yaitu jurnal yang berjudul “Penyelesaian *Travelling Salesman Problem* (TSP) dengan Menggunakan *Artificial Bee Colony*” yang ditulis oleh Rendra Firman Pratama pada tahun 2013. Jurnal ini menjelaskan tentang Algoritma *Artificial Bee Colony* untuk menyelesaikan masalah *Travelling Salesman Problem*. Secara garis besar penelitian ini menggunakan langkah kerja dalam algoritma dengan metode *Greedy Selection*. Sedangkan skripsi ini menggunakan metode heuristik *2-Opt* sehingga proses pencarian menjadi lebih cepat.

Referensi ketiga yaitu jurnal yang berjudul “*Artificial Bee Colony* untuk menyelesaikan *Travelling Salesman Problem*” yang ditulis oleh Faisal Amri pada tahun 2012. Inti dari jurnal tersebut adalah penyelesaian TSP menggunakan inisial solusi *Random Method* dan metode *improvement solution ABC Algorithm* dan *neighbor operator (swap operator, swap sequence, insert operator, and insert sequence)*. Sedangkan skripsi ini menggunakan perluasan solusi dengan metode heuristik *2-Opt* sehingga didapat solusi yang lebih optimal.

Perbedaan penelitian ini dapat disajikan dalam tabel sebagai berikut.

Tabel 1.1 Perbedaan Penelitian

No	Nama	Judul	Perbedaan
1	Filman Ferdian	Penyelesaian Travelling Salesman Problem dengan Algoritma Heuristik.	Dalam menyelesaikan TSP penelitian tersebut menggunakan Algoritma Heuristik dan dengan metode <i>Minimum Spanning Tree</i> sedangkan penelitian ini menggunakan metode Metaheuristik dengan Algoritma

			<i>Artificial Bee Colony</i> (program menggunakan Borland Delphi).
2	Rendra Firman Pratama	Penyelesaian <i>Travelling Salesman Problem</i> (TSP) dengan Menggunakan <i>Artificial Bee Colony</i> .	Penelitian tersebut menggunakan Algoritma <i>Artificial Bee Colony</i> dengan langkah kerja yang sama tanpa adanya metode tambahan pada perluasan solusi sedangkan skripsi ini menggunakan Algoritma <i>Artificial Bee Colony</i> dengan <i>Improvement Solution</i> yaitu metode <i>2-Opt</i> sehingga perluasan solusi lebih optimal.
3	Faisal Amri	<i>Artificial Bee Colony</i> untuk menyelesaikan <i>Travelling Salesman Problem</i> .	Langkah kerja Algoritma <i>Artificial Bee Colony</i> menggunakan metode perluasan <i>neighbor operator</i> (<i>swap operator</i> , <i>swap sequence</i> , <i>insert operator</i> , dan <i>insert sequence</i>) tetapi menghilangkan langkah umum algoritma. Sedangkan skripsi ini menggunakan perluasan solusi tanpa menghilangkan langkah umum algoritma.

Skripsi yang berjudul “Penyelesaian *Travelling Salesman Problem* pada Masalah Pendistribusian Qurban PPHQ AMM dengan Algoritma *Artificial Bee Colony*” ini, menggunakan tiga judul penelitian di atas sebagai rujukan. Penelitian ini akan menyelesaikan permasalahan TSP pada studi kasus pendistribusian qurban PPHQ AMM dengan menggunakan Algoritma *Artificial Bee Colony* dengan metode perluasan *2-Opt* dan program menggunakan Matlab 8.1 (R2013a). Sehingga diharapkan penelitian ini dapat memberikan hasil yang lebih optimal.

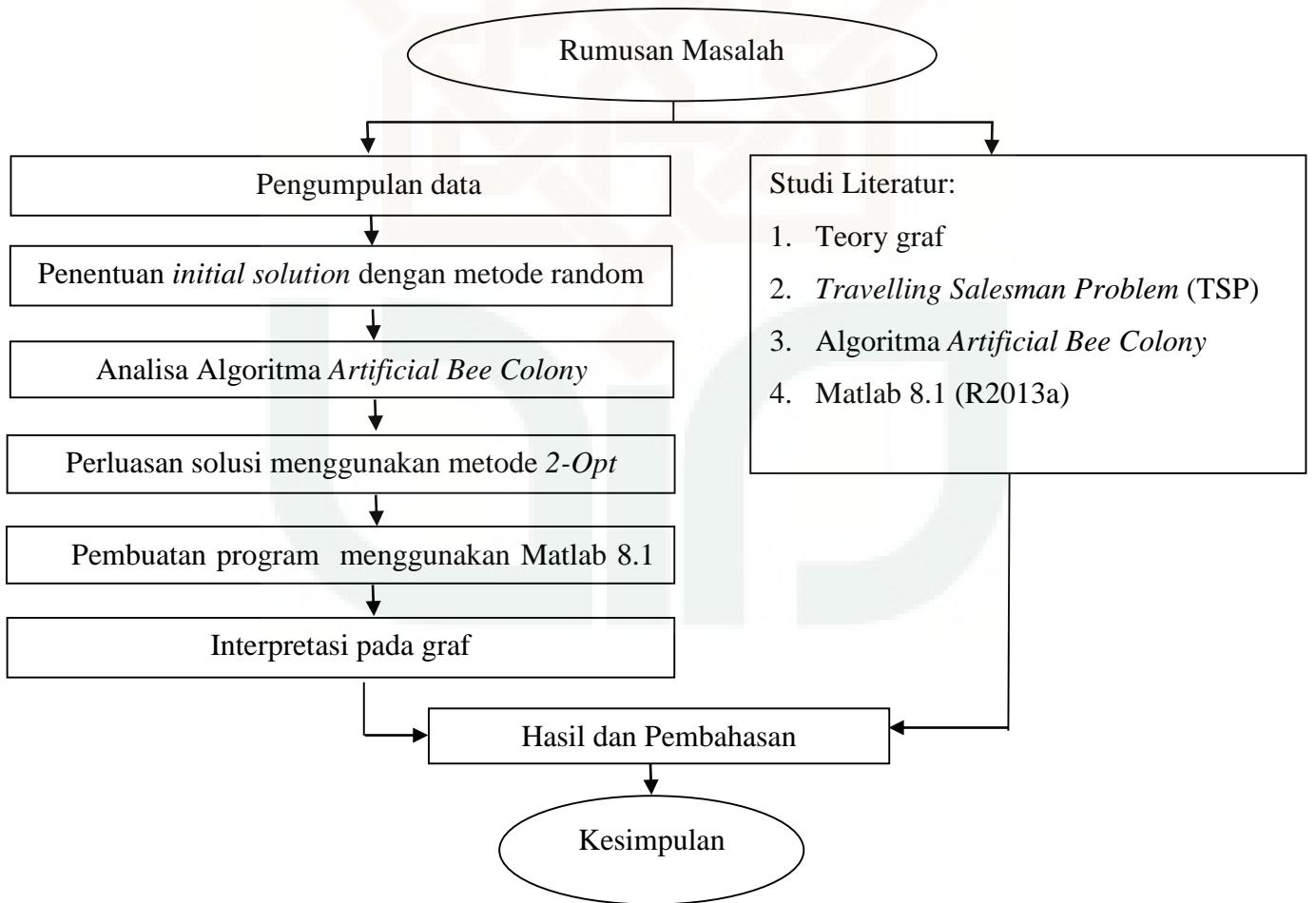
G. Metode Penelitian

Penelitian ini termasuk jenis penelitian deskriptif kualitatif, yakni penelitian yang cenderung mencari fakta yang terjadi saat ini maupun sudah lampau. Proses penelitian ini tidak dapat dimanipulasi atau dilakukan perubahan variabel bebas,

tetapi menunjukkan suatu kondisi yang ada. Tujuan dari metode penelitian ini ialah pemahaman secara lebih mendalam terhadap suatu permasalahan yang akan diteliti.

Objek yang digunakan adalah *Travelling Salesman Problem* dengan menggunakan Algoritma *Artificial Bee Colony*. Data yang digunakan adalah data pada pendistribusian qurban PPHQ AMM. Program Matlab 8.1 (R2013a) diharapkan dapat memudahkan pencarian solusi optimal, dimana solusi yang dicari dalam masalah ini adalah rute terpendek.

Berikut skema langkah penelitian yang meliputi pengumpulan data, studi literatur, hasil solusi, dan pembahasan serta kesimpulan yaitu sebagai berikut :



Gambar 1.1 Skema langkah penelitian

H. Sistematika Penulisan

Penulisan skripsi ini dibagi menjadi empat bab dengan sistematika sebagai berikut:

BAB I PENDAHULUAN

Bab ini membahas mengenai Latar Belakang, Rumusan Masalah, Tujuan Penelitian, Batasan Masalah, Manfaat Penelitian, Tinjauan Pustaka, Metode Penelitian , dan Sistematika Penulisan.

BAB II LANDASAN TEORI

Bab ini berisi teori dan konsep-konsep tentang Teori Graf, TSP, dan Algoritma *Artificial Bee Colony* yang menjadi dasar pembahasan.

BAB III PEMBAHASAN

Bab ini membahas mengenai konsep dan langkah Algoritma *Artificial Bee Colony* dalam menyelesaikan TSP untuk mencari solusi permasalahan pendistribusian qurban serta pengolahan solusi menggunakan program Matlab 8.1 (R2013a).

BAB IV PENUTUP

Bab ini berisi kesimpulan dan saran dari pokok bab–bab sebelumnya.

BAB IV

PENUTUP

A. Kesimpulan

Berdasarkan hasil pembahasan tentang penerapan Algoritma *Artificial Bee Colony* pada *Travelling Salesman Problem* dapat diambil kesimpulan sebagai berikut:

1. Proses perhitungan Algoritma *Artificial Bee Colony* terdiri dari 7 langkah. Pertama adalah pembentukan solusi awal atau rute pendistribusian sejumlah 8 solusi sebagai iterasi 0 dan dihitung jarak masing-masing rute. Setelah dihitung masing-masing jarak langkah kedua yaitu menentukan batas atas dan batas bawah solusi pencarian yang merupakan jarak tertinggi dan jarak terendah dari rute pendistribusian. Tujuan ditentukan batas atas dan batas bawah solusi adalah untuk melakukan proses inisialisasi. Langkah ketiga yaitu perbaikan solusi dengan mencari solusi tetangga, jika solusi yang baru lebih baik maka menggantikan solusi yang lama. Langkah keempat dihitung nilai *fitness* masing-masing solusi, nilai *fitness* yang diperoleh selanjutnya digunakan untuk menghitung nilai probabilitas. Langkah kelima setelah diperoleh nilai probabilitas selanjutnya dihitung nilai kumulatif dari masing masing solusi. Nilai kumulatif yang diperoleh kemudian dibandingkan dengan bilangan acak antara 0 sampai dengan 1, jika nilai kumulatif lebih besar maka solusi dianggap tidak mengalami peningkatan. Langkah keenam yaitu melakukan perluasan

- solusi yang tidak mengalami peningkatan dengan metode *2-Opt*. Solusi baru yang diperoleh dibandingkan dengan solusi lama, jika solusi baru lebih baik maka menggantikan solusi lama. Langkah terakhir yaitu apabila kriteria pemberhentian dipenuhi maka proses perhitungan Algoritma *Artificial Bee Colony* berhenti dan diperoleh solusi optimum, jika tidak dipenuhi maka proses kembali berulang dimulai pada langkah kedua.
2. Pada kasus pendistribusian hewan qurban PPHQ AMM diperoleh beberapa solusi dengan menggunakan Algoritma *Artificial Bee Colony*. Jarak optimum diperoleh dengan melakukan perhitungan manual 11,8 km. Sedangkan perhitungan dengan menggunakan program didapat beberapa rute penyelesaian dengan jarak optimum sebesar 11,8 km. Berdasarkan hasil perhitungan dihasilkan dua solusi rute perjalanan optimum alternatif menggunakan perhitungan manual dan program sebagai berikut (lihat Tabel 4.1):

Tabel 4.1 Solusi optimum perhitungan manual dan program

Metode	Rute	Jarak
Manual	1-6-5-14-13-3-2-4-16-7-12-8-10-9-11-15-1	11,8 km
Program	1-6-5-14-13-3-2-4-16-12-7-8-10-9-11-15-1	11,8 km

B. Saran

1. Program kurang efisien dalam melakukan *input* jarak karena harus menggunakan matriks jarak. Untuk itu bagi peneliti selanjutnya dapat mengembangkan program agar lebih efisien dalam melakukan *input* jarak.
2. Bagi peneliti selanjutnya dapat menggunakan Algoritma *Artificial Bee Colony* untuk menyelesaikan permasalahan *Vehicle Routing Problem*, pewarnaan graf dan lain-lain.

DAFTAR PUSTAKA

- Alkallak, I. S & Sha'ban, R. Z. (2008). *Tabu Search Method For Solving The Traveling Salesman Problem*. *Raf. J. of Comp. & Math's.* 5:141-153
- Amin, Aulia A dkk. (2003). *Travelling Salesman Problem*. Bandung : Institut Teknologi Bandung.
- Amri, F dkk. (2012). *Artificial Bee Colony untuk Menyelesaikan Travelling Salesman Problem*. Sumatra Utara : Universitas Sumatera Utara.
- Bhuwana, Anak Agung. (2013). *Optimasi Pendistribusian Barang berdasarkan Rute dan Daya Tampung Kendaraan dengan Algoritma Artificial Bee Colony*. Bali: Universitas Udayana.
- Bolaji, Asaju La'aro dkk. (2013). *Artificial Bee Colony Algorithm, It's Variants and Applications : A Survey*. Malaysia : Universiti Sains Malaysia.
- Diestel, Reinhard. (2005). *Graph Theory*. Germany: Springer.
- Ferdian, F. (2009). *Penyelesaian Travelling Salesman Problem dengan Algoritma Heuristik*. Bandung : Institut Teknologi Bandung.
- Goddairrie, Edgar G. &Parmenter, Michael M. (2002). *Discrete Mathematics with Graph Theory Second Edition*. United States of America: Prentice-Hall, Inc.
- Irsalinda, Nursyiva. (2013). *Penyelesaian Permasalahan Optimasi Global Menggunakan Algoritma Koloni Lebah Buatan*. Yogyakarta: Universitas Ahmad Dahlan.
- Karaboga, D & Akay, B. (2009). *Applied Mathematics and Computation : A Comparative Study of Artificial Bee Colony algorithm*. Turkey : Erciyes University.
- Karaboga, D & Basturk, B. (2012). *On the Performance Artificial Bee Colony (ABC) Algorithm*. Applied and Soft Computing.
- Kusumadewi, Sri. (2003). *Artificial Intelegence: Teknik dan aplikasinya*. Yogyakarta: Graha Ilmu.
- Munir, Rinaldi. (2005). *Matematika Diskrit*. Bandung: Penerbit Informatika.

- Pratama, Rendra Firman. (2013). Penyelesaian *Travelling Salesman Problem* (TSP) dengan Menggunakan *Artificial Bee Colony*. Malang: Universitas Negeri Malang
- Rosen, Kenneth H. (2012). *Discrete Mathematics and Its Application Seventh Edition*. New York: Mc-Graw-Hill.
- Sharma, Kumar. dkk. (2012). *Adaptive Bee Colony in an Artificial Bee Colony for Solving Engineering Design Problem*. India : Indian Institute of Technology Roorkee.
- Sugiharto, Aris. (2006). *Pemrograman GUI dengan MATLAB*. Yogyakarta: Penerbit Andi.
- Suyanto. (2010). *Algoritma Optimasi Deterministik atau Probabilitik*. Yogyakarta: Graha Ilmu.
- Taha, H. A. (2003). *Operations Research: An Introduction seventh Edition*. Prentice Hall, Inc.
- Tsai, Pei-Wei. Dkk. (2009). *Enhanced Artificial Bee Colony Optimization*. Taiwan: Cheng Shiu University.

LAMPIRAN 1

ITERASI 1

Jarak masing-masing solusi:

	Jarak total
1	19,9
2	19
3	23,6
4	20,3
5	14,5
6	21
7	12,5
8	20,3

xj max	23,6
xj min	12,5

Nilai probabilitas	
1	0,1114
2	0,1220
3	0,1583
4	0,1089
5	0,1105
6	0,1121
7	0,1382
8	0,1386

Probabilitas tertinggi: 0,1583

Rute terbaik :

A F E C M N B D P G H L I K O J A

Jarak terbaik :

12,5 km

Nilai probabilitas yang digunakan : 0,1583

Rute yang diperbarui:

	Rute	Jarak
5	A F E N M C B D P G L H J I K O A	11,8
6	A E B D P G O K L C F H I J M N A	19,8
8	A I E F G H J M N P O L K B C D A	20

ITERASI 2

Solusi baru yang lebih baik maka menggantikan solusi lama, jika tidak maka solusi lama dianggap menjadi solusi baru.

Jarak masing-masing solusi:

	Jarak total
1	19,9
2	19
3	23,6
4	20,3
5	11,8
6	19,8
7	12,5
8	20

xj max	23,6
xj min	11,8

Nilai Probabilitas	
1	0,1415
2	0,1066
3	0,1012
4	0,1028
5	0,1629
6	0,1788
7	0,0913
8	0,1150
Probabilitas tertinggi: 0,1788	

Rute terbaik :

A F E N M C B D P G L H J I K O A

Jarak terbaik :

11,8 km

Nilai probabilitas yang digunakan : 0,1788

Rute yang diperbarui:

	Rute	Jarak
1	A F E D C B G H I J K L M N O P A	17,5
3	A J C E B D N K I G P O F H L M A	21,4
4	A B M L G E C F D K J O P N H I A	23,2

ITERASI 3

Solusi baru yang lebih baik maka menggantikan solusi lama, jika tidak maka solusi lama dianggap menjadi solusi baru.

Jarak masing-masing solusi:

	Jarak total
1	17,5
2	19
3	21,4
4	20,3
5	11,8
6	19,8
7	12,5
8	20

x _j max	21,4
x _j min	11,8

Nilai Probabilitas	
1	0,1186
2	0,1220
3	0,1280
4	0,0983
5	0,1500
6	0,1211
7	0,1471
8	0,1149
Probabilitas tertinggi: 0,1500	

Rute terbaik :

A F E N M C B D P G L H J I K O A

Jarak terbaik :

11,8 km

Nilai probabilitas yang digunakan : 0,1788

Rute yang diperbarui:

	Rute	Jarak
2	A F E B D C H J L K P G I O N M A	18,4
3	A J N D B E C K I G P O F H L M A	22,4
5	A L G F E O K P D B M N C H I J A	16,4

ITERASI 4

Solusi baru yang lebih baik maka menggantikan solusi lama, jika tidak maka solusi lama dianggap menjadi solusi baru.

Jarak masing-masing solusi:

	Jarak total
1	17,5
2	18,4
3	21,4
4	20,3
5	11,8
6	19,8
7	12,5
8	20

x _j max	21,4
x _j min	11,8

Nilai Probabilitas	
1	0,1290
2	0,1675
3	0,1060
4	0,1057
5	0,1178
6	0,1078
7	0,1653
8	0,1009
Probabilitas tertinggi: 0,1675	

Rute terbaik :

A F E N M C B D P G L H J I K O A

Jarak terbaik :

11,8 km

Nilai probabilitas yang digunakan : 0,1788

Rute yang diperbarui:

	Rute	Jarak
1	A F E D C B G H I J K P O N M L A	17,7
3	A J N D B E C K I G P M L H F O A	21,3
8	A I E F G H J M N P O D C B K L A	20,5

ITERASI 5

Solusi baru yang lebih baik maka menggantikan solusi lama, jika tidak maka solusi lama dianggap menjadi solusi baru.

Jarak masing-masing solusi:

	Jarak total
1	17,5
2	18,4
3	21,3
4	20,3
5	11,8
6	19,8
7	12,5
8	20

x _j max	21,3
x _j min	11,8

Nilai Probabilitas	
1	0,1115
2	0,1214
3	0,1060
4	0,1491
5	0,1256
6	0,1729
7	0,1034
8	0,1100
Probabilitas tertinggi: 0,1729	

Rute terbaik :

A F E N M C B D P G L H J I K O A

Jarak terbaik :

11,8 km

Nilai probabilitas yang digunakan : 0,1788

Rute yang diperbarui:

	Rute	Jarak
4	A B C E G L M F D K J O P N H I A	23,8
6	A E B K O G P D L C F H I J M N A	21,8
8	A I E F G H J M N P O L K B C D A	20

ITERASI 6

Solusi baru yang lebih baik maka menggantikan solusi lama, jika tidak maka solusi lama dianggap menjadi solusi baru.

Jarak masing-masing solusi:

	Jarak total
1	17,5
2	18,4
3	21,3
4	20,3
5	11,8
6	19,8
7	12,5
8	20

x _j max	21,3
x _j min	11,8

Nilai Probabilitas	
1	0,1221
2	0,1625
3	0,1101
4	0,1129
5	0,1058
6	0,1460
7	0,0956
8	0,1450
Probabilitas tertinggi: 0,1625	

Rute terbaik :

A F E N M C B D P G L H J I K O A

Jarak terbaik :

11,8 km

Nilai probabilitas yang digunakan : 0,1788

Rute yang diperbarui:

	Rute	Jarak
4	A B M F C E G L D K J O P N H I A	22,8
6	A E P G O K B D L C F H I J M N A	21,6
7	A F E C M N B D P K I L H G O J A	13,5

ITERASI 7

Solusi baru yang lebih baik maka menggantikan solusi lama, jika tidak maka solusi lama dianggap menjadi solusi baru.

Jarak masing-masing solusi:

	Jarak total
1	17,5
2	18,4
3	21,3
4	20,3
5	11,8
6	19,8
7	12,5
8	20

x _j max	21,3
x _j min	11,8

Nilai Probabilitas	
1	0,1379
2	0,1239
3	0,1372
4	0,1234
5	0,1164
6	0,1218
7	0,1001
8	0,1394
Probabilitas tertinggi: 0,1394	

Rute terbaik :

A F E N M C B D P G L H J I K O A

Jarak terbaik :

11,8 km

Nilai probabilitas yang digunakan : 0,1788

Rute yang diperbarui:

	Rute	Jarak
4	A B M F C E G L D K J O P N H I A	22,8
6	A E P G O K B D L C F H I J M N A	21,6
7	A F E C M N B D P K I L H G O J A	13,5

ITERASI 8

Solusi baru yang lebih baik maka menggantikan solusi lama, jika tidak maka solusi lama dianggap menjadi solusi baru.

Jarak masing-masing solusi:

	Jarak total
1	17,5
2	18,4
3	21,3
4	20,3
5	11,8
6	19,8
7	12,5
8	20

x _j max	21,3
x _j min	11,8

Nilai Probabilitas	
1	0,1185
2	0,1634
3	0,1028
4	0,1190
5	0,1231
6	0,1142
7	0,1095
8	0,1494
Probabilitas tertinggi: 0,1634	

Rute terbaik :

A F E N M C B D P G L H J I K O A

Jarak terbaik :

11,8 km

Nilai probabilitas yang digunakan : 0,1788

Rute yang diperbarui:

	Rute	Jarak
1	A F E D C J I H G B K P O N M L A	20,5
3	A J N D B E C K I G P O F H L M A	23,5
8	A I E F G P N M J H O D C B K L A	21,9

ITERASI 9

Solusi baru yang lebih baik maka menggantikan solusi lama, jika tidak maka solusi lama dianggap menjadi solusi baru.

Jarak masing-masing solusi:

	Jarak total
1	17,5
2	18,4
3	21,3
4	20,3
5	11,8
6	19,8
7	12,5
8	20

x _j max	21,3
x _j min	11,8

Nilai Probabilitas	
1	0,1033
2	0,1417
3	0,1136
4	0,1069
5	0,1408
6	0,1549
7	0,1346
8	0,1041
Probabilitas tertinggi: 0,1549	

Rute terbaik :

A F E N M C B D P G L H J I K O A

Jarak terbaik :

11,8 km

Nilai probabilitas yang digunakan : 0,1788

Rute yang diperbarui:

	Rute	Jarak
2	A F E J H C D B L K P G I O N M A	17,9
5	A L G P K O E F D B M N C H I J A	17,1
6	A E B K O G P D L C F H I J M N A	21,8

ITERASI 10

Solusi baru yang lebih baik maka menggantikan solusi lama, jika tidak maka solusi lama dianggap menjadi solusi baru.

Jarak masing-masing solusi:

	Jarak total
1	17,5
2	17,9
3	21,3
4	20,3
5	11,8
6	19,8
7	12,5
8	20

xj max	21,3
xj min	11,8

Nilai Probabilitas	
1	0,0965
2	0,1467
3	0,1437
4	0,1214
5	0,1114
6	0,1070
7	0,1057
8	0,1676
Probabilitas tertinggi: 0,1676	

Rute terbaik :

A F E N M C B D P G L H J I K O A

Jarak terbaik :

11,8 km

Nilai probabilitas yang digunakan : 0,1788

LAMPIRAN 2

Source Code M.Files

```

function varargout = gui_ABC(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name', '', 'mfilename', ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', @gui_ABC_OpeningFcn, ...
    'gui_OutputFcn', @gui_ABC_OutputFcn, ...
    'gui_LayoutFcn', [], ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
function gui_ABC_OpeningFcn(hObject, eventdata, handles,
varargin)
handles.output = hObject;

hback = axes('unit','normalized','position',[0 0 1 1]);
uistack(hback,'bottom');
[back, map]=imread('BG.jpg');
image(back)
colormap(map)
set(hback,'handlevisibility','off','visible','off')
guidata(hObject, handles);
set(handles.Mjarak,'visible', 'off');
set(handles.text3,'visible', 'off');
set(handles.help,'visible', 'off');
function varargout = gui_ABC_OutputFcn(hObject, eventdata,
handles)
varargout{1} = handles.output;

function Jumlahkota_Callback(hObject, eventdata, handles)
function Jumlahkota_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function Jumlahsolusi_Callback(hObject, eventdata, handles)
function Jumlahsolusi_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function Mjarak_Callback(hObject, eventdata, handles)

```

```

function Mjarak_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function iterasi_Callback(hObject, eventdata, handles)
function iterasi_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function Solusioptimal_Callback(hObject, eventdata, handles)
function Solusioptimal_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function Proses_Callback(hObject, eventdata, handles)
set(handles.Mjarak,'visible', 'off');
set(handles.text3,'visible', 'off');
MaxIterasi=str2num(get(handles.iterasi,'string'));
jumlahsolusi=str2num(get(handles.Jumlahsolusi,'string'));
JK=str2num(get(handles.Jumlahkota,'string'));
Jarak=str2num(get(handles.Mjarak,'string'));
Jumlahkota=JK-1;
for A=1:jumlahsolusi
    Solusi(A,:)=[1 randperm(Jumlahkota)+1 1];
end
numberofjourney = JK - 1;
iterasi=0;

for i=1:MaxIterasi
for B=1:jumlahsolusi
    AD=Solusi(B,:);
    totaldistance(B,:) = 0 ;
    for i = 1 : JK
        totaldistance(B,:) = totaldistance(B,:) + Jarak (AD (i),
AD (i+1));
    end
end

Xmin=min(totaldistance(:,1));
Xmax=max(totaldistance(:,1));
for D=1:jumlahsolusi
r(D,:)=rand;
end

for E=1:jumlahsolusi
x(E,1)=Xmin + (r(E,1)*(Xmax-Xmin));
end
for D=1:jumlahsolusi
T(D,:)=-1 + (1+1)*rand;
end

```

```

for F=1:jumlahsolusi
    if F <=jumlahsolusi-1
        v(F,1)=x(F,1)+ (T(F,1)*(x(F,1)-x(F+1,1)));
    else
        v(F,1)=x(F,1)+ (T(F,1)*(x(F,1)-x(1,1)));
    end
end
for G=1:jumlahsolusi
    if v(G,1)>Xmax
        v(G,1)=Xmax;
    end
    if v(G,1)<Xmin
        v(G,1)=Xmin;
    end
end
for H=1:jumlahsolusi
    if v(H,1) < x(H,1)
        x(H,1)=v(H,1);
    end
end
for J=1:jumlahsolusi
    fit(J,1)=1/(1+x(J,1));
end
sumfit=0;
for K=1:jumlahsolusi
    sumfit=sumfit + fit(K,1);
end

for L=1:jumlahsolusi
    p(L,1)=fit(L,1)/sumfit;
end
d=0;
for M=1:jumlahsolusi
    q(M,1)=d+p(M,1);
    d=q(M,1);
end

for N=1:jumlahsolusi
    rr(N,:)=rand;
end
for O=1:jumlahsolusi
    if q(O,1)>rr(O,1)
        PS(O,1)=O;
    end
end
for P=1:jumlahsolusi
    if PS(P,1) ~= 0
        JumlahKota = JK - 2;
        Index_1 = randi(JumlahKota) + 1;
        Index_2 = Index_1;
        SolusiP=Solusi(P,:);
        while Index_1 >= Index_2
            Index_1 = randi(JumlahKota) + 1;
            Index_2 = randi(JumlahKota) + 1;
        end
        Kota_1 = SolusiP(Index_1);
    end
end

```

```

Kota_2 = SolusiP(Index_2);
Solusi2Opt = SolusiP;
Solusi2Opt(Index_1:Index_2) = SolusiP(Index_2:-
1:Index_1);
AD=Solusi2Opt;
newtotaldistance(P,:) = 0 ;
for i = 1 : JK
    newtotaldistance(P,:) = newtotaldistance(P,:)
+ Jarak (AD (i), AD (i+1));
end
if newtotaldistance(P) <= totaldistance(P)
    Solusi(P,:)=AD;
end
end

for Q=1:jumlahsolusi
newtotaldistanceS(Q,:)=0;
AF=Solusi(Q,:);
for j = 1 : JK
    newtotaldistanceS(Q,:) = newtotaldistanceS(Q,:) + Jarak (AF
(j), AF (j+1));
end
end
totalmin=min(newtotaldistances(:,1))
yy=find(newtotaldistanceS==totalmin);
k=Solusi(yy,:);
y=rot90(k)
e=newtotaldistanceS(yy,:)
iterasi=iterasi+1

end

set(handles.Solusioptimal,'string',y);
set(handles.totaljarak,'string',e);
function Reset_Callback(hObject, eventdata, handles)
set(handles.Jumlahkota,'string','');
set(handles.Jumlahsolusi,'string','');
set(handles.Mjarak,'string','');
set(handles.iterasi,'string','');
set(handles.totaljarak,'string','');
set(handles.Solusioptimal,'string','');
function PJarak_Callback(hObject, eventdata, handles)
set(handles.Mjarak,'visible', 'on');
set(handles.text3,'visible', 'on');

function totaljarak_Callback(hObject, eventdata, handles)
function totaljarak_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function buka_Callback(hObject, eventdata, handles)
set(handles.help,'visible', 'on');
function tutup_Callback(hObject, eventdata, handles)
set(handles.help,'visible', 'off');

```

```
function figure1_WindowButtonDownFcn(hObject, eventdata, handles)
function help_CreateFcn(hObject, eventdata, handles)
```