MILIK
PERPUSTAKAAN-FTI-UII
YOGYAKARTA

Penyelesaian Tata Letak Fasilitas (*Facility Layout*) Menggunakan Formula QAP (*Quadratic Assignment Problem*)

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat untuk Memperoleh Gelar Sarjana Teknik Informatika

No. Inv 605/AB/FTI-UII/05
Tanggal 25 OKT 05
Asal F. TEKN. INDUSTRI - UII
Harga ~~Rp~~
PERPUSTAKAAN
FAK. TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA

disusun oleh :

Nama : Meyta Nindiasari

No. Mahasiswa : 00 523 019

JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ISLAM INDONESIA

YOGYAKARTA

2004

MILIK
PERPUSTAKAAN-FTI-UII
YOGYAKARTA

LEMBAR PENGESAHAN PEMBIMBING

Penyelesaian Tata Letak Fasilitas (*Facility Layout*) Menggunakan Formula QAP (*Quadratic Assignment Problem*)

Tugas Akhir

Disusun oleh :

Nama : Meyta Nindiasari
No. Mahasiswa : 00 523 019

Yogyakarta, 10 September 2004
Pembimbing

(Sri Kusumadewi, SSi., MT)

LEMBAR PENGESAHAN PENGUJI

Penyelesaian Tata Letak Fasilitas (*Facility Layout*) Menggunakan Formula QAP (*Quadratic Assignment Problem*)

Tugas Akhir

Disusun oleh :

Nama : Meyta Nindiasari
No. Mahasiswa : 00 523 019

Telah Dipertahankan di Depan Sidang Penguji Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 01 Oktober 2004

Tim penguji :

Sri Kusumadewi, Ssi., MT
Ketua

Fathul Wahid, ST., MSc
Anggota I

Yudi Prayudi, Ssi., MKom
Anggota II

Mengetahui,
Dekan Fakultas Teknologi Industri
Universitas Islam Indonesia

UNIVERSITAS ISLAM INDONESIA ★ YOGYAKARTA ★ FAKULTAS TEKNOLOGI INDUSTRI

Ir. Bachrun Sutrisno, MSc

Dengan hati yang tulus dan ikhlas
serta penuh rasa syukur
kupersembahkan karya sederhana ini kepada :

Allah, Tuhan Yang Maha Esa
Untuk semua anugrah dan karunia-Nya sampai detik ini
Thank You for blessing my roads

My Sincerer parents
Ibunda tercinta, Nursechah dan Ayahanda, Eko Widayat
Atas semua dukungan, nasehat, petuah, pengorbanan, dan kasih sayang
yang selalu tercurah kepada ananda selama ini

My Lovely siblings
My little brother, Indra Eka Cahya 'n my little sister, Imas Eriane Zuhrida
Who always made me laugh with their stories and jokes

My Beloved soulmate
Mas "Pebyanto Abadi", The one and only thank's to showing me
the power of love, for being such a great partner and friend for all this years.
I wish we always together 'n forever.....AMIEN

Kalian Semua adalah Anugrah yang Tidak Akan Pernah Terganti dan Ternilai Harganya, Semoga Kita Selalu Dalam Lindungan Allah SWT. AMIEN.....

UCAPAN TERIMAKASIH

Seiring kebersamaan yang telah tercipta dikala kita punya cerita, dalam setiap kesempatan yang hadir tanpa sengaja menjadikan kita kadang tertawa, kadang menangis dalam laku dan karakter yang masing-masing kita miliki, bukanlah hitam atau putih tujuan akhir tapi mempelajari, memahami, dan menghayati yang perlu kita sadari.
Mengingat banyaknya celah yang telah membentuk perjalanan spiritual penyusun, maka dengan segala kesederhanaan dan kerendahan hati, saya Meyta Nindiasari mengucapkan terimakasih kepada :

Segenap keluarga dari ayah dan bunda yang tidak dapat disebutkan satu-persatu, terimakasih atas segala nasehat dan dukungannya. Serta mba Atun dan Mum makasih atas doanya…

Sahabatku :(Danto, Eva, Serly, Tria, Sukma, Sinchan,Ivan_Mamoru, Ari, Ria, Flory, Puji, Fatma, Yana, Kris, Safi, Ukas, Pion, Yogo, Dimaz, Riski, Plenthong, Lelia, Ina, Rio, bety la betet, dianawati, dini) tawa, canda, suka duka, kebersamaan, datang dan pergi adalah hari-hari kita.

Peri-peri ceria Puri Gading : Nina_bobo, Arumba_miaow, SiLia_BuluGetar, Ulin_BuDokter, Nisa_kalempitLempit always ceria 'n jail, tetep jalan terus ya prend…

PergiWati Boarding house crew : Anggi, thanks bgt buat supportnya 'n dah nemenin ampe skripsi meta kelar, cayoo…buruan selesein skripsinya, dah ditunggu nih undangannya…. Reren, thanks sering bawa ole2 tiap pulang, helie… Icha, Thanks dah banyak kasih pelajaran… Umi, thanks buat nasehatnya…. Masak, makan, pergi, ngerumpi ampe pagi ternyata asik juga ya…

Temen-temen KKN : Mas Iqbal, Mas Anto, Mas Wawan, Asep, Diaz, Galih, Aji, Melly, Anik, Tuti, Nurul always kompak ya prend 'n kapan nih qta maen lg….

Temen-temen KP : Mas Fery, Gde, Aji, Kak tau, Helma, Adi, Toim, Lupi, Qoni, Dedy, Woro, Sigit,Hatta kapn qta ke nusakambangan lg ngeliat gunung kapur di bom…

Temen-temen Kost Mamaz : Tukul, Agus, Mas Gondronk, Sarno, Boim, Alam, Sabar, Diding, Sugi, Mas Kiting, Aji, Temo semuanya lucu-lucu deh… meta jadi pengen ketawa terus kalo liat tingkah kalian….

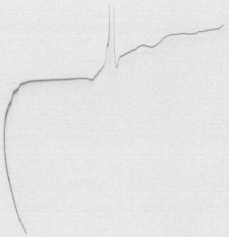
Temen-temen Informatika : Pinoe, Iid, Anjar, Aal, Febri, Arief, Anto, Ambar2 picang, Camel, Ivan'solo', Pasha_ah, Novan, Beta, Adi_ikam, Qiqi_upil, Popoy, Vitha, Dita_Cilik Aal kapan qta maen2 lagi… 'n semua temen2 Informatika 2000

Kakak-kakak qu : Mas Hambali, Mas Kiki, Mas Agung, Mas Faisal, Mas Helmi, Mas Fatur, Mas Yuni, Mba Febi, Mba Novir, Mas Rama_Junkis makasih untuk semua nasehatnya

Kadang dengan bercanda akan mewujudkan kenyataan yang tidak dapat dibayangkan, mungkin inilah kenangan manis yang selalu dikenang dengan suka ria, tapi tidak semua kenangan pahit dapat dilupakan dengan begitu saja, kesedihan akan larut dengan rasa kegembiraan dan keceriaan dengan kebersamaan yang tak pernah putus.
Terimakasih atas persahabatan, kekompakkan, kebersamaan, yang erat selama ini yang merupakan kenagan indah. Semoga dapat menjadi pengalaman yang mengesankan dan membanggakan. Amien…

MOTTO

"Allah tidak akan merubah nasib suatu kaum kecuali kaum itu yang berusaha untuk merubahnya"



"Ridho ibu merupakan rezeki yang tidak ada nilainya"

"Sesungguhnya Allah SWT. Akan membantu orang-orang yang berusaha, sekalipun ia tidak memiliki kekuatan dan kemampuan, melainkan kemauan yang kuat serta niat yang tulus dan ikhlas"

"Sholat dapat menjernihkan fikiran, dan hanya sholatlah yang dapat meninggikan derajatmu di hadapan-Nya"

"For every step you take, for every doubt you face, for every choice you made, dream aren't made to be erase cause you are the master of your destiny"

Kata Pengantar

بِسْمِ اللهِ الرَّحْمٰنِ الرَّحِيْمِ

Assalamu'alaikum Wr. Wb

Alhamdulillah, segala puji syukur hanyalah kepada Allah SWT dan semoga sholawat serta salam dilimpahkan kepada junjungan kita Nabi Muhammad SAW, keluarga, sahabat, dan pengikut-pengikut beliau (amin). Sehingga penulisan laporan tugas akhir yang berjudul : **Penyelesaian Tata Letak Fasilitas (*Facility Layout*) Menggunakan Formula QAP (*Quadratic Assignment Problem*)** dapat penulis selesaikan dengan baik.

Laporan tugas akhir ini disusun untuk melengkapi salah satu syarat guna memperoleh gelar Sarjana Teknik Informatika pada Universitas Islam Indonesia dan atas apa yang telah diajarkan selama perkuliahan baik teori maupun praktek, disamping laporan itu sendiri yang merupakan rangkaian kegiatan yang harus dilakukan setelah tugas akhir ini selesai.

Untuk itu saya menyampaikan ucapan terimakasih dan penghargaan yang setinggi-tingginya kepada :

1. Ibu Sri Kusumadewi Ssi., MT selaku Ketua Jurusan Teknik Informatika serta Pembimbing Tugas Akhir. Terimakasih atas segala bantuan, dukungan, semangat, dan pengetahuannya, serta kemudahan yang telah diberikan.

2. Bapak Zainuddin Zukhri, ST selaku Sekretaris Jurusan Teknik Informatika dan seluruh dosen pengajar Jurusan Teknik Informatika.
3. Teman-teman Informatika 2000 dan seluruh pihak yang tidak dapat saya sebutkan satu persatu.

Semoga apa yang telah mereka berikan dengan keikhlasan, mendapat pahala yang setimpal dari Allah SWT. Penulis menyadari dalam penulisan laporan tugas akhir ini masih jauh dari sempurna, karena keterbatasan kemampuan dan pengalaman. Penulis mengharapkan saran dan kritik yang bersifat membangun untuk memperbaiki tugas akhir ini semoga dapat bermanfaat bagi penulis khususnya dan pembaca pada umumnya.

Wassalamu'alaikum Wr. Wb

Yogyakarta, September 2004

Penulis

DAFTAR ISI

Halaman judul i
Lembar Pengesahan Pembimbing ii
Lembar Pengesahan Penguji iii
Lembar Persembahan iv
Lembar Ucapan Terimakasih v
Motto. vi
Kata Pengantar vii
Daftar Isi ix
Daftar Gambar xii
Abstraksi xiv

BAB I PENDAHULUAN

1.1 Latar Belakang Masalah 1
1.2 Rumusan Masalah 3
1.3 Batasan Masalah 4
1.4 Tujuan Penelitian 4
1.5 Manfaat Penelitian 5
1.6 Metodologi Penelitian 5
1.6.1 Metode Pengumpulan Data 5
1.6.2 Metode Pengembangan Sistem 6
1.7 Sistematika Penulisan 7

BAB II LANDASAN TEORI

2.1 Definisi Tata Letak Fasilitas 9
2.2 Penyelesaian Tata Letak fasilitas 12
2.3 Interaksi antara Fasilitas-fasilitas Baru 14
2.4 Formula QAP (*Quadratic Assignment Problem*) 15
2.4.1 *Construction Heuristics* 17
2.4.2 *Improvement Heuristics* 17

2.5 Simulasi *Heuristics* untuk QAP (*Quadratic Assignment Problem*) 18

BAB III ANALISIS KEBUTUHAN SISTEM

3.1 Metode Analisis 19
3.2 Hasil Analisis Kebutuhan 19
3.2.1 Masukan Sistem 20
3.2.2 Keluaran Sistem 20
3.2.3 Kebutuhan Fungsi 21
3.2.4 Kebutuhan Perangkat Keras 21
3.2.5 Kebutuhan Perangkat Lunak 22
3.2.6 Antarmuka Sistem 22
3.2.7 Kinerja yang diharapkan 22

BAB IV PERANCANGAN PERANGKAT LUNAK

4.1 Metode Perancangan 23
4.2 Hasil Perancangan 23
4.2.1 Perancangan Diagram Alir Sistem 23
4.2.2 Perancangan Antarmuka 29
4.2.2.1 Rancangan Antarmuka Menu Utama 30
4.2.2.2 Rancangan Antarmuka *Input* Data 31
4.2.2.3 Rancangan Antarmuka *Loading* Data 31
4.2.2.4 Rancangan Antarmuka Grafik 32
4.2.2.5 Rancangan Antarmuka Laporan (*Report*) 33
4.2.2.6 Rancangan Antarmuka Perbandingan Metode 34
4.2.2.7 Rancangan Antarmuka *Help* 34
4.2.3 Perancangan *File* 35
4.2.3.1 Perancangan *Input* 35
4.2.3.2 Perancangan Proses 35
4.2.3.4 Perancangan Output 35

BAB V IMPLEMENTASI PERANGKAT LUNAK

5.1 Implementasi 36

5.1.1 Pemilihan Bahasa Pemrograman 36

5.1.2 Implementasi Antarmuka 37

5.1.2.1 Antarmuka Menu Utama 37

5.1.2.2 Antarmuka Data Input/Data Baru 40

5.1.2.3 Antarmuka Ambil Data 41

5.1.2.4 Antarmuka Simpan Data Input 42

5.1.2.5 Antarmuka Grafik Selisih Biaya Penugasan 43

5.1.2.6 Antarmuka Grafik Biaya Penugasan 44

5.1.2.7 Antarmuka Peta Lokasi Departemen/Fasilitas 45

5.1.2.8 Antarmuka Laporan/*Report* 46

5.1.2.9 Antarmuka Perbandingan Metode 47

5.1.2.10 Antarmuka *Help* 48

BAB VI ANALISIS KINERJA PERANGKAT LUNAK

6.1 Proses Masukan/*Input* 49

6.2 Proses Peatihan 50

6.3 Proses Keluaran/*Output* 51

6.4 Uji Coba Perangkat Lunak 52

6.5 Hasil Uji Coba Perangkat Lunak 52

6.6 Analisis Perbandingan Metode 66

6.7 Kelebihan dan Kekurangan Perangkat Lunak 67

BAB VII PENUTUP

7.1 Kesimpulan 68

7.2 Saran 69

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar 4.1 Bagan Alir Perangkat Lunak untuk Optimasi Tata Letak Fasilitas24
Gambar 4.2 Bagan Alir Metode *Improvement Heuristics* 25
Gambar 4.3(a) Bagan Alir Metode *Construction Heuristics* 26
Gambar 4.3(b) Bagan Alir Metode *Construction Heuristics* 27
Gambar 4.4 Bagan Alir PosisiDepartemen Secara Acak 28
Gambar 4.5 Bagan Alir Pemindahan Posisi Departemen untuk Diurutkan 29
Gambar 4.6 Rancangan Antarmuka Menu Utama 30
Gambar 4.7 Rancangan Antarmuka *Input* Data 31
Gambar 4.8 Rancangan Antarmuka *Loading* Data 31
Gambar 4.9 Rancangan Antarmuka Grafik Selisih Biaya Penugasan 32
Gambar 4.10 Rancangan Antarmuka Grafik Total Biaya 32
Gambar 4.11 Rancangan Antarmuka Peta Lokasi Departemen 33
Gambar 4.12 Rancangan Antarmuka Laporan 33
Gambar 4.13 Rancangan Antarmuka Hasil Perbandingan 34
Gambar 4.14 Rancangan Antarmuka *Help* 34
Gambar 5.1 Antarmuka Menu Utama 39
Gambar 5.2 Antarmuka *Input* Data/Data Baru 40
Gambar 5.3 Antarmuka Ambil Data 41
Gambar 5.4 Antarmuka Simpan Data Input 42
Gambar 5.5 Antarmuka Grafik Selisih Biaya Penugasan 43
Gambar 5.6 Antarmuka Grafik Biaya Penugasan 44
Gambar 5.7 Antarmuka Grafik Pemetaan Departemen/Fasilitas 45
Gambar 5.8 Antarmuka Laporan/*Report* 46
Gambar 5.9 Antarmuka Laporan Perbandingan Metode 47
Gambar 5.10 Antarmuka *Help* 48
Gambar 6.1 Pesan File Data yang Dibuka tidak Dapat Didefinisikan 49
Gambar 6.2 Pesan Nilai Data tidak Boleh Bernilai Nol 49
Gambar 6.3 Pesan Nilai Data Ada yang Kosong 50
Gambar 6.4 Pesan Kesalahan Parameter Pelatihan Ada yang Kosong 50

Gambar 6.5 Pesan Masukan Jumlah Departemen di Luar *Range* 50
Gambar 6.6 Pesan Nilai Parameter Maksimum Iterasi di Luar *Range* 51
Gambar 6.7 Pesan Nilai Parameter Maksimum Cacah di Luar *Range* 51
Gambar 6.8 Pesan Kesalahan Penyimpanan 51
Gambar 6.9 Tampilan Awal Kasus-1(a) 53
Gambar 6.10 Tampilan Grafik Selisih Biaya Penugasan Kasus-1(a) 54
Gambar 6.11 Tampilan Grafik Total Biaya Kasus-1(a) 54
Gambar 6.12 Tampilan Peta Lokasi Departemen Kasus-1(a) 55
Gambar 6.13 Tampilan Laporan Hasil Proses Pelatihan Kasus-1(a) 55
Gambar 6.14 Tampilan Laporan Perbandingan Metode Kasus-1(a) 56
Gambar 6.15 Tampilan Awal Kasus-1(b) 57
Gambar 6.16 Tampilan Grafik Total Biaya Kasus-1(b) 57
Gambar 6.17 Tampilan Peta Lokasi Departemen Kasus-1(b) 58
Gambar 6.18 Tampilan Laporan Hasil Proses Pelatihan Kasus-1(b) 58
Gambar 6.19 Tampilan Laporan Perbandingan Metode Kasus-1(b) 59
Gambar 6.20 Tampilan Awal Kasus-2(a) 60
Gambar 6.21 Tampilan Grafik Selisih Biaya Penugasan Kasus-2(a) 60
Gambar 6.22 Tampilan Grafik Total Biaya Kasus-2(a) 61
Gambar 6.23 Tampilan Peta Lokasi Departemen Kasus-2(a) 61
Gambar 6.24 Tampilan Laporan Hasil Proses Pelatihan Kasus-2(a) 62
Gambar 6.25 Tampilan Laporan Perbandingan Metode Kasus-2(a) 62
Gambar 6.26 Tampilan Awal Kasus-2(b) 63
Gambar 6.27 Tampilan Grafik Total Biaya Kasus-2(b) 64
Gambar 6.28 Tampilan Peta Lokasi Departemen Kasus-2(b) 64
Gambar 6.29 Tampilan Laporan Hasil Proses Pelatihan Kasus-2(b) 65
Gambar 6.30 Tampilan Laporan Perbandingan Metode Kasus-2(b) 65

Abstraksi

Tata Letak Fasilitas (*facility layout*) merupakan persoalan yang mempunyai konsep sederhana dan mudah dipahami. Pada penyelesaian tata letak fasilitas, optimasi yang diinginkan agar ditemukan letak fasilitas dengan total biaya penugasan yang minimum. Perhitungan biaya dengan menggunakan formula QAP (*Quadratic Assignment Problem*) dan penentuan penugasan (letak fasilitas) dengan biaya yang paling minimum menggunakan metode *heuristics*.

Tujuan yang ingin dicapai adalah mengaplikasikan formula QAP (*Quadratic Assignment Problem*) dengan menggunakan metode *Improvement heuristics* dan *Construction heuristics* untuk mendapatkan tata letak fasilitas dengan biaya yang minimum.

Proses penelitian menggunakan data input berupa jumlah departemen/fasilitas, nilai jarak antar lokasi, nilai aliran perpindahan bahan antar departemen/fasilitas, maksimum iterasi, maksimum cacah yang kemudian akan diolah menggunakan formula QAP (*Quadratic Assignment Problem*) dengan metode *Improvement heuristics* atau *Construction heuristics* untuk menentukan tata letak fasilitas dengan biaya yang paling minimum.

Kata kunci : Tata Letak Fasilitas, QAP

MILIK
PERPUSTAKAAN-FTI-UII
YOGYAKARTA

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan teknologi komputer dari tahun ke tahun semakin akrab menyentuh kehidupan dari pekerjaan profesional, alat rumah tangga, hobi sampai ke permainan. Harganya pun semakin terjangkau oleh masyarakat umum. Sehingga banyak perusahaan, industri maupun instansi-instansi yang mengalihkan/memanfaatkan teknologi komputer. Ketepatan, keakuratan dan kecepatan dalam pengolahan data yang sangat efektif dan efisien sangat diperlukan untuk menghemat waktu dan biaya. Banyak manfaat dan keuntungan yang dapat diperoleh karena kecanggihan komputer dalam bidang menghitung, khususnya untuk memproses data dengan jumlah yang banyak dan rumit.

Permasalahan tata letak fasilitas merupakan masalah yang masih sering dijumpai di dalam dunia industri maupun perusahaan-perusahaan, karena tata letak fasilitas merupakan salah satu faktor penting dalam sistem produksi. Tata letak fasilitas (*facility layout*) atau sering disebut dengan susunan letak fasilitas produksi di dalam pabrik, sangat perlu direncanakan dan diatur dengan baik dan sesuai dengan pelaksanaan produksi yang ada di dalam pabrik tersebut. Hal ini disebabkan oleh karena dengan susunan letak fasilitas produksi yang tepat, maka para karyawan akan dapat melaksanakan tugasnya dengan baik sementara aliran produksi dari bahan baku sampai dengan menjadi produksi akhir akan dapat berjalan dengan lancar.

Kelancaran aliran produksi akan dapat menunjang efisiensi produksi yang dilaksanakan oleh perusahaan tersebut.

Tata letak fasilitas adalah landasan utama dalam dunia industri. Tata letak fasilitas (*facility layout*) dapat didefinisikan sebagai tata cara pengaturan fasilitas-fasilitas fisik pabrik guna menunjang kelancaran proses produksi. Tata letak yang baik akan mempermudah aliran dan pemindahan bahan pada suatu tempat.

Perencanaan fasilitas akan menentukan bagaimana aktivitas-aktivitas dari fasilitas-fasilitas produksi dapat diatur sedemikian rupa sehingga mampu menunjang upaya pencapaian tujuan pokok secara efektif dan efisien. Tata letak yang efektif akan mengurangi ongkos pemindahan material dan ongkos operasi. Namun kebanyakan permasalahan yang terjadi pada departemen produksi dikebanyakan perusahaan, tidak menunjukkan adanya tata letak fasilitas yang direncanakan sesuai aturan tata letak fasilitas. Dengan adanya pengaturan tata letak fasilitas, diharapkan dapat mengurangi ongkos pemindahan barang dan ongkos produksi secara keseluruhan. Disamping itu juga untuk meningkatkan efisiensi dan efektivitas produksinya.

Pengaturan tata letak fasilitas suatu industri hasilnya akan dinikmati dalam jangka waktu yang lama, sehingga perencanaan pengaturan tata letak fasilitas harus dilakukan dengan baik dan kokoh. Secara khusus tujuan penentuan tata letak fasilitas adalah meminimalkan jumlah biaya konstruksi serta instalasi, pemindahan bahan, biaya perawatan dan membuat operasi kerja yag lebih efektif dan efisien.

Ketentuan tata letak fasilitas ini mempertimbangkan faktor jarak, biaya dan jumlah fasilitas. Ada 2 tipe penyelesaian pada permasalahan tata letak fasilitas untuk mendapatkan penugasan dengan biaya yang minimum yaitu *procedure* tetap dan *procedure* heuristik. *Procedure* tetap amat teoritis dan tidak dapat memecahkan masalah jika jumlah fasilitas (n) lebih besar dari 15. Pencarian biaya penugasan yang digunakan sebagai pertimbangan tata letak fasilitas ada n!. Jika menggunakan sistem manual atau secara eksak berdasarkan basis pengetahuan akan sangat sulit, yaitu pencarian biaya penugasan memerlukan perhitungan dari semua n! penugasan dan membutuhkan waktu yang cukup lama. Sebagai ilustrasi, untuk 12 fasilitas saja, diperlukan proses perhitungan sebanyak $4{,}79.10^8$. Penjelasan ini menunjukkan bahwa solusi eksak terhadap masalah tata letak fasilitas sangat sulit dilakukan. Hal inilah yang mendorong penulis untuk membangun software dengan menggunakan formula QAP (*Quadratic Assignment Problem*) yang dapat menghitung dan mencari penugasan dengan biaya yang minimum untuk menentukan tata letak fasilitas.

1.2 Rumusan Masalah

Sesuai dengan latar belakang masalah yang telah diuraikan sebelumnya, maka dapat dirumuskan permasalahannya, yaitu : Bagaimana penerapan metode *Construction Heuristics* dan *Improvement Heuristics* dengan menggunakan formula QAP (*Quadratic Assignment Problem*) pada penentuan lokasi dan tata letak fasilitas agar dapat menghasilkan biaya penugasan yang minimum.

1.3 Batasan Masalah

Agar lebih mudah dalam penarikan kesimpulan serta menjaga agar lebih mencerminkan permasalahan yang sedang dihadapi, maka diperlukan penyederhanaan masalah sebagai berikut :

1. Ada 5 input, yaitu :
   a. Jumlah fasilitas/departemen.
   b. Nilai jarak antar lokasi yang disimpan dalam matrik jarak (d).
   c. Nilai aliran bahan antar fasilitas/departemen dan disimpan dalam suatu matrik (r), sehingga dapat diketahui nilai bobot/ biaya antar departemen (w).
   d. Maksimum iterasi.
   e. Maksimum cacah.
2. Menggunakan metode *Construction Heuristics* dan *Improvement Heuristics* dengan menggunakan formula QAP (*Quadratic Assignment Problem*).
3. Bahasa pemrograman yang dipakai untuk menentukan lokasi dan tata letak fasilitas adalah dengan menggunakan Delphi 6.0 yang merupakan perangkat lunak dalam melakukan perhitungan.

1.4 Tujuan Penelitian

Penelitian yang dilakukan bertujuan untuk merealisasikan sebuah program aplikasi sederhana pada permasalahan tata letak fasilitas dengan menggunakan formula QAP (*Quadratic Assignment Problem*) untuk mendapatkan tata letak fasilitas dengan total biaya penugasan yang minimum.

1.5 Manfaat Penelitian

Adapun manfaat yang diperoleh dari penelitian ini adalah :

1. Dapat mengimplementasikan dan membandingkan dua metode yaitu *Construction Heuristics* dengan *Improvement Heuristics* dalam penentuan lokasi dan tata letak fasilitas sehingga diperoleh biaya penugasan yang paling minimum.
2. Memberi sumbangan pemikiran serta sebagai bahan pertimbangan kepada dunia industri atau pihak perusahaan dalam pengambilan keputusan khususnya dalam penentuan lokasi dan tata letak fasilitas sehingga menghasilkan penugasan dengan total biaya yang minimum.

1.6 Metodologi Penelitian

Adapun metode-metode yang akan digunakan dalam penyelesaian tugas akhir ini adalah sebagai berikut :

1.6.1 Metode Pengumpulan Data

Metode pengumpulan data adalah metode yang digunakan untuk mengumpulkan data yang diperlukan dalam penelitian. Metode ini meliputi : studi pustaka, yaitu pengumpulan data dengan cara melakukan studi, analisis dan dokumentasi literatur, dan sumber catatan lain yang berkaitan dengan permasalahan yang dibahas.

1.6.2 Metode Pengembangan Sistem

Metode pengembangan sistem disusun berdasarkan hasil dari data yang sudah diperoleh. Metode ini meliputi :

1. Analisa data

   Analisa ini dilakukan untuk mengolah data yang sudah didapat dan mengelompokkan data sesuai dengan kebutuhan perancangan.

2. Desain

   Tahap ini merupakan tahap perancangan sistem, yaitu mendefinisikan kebutuhan yang ada, menggambarkan bagaimana sistem dibentuk dan persiapan untuk rancang bangun aplikasi.

3. Pengkodean

   Tahap ini adalah penerjemahan rancangan dalam tahap desain ke dalam bahasa pemrograman komputer yang telah ditentukan sebelumnya.

4. Pengujian

   Setelah aplikasi selesai dibuat, maka pada tahap ini merupakan uji coba terhadap program tersebut. Sehingga analisis hasil implementasi yang didapat dari sistem disesuaikan dengan kebutuhan sistem tersebut. Jika penerapan sistem sudah berjalan dengan lancar, maka sistem dapat diimplementasikan untuk melakukan perhitungan biaya penugasan dalam penentuan lokasi dan tata letak fasilitas.

1.7 Sistematika Penulisan

Untuk memudahkan dalam memahami laporan Tugas Akhir, dikemukakan sistematika penulisan yang terdiri dari 7 bab, yaitu :

BAB I Pendahuluan

Bab ini merupakan pengantar terhadap permasalahan yang akan dibahas. Di dalamnya menguraikan tentang gambaran suatu penelitian yang terdiri dari : latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

BAB II Landasan Teori

Bab ini merupakan bagian yang menjadi landasan teori yang digunakan dalam memecahkan masalah dan membahas masalah yang ada. Teori-teori yang dicantumkan dalam bab ini , seperti teori interaksi antara fasilitas-fasilitas baru, formula QAP (*Quadratic Assignment Problem*), *construction heuristics*, *improvement heuristics*, simulasi *heuristics* untuk QAP (*Quadratic Assignment Problem*).

BAB III Analisis Kebutuhan Sistem

Mengemukakan analisis kebutuhan sistem yang meliputi metode analisis, analisis kebutuhan berupa input, output, metode-metode yang dibutuhkan serta antarmuka yang diinginkan.

BAB IV Perancangan Perangkat Lunak

Bab ini mengemukakan tahapan perancangan yang meliputi metode perancangan sistem, *flowchart*, perhitungan biaya penugasan untuk penentuan lokasi dan tata letak fasilitas serta visualisasinya yang meliputi perancangan antarmuka masukan sistem dan rancangan antarmuka keluaran sistem.

BAB V Implementasi Perangkat Lunak

Bab ini merupakan kelanjutan dari tahap perancangan yang dibuat. Tahap implementasi ini berisi tentang pembuatan *software* berdasarkan hasil analisis dan perancangan bab sebelumnya, dengan menggunakan prosedur-prosedur dalam bahasa pemrograman yang dipilih.

BAB VI Analisis Kinerja Perangkat Lunak

Bab ini merupakan penguraian dari hasil implementasi yang dibuat, yaitu berisi tentang tata cara kerja program, pengujian program dan analisis kesalahan.

BAB VII Penutup

Bab ini merupakan bab terakhir yang berisikan kesimpulan yang diperoleh dari pemecahan masalah maupun penelitian serta saran-saran sebagai masukan untuk perbaikan di masa yang akan datang.

BAB II

LANDASAN TEORI

2.1 Definisi Tata Letak Fasilitas

Tata letak fasilitas (*facility layout*) dapat didefinisikan sebagai tata cara pengaturan fasilitas-fasilitas fisik guna menunjang kelancaran proses produksi. Tata letak yang baik akan mempermudah aliran dan pemindahan bahan pada suatu tempat. Tata letak fasilitas berfungsi untuk menggambarkan sebuah susunan yang ekonomis dari tempat-tempat kerja yang berkaitan[PUR01].

Tipe-tipe tata letak fasilitas yang sesuai akan menjadikan efisiensi proses produksi untuk jangka waktu yang cukup panjang, yaitu [PUR01] :

a. Tata letak fasilitas berdasarkan aliran produksi (*product layout*)

   Dalam *poduct layout,* mesin-mesin atau alat bantu disusun menurut urutan proses dari suatu produk. *Poduct layout* dapat didefinisikan sebagai metode atau cara pengaturan dan penempatan semua fasilitas produksi yang diperlukan ke dalam suatu departemen tertentu atau khusus. Suatu produk dapat dibuat/diproduksi sampai selesai di dalam departemen tersebut. Bahan baku dipindahkan dari satu stasiun kerja ke stasiun kerja lainnya di dalam departemen tersebut, dan tidak perlu dipindah-pindahkan ke departemen yang lain. Dengan demikian dapat disimpulkan, bahwa tujuan dari tata letak ini adalah untuk mengurangi proses pemindahan bahan dan memudahkan pengawasan di dalam aktivitas produksi, sehingga pada akhirnya terjadi penghematan biaya.

b. Tata letak fasilitas berdasarkan fungsi atau macam proses (*Process Layout*)

Dalam *Process/Functional Layout* semua operasi dengan sifat yang sama dikelompokkan dalam departemen yang sama pada suatu pabrik/industri. Mesin, peralatan yang mempunyai fungsi yang sama dikelompokkan menjadi satu. Dengan kata lain material dipindah menuju departemen-departemen sesuai dengan urutan proses yang dilakukan.

Process Layout dilakukan bila volume produksi lebih kecil, dan terutama untuk jenis produk yang tidak *standart*, biasanya berdasarkan *order*. Kondisi ini disebut sebagai *"job-shop"*. Tata letak tipe *process layout* banyak dijumpai pada sektor industri manufakturing maupun jasa, misal Bank, Rumah Sakit, Universitas dan industri jasa lainnya yang mengatur segala fasilitas berdasarkan kelompok-kelompok fungsionalnya. Begitu pula pada sektor industri manufakturing, beberapa bengkel permesinan akan mengatur tata letak mesinnya berdasarkan kelompok-kelompok mesin yang memiliki fungsi sejenis.

c. Tata letak fasilitas berdasarkan kelompok produk (*Group Technology Layout*)

Tipe tata letak ini, biasanya komponen yang tidak sama dikelompokkan ke dalam satu kelompok berdasarkan kesamaan bentuk komponen, mesin atau peralatan yang dipakai. Kelebihan tata letak berdasarkan kelompok teknologi ini adalah dengan adanya pengelompokkan produk sesuai dengan proses pembuatannya maka akan dapat diperoleh pendayagunaan mesin yang maksimal. Juga lintasan aliran kerja menjadi lebih lancar dan jarak perpindahan material akan lebih pendek bila dibandingkan tata letak berdasarkan fungsi atau macam *process*

(*process layout*). Tata letak berdasarkan kelompok teknologi/produk dapat pula menciptakan suasana kerja yang lebih baik.

d. *Layout* yang berposisi tetap (*Fixed Position Layout*)

Sistem berdasarkan "*product layout*" maupun "*process layout*", produk bergerak menuju mesin sesuai dengan urutan proses yang dijalankan. *Layout* yang berposisi tetap ditunjukkan bahwa mesin, manusia, serta komponen-komponen kecil bergerak menuju lokasi material untuk menghasilkan produk. *Layout* ini biasanya digunakan untuk memproses barang yang relatif besar dan berat sedangkan peralatan yang digunakan mudah untuk dilakukan pemindahan. Contoh dari proses ini adalah industri pesawat terbang, penggalangan kapal, pekerjaan konstruksi bangunan.

Pada dasarnya tujuan utama tata letak adalah optimasi pengaturan fasilitas-fasilitas operasi sehingga nilai yang diciptakan oleh sistem produksi akan maksimal. Adapun secara rinci beberapa tujuan tata letak fasilitas diantaranya adalah[PUR04] :

a. Memanfaatkan area yang ada

Tata letak yang optimal akan memberikan solusi dalam penghematan penggunaan area yang ada, baik area untuk produksi, gudang, *service*, dan juga untuk departemen lainnya.

b. Pendayagunaan pemakaian mesin, tenaga kerja dan fasilitas produksi lebih besar

Pengaturan yang tepat akan dapat mengurangi investasi di dalam peralatan dan perlengkapan produksi. Peralatan-peralatan dan perlengkapan dalam proses produksi dapat dipergunakan di dalam tingkat effisiensi yang cukup tinggi. Begitu juga tenaga kerja dan fasilitas produksi lainnya akan lebih berdayaguna.

c. Meminimumkan *material handling*

Selama proses produksi/operasi perusahaan akan selalu terjadi aktivitas perpindahan baik itu bahan baku, tenaga kerja, mesin ataupun peralatan produksi lainnya. Proses perpindahan ini memerlukan biaya yang relatif cukup besar. Dengan demikian tata letak yang baik harus mampu meminimalkan aktivitas-aktivitas pemindahan bahan.

2.2 Penyelesaian Tata Letak Fasilitas

Peneyelesaian tata letak fasilitas dalam perancangan teknik, khususnya pada proses manufaktur dapat dilakukan dengan tahapan-tahapan sebagai berikut [PUR04] :

a. Membuat definisi tentang tujuan dari fasilitas.

Pada tahap ini ditentukan produk apa yang dibuat, bagaimana produk tersebut dibuat, apakah dengan menggunakan tata letak fasilitas yang baru atau mengembangkan yang sudah ada. Volume dan tingkat aktivitas diidentifikasi juga jika perlu.

b. Menentukan aktivitas utama dan aktivitas penunjang yang dibutuhkan dalam proses manufaktur.

Termasuk di dalam hal ini adalah penentuan pengoperasian, peralatan, personil, dan aliran bahan. Yang dimaksud aktivitas penunjang adalah kegiatan-kegiatan yang menunjang aktivitas utama sehingga akan membantu kelancaran proses kerja.

c. Menentukan hubungan diantara aktivitas yang ada.

Jika suatu departemendengan departemen lainnya mempunyai kaitan yang sangat erat, keduanya didekatkan agar aliran operasi akan lebih efisien.

d. Menentukan kebutuhan ruangan diantara aktivitas-aktivitas pada proses manufaktur.

Seluruh kebutuhan peralatan, bahan dan tenaga kerja harus diperhitungkan dengan cermat, hal ini terkait dengan perencanaan kebutuhan ruang yang digunakan masing-masing aktivitas.

e. Buat beberapa alternatif tata letak fasilitas.

Termasuk di sini adalah alternatif lokasi fasilitas dan alternatif perancangan fasilitas. Alternatif perancangan fasilitas meliputi alternatif perancangan tata letak (layout), perancangan sistem fasilitas danperancangan sistem penanganan bahan.

f. Lakukan evaluasi terhadap alternatif tata letak fasilitas, dengan memberi *ranking* terhadap alternatif tata letak fasilitas yang ada.

Masing-masing tata letak ditentukan faktor-faktor subyektifnya dan dilakukan evaluasi jika dan bagaimana faktor-faktor tersebut mempunyai pengaruh terhadap fasilitas atau operasi.

g. Dari hasil evaluasi pilih tata letak fasilitas.

Permasalahannya adalah untuk menentukan tata letak fasilitas jika ada yang sesuai dengan sasaran dan tujuan organisasi. Sering biaya bukan satu-satunya pertimbangan utama ketika mengevaluasi tata letak fasilitas. Informasi yang

disampaikan pada langkah f dipergunakan sebagai dasar seleksi akhir dari suatu rencana.

h. Implementasikan tata letak fasilitas yang dipilih. Implementasi merupakan realisasi dari langkah-langkah sebelumnya.

i. Pembiayaan dan adaptasi tata letak fasilitas.

Keseluruhan dari tata letak fasilitas harus dimodifikasi secara serasi. Tata letak fasilitas merupakan refleksi dari penghematan energi atau perbaikan dari peralatan penanganan bahan menjadi lebih berguna.

j. Definisikan kembali tujuan dari fasilitas.

Perlu kiranya untuk mengidentifikasi kembali proses produksi dari produk atau jasa.

2.3 Interaksi antara Fasilitas-fasilitas Baru

Interaksi antara fasilitas-fasilitas baru merupakan penempatan fasilitas-fasilitas dengan menukarkan pasangan-pasangan dari fasilitas-fasilitas yang baru. Satu fasilitas pasti menempati satu tempat. Sebagai contoh, tempat dapat berupa ruangan di dalam pabrik dan fasilitas-fasilitasnya berupa departemen untuk ditempatkan di suatu ruangan. Contoh yang lainnya fasilitas berupa mesin-mesin baru untuk dilokasikan pada tempat produksi dan tempatnya berupa lokasi yang memungkinkan untuk fasilitas yang baru. Penyelesaian masalah penempatan fasilitas-fasilitas dengan QAP (*Quadratic Assignment Problem*) dengan menggunakan metode heuristik agar dapat menghasilkan penyelesaian yang optimum untuk jumlah fasilitas yang besar[FMW92].

2.4 Formula QAP (*Quadratic Assignment Problem*)

Penugasan yang optimum untuk meminimalkan total jarak dari perpindahan bahan dan nilai aliran perpindahan bahan antar stasiun kerja/departemen dapat dirumuskan sebagai masalah optimasi sebagai berikut [FMW92] :

r_{ij} : nilai aliran perpindahan bahan dari stasiun kerja/departemen i ke stasiun kerja/departemen j

d_{rs} : jarak antara lokasi r dan s

x_{ir} : $\begin{cases} 1 \\ 0 \end{cases}$ jika stasiun kerja/departemen i diletakan di lokasi r, selain itu = 0

Setiap stasiun kerja/departemen harus diletakkan di sebuah lokasi dan setiap lokasi harus mempunyai stasiun kerja/departemen yang ditempatkan pada lokasi tersebut. Total jarak perpindahan bahannya adalah [FMW92] :

$$\text{Min } z = \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{r=1}^{n}\sum_{s=1}^{n} r_{ij}d_{rs}x_{ir}x_{js} \qquad (2.1)$$

$$\sum_{r=1}^{n} x_{ir} = 1, \quad i = 1,.....,n$$

$$\sum_{i=1}^{n} x_{ir} = 1, \quad r = 1,.....,n$$

$$x_{ir} \in \{0,1\} \quad \forall\ i,r$$

Dalam pemecahan masalah lokasi yang mempunyai ciri-ciri tertentu dikerjakan secara langsung dengan variabel-variabel pasangan lokasi x_{ir}. Dalam QAP (*Quadratic Assignment Problem*) pemecahan masalah lokasi dikerjakan dengan permutasi (perubahan secara berurutan).

a(i) merupakan vektor penugasan yang menunjukkan lokasi dimana stasiun kerja/departemen i ditempatkan. TC(a) merupakan total biaya dari penyelesaian permasalahan penugasan. Dalam membangun TC(a), arah perpindahan material/bahan tidak penting atau tidak berpengaruh, maka bobot antara departemen i dan j adalah $w_{ij} = r_{ij} + r_{ji}$. Jarak antara lokasi r dan s adalah d(r,s) dan matrik jarak adalah simetri. Dengan notasi ini, maka total biaya dari penyelesaian permasalahan penugasan untuk QAP (*Quadratic Assignment Problem*) adalah [FMW92]:

$$TC(a) = \sum_{1 \le i < j \le n} w_{ij}\, d(a(i),a(j)) \qquad (2.2)$$

Sedangkan total biaya untuk fasilitas k dari sebuah vektor penugasan untuk dilokasikan adalah [FMW92] :

$$P_k(a) = \sum_{j=1}^{n} w_{kj}\, d(a(k),a(j)) \qquad \text{dengan } w_{kk} = 0 \qquad (2.3)$$

Pertukaran pasangan dilakukan untuk mencari solusi yang lebih baik dalam penyelesaian tata letak fasilitas pada QAP (*Quadratic Assignment Problem*). Pertukaran ini dilakukan untuk mengevaluasi perubahan dalam total biaya penugasan. a' merupakan vektor penugasan dari pertukaran fasilitas u dan v dari vektor penugasan a, dimana fasilitas u akan dilokasikan pada tempat a(v) dan fasilitas v akan ditempatkan pada tempat a(u). Perubahan dalam total biaya penugasan ditunjukkan oleh $DTC_{uv}(a) = TC(a) - TC(a')$ [FMW92].

2.4.1 Construction Heuristics

Dalam membangun penyelesaian tata letak fasilitas dengan menggunakan procedure *construction heuristics* dimulai dari data masalah dasar dengan permutasi (perubahan yang berulang). Penugasan awal diberikan secara acak atau random. Pengacakan pemilihan fasilitas secara random menghasilkan penyelesaian yang berbeda, setiap kali *procedure Construction heuristics* dijalankan [FMW92].

2.4.2 Improvement Heuristics

Improvement heuristics digunakan untuk memodifikasi solusi/penyelesaian yang diberikan sehingga total biaya penugasan dapat berkurang. Modifikasi yang paling sederhana adalah dengan menukarkan antara 2 fasilitas/elemen. Pertukaran ini merupakan dasar untuk *Pairwise Interchange Improvement Heuristics*, yang mempunyai bentuk umum sebagai berikut [FMW92] :

```
LOOP
0. a ← a0
1. Pilih pasangan dari fasilitas (u,v)
2. Evaluasi nilai DTCuv(a)
3. Putuskan bisa atau tidak dilakukan pertukaran
4. Putuskan bisa atau tidak untuk dilanjutkan
ENDLOOP
```

a_0 merupakan penyelesaian inisialisasi yang dapat diperbaiki dan a adalah penyelesaian akhir. Metode yang digunakan untuk memilih pasangan dari fasilitas pada langkah 1, untuk menunjukkan satu per satu semua kemungkinan pasangan dari fasilitas, yaitu [FMW92] :

```
for i = 1 to n - 1
   for j = i + 1 to n
   endfor
endfor
```

Terdapat 3 cara untuk digunakan pada langkah ke-3 pada perulangan, yaitu [FMW92] :

1. Tukar u dan v bila $DTC_{uv}(a) > 0$ (pertukaran ini menggunakan perulangan *for* j).
2. Menukar u dan v selama $DTC_{uv}(a)$ adalah nilai yang paling besar untuk fasilitas u (menggunakan perulangan *for* i tanpa perulangan *for* j).
3. Evaluasi semua kemungkinan pasangan, dan pilih nilai $DTC_{uv}(a)$ yang paling besar.

2.5 Simulasi *Heuristics* untuk QAP (*Quadratic Assignment Problem*)

Gambaran untuk menunjukkan permasalahan tata letak fasilitas menggunakan QAP (*Quadratic Assignment Problem)* adalah dengan menukarkan pasangan. Dalam metode pendekatan simulasi ada 3 konsep, yaitu [FMW92] :

1. Parameter yang mengendalikan kemungkinan pertukaran penambahan biaya penugasan.
2. Penugasan dengan biaya yang lebih minimum merupakan perubahan yang penting.
3. Banyak pertukaran yang dilakukan untuk mendapatkan biaya penugasan yang paling minimum dilakukan sampai tidak ada lagi kemungkinan penyelesaian yang lebih baik dalam setiap iterasi dengan batasan maksimum iterasi.

BAB III

ANALISIS KEBUTUHAN SISTEM

3.1 Metode Analisis

Sistem yang akan dirancang dan digunakan dalam penyelesaian tata letak fasilitas dirancang dengan menggunakan pendekatan terstruktur (*Structured Approach*). Selain itu bersifat GUI (*Graphical User Interface)* dan memiliki komponen-komponen yang mudah dalam pemakaiannya. Pada tahap transformasi input, proses, dan output dinyatakan dengan bagan alir (*flowchart*) sebagai pernyataan algoritma dalam menentukan perhitungan total biaya penugasan dan penyelesaian/penentuan tata letak (posisi) fasilitas.

3.2 Hasil Analisis Kebutuhan

Berdasarkan analisis yang telah dilakukan maka dapat diketahui apa saja yang akan menjadi masukan sistem, keluaran sistem, metode yang digunakan oleh sistem, kebutuhan perangkat keras, kebutuhan perangkat lunak serta antarmuka sistem yang akan dibuat, sehingga sistem yang dibangun sesuai dengan apa yang diharapkan.

3.2.1 Masukan Sistem

Masukan data untuk perangkat lunak pada penyelesaian tata letak fasilitas (*facility layout*) dengan menggunakan formula QAP (*Quadratic Assignment Sistem*) adalah sebagai berikut :

1. Jumlah fasilitas/departemen.
2. Nilai jarak antar lokasi yang disimpan dalam matrik jarak (d).
3. Nilai aliran bahan antar departemen dan disimpan dalam suatu matrik (r).
4. Maksimum iterasi : jumlah maksimum iterasi yang dilakukan untuk mencari total biaya penugasan yang paling minimum.
5. Maksimum cacah : jumlah maksimum cacah yang dilakukan untuk mencari biaya penugasan yang paling minimum pada metode *construction heuristics*.

3.2.2 Keluaran Sistem

Keluaran dari perangkat lunak penyelesaian tata letak fasilitas dengan menggunakan formula QAP (*Quadratic Assignment Problem*) adalah sebagai berikut :

1. Laporan hasil proses pelatihan.
2. Laporan perbandingan metode.
3. Grafik nilai maksimum dari selisih nilai diantara penugasan pada setiap iterasi.
4. Grafik total biaya penugasan minimum.
5. Pemetaan lokasi tata letak fasilitas.

3.2.3 Kebutuhan Fungsi

Fungsi yang digunakan untuk menyelesaikan masalah tata letak fasilitas agar mendapat tata letak fasilitas dengan total biaya yang paling minimum menggunakan formula QAP (*Quadratic Assignment Problem*) untuk menghitung total biaya penugasan dan menggunakan metode *Construction heuristics* dan *Improvement heuristics* untuk menentukan tata letak fasilitas dengan total biaya yang paling minimum.

3.2.4 Kebutuhan Perangkat Keras

Perangkat keras yang digunakan dalam penyelesaian tata letak fasilitas dengan menggunakan formula QAP (*Quadratic Assignment Problem*) minimal memiliki spesifikasi sebagai berikut :

1. Satu unit komputer dengan spesifikasi minimum *Processor Pentium* II 350 MHz, RAM 64 MB dan *HardDisk* 500 MB.
2. Monitor VGA atau SVGA atau yang lebih tinggi.
3. *Mouse*.
4. *Keyboard*.
5. Satu unit *printer dot* matrik, *deskjet* atau laser jet atau yang lebih tinggi.

3.2.5 Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan untuk pengembangan dan implementasi perangkat lunak (*software*) untuk penyelesaian tata letak fasilitas ini menggunakan :

1. Sistem operasi *Windows* 98, 2000 atau XP.
2. Bahasa pemrograman Borland Delphi 6.0 untuk pembuatan perangkat lunak.

3.2.6 Antarmuka Sistem

Sistem penyelesaian tata letak fasilitas akan diakses oleh pengguna (*user*) dengan tingkat kemampuan menggunakan komputer yang berbeda. Untuk itu diperlukan antarmuka (*interface*) yang dapat dimengerti dan mudah digunakan oleh pengguna dengan berbagai tingkat kemampuan dalam menggunakan komputer. Kriteria tersebut dapat dipenuhi dengan menggunakan antarmuka (*interface*) berbasis grafis yang lebih mudah dimengerti oleh banyak pengguna.

3.2.7 Kinerja yang diharapkan

Kinerja yang diharapkan dari hasil analisis di atas adalah perangkat lunak yang dibangun mampu menangani berbagai kemungkinan masukan data sehingga fleksibel untuk digunakan dan untuk mengetahui tata letak fasilitas yang mempunyai total biaya penugasan yang paling minimum. Jumlah data yang dapat diolah perangkat lunak ini tidak dibatasi tetapi hanya dibatasi oleh kemampuan perangkat lunak maupun perangkat keras (*Processor*, RAM dan lai-lain) dan berupa bilangan bulat maupun desimal.

BAB IV

PERANCANGAN PERANGKAT LUNAK

4.1 Metode Perancangan

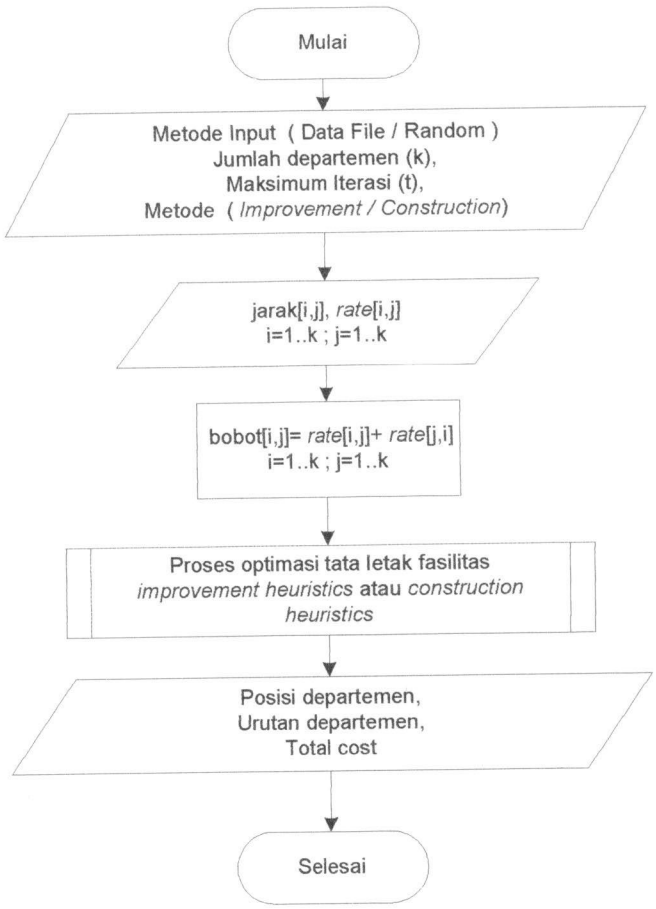
Metode perancangan dalam pembuatan aplikasi ini menggunakan metode berarah alir data (*flowchart*).

4.2 Hasil Perancangan

Hasil pada tahap perancangan berkaitan erat dengan hasil tahap analisis. Karena pada tahap analisis telah ditentukan metode, fungsi-fungsi yang digunakan, perangkat lunak yang dipakai, serta antarmuka yang diharapkan. Dari hasil tahap analisis tersebut didapat suatu gambaran tentang optimasi Penyelesaian Tata Letak Fasilitas (*facility layout*) dengan Formula QAP (*Quadratic Assignment Problem*).

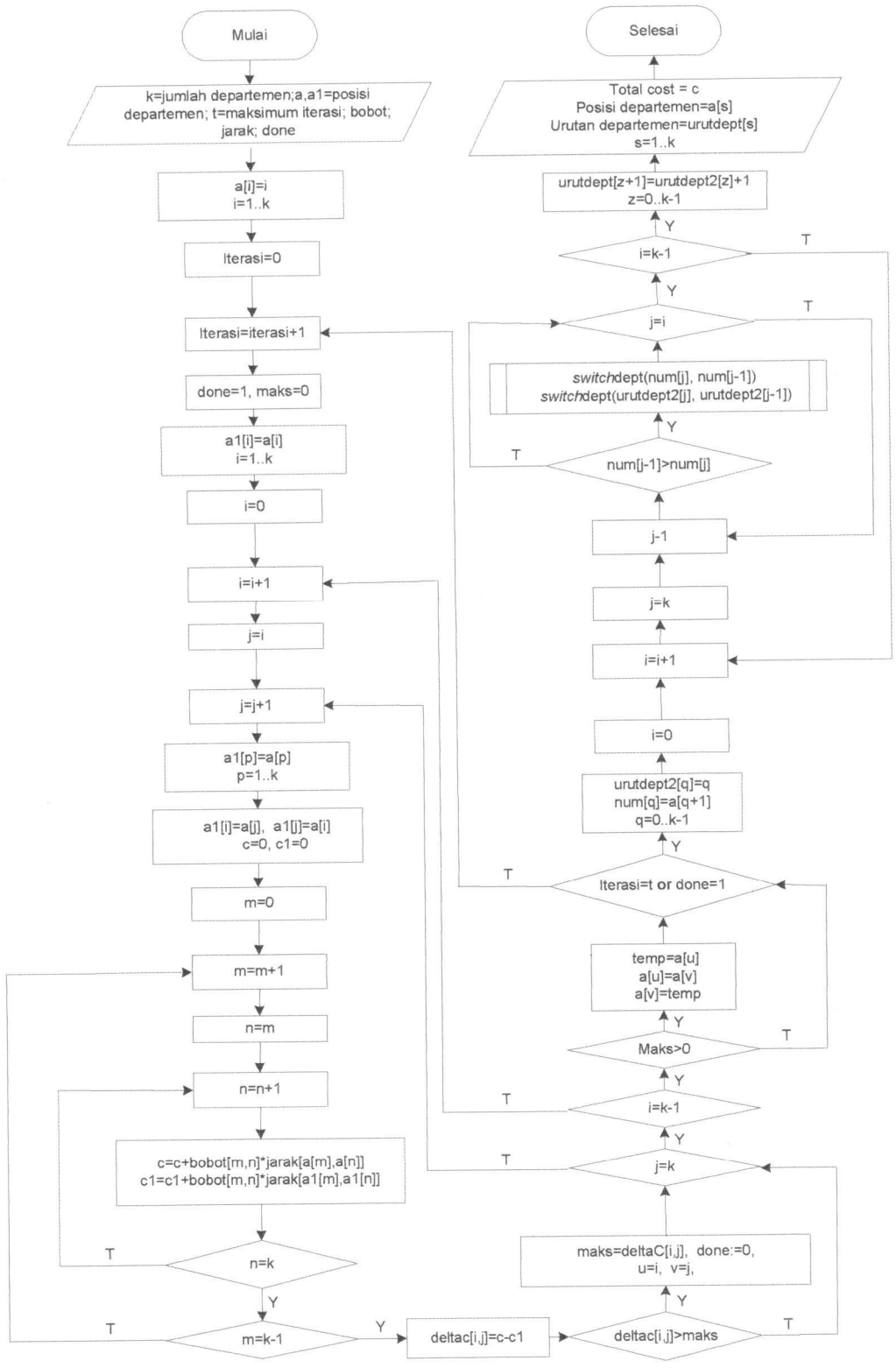
4.2.1 Perancangan Diagram Alir Sistem

Pada subbab ini, sistem akan digambarkan sebagai sebuah diagram alir (*flowchart*) secara keseluruhan. Gambar (4.1) sampai gambar (4.5) merupakan diagram alir perangkat lunak optimasi Tata Letak Fasilitas dengan menggunakan formula QAP (*Quadratic Assignment Problem*) serta diagram alir penjabarannya.

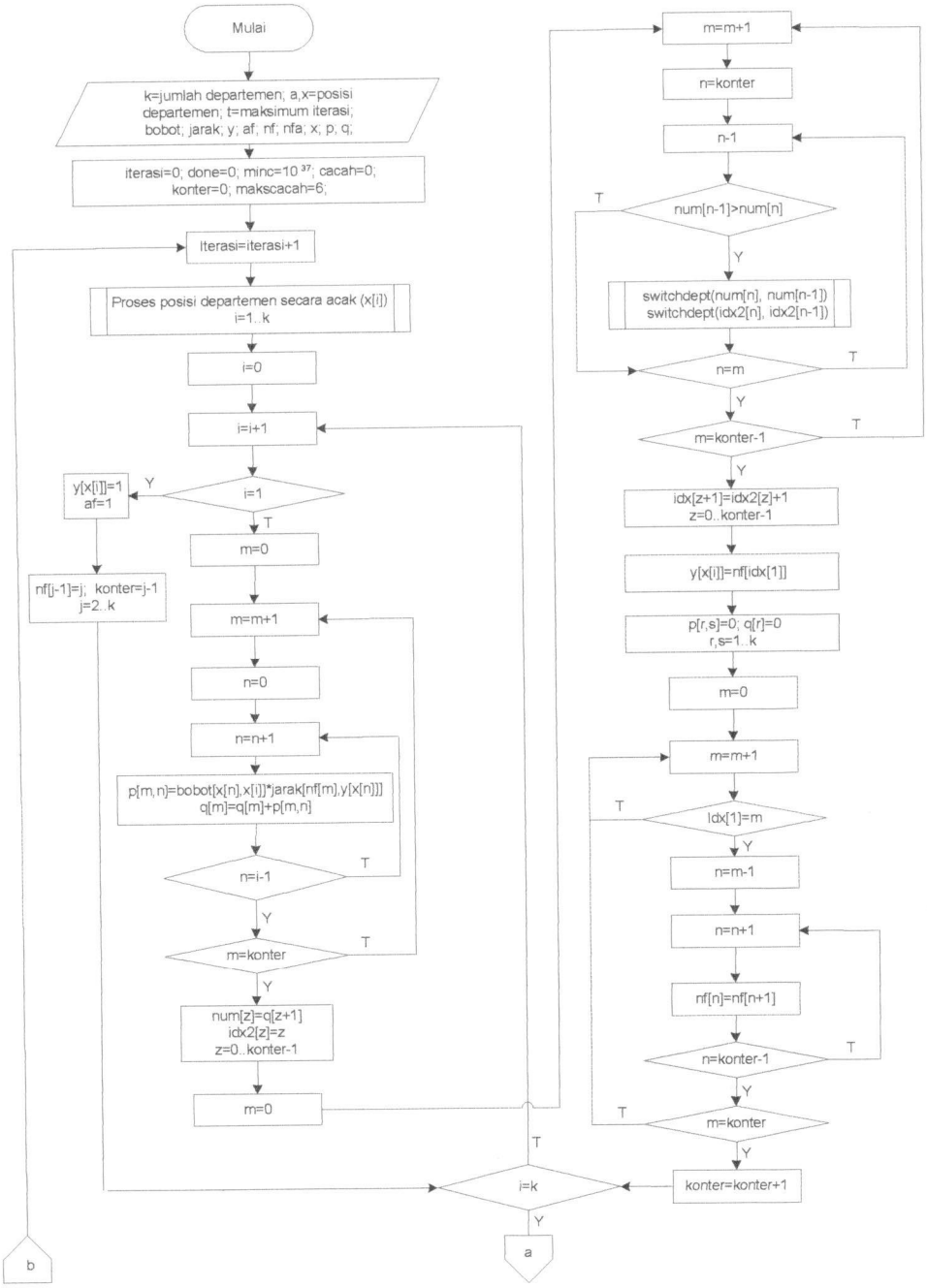


Gambar 4.1 Bagan alir perangkat lunak untuk optimasi tata letak fasilitas

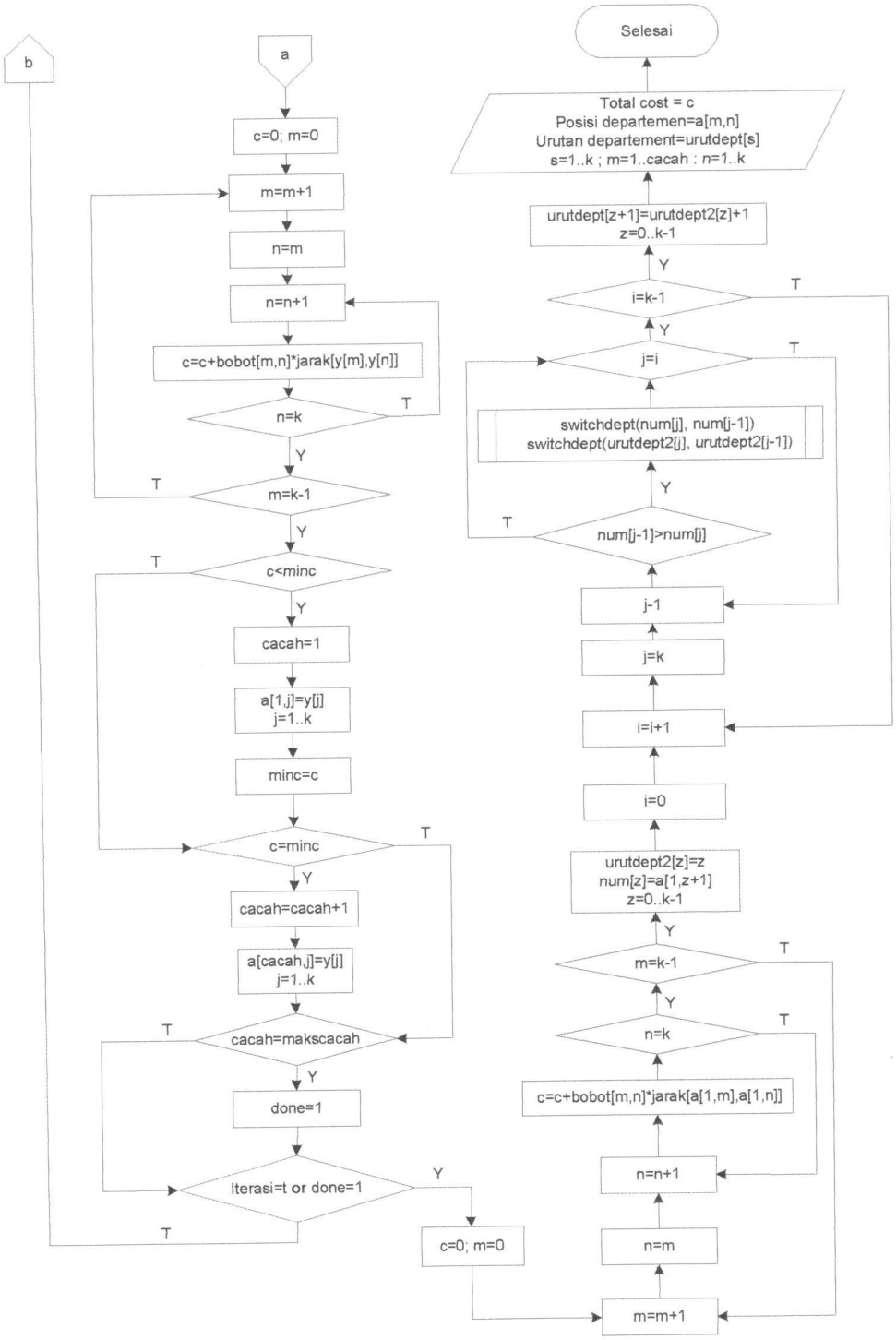
Pada penginputan melalui *data file,* dapat dilakukan pemanggilan data (ambil data) atau pemasukan data (data baru) berupa nilai jarak antar lokasi dan nilai aliran perpindahan bahan antar fasilitas/departemen. Sedangkan pada penginputan secara random, nilai jarak antar lokasi dan nilai aliran perpindahan bahan antar fasilitas/departemen akan diinputkan secara acak. Dari diagram alir (*flowchart*) di atas akan dijabarkan pemrosesan optimasi Tata Letak Fasilitas dengan formula QAP (*Quadratic Assignment Problem*) menggunakan metode *improvement heuristics* dan *construction heuristics*, sebagai berikut :



Gambar 4.2 Bagan alir metode *improvement heuristics*

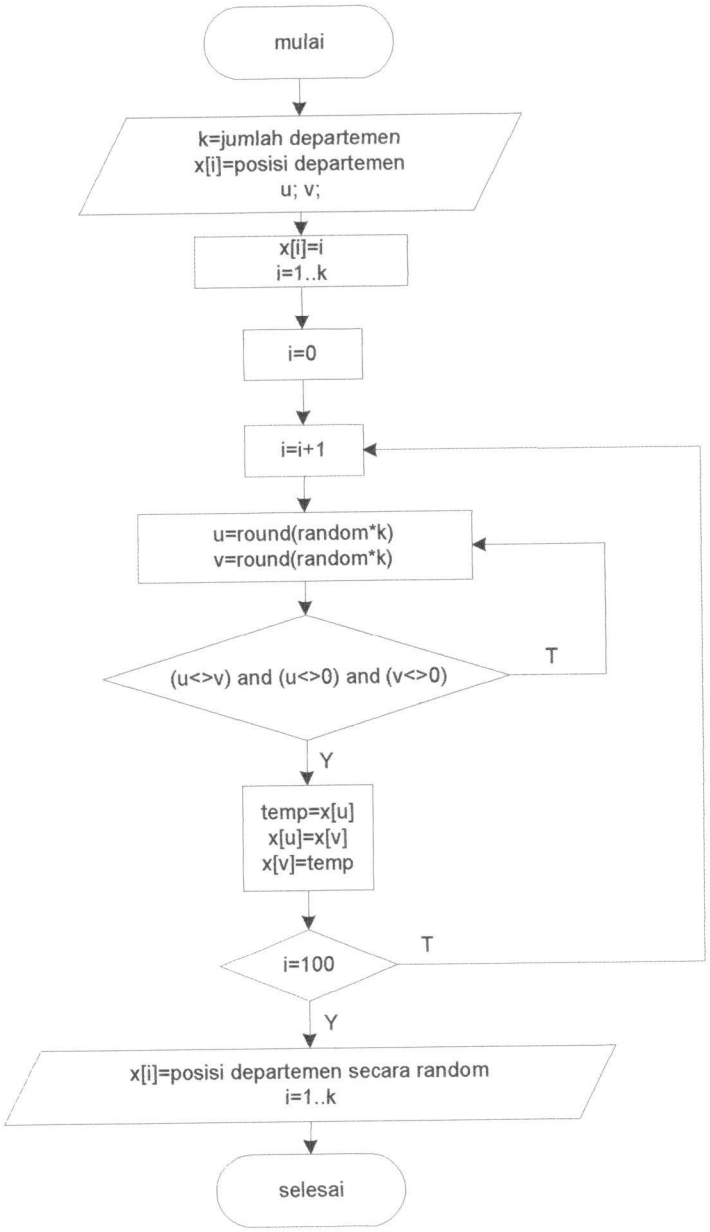


Gambar 4.3(a) Bagan alir metode *construction heuristics*



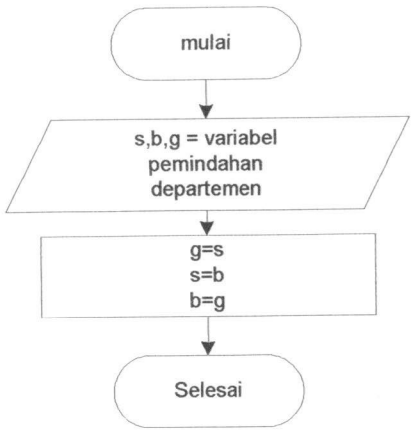
Gambar 4.3(b) Bagan alir metode *construction heuristics*

Dari diagram alir (*flowchart*) optimasi Tata Letak Fasilitas dengan menggunakan metode *construction heuristics* Gambar 4.3(a), pada posisi departemen/fasilitas secara acak dapat dijabarkan ke dalam diagram alir (*flowchart*) sebagai berikut :



Gambar 4.4 Bagan alir posisi departemen secara acak

Dari diagram alir (*flowchart*) metode *improvement heuristics* Gambar (4.2) dan diagram alir (*flowchart*) metode *construction heuristics* Gambar 4.3(a) dan gambar 4.3(b), pada *switchdept* yang merupakan salah satu prosedur untuk pengurutan departemen/fasilitas, dapat dijabarkan ke dalam diagram alir (*flowchart*) sebagai berikut :



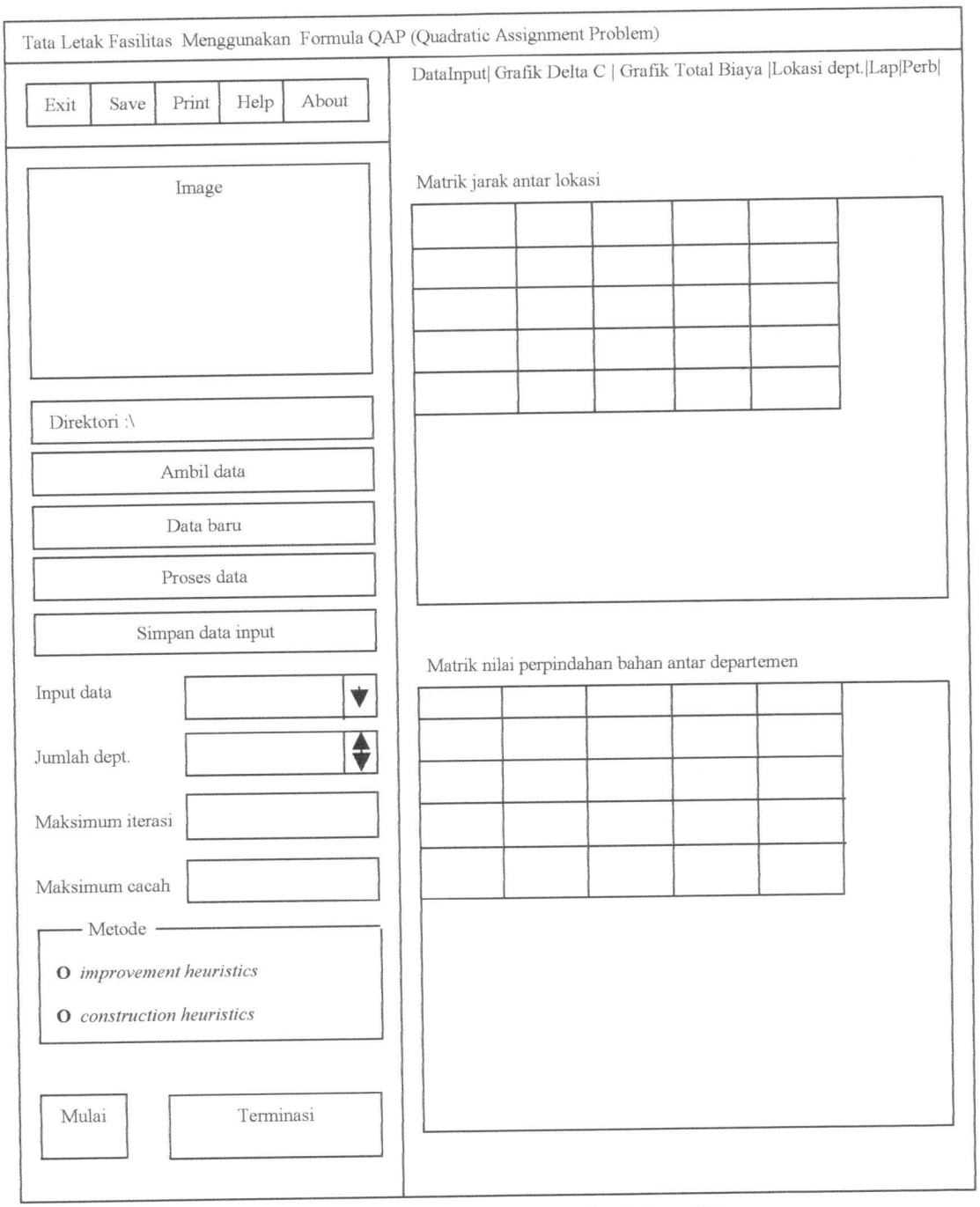
Gambar 4.5 Bagan alir pemindahan posisi departemen untuk diurutkan

4.2.2 Perancangan Antarmuka

Rancangan antarmuka dari perangkat lunak untuk optimasi Tata Letak fasilitas dengan formula QAP (*Quadratic Assignment Problem*) ini menggunakan perancangan model grafis. Terdapat beberapa *form* terpisah yaitu *form* menu utama, *form loading*, *form help*, dan lain-lain. *Form* menu utama terdiri dari input data, tampilan grafik, dan laporan hasil pelatihan.

4.2.2.1 Rancangan Antarmuka Menu Utama

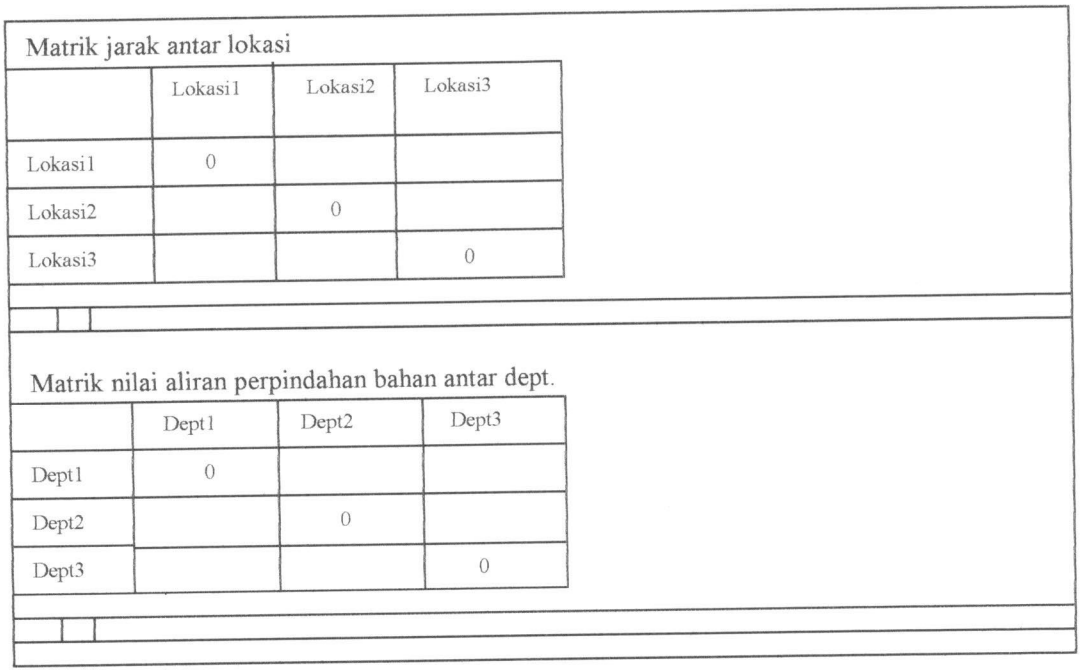
Antarmuka ini merupakan form utama yang menghubungkan dengan form-form lain dan merupakan antarmuka untuk memulai proses pelatihan. Gambar (4.6) mengilustrasikan perancangan antarmuka menu utama.



Gambar 4.6 Rancangan Antarmuka Menu Utama

4.2.2.2 Rancangan Antarmuka *Input* Data

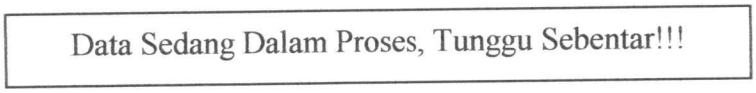
Form ini digunakan untuk memasukkan nilai jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen. Gambar (4.7) adalah ilustrasi dari rancangan antarmuka *input* data.



Gambar 4.7 Rancangan Antarmuka *Input* Data

4.2.2.3 Rancangan Antarmuka *Loading* Data

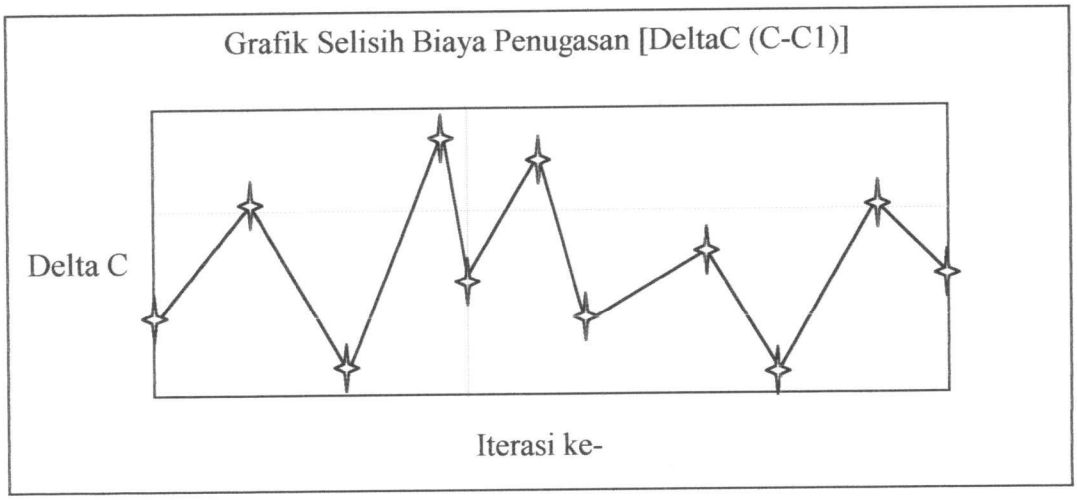
Form ini akan muncul pada saat hasil pelatihan data sedang diproses untuk ditampilkan pada menu laporan (*report*). *Form* ini akan muncul apabila data yang diproses besar. Gambar (4.8) adalah ilustrasi dari rancangan antarmuka *loading* data.



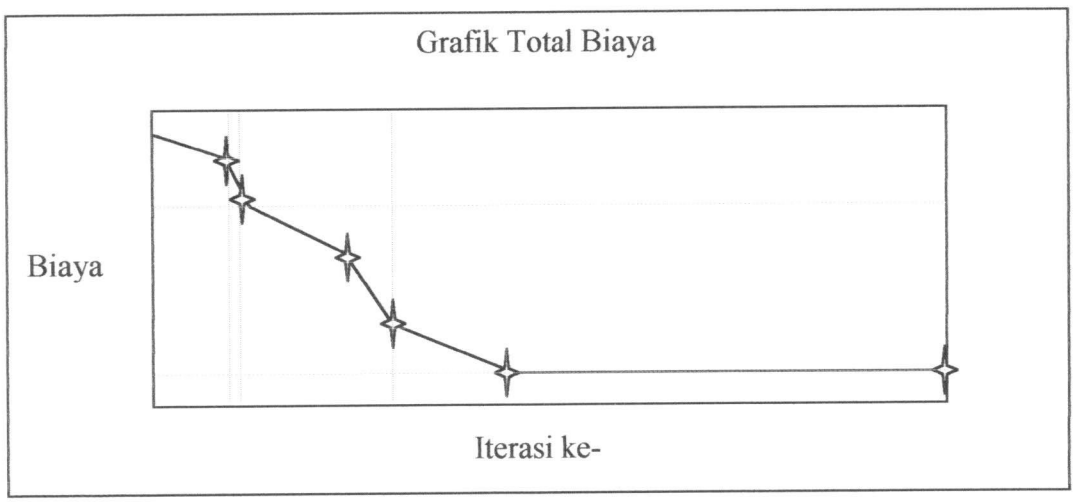
Gambar 4.8 Rancangan Antarmuka *Loading* Data

4.2.2.4 Rancangan Antarmuka Grafik

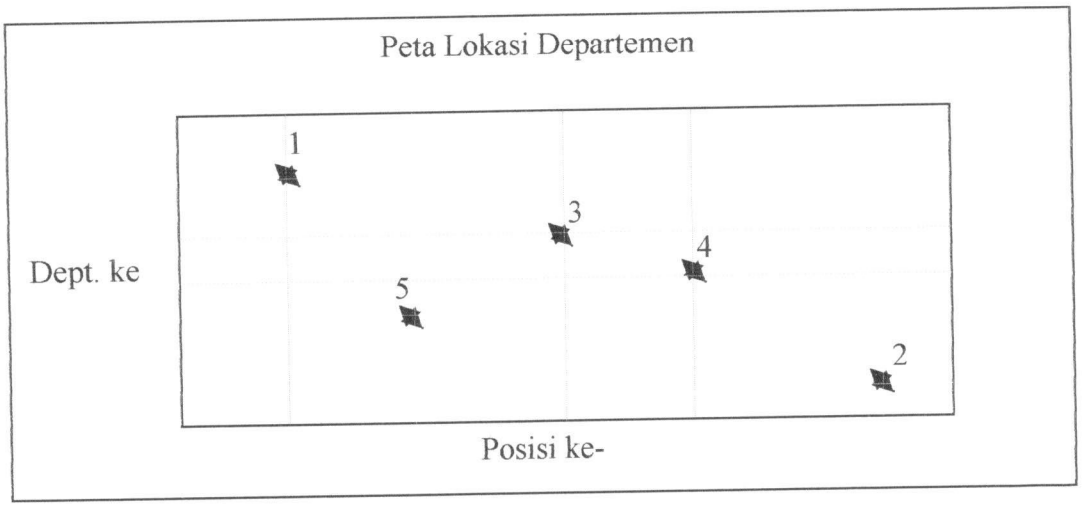
Form grafik digunakan untuk menampilkan informasi hasil pelatihan untuk nilai selisih biaya penugasan, nilai total biaya, dan peta lokasi departemen/fasilitas. Dalam bentuk grafik. Gambar (4.9), gambar (4.10), dan gambar (4.11) adalah gambar rancangan antarmuka grafik.



Gambar 4.9 Rancangan Antarmuka Grafik Selisih Biaya Penugasan



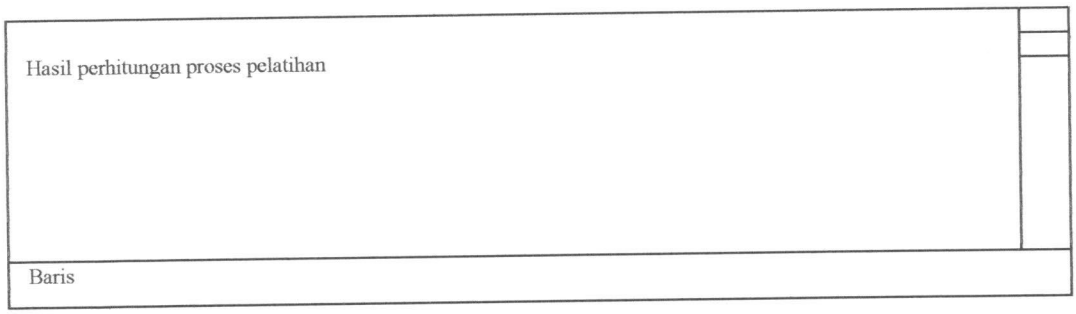
Gambar 4.10 Rancangan Antarmuka Grafik Total Biaya



Gambar 4.11 Rancangan Antarmuka Peta Lokasi Departemen

4.2.2.5 Rancangan Antarmuka Laporan (*Report*)

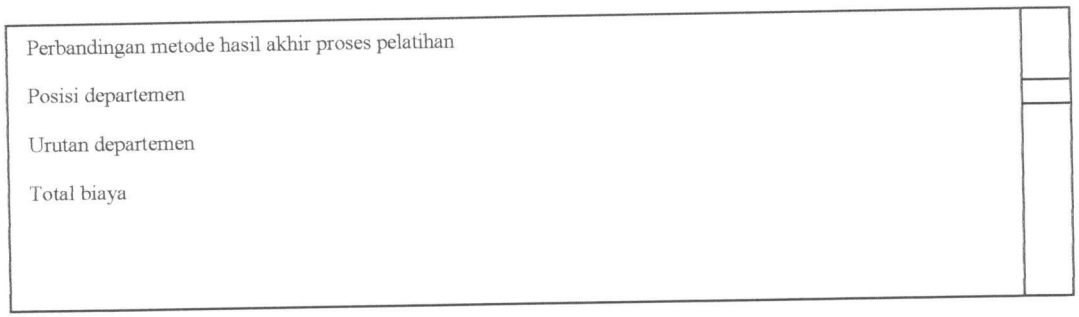
Form ini digunakan untuk menampilkan informasi hasil proses pelatihan. Gambar (4.12) adalah rancangan antarmuka laporan (*report*).



Gambar 4.12 Rancangan Antarmuka Laporan

4.2.2.6 Rancangan Antarmuka Perbandingan Metode

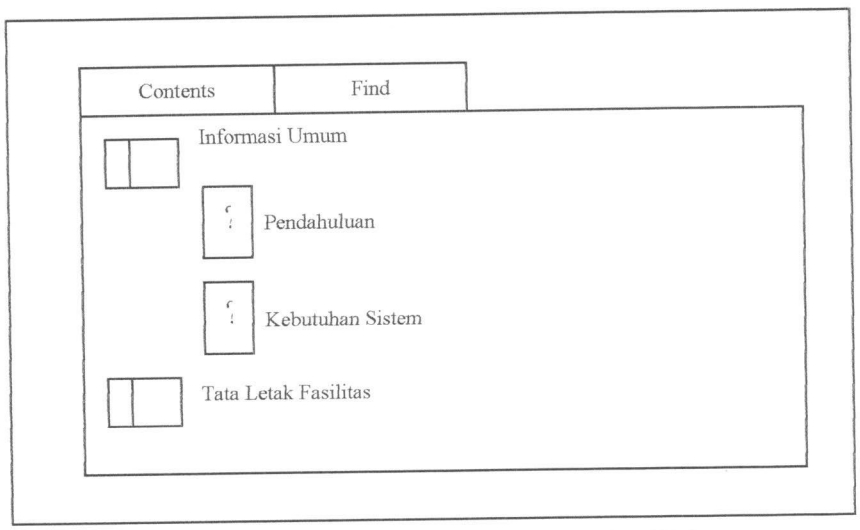
Form ini digunakan untuk menampilkan perbandingan metode dari hasil akhir proses pelatihan dengan data yang sama. Gambar (4.13) adalah rancangan antarmuka hasil perbandingan.



Gambar 4.13 Rancangan Antarmuka Hasil Perbandingan

4.2.2.7 Rancangan Antarmuka *Help*

Form ini digunakan untuk membantu user dalam menggunakan sistem dan beberapa informasi lainnya yang berhubungan dengan sistem. Gambar (4.14) adalah rancangan antarmuka *help*.



Gambar 4.14 Rancangan Antarmuka *Help*

4.2.3 Perancangan *File*

4.2.3.1 Perancangan *Input*

Pada bagian *input* data, *file* yang digunakan sebagai masukan *file* adalah *file* yang berekstensi .dat yang diambil dari *file* yang telah disimpan sebelumnya. Semua jenis data seperti jarak antar lokasi, nilai aliran perpindahan bahan, jumlah departemen, dan maksimum iterasi dimasukkan pada sebuah parameter yang telah ditetapkan.

4.2.3.2 Perancangan Proses

Pada bagian proses data, hasilnya akan disimulasikan dan didokumentasikan dalam bentuk grafik maupun informasi tertulis, data yang diproses adalah semua data yang telah dimasukkan pada parameter yang telah ditetapkan beserta metodenya.

4.2.3.3 Perancangan *Output*

Pada bagian *output* data, *file* yang digunakan untuk menyimpan hasil dari proses pelatihan adalah *file* teks dengan ekstensi .txt untuk menyimpan informasi hasil pemrosesan, dan *file* bitmap dengan ekstensi .bmp untuk penyimpanan grafik hasil proses pelatihan.

BAB V

IMPLEMENTASI PERANGKAT LUNAK

5.1 Implementasi

Tujuan dari tahap implementasi ini untuk memastikan perangkat lunak yang dibuat dapat bekerja secara efektif dan efisien sesuai yang diinginkan, untuk itu pada tahap implementasi ini akan menjelaskan, antara lain, bahasa pemrograman yang dipakai, lingkungan pengembangan dan uji coba terhadap perangkat lunak. Berikut implementasi dari sistem perangkat lunak untuk optimasi Penyelesaian Tata Letak Fasilitas (*facility layout*) dengan Menggunakan Formula QAP (*Quadratic Assignment Problem*).

5.1.1 Pemilihan Bahasa Pemrograman

Sistem perangkat lunak yang akan dibuat adalah sebuah perangkat lunak yang berisi banyak perhitungan, peraturan logika, pengembangan visual, pola desain dan pemakaian pada lingkup kerjanya, dan fleksibilitas. Maka dibutuhkan sebuah bahasa pemrograman yang terstruktur, handal, praktis, mudah digunakan dan juga mendukung batasan sistem operasi secara umum. Borland Delphi 6.0 adalah salah satu bahasa pemrograman yang memiliki semua persyaratan yang dibutuhkan dalam proses pembuatan perangkat lunak ini.

5.1.2 Implementasi Antarmuka

Berikut ini adalah gambaran antarmuka-antarmuka yang terdapat pada sistem perangkat lunak untuk optimasi Penyelesaian Tata Letak Fasilitas (*facility layout*) dengan Menggunakan Formula QAP (*Quadratic Assignment Problem*).

5.1.2.1 Antarmuka Menu Utama

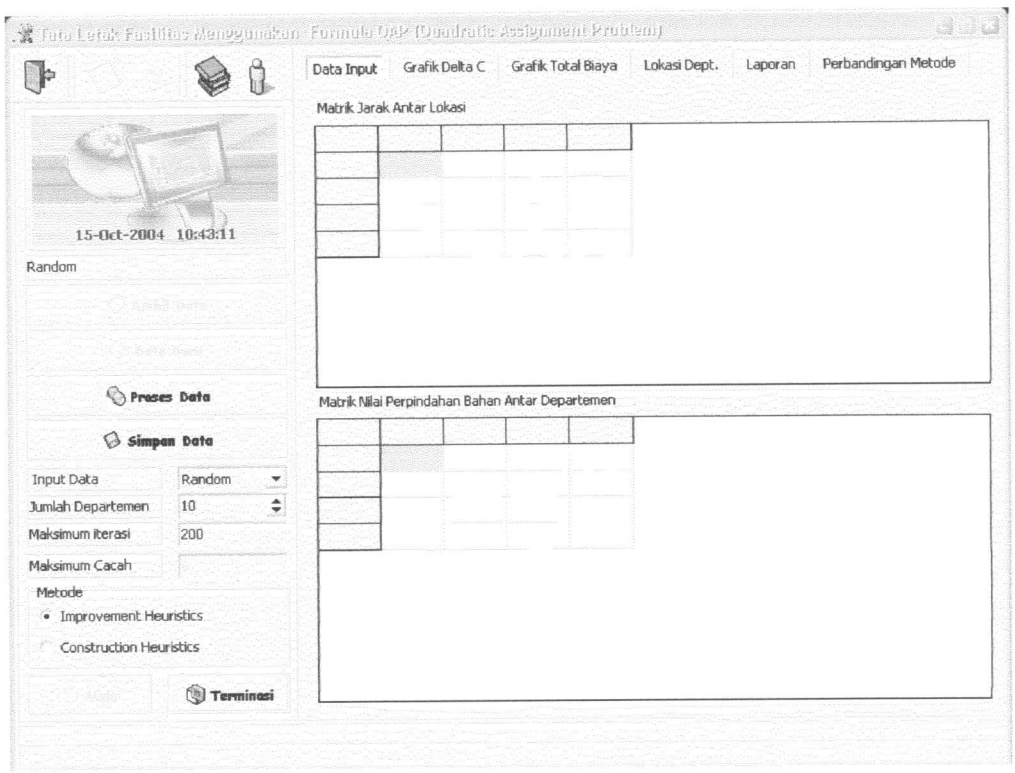
Antarmuka ini merupakan menu utama untuk melakukan pelatihan dan simulasi. Terdapat beberapa komponen pada antarmuka ini :

1. Menu Utama : terdiri dari beberapa menu diantaranya adalah :
   a. Data Input : untuk menampilkan input data yang akan diproses.
   b. Grafik Delta C : untuk mensimulasikan pergerakan nilai selisih biaya penugasan pada saat proses pelatihan berlangsung.
   c. Grafik Total Biaya : untuk mensimulasikan pergerakan nilai total biaya penugasan pada saat proses pelatihan berlangsung.
   d. Lokasi dept. : untuk menampilkan peta lokasi departemen dari hasil akhir pelatihan
   e. Laporan : untuk menampilkan informasi hasil pelatihan.
   f. Perbandingan metode : untuk membandingkan metode yang digunakan dengan menampilkan informasi hasil akhir dari proses pelatihan.
2. Tombol Ambil Data : untuk pemanggilan dan mengedit *file* yang berisi data jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen yang pernah dilakukan penyimpanan sebelumnya.

3. Tombol Data Baru : untuk membuka *form* data input untuk memasukkan data jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen.
4. Tombol Proses Data : untuk memasukkan data jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen secara random.
5. Tombol Simpan Data : untuk menyimpan data jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen yang ada pada form data input.
6. Input data : merupakan cara penginputan nilai jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen, yang terdiri dari *Random* dan Data *file*.
7. Jumlah departemen, Maksimum iterasi, Maksimum cacah : merupakan tempat penginputan parameter pelatihan yang digunakan.
8. Metode : merupakan metode yang digunakan dalam proses pelatihan, yang terdiri dari *improvement heuristics* dan *construction heuristics*.
9. Tombol Mulai dan Berhenti : Mulai, untuk memulai atau melanjutkan proses pelatihan. Sedangkan Berhenti, untuk menghentikan sementara proses pelatihan.
10. Tombol Terminasi : untuk menghentikan proses pelatihan.
11. *Icon Exit* : untuk keluar dari sistem.
12. *Icon Save* : untuk menyimpan hasil proses pelatihan yang terdiri dari grafik selisih biaya penugasan, grafik biaya penugasan, dan peta lokasi departemen/fasilitas dengan ekstensi .bmp serta laporan proses pelatihan dan perbandingan metode dengan ekstensi .txt.

13. *Icon Print* : untuk mendokumentasikan hasil *output* dari proses pelatihan yang terdiri dari grafik selisih biaya penugasan, grafik biaya penugasan, peta lokasi departemen/fasilitas, laporan perhitungan, dan perbandingan metode dalam bentuk *hard copy*.
14. *Icon Help* : untuk pemanggilan antarmuka pertolongan seputar penggunaan sistem.
15. *Icon About* : untuk pemanggilan antarmuka tentang pembuat sistem.

Gambar (5.1) adalah gambar antarmuka menu utama

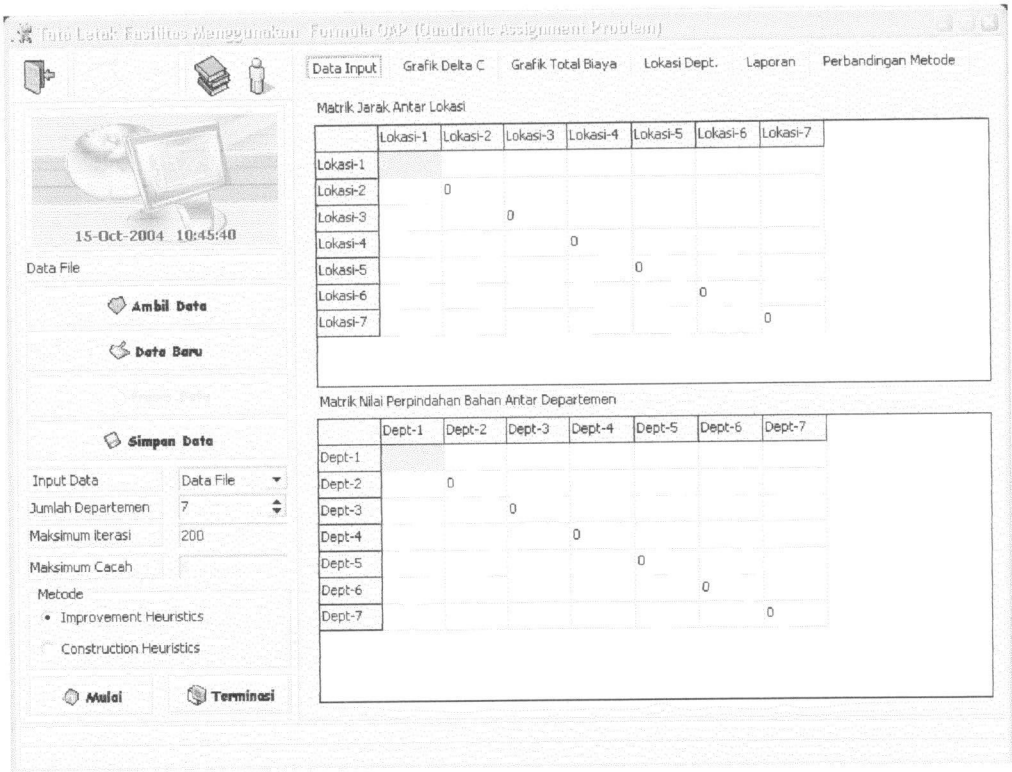


Gambar 5.1 Antarmuka Menu Utama

MILIK
PERPUSTAKAAN-FTI-UII
YOGYAKARTA

5.1.2.2 Antarmuka Data Input / Data Baru

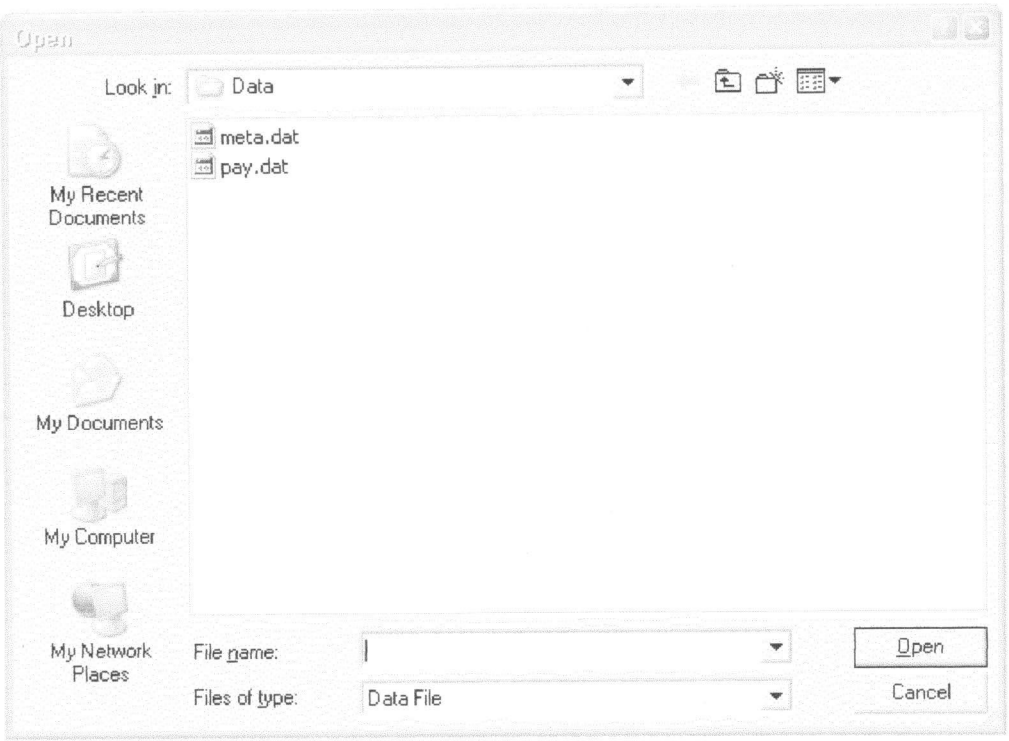
Antarmuka ini digunakan untuk membuat atau melihat data input. Pada antarmuka ini terlihat menyatu dengan antarmuka menu utama dan terdapat komponen *StringGrid* yang digunakan untuk memasukkan nilai jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen/fasilitas sebagai data input. Gambar (5.2) adalah gambar dari antarmuka Data Input.



Gambar 5.2 Antarmuka Input Data / Data Baru

5.1.2.3 Antarmuka Ambil Data

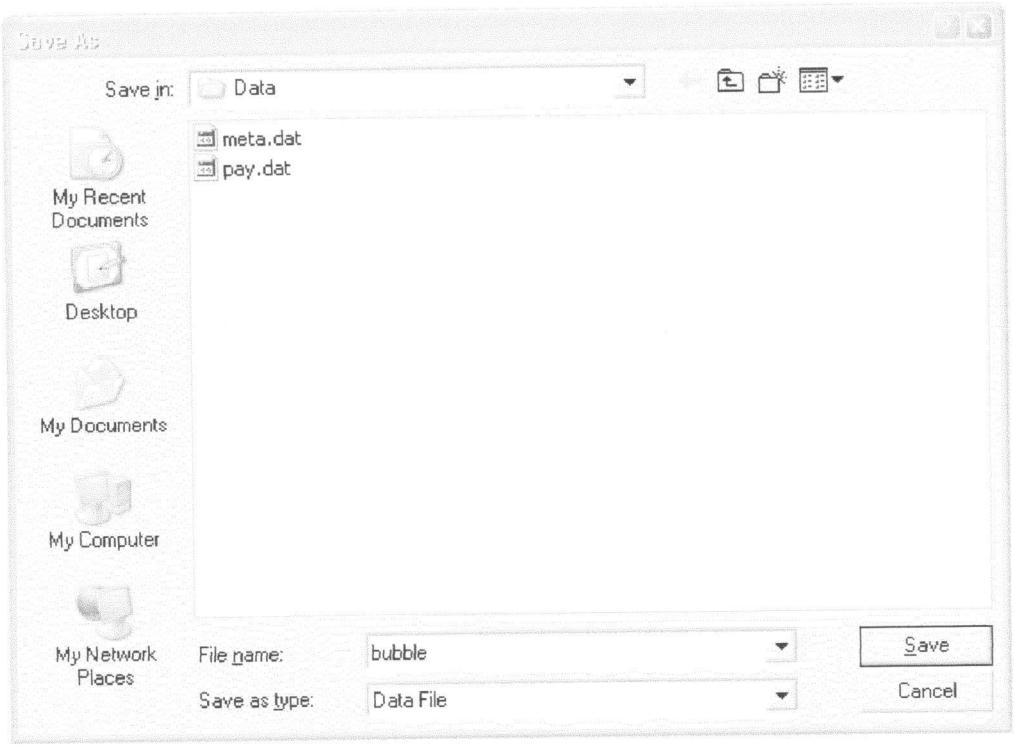
Antarmuka ini digunakan untuk mengambil data jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen yang sebelumnya sudah pernah disimpan. Gambar (5.3) adalah antarmuka Ambil Data (*load data*).



Gambar 5.3 Antarmuka Ambil Data

5.1.2.4 Antarmuka Simpan Data Input

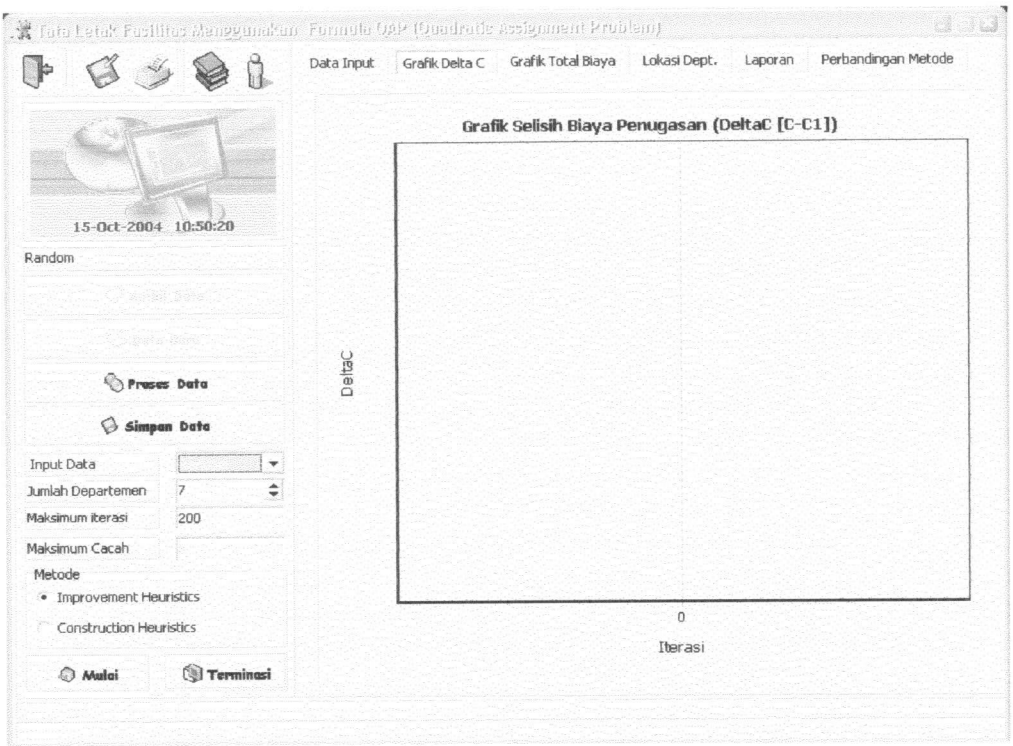
Antarmuka ini digunakan untuk menyimpan data *input* jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen. Gambar (5.4) adalah antarmuka Simpan Data Input.



Gambar 5.4 Antarmuka Simpan Data Input

5.1.2.5 Antarmuka Grafik Selisih Biaya Penugasan

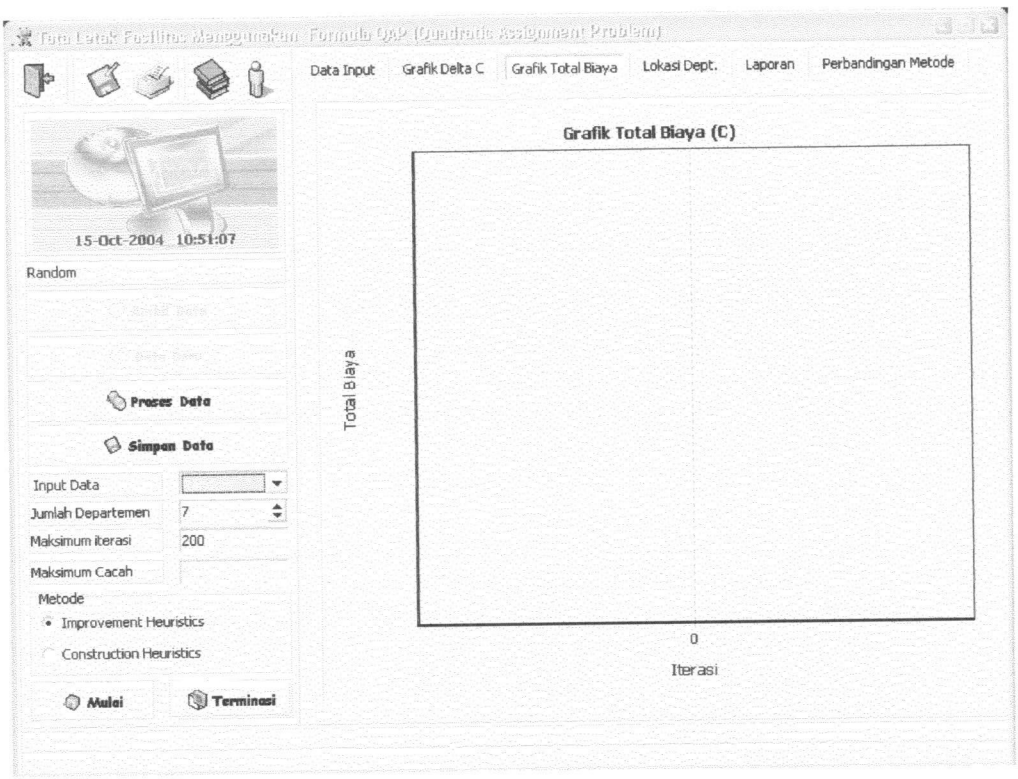
Antarmuka ini untuk menampilkan informasi proses pelatihan dalam bentuk grafik pergerakan dari selisih biaya penugasan. Pada antarmuka ini terlihat menyatu dengan antarmuka menu utama. Gambar (5.5) adalah gambar antarmuka grafik untuk selisih biaya penugasan (Delta C).



Gambar 5.5 Antarmuka Grafik Selisih Biaya Penugasan

5.1.2.6 Antarmuka Grafik Biaya Penugasan

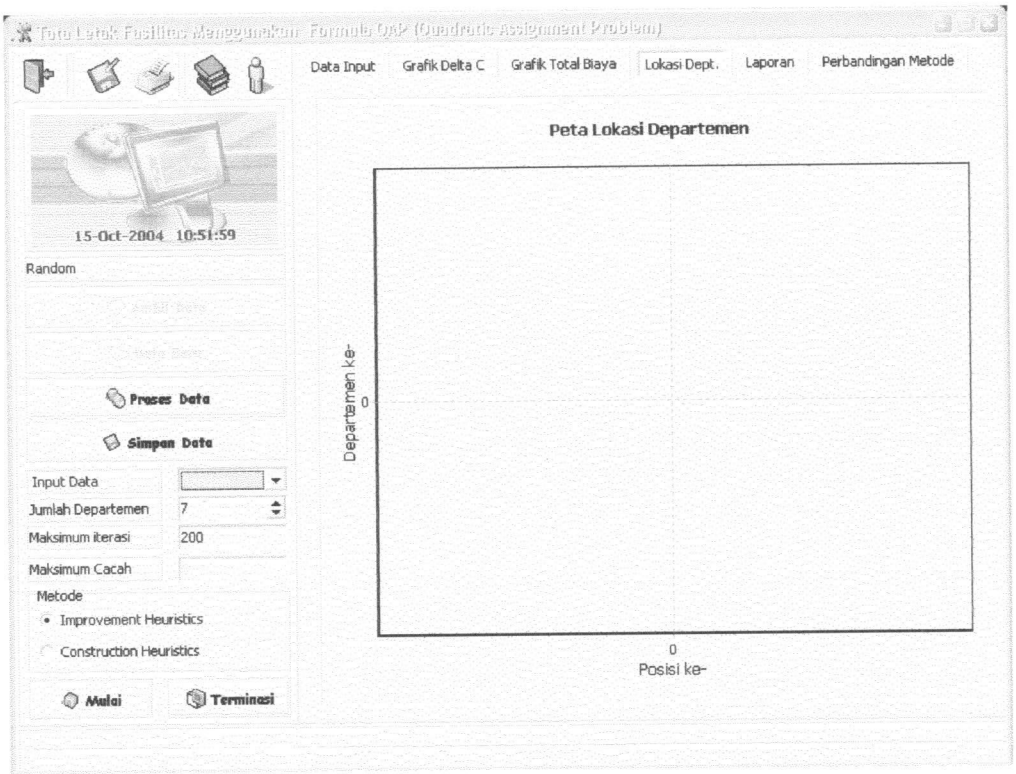
Antarmuka ini untuk menampilkan informasi proses pelatihan dalam bentuk grafik pergerakan biaya penugasan. Pada antarmuka ini terlihat menyatu dengan antarmuka menu utama. Gambar (5.6) adalah gambar antarmuka grafik untuk biaya penugasan.



Gambar 5.6 Antarmuka Grafik Biaya Penugasan

5.1.2.7 Antarmuka Peta Lokasi Departemen / Fasilitas

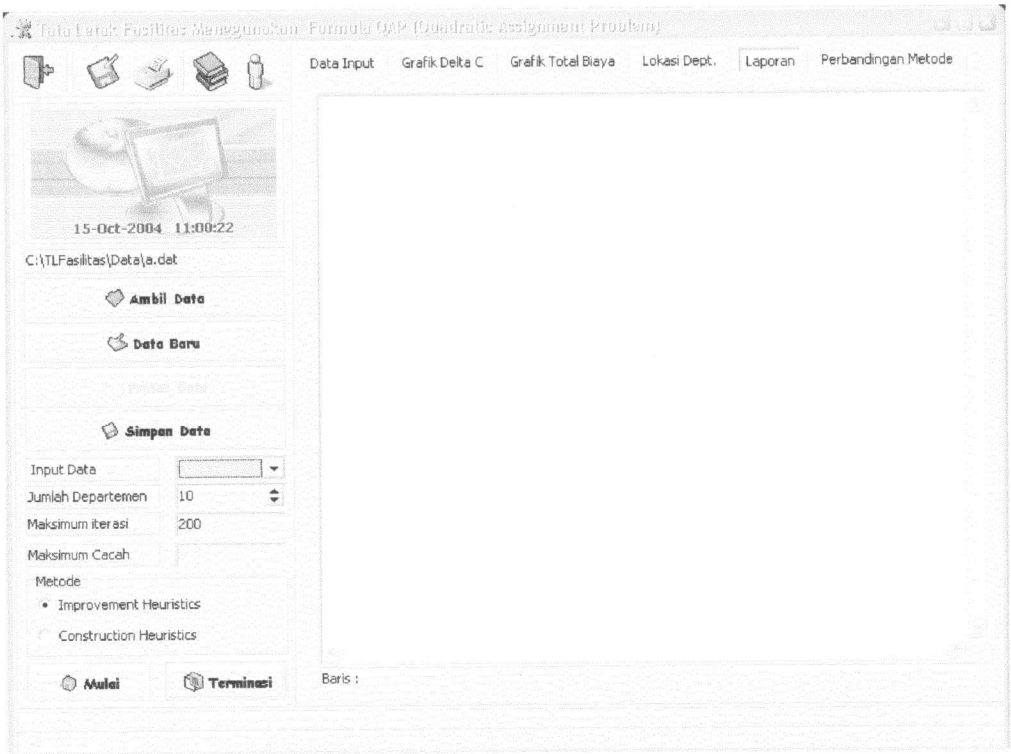
Antarmuka ini menampilkan grafik peta lokasi tata letak fasilitas dari hasil akhir pada proses pelatihan. Gambar (5.7) adalah gambar grafik pemetaan lokasi departemen/fasilitas.



Gambar 5.7 Antarmuka Grafik Pemetaan Lokasi Departemen/Fasilitas

5.1.2.8 Antarmuka Laporan (*Report*)

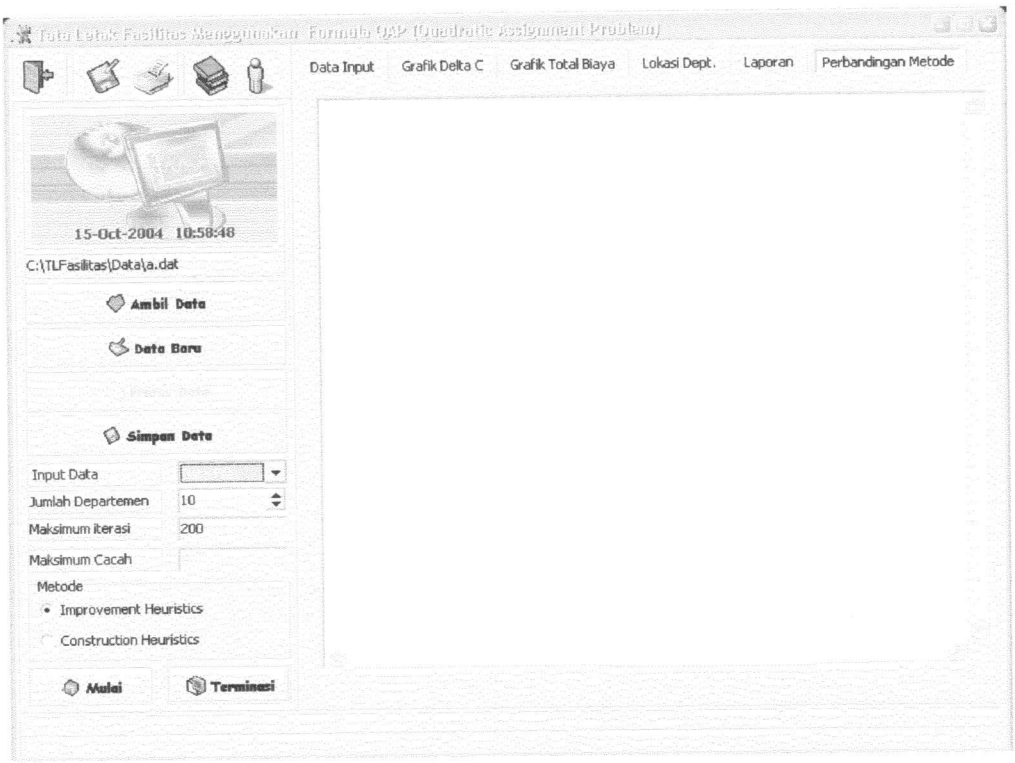
Antarmuka ini untuk menampilkan hasil perhitungan biaya penugasan dan alternatif penugasan pada setiap iterasi dari proses pelatihan. Gambar (5.8) adalah gambar antarmuka laporan/*report*.



Gambar 5.8 Antarmuka Laporan/*Report*

5.1.2.9 Antarmuka Perbandingan Metode

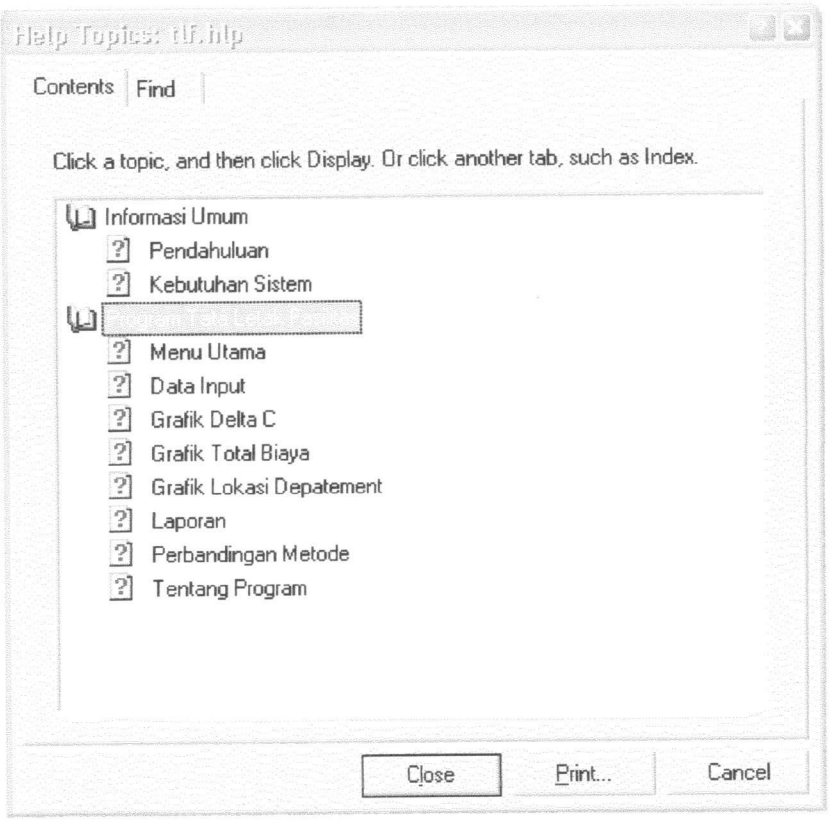
Antarmuka ini menampilkan alternatif penugasan yang terbaik dengan total biaya penugasan pada 2 metode yang digunakan, yaitu *improvement heuristics* dan *construction heuristics* dari proses pelatihan. Gambar 5.9 adalah gambar antarmuka laporan perbandingan metode.



Gambar 5.9 Antarmuka Laporan Perbandingan Metode

5.1.2.10 Antarmuka *Help*

Antarmuka ini untuk membantu pengguna dalam mengoperasikan sistem yang dibuat. Gambar (5.10) adalah gambar antamuka *Help*.



Gambar 5.10 Antarmuka Help

BAB VI

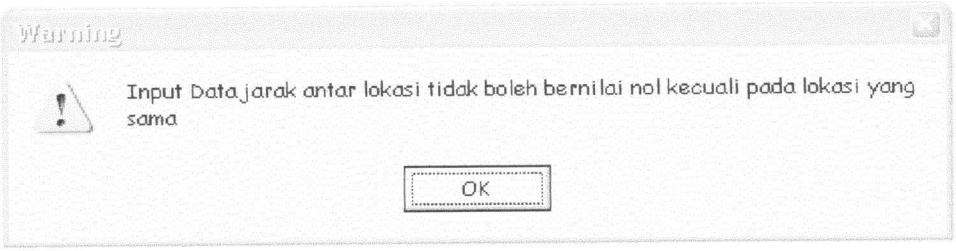
ANALISIS KINERJA PERANGKAT LUNAK

6.1 Proses Masukan / *Input*

Untuk melakukan pemanggilan *file* data matrik jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen melalui *data file* dan ambil data (*load data*), sistem akan membatasi jika *file* yang dibuka bukan *file* dari sistem atau sistem tidak dapat mendefinisikan *file* yang dibuka. Jika terjadi kesalahan pada pemanggilan *file* maka sistem akan memberikan pesan kesalahan. Gambar (6.1) memperlihatkan pesan kesalahan pada pemanggilan *file*.

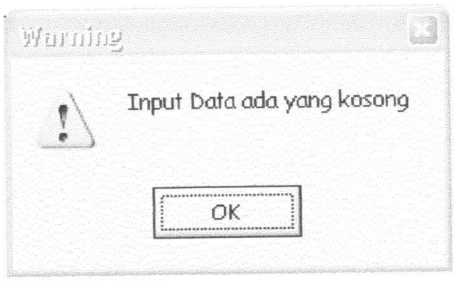
Gambar 6.1 Pesan *file* data yang dibuka tidak dapat didefinisikan

Pada penginputan data jarak antar lokasi melalui *data file* dan data baru, apabila nilai pada matrik jarak antar lokasi ada yang bernilai nol, maka sistem akan memberikan pesan kesalahan seperti terlihat pada gambar (6.2)



Gambar 6.2 Pesan nilai data tidak boleh bernilai nol

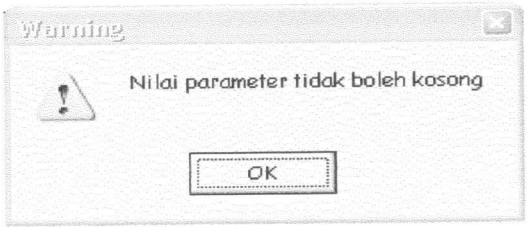
Pada penginputan nilai jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen, apabila ada data yang kosong, maka sistem akan memberikan pesan kesalahan seperti terlihat pada gambar (6.3)



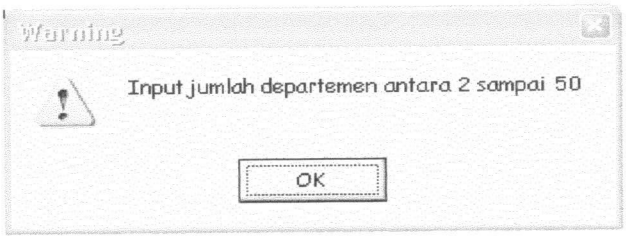
Gambar 6.3 Pesan nilai data ada yang kosong

6.2 Proses Pelatihan

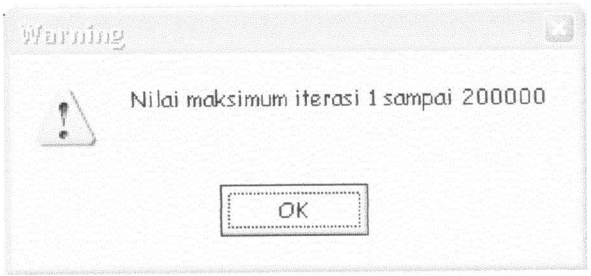
Pada proses pelatihan, sistem akan membatasi apabila dalam penginputan parameter pelatihan diluar dari *range* yang telah ditentukan oleh sistem atau kosong. Jika terjadi kesalahan sistem akan memberikan pesan kesalahan seperti pada gambar (6.4) sampai (6.7).



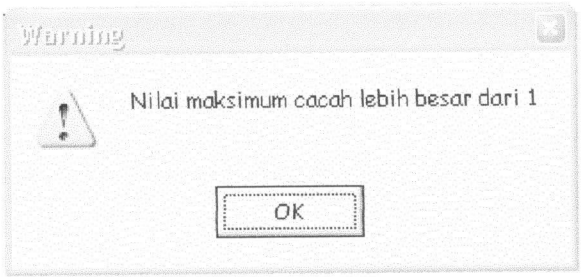
Gambar 6.4 Pesan kesalahan parameter pelatihan ada yang kosong



Gambar 6.5 Pesan masukan jumlah departemen di luar *range*



Gambar 6.6 Pesan nilai parameter maksimum iterasi di luar *range*



Gambar 6.7 Pesan nilai parameter maksimum cacah di luar *range*

6.3 Proses Keluaran / *Output*

Output dari sistem berupa hasil pelatihan yang meliputi informasi nilai bobot antar departemen, hasil perhitungan total biaya, urutan departemen, posisi departemen dan lain-lain. Output dari sistem yang berupa informasi text akan disimpan dalam text file dan yang berupa grafik akan disimpan dalam bentuk bitmap.

Gambar 6.8 Pesan kesalahan penyimpanan

6.4 Uji Coba Perangkat Lunak

Pada bagian ini akan dilakukan uji coba perangkat lunak dengan beberapa contoh kasus, berikut data kasus yang digunakan untuk melakukan uji coba perangkat lunak :

1. **Kasus-1** : penginputan data jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen menggunakan *data file*, dengan jumlah departemen = 6, maksimum iterasi = 200, menggunakan metode :
   a. *improvement heuristics*
   b.*construction heuristics* dengan maksimum cacah = 6
2. **Kasus-2** : penginputan data jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen secara *random*, dengan jumlah departemen 10, maksimum iterasi = 100, menggunakan metode :
   a. *improvement heuristics*
   b. *construction heuristics* dengan maksimum cacah = 5

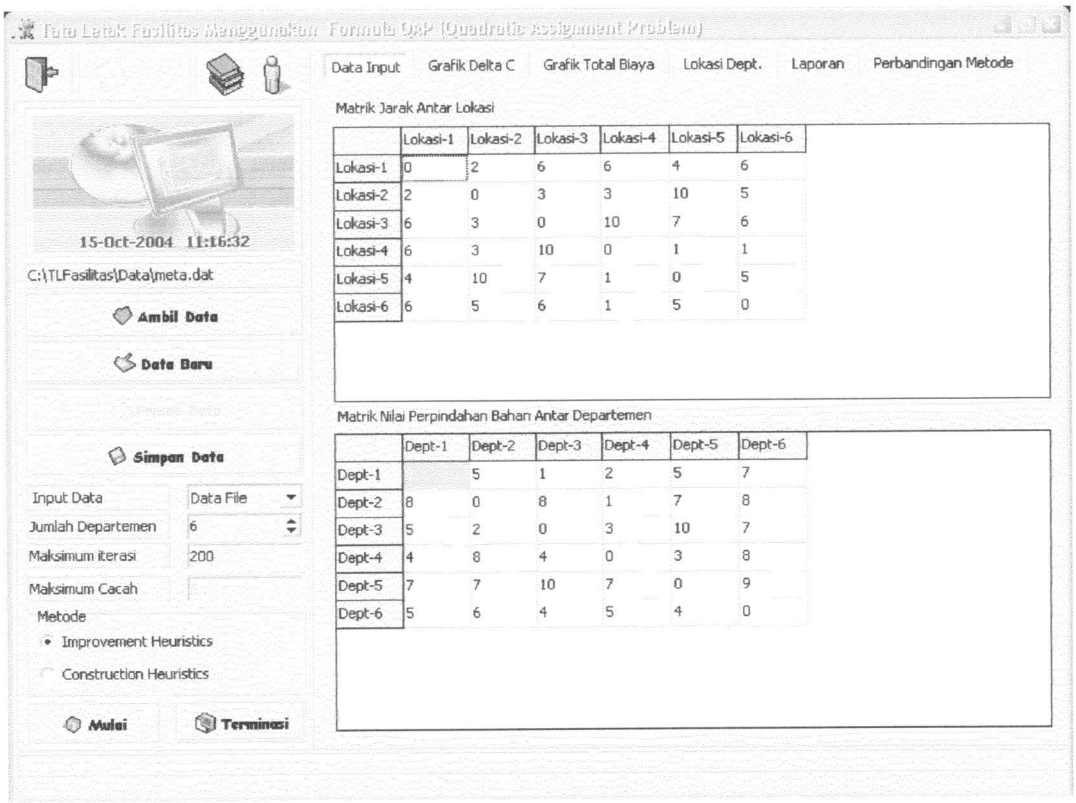
6.5 Hasil Uji Coba Perangkat Lunak

Berdasarkan data kasus pada uji coba terhadap perangkat lunak yang dibuat, hasil uji coba dari beberapa kasus tersebut akan diperlihatkan dalam bentuk antarmuka sebagai berikut :

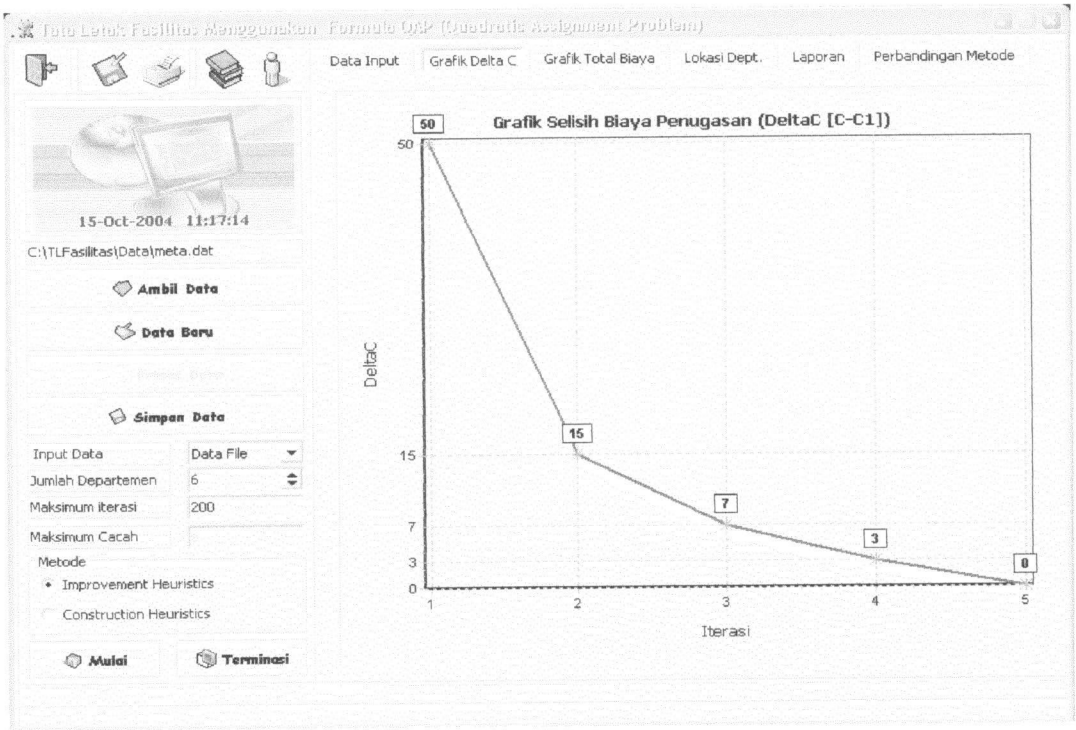
1. **Kasus-1**
   a. Menggunakan metode *improvement heuristics*

Pada uji coba kasus-1 dengan menggunakan metode *improvement heuristics* dari beberapa kali proses pelatihan, total biaya penugasan paling minimum yang

diperoleh adalah 774, dengan vektor penugasan/posisi departemen a = (2, 6, 5, 3, 4, 1). Gambar (6.9) sampai (6.14) berurutan merupakan antarmuka hasil uji perangkat lunak untuk kasus-1.

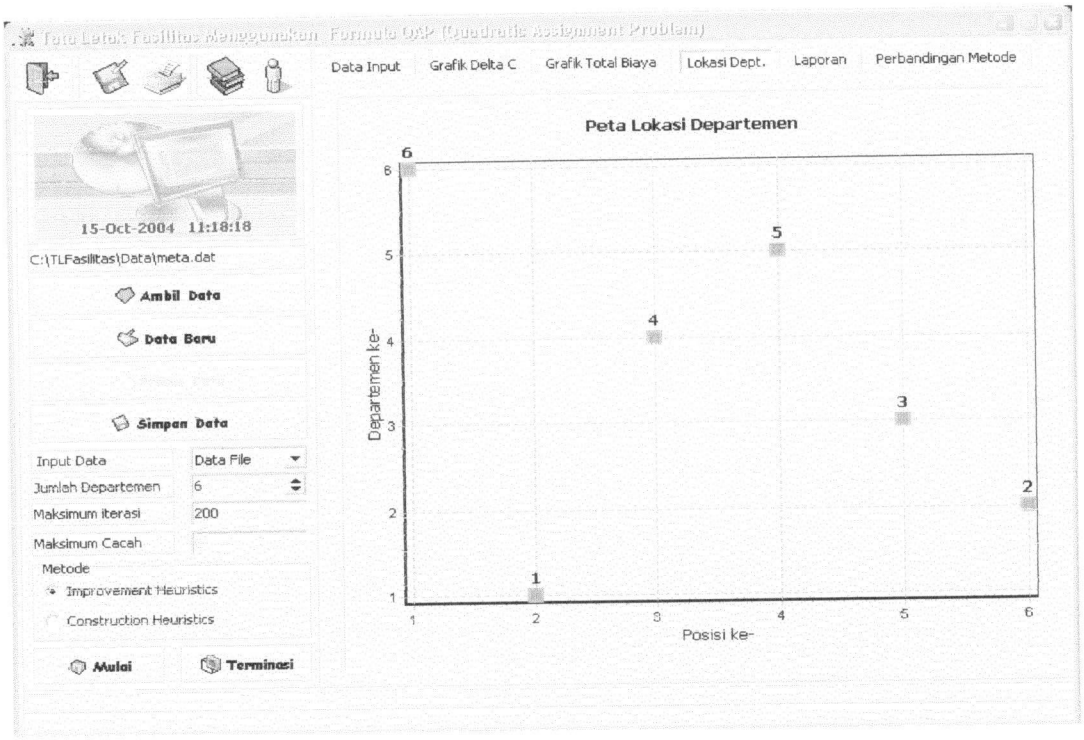


Gambar 6.9 Tampilan Awal Kasus-1(a)



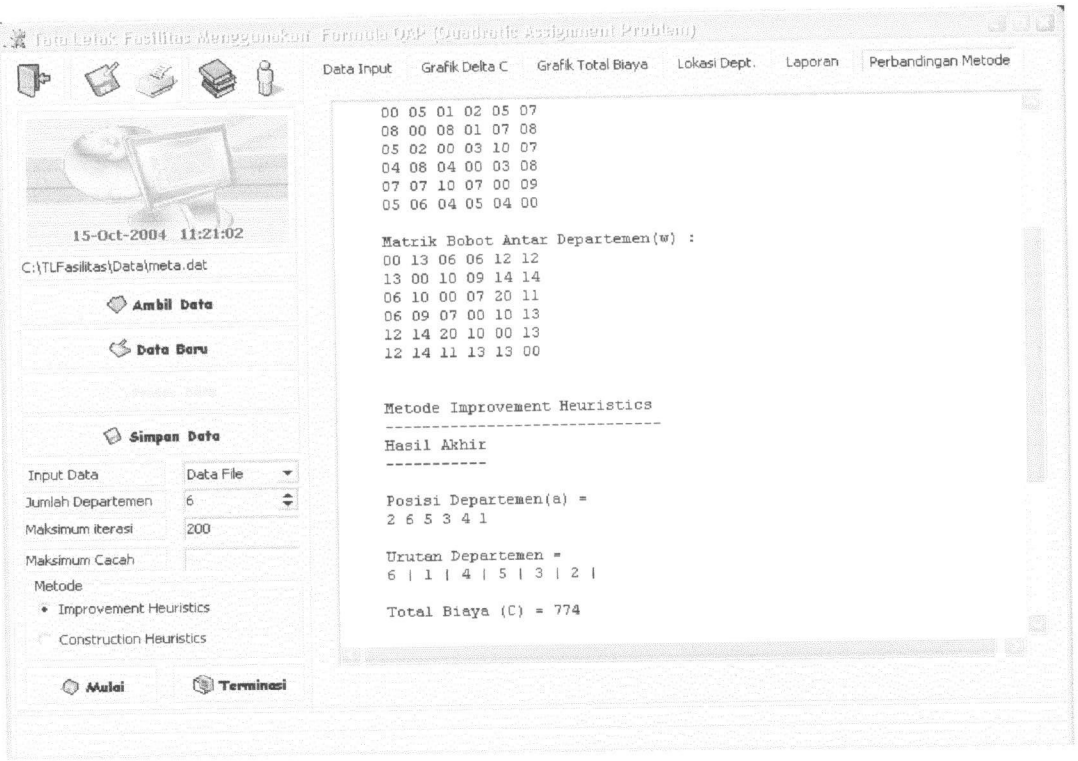
Gambar 6.10 Tampilan Grafik Selisih Biaya Penugasan Kasus-1(a)

Gambar 6.11 Tampilan Grafik Total Biaya Kasus-1(a)



Gambar 6.12 Tampilan Peta Lokasi Departemen Kasus-1(a)

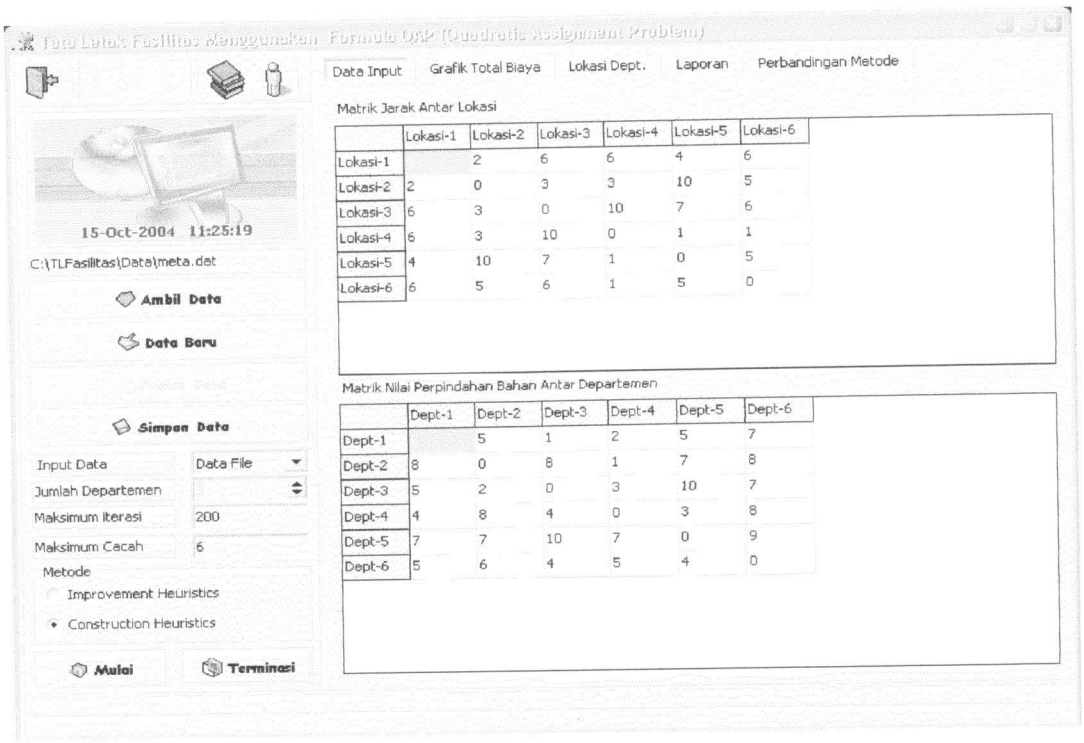
Gambar 6.13 Tampilan Laporan Hasil Proses Pelatihan Kasus-1(a)



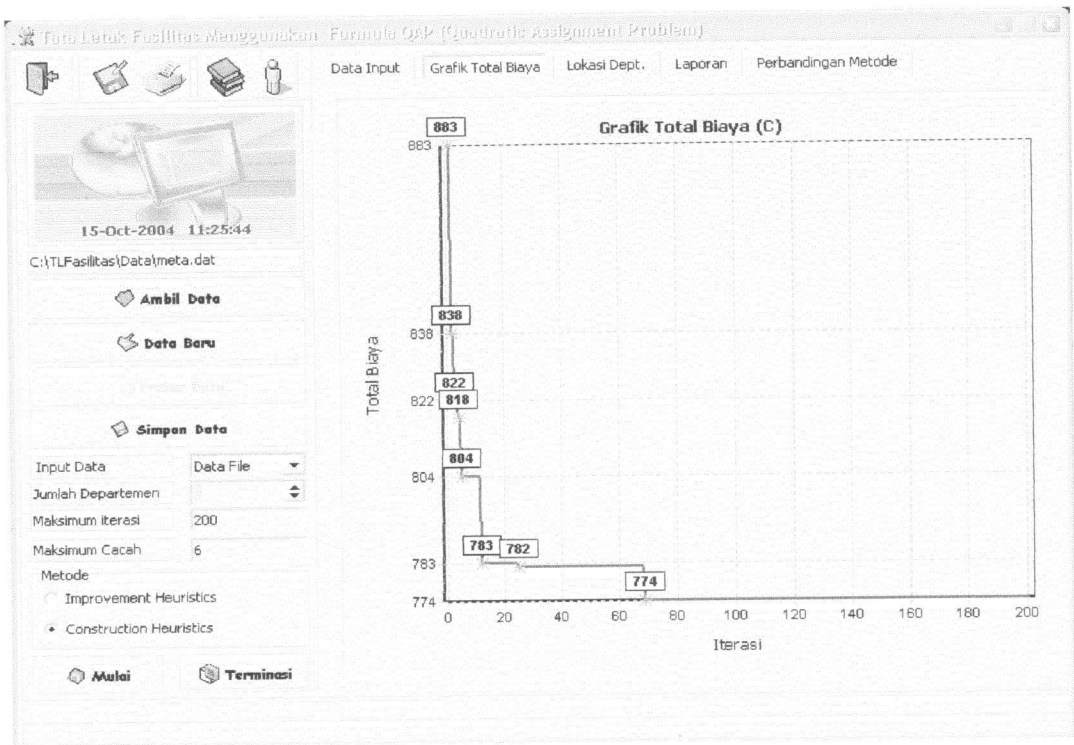
Gambar 6.14 Tampilan Laporan Perbandingan Metode Kasus-1(a)

b. Menggunakan metode *construction heuristics*

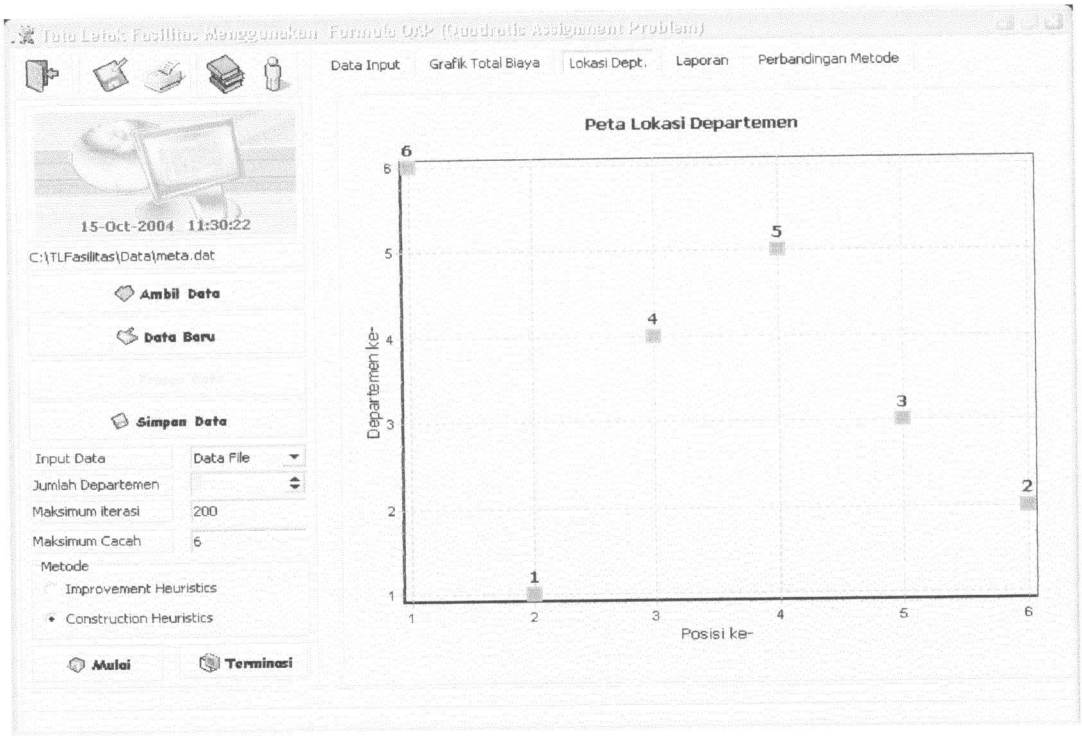
Pada uji coba kasus-1 dengan menggunakan metode *construction heuristics* dari beberapa kali proses pelatihan, total biaya penugasan paling minimum yang diperoleh adalah 774, dengan vektor penugasan/posisi departemen a = (2, 6, 5, 3, 4, 1). Hasil akhir dengan menggunakan metode *construction heuristic* dengan data yang sama selalu berbeda setiap kali dilakukan proses pelatihan, hal ini terjadi karena pemilihan departemen untuk diproses dilakukan secara random. Gambar (6.15) sampai (6.19) berurutan merupakan antarmuka hasil uji perangkat lunak untuk kasus-1.



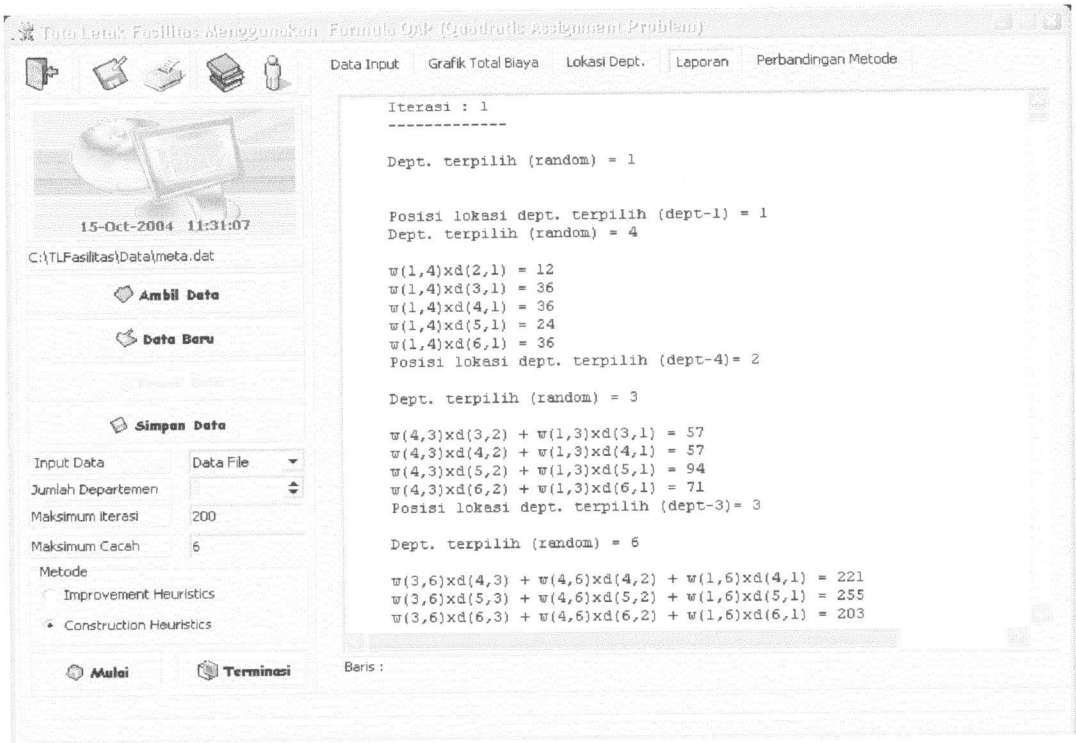
Gambar 6.15 Tampilan Awal Kasus-1(b)



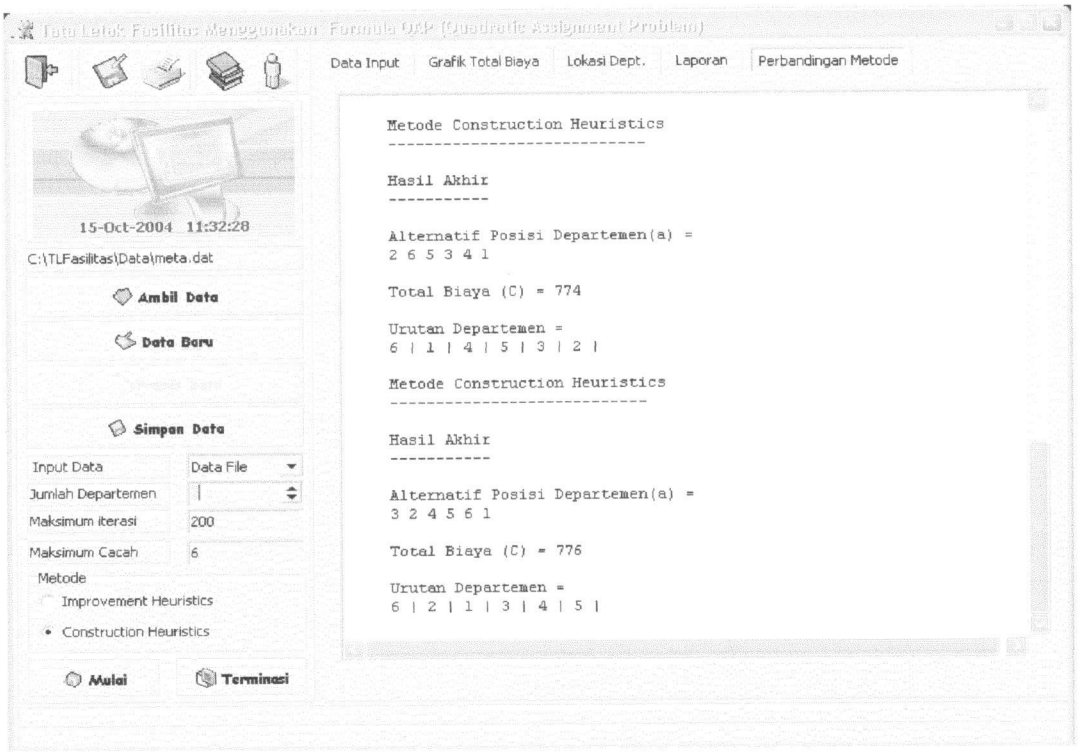
Gambar 6.16 Tampilan Grafik Total Biaya Kasus-1(b)



Gambar 6.17 Tampilan Peta Lokasi Departemen Kasus-1(b)



Gambar 6.18 Tampilan Laporan Hasil Proses Pelatihan Kasus-1(b)

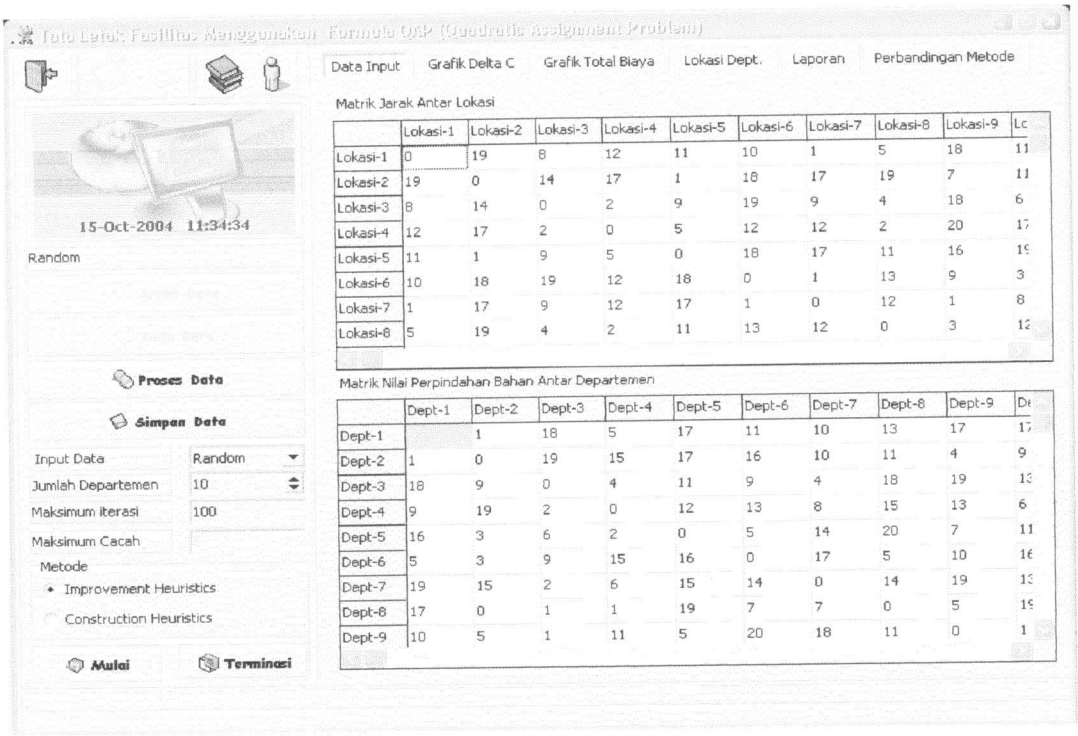


Gambar 6.19 Tampilan Laporan Perbandingan Metode Kasus-1(b)

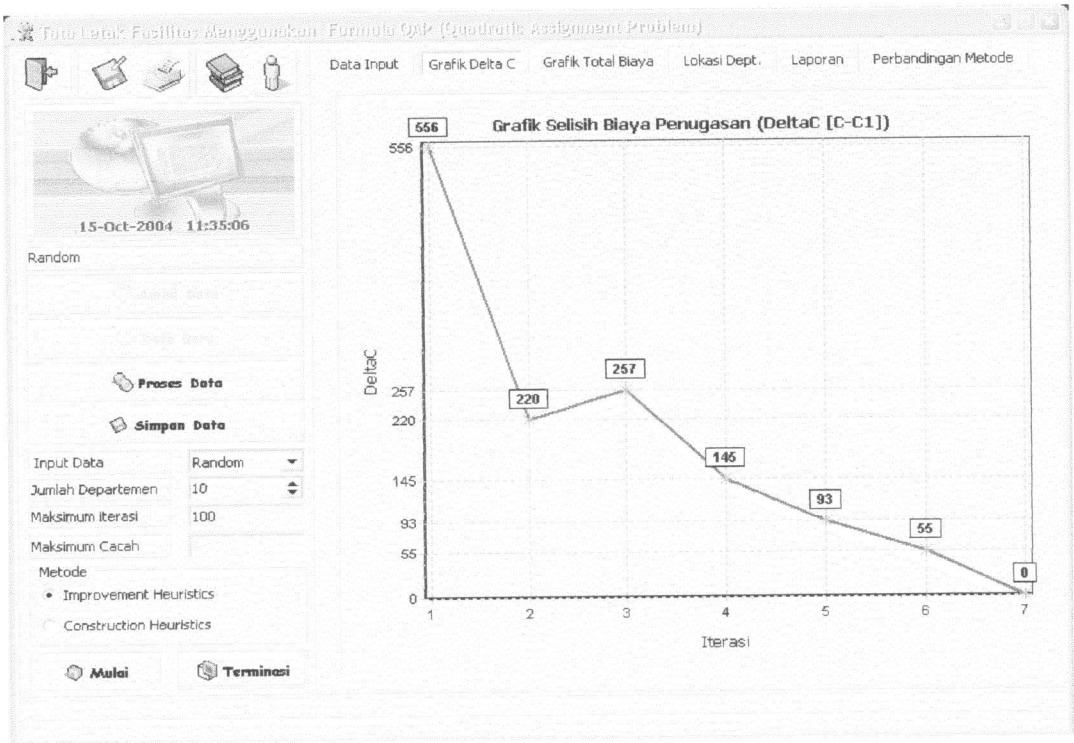
2. **Kasus-2**

a. Menggunakan metode *improvement heuristics*

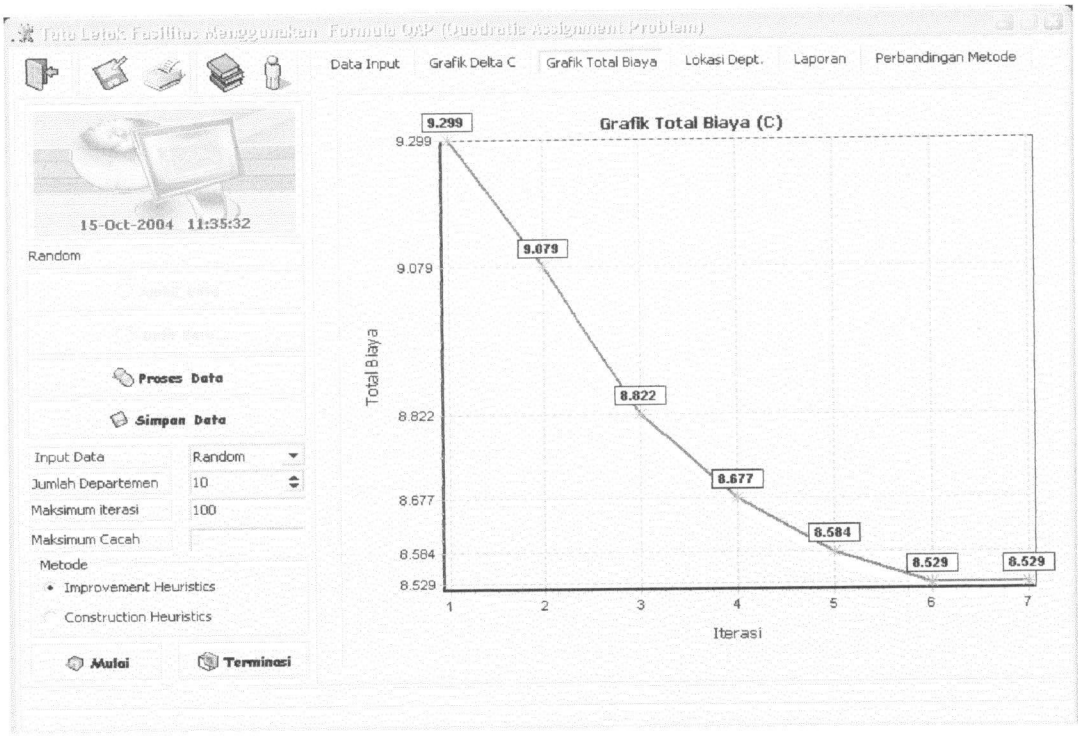
Pada uji coba kasus-2 dengan menggunakan metode *improvement heuristics* dari beberapa kali proses pelatihan, total biaya penugasan paling minimum yang diperoleh adalah 8529, dengan vektor penugasan/posisi departemen a = (4, 2, 5, 10, 8, 9, 7, 1, 6, 3). Gambar (6.20) sampai (6.25) berurutan merupakan antarmuka hasil uji perangkat lunak untuk kasus-2.



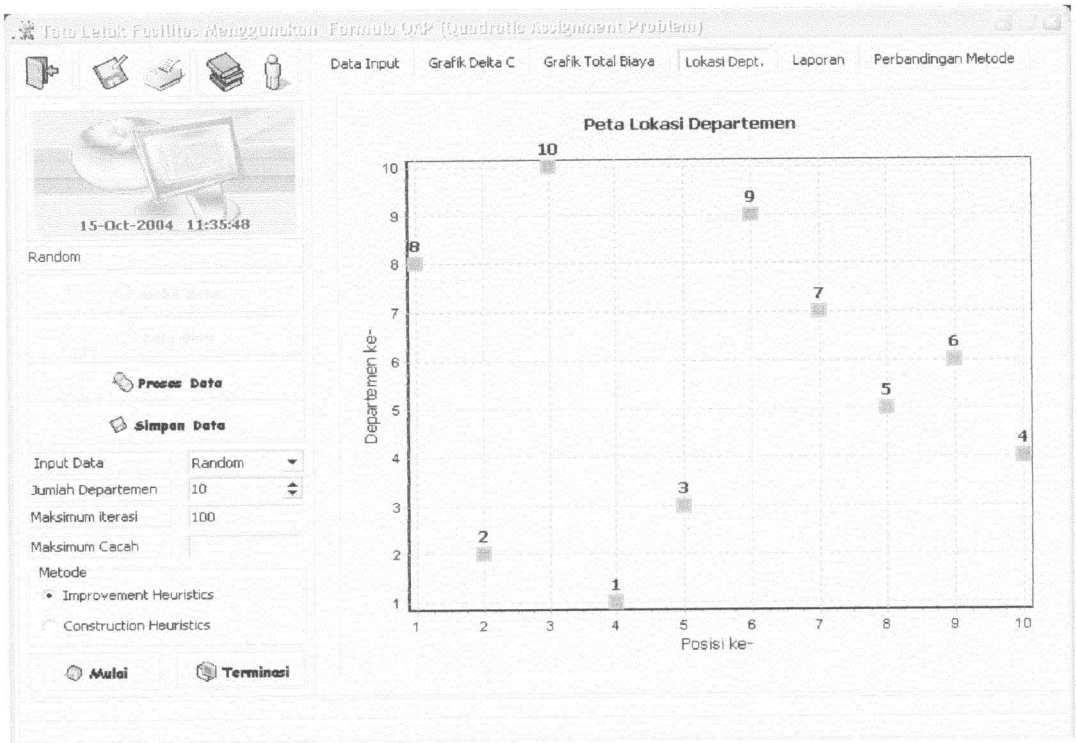
Gambar 6.20 Tampilan Awal Kasus-2(a)



Gambar 6.21 Tampilan Grafik Selisih Biaya Penugasan Kasus-2(a)

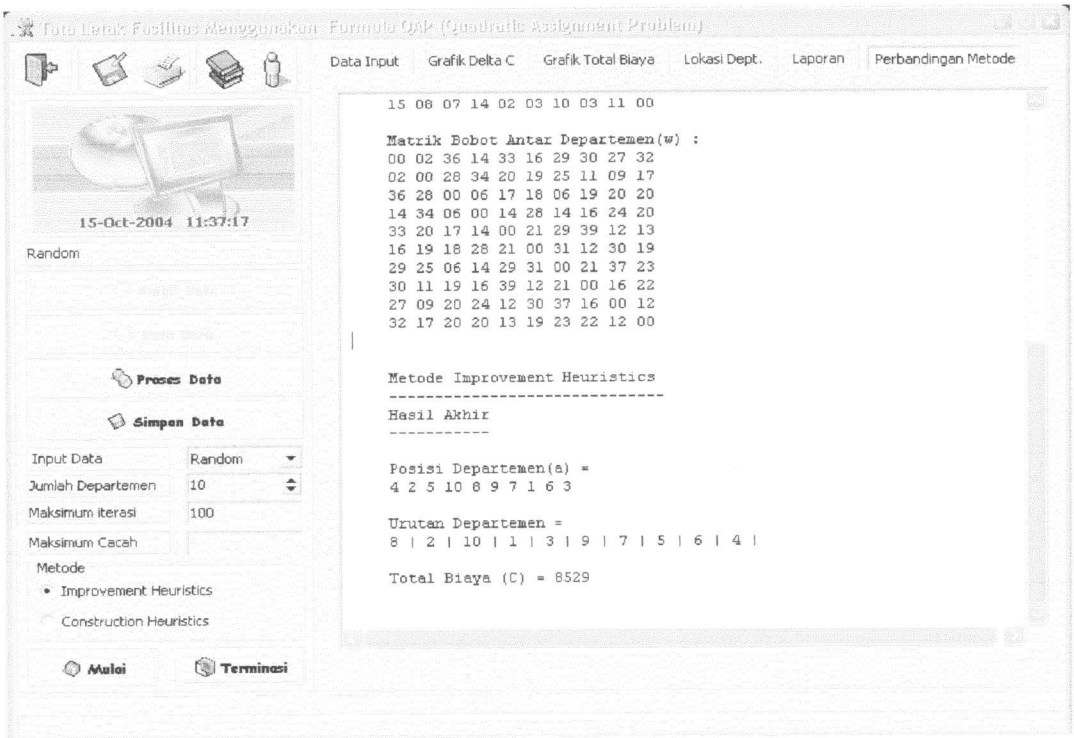


Gambar 6.22 Tampilan Grafik Total Biaya Kasus-2(a)



Gambar 6.23 Tampilan Peta Lokasi Departemen Kasus-2(a)

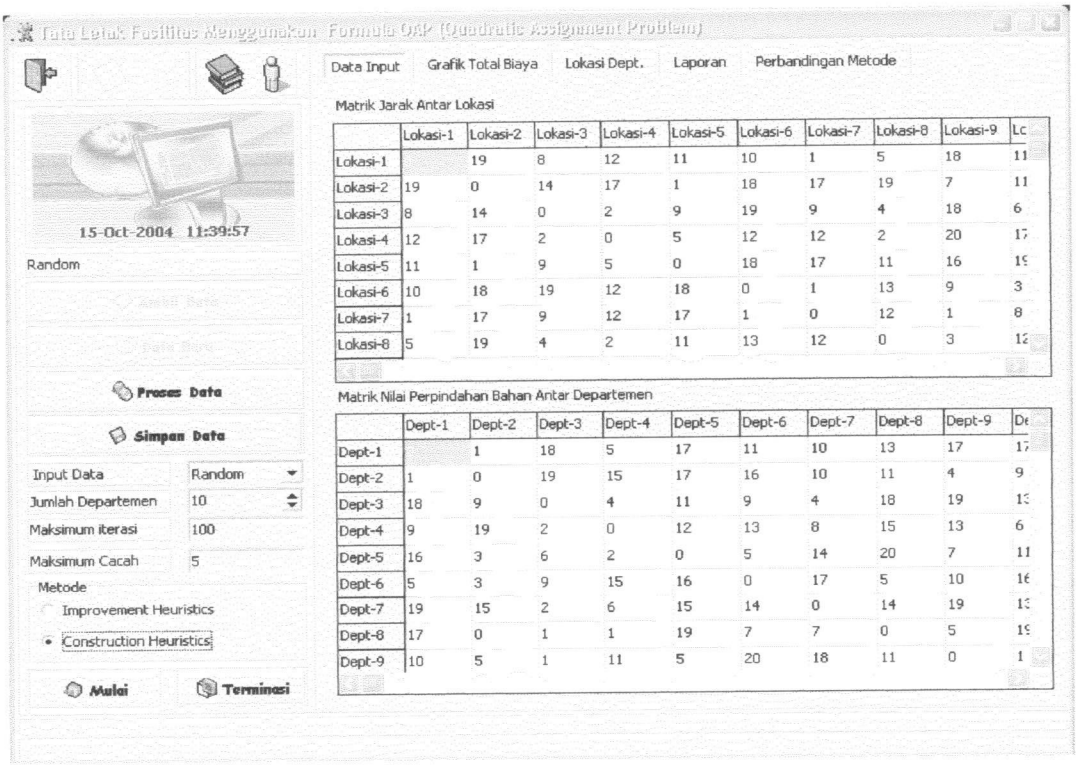
Gambar 6.24 Tampilan Laporan Hasil Proses Pelatihan Kasus-2(a)



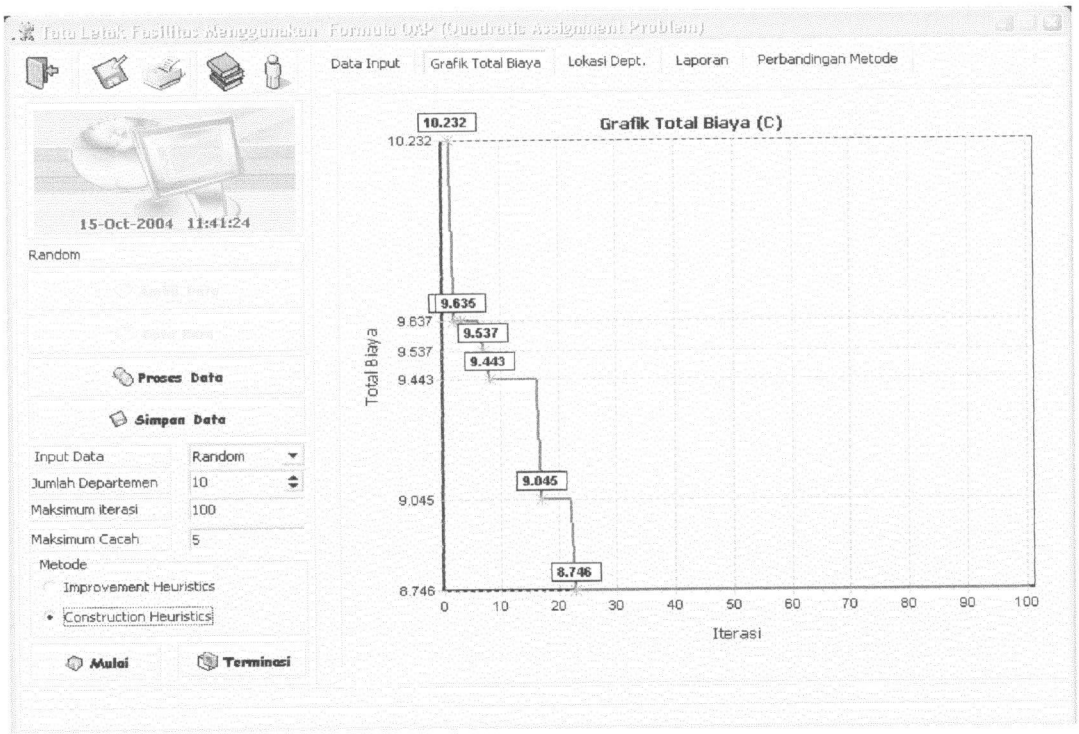
Gambar 6.25 Tampilan Laporan Perbandingan Metode Kasus-2(a)

b. Menggunakan metode *construction heuristics*

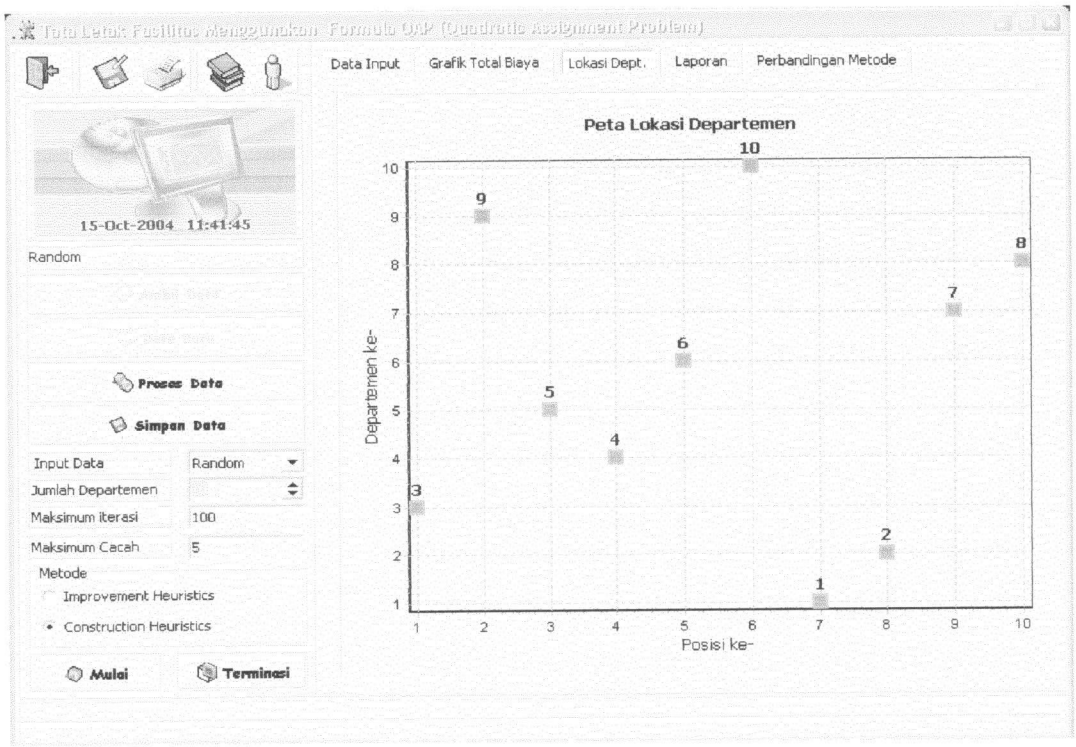
Pada uji coba kasus-2 dengan menggunakan metode *construction heuristics* dari beberapa kali proses pelatihan, total biaya penugasan paling minimum yang diperoleh adalah 8746 , dengan vektor penugasan/posisi departemen a = (7, 8, 1, 4, 3, 5, 9, 10, 2, 6). Hasil akhir dengan menggunakan metode *construction heuristic* dengan data yang sama selalu berbeda setiap kali dilakukan proses pelatihan, hal ini terjadi karena pemilihan departemen untuk diproses dilakukan secara random. Gambar (6.26) sampai (6.30) berurutan merupakan antarmuka hasil uji perangkat lunak untuk kasus-2.



Gambar 6.26 Tampilan Awal Kasus-2(b)



Gambar 6.27 Tampilan Grafik Total Biaya Kasus-2(b)



Gambar 6.28 Tampilan Peta Lokasi Departemen Kasus-2(b)

Gambar 6.29 Tampilan Laporan Hasil Proses Pelatihan Kasus-2(b)

Gambar 6.30 Tampilan Laporan Perbandingan Metode Kasus-2(b)

6.6 Analisis Perbandingan Metode

Setelah dilakukan uji coba terhadap perangkat lunak, dua metode yang digunakan dalam penyelesaian tata letak fasilitas yaitu *construction heuristics* dan *improvement heuristics* dapat dibandingkan. Dengan jumlah fasilitas dan data jarak antar lokasi serta nilai aliran perpindahan bahan yang sama, apabila menggunakan metode *improvement heuristics* walaupun parameter pelatihan yang berupa maksimum iterasi diperbesar hasil akhir total biaya penugasan yang dihasilkan dari proses pelatihan akan tetap sama. Tetapi, dengan menggunakan metode *construction heuristics* bila parameter pelatihan yang berupa maksimum iterasi diperbesar maka hasil akhir total biaya penugasan yang dihasilkan dari proses pelatihan dapat berkurang. Hal ini mengasumsikan bahwa metode *construction heuristics* menghasilkan total biaya penugasan lebih kecil bila dibandingkan dengan metode *improvement heuristics*, tetapi harus dengan parameter pelatihan yang besar dan proses pelatihan dilakukan berulangkali dengan data yang sama agar diperoleh hasil akhir yang seminimal mungkin.

6.7 Kelebihan dan Kekurangan Perangkat Lunak

Dari hasil analisis di atas, perangkat lunak yang dibangun mampu menangani berbagai kemungkinan masukan data sehingga fleksibel untuk digunakan dan untuk mengetahui tata letak fasilitas yang mempunyai total biaya penugasan yang paling minimum. Jumlah data yang dapat diolah perangkat lunak ini maksimal 50 fasilitas. Input jumlah fasilitas ini bisa ditambah sampai tidak terbatas, tetapi hanya dibatasi oleh kemampuan perangkat lunak maupun perangkat keras (*Processor*, RAM dan lai-lain) dan berupa bilangan bulat maupun desimal.

BAB VII

PENUTUP

7.1 Kesimpulan

Pada bab ini dijelaskan beberapa kesimpulan sesuai dengan uraian yang telah dijelaskan pada bab-bab sebelumnya bagi pengembangan terhadap perangkat lunak yang dibuat. Dengan memperhatikan program yang telah dibuat didapatkan beberapa kesimpulan, antara lain :

1. Perangkat lunak yang telah dibuat dapat digunakan untuk menyelesaikan permasalahan tata letak fasilitas (*facility layout*) sebagai optimasi mendapatkan alternatif tata letak fasilitas dengan total biaya penugasan yang minimum.
2. Penyelesaian masalah tata letak fasilitas (*facility layout*) menggunakan formula QAP (*Quadratic Assignment Problem*) untuk mendapatkan total biaya yang minimum sangat dipengaruhi oleh metode yang digunakan, yaitu *improvement heuristics* atau *construction heuristics,* serta sangat dipengaruhi oleh parameter pelatihan yang berupa jumlah departemen dan maksimum iterasi. Apabila proses pelatihan dilakukan beberapa kali dengan data input jarak antar lokasi dan nilai aliran perpindahan bahan antar departemen yang sama dengan menggunakan metode *improvement heuristics* maka hasil akhir alternatif urutan departemen/fasilitas dan total biaya yang dihasilkan akan sama. Tetapi apabila menggunakan metode *construction heuristics* hasil akhir alternatif urutan departemen/fasilitas dan total biaya yang dihasilkan akan

berbeda, karena pemilihan departemen untuk diproses dilakukan secara random. Hal ini mengasumsikan bahwa untuk mendapatkan hasil yang baik dengan menggunakan metode *construction heuristics* perlu dilakukan beberapa kali pelatihan sebagai perbandingan untuk mendapatkan total biaya penugasan yang paling minimum.

7.2 Saran

Berdasarkan pada pengujian yang telah dilakukan pada perangkat lunak yang dibuat, masih banyak kekurangan dan kelemahan sehingga perlu dikembangkan lagi agar kinerjanya lebih baik, oleh karena itu disarankan :

1. Pada proses pelatihan dengan jumlah departemen dan maksimum iterasi yang besar akan membutuhkan memori yang cukup besar dan pelatihan akan berjalan lambat, hal ini memungkinkan pengembangan terhadap perangkat lunak yang dibuat sehingga dapat efisien dalam penggunaan memori, dan pengembangan pada visualisasinya supaya lebih baik.
2. Perlu ditambahkan data yang lebih akurat untuk mendapatkan hasil yang maksimal selama pelatihan.

DAFTAR PUSTAKA

[PUR01] Purnomo, Hari. *Diktat Perencanaan Tata Letak Fasilitas.* Yogyakarta : Jurusan Teknik dan Manajemen Industri, 2001.

[PUR04] Purnomo, Hari. *Diktat Perencanaan Tata Letak Fasilitas.* Yogyakarta : Jurusan Teknik dan Manajemen Industri, 2004.

[RLJ92] Richard, L. Francis. Leon, F. McGinnis, Jr. John, A. White. *Facility Layout and Location 2nd Edition : An Analytical Approach.* New Jersey : Prentice-Hall, 1992.

LAMPIRAN

Pseudo Code Metode *Improvement Heuristics*

```
procedure imp.Execute;
var
i,j,k,m,n,u,v : integer;
maks,c,c1,cg : real;
deltac : array[1..50, 1..50] of real;
jdept : integer;
jarak,rate,bobot: ARRAY[1..50, 1..50] OF REAL;
a,a1,num,urutdept,urutdept2 : array[byte] of integer;
temp,iterasi,maksiterasi,done : integer;

procedure switchdata(var o, p : integer);
var q: integer;
begin
q:=o; o:=p; p:=q;
end;

begin

//hitung bobot
for i:= 1 to jdept do
  begin
    for j:= i to jdept do
      begin
          bobot[i,j]:=rate[i,j])+rate[j,i]));
          bobot[j,i]:=bobot[i,j];
      end;
  end;

for i:=1 to jdept do
  begin
   a[i]:=i;
  end;
iterasi:=0;
maksiterasi:=maksimum_iterasi;
done:=0;

///mulai iterasi
repeat
iterasi:=iterasi+1;
done:=1;
maks:=0;
for i:=1 to jdept do
  begin
   a1[i]:=a[i];
  end;

//tentukan posisi sementara dept yang akan ditukar//
for i:=1 to jdept-1 do
   begin
     for j:=i+1 to jdept do
       begin
          for k:=1 to jdept do
            begin
              a1[k]:=a[k];
            end;
          a1[i]:=a[j];
          a1[j]:=a[i];
          c:=0;
          c1:=0;
```

```
      //hitung Cost(c) dan (c1)//
           for m:=1 to jdept-1 do
              begin
                 for n:=m+1 to jdept do
                   begin
                      c:=c+bobot[m,n]*jarak[a[m],a[n]];
                      c1:=c1+bobot[m,n]*jarak[a1[m],a1[n]];
                   end;
               end;

    //hitung delta c ---> c-c1//
           deltac[i,j]:=c-c1;

     //cari nilai maksimum delta c, posisi ini akan ditukar//
           if deltaC[i,j]>maks then
              begin
              maks:=deltaC[i,j];
              done:=0;
              u:=i;
              v:=j;
              cg:=c1;
              end
              else
              begin
              cg:=cg;
              end;
        end;
    end;


//tukar posisi dept u dan dept v//
if maks>0 then
  begin
  temp:=a[u];
  a[u]:=a[v];
  a[v]:=temp;
  end;

until (iterasi=maksiterasi) or (done=1);
for k:=0 to jdept-1 do
 begin
 num[k]:=a[k+1];
 end;

for k:=0 to jdept-1 do
  begin
  urutdept2[k]:=k;
  end;

begin
for k:=1 to jdept-1 do
for m:=jdept-1 downto k do
    if num[m-1]>num[m] then
    begin
    switchdata(num[m], num[m-1]);
    switchdata(urutdept2[m], urutdept2[m-1]);
    end;
end;

for k:=0 to jdept-1 do
  begin
  urutdept[k+1]:=urutdept2[k]+1;
```

```
  end;
//hitung cost(c)
c:=0;
for k:=1 to jdept-1 do
  begin
    for m:=k+1 to jdept do
      begin
        c:=c+bobot[k,m]*jarak[a[k],a[m]];
      end;
  end;

end;
end.
```

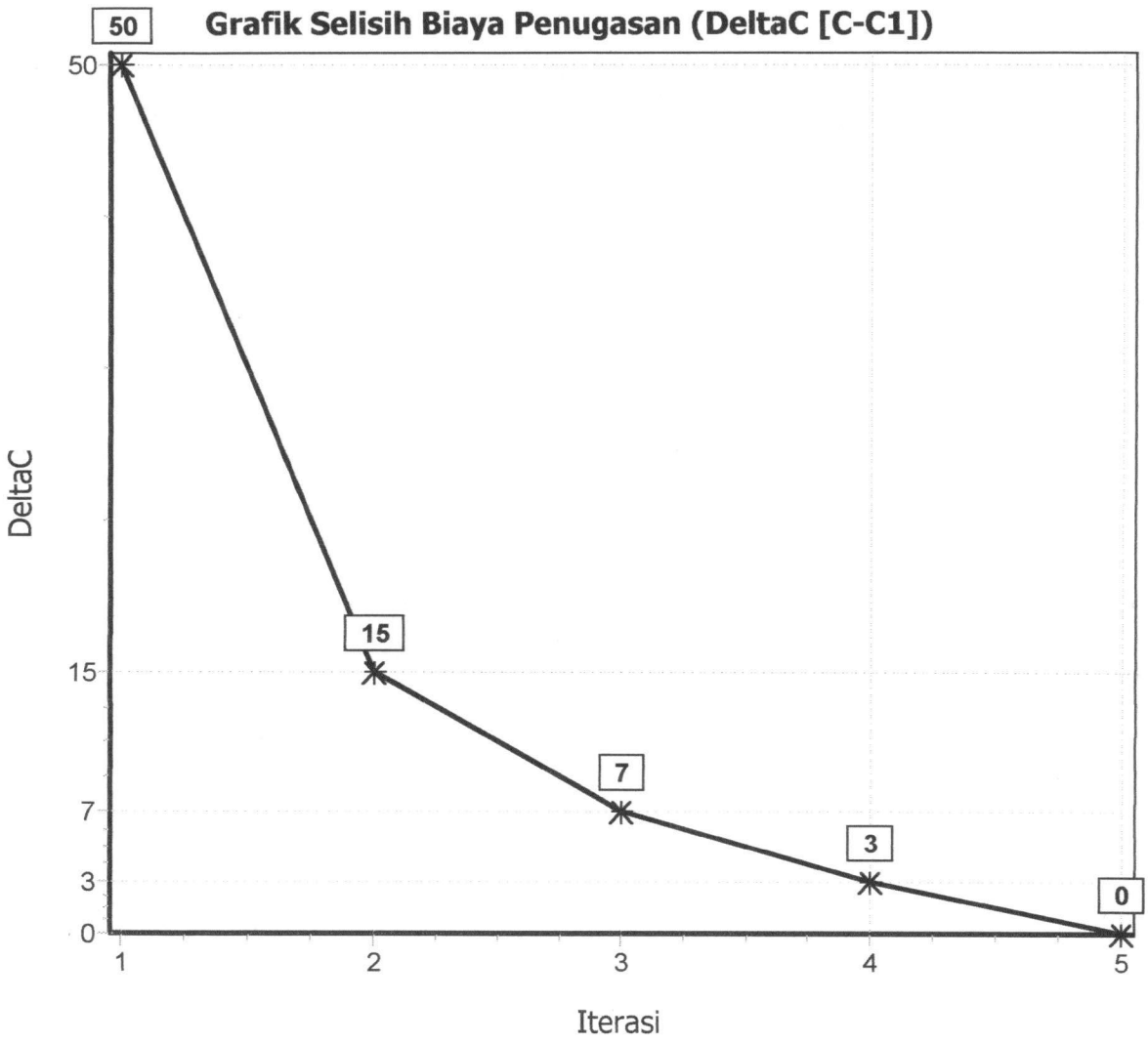
Pseudo Code Metode *Construction Heuristics*

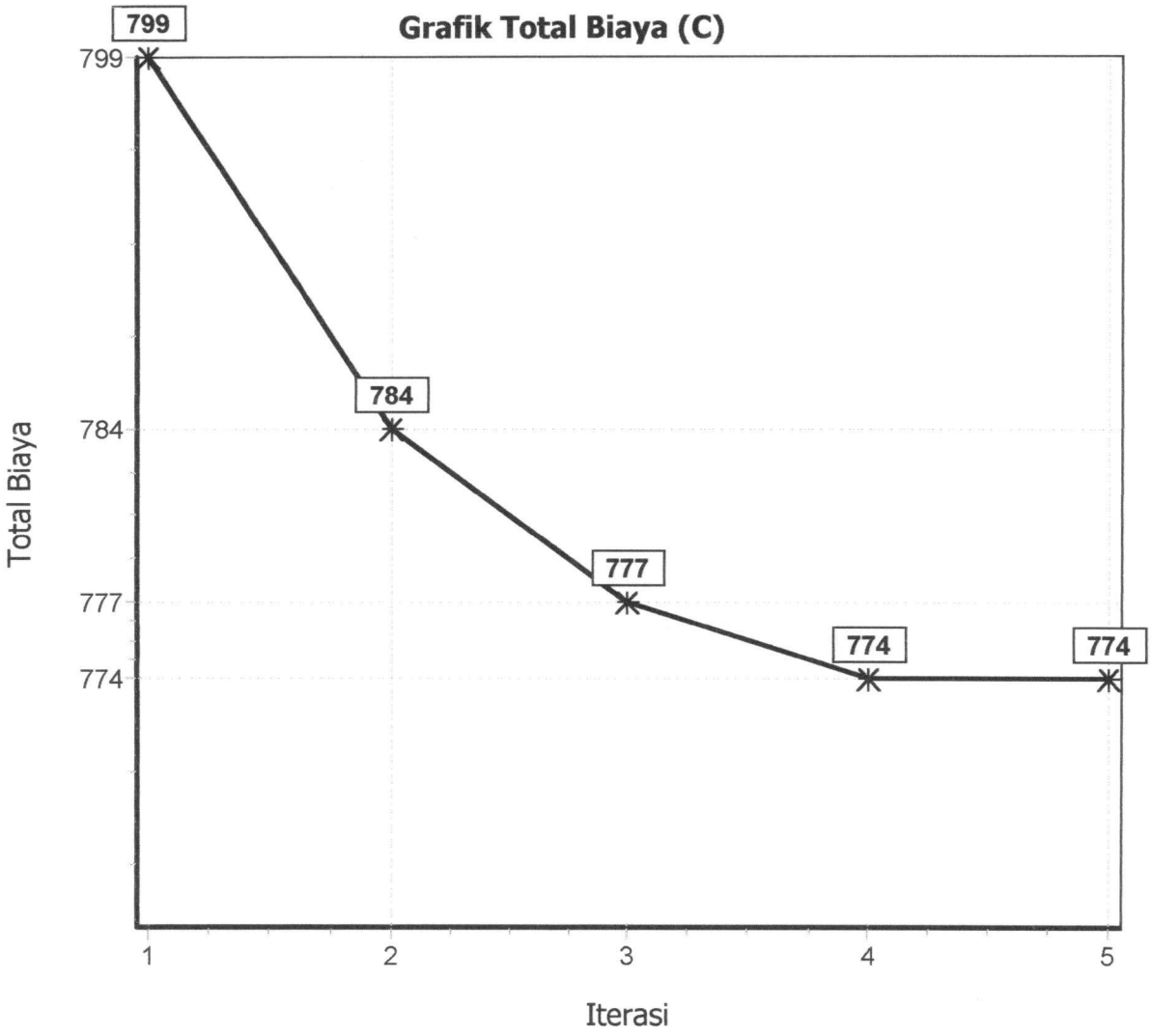
```
procedure con.Execute;
var
konter,ti,af,i,l,j,k,m,n,u,v : integer;
minc,maks,c,c1,cg : real;
jdept : integer;
jarak,rate,bobot: ARRAY[1..50, 1..50] OF REAL;
a,posisidept : array[1..50, 1..50] of integer;
p : array[1..50, 1..50] of real;
urutdept2,urutdept,nfa,q,x,y,nf,num,idx,idx2 : array[byte] of integer;
makscacah,cacah,temp,iterasi,maksiterasi,done : integer;

procedure switchdata(var s, b : integer);
var g: integer;
begin
g:=s; s:=b; b:=g;
end;

begin

//hitung bobot
for i:= 1 to jdept do
  begin
    for j:= i to jdept do
      begin
          bobot[i,j]:=rate[i,j])+rate[j,i]));
          bobot[j,i]:=bobot[i,j];
      end;
  end;

iterasi:=0;
maksiterasi:=maksimum_iterasi;
done:=0;
minC:=10e37;
cacah:=0;
konter:=0;
makscacah:=maksimum_cacah;

//mulai iterasi
repeat
iterasi:=iterasi+1;
for j:=1 to 100 do
begin
repeat
```

```
u:=round(random*jdept);
v:=round(random*jdept);
until (u<>v) and (u<>0) and (v<>0);
temp:=x[u];
x[u]:=x[v];
x[v]:=temp;
end;
temp:=0;

for i:=1 to jdept do
   begin
     if i=1 then
        begin
          y[x[i]]:=1;
          af:=1;
             for k:=2 to jdept do
               begin
               nf[k-1]:=k;
               konter:=k-1;
               end;
         end
         else
         begin
           for j:=1 to konter do
             begin
               for ti:=1 to i-1 do
                   begin
                      p[j,ti]:=bobot[x[ti],x[i]]*jarak[nf[j],y[x[ti]]];
                      q[j]:=round(q[j]+p[j,ti]);
                   end;
             end;

             //posisidept=a
             for k:=0 to konter-1 do
                begin
                 num[k]:=q[k+1];
                end;

             for k:=0 to konter-1 do
                begin
                  idx2[k]:=k;
                end;

             begin
                for k:=1 to konter-1 do
                   for m:=konter-1 downto k do
                     if num[m-1]>num[m] then
                       begin
                        switchdata(num[m], num[m-1]);
                        switchdata(idx2[m], idx2[m-1]);
                       end;
             end;

             for k:=0 to konter-1 do
                begin
                  idx[k+1]:=idx2[k]+1;
                end;
          y[x[i]]:=nf[idx[1]];

          for j:=1 to jdept do
             begin
               for ti:=1 to jdept do
                   begin
```

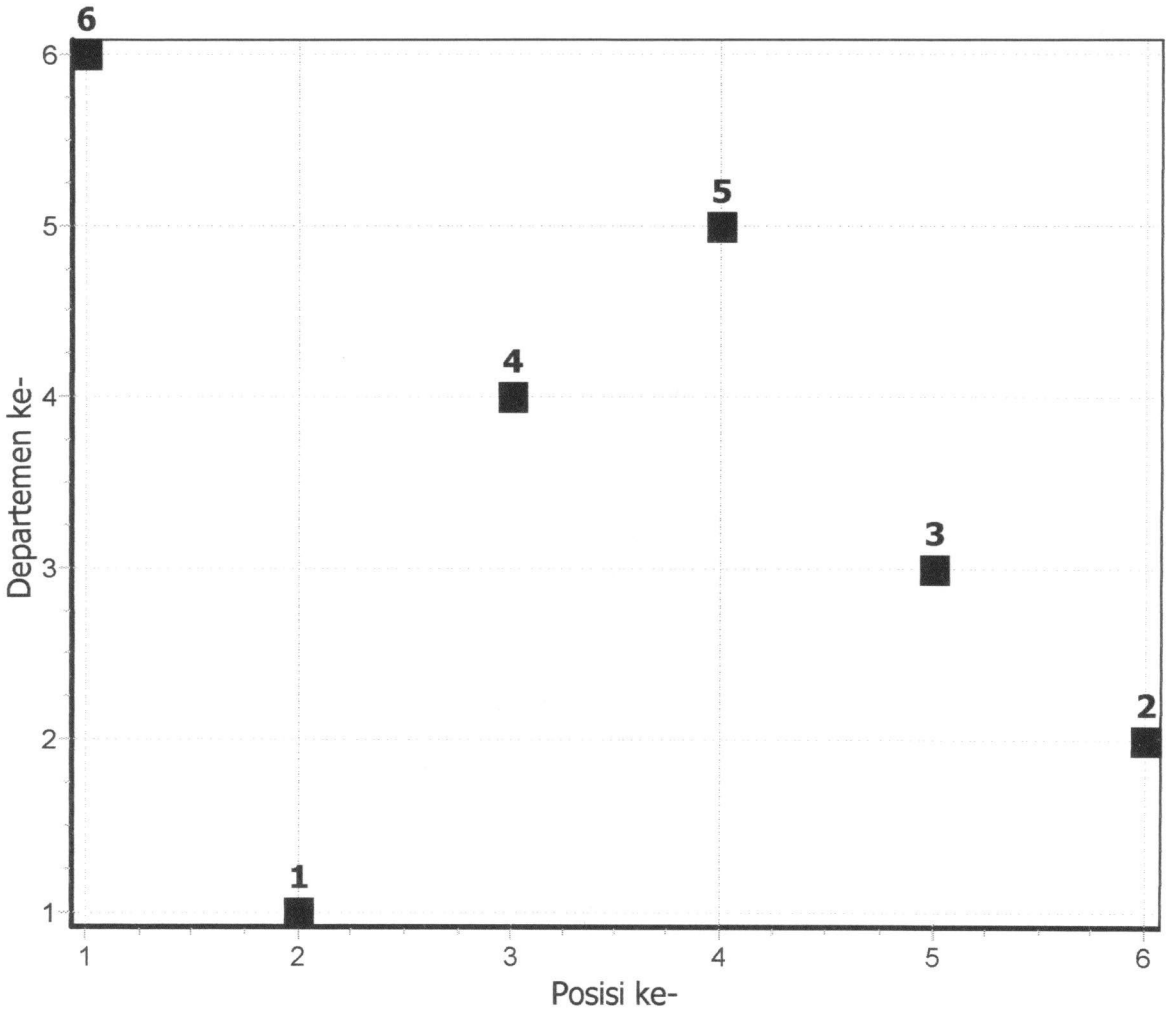
```
                        p[j,ti]:=0;
                        q[j]:=0;
                    end;
            end;

            for k:=1 to konter do
              begin
                if idx[1]=k then
                begin
                   for m:=k to konter-1 do
                      begin
                        nf[m]:=nf[m+1];
                      end;
                end;
              end;
            konter:=konter-1;
        end;
    end;

     // hitung cost (c)
     c:=0;
     for k:=1 to jdept-1 do
     begin
        for l:=k+1 to jdept do
           begin
               c:=c+bobot[k,l]*jarak[y[k],y[l]];
           end;
     end;

    if c<minc then
    begin
      cacah:=1;
      for k:=1 to 1 do
        begin
           for l:=1 to jdept do
              begin
                 a[1,l]:=y[l];
              end;
         end;

      minc:=c;
    end
    else
    begin
       if c=minc then
         begin
            cacah:=cacah+1;
            for l:=1 to jdept do
              begin
                a[cacah,l]:=y[l];
              end;
         end;
    end;

    if cacah > makscacah then
    begin
     done:=1;
    end;

until (iterasi=maksiterasi) or (done=1);

c:=0;
for k:= 1 to jdept-1 do
```

```
  begin
    for l:=k+1 to jdept do
       begin
          c:=c+bobot[k,l]*jarak[a[1,k],a[1,l]];
       end;
  end;

for k:=0 to jdept-1 do
 begin
 num[k]:=a[1,k+1];
 end;

for k:=0 to jdept-1 do
  begin
  urutdept2[k]:=k;
  end;

begin
for k:=1 to jdept-1 do
for m:=jdept-1 downto k do
    if num[m-1]>num[m] then
    begin
    switchdata(num[m], num[m-1]);
    switchdata(urutdept2[m], urutdept2[m-1]);
    end;
end;

for k:=0 to jdept-1 do
  begin
  urutdept[k+1]:=urutdept2[k]+1;
  end;

end;
end.
```


Grafik Selisih Biaya Penugasan (DeltaC [C-C1])
50
15
7
3
0
DeltaC
Iterasi
1
2
3
4
5


Grafik Total Biaya (C)
799
784
777
774
774
799
784
777
774
Total Biaya
1
2
3
4
5
Iterasi

Peta Lokasi Departemen



```
Jumlah Departemen : 6

Posisi Awal Departemen :
1 2 3 4 5 6

Matrik Jarak Antar lokasi(d) :
00 02 06 06 04 06
02 00 03 03 10 05
06 03 00 10 07 06
06 03 10 00 01 01
04 10 07 01 00 05
06 05 06 01 05 00


Matrik Nilai Aliran Perpindahan Bahan Antar Departemen(r) :
00 05 01 02 05 07
08 00 08 01 07 08
05 02 00 03 10 07
04 08 04 00 03 08
07 07 10 07 00 09
05 06 04 05 04 00


Matrik Bobot Antar Departemen(w) :
00 13 06 06 12 12
13 00 10 09 14 14
06 10 00 07 20 11
06 09 07 00 10 13
12 14 20 10 00 13
12 14 11 13 13 00


Iterasi : 1
-------------

Set Maks = 0

Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 2 1 3 4 5 6
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 860
Selisih biaya penugasan [DeltaC (C-C1)] = -11


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 3 2 1 4 5 6
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 824
Selisih biaya penugasan [DeltaC (C-C1)] = 25

(DeltaC = 25) > (Maks =  0) maka Maks =25
* u = 1
* v = 3
```

```
Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 4 2 3 1 5 6
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 848
Selisih biaya penugasan [DeltaC (C-C1)] = 1


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 5 2 3 4 1 6
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 848
Selisih biaya penugasan [DeltaC (C-C1)] = 1


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 6 2 3 4 5 1
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 880
Selisih biaya penugasan [DeltaC (C-C1)] = -31


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 1 3 2 4 5 6
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 912
Selisih biaya penugasan [DeltaC (C-C1)] = -63


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 1 4 3 2 5 6
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 858
Selisih biaya penugasan [DeltaC (C-C1)] = -9


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 1 5 3 4 2 6
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 813
Selisih biaya penugasan [DeltaC (C-C1)] = 36

(DeltaC = 36) > (Maks =  25) maka Maks =36
* u = 2
* v = 5


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 1 6 3 4 5 2
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 853
Selisih biaya penugasan [DeltaC (C-C1)] = -4


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 1 2 4 3 5 6
```

```
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 799
Selisih biaya penugasan [DeltaC (C-C1)] = 50

(DeltaC = 50) > (Maks =  36) maka Maks =50
* u = 3
* v = 4


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 1 2 5 4 3 6
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 862
Selisih biaya penugasan [DeltaC (C-C1)] = -13


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 1 2 6 4 5 3
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 881
Selisih biaya penugasan [DeltaC (C-C1)] = -32


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 1 2 3 5 4 6
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 865
Selisih biaya penugasan [DeltaC (C-C1)] = -16


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 1 2 3 6 5 4
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 843
Selisih biaya penugasan [DeltaC (C-C1)] = 6


Posisi awal(a) = 1 2 3 4 5 6
posisi baru(a1)= 1 2 3 4 6 5
Biaya posisi awal(C) = 849
Biaya posisi baru(C1) = 840
Selisih biaya penugasan [DeltaC (C-C1)] = 9


Tukar departemen 3 dengan departemen 4
Posisi baru : a = 1 2 4 3 5 6

Iterasi : 2
-------------

Set Maks = 0

Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 2 1 4 3 5 6
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 810
```

```
Selisih biaya penugasan [DeltaC (C-C1)] = -11


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 4 2 1 3 5 6
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 817
Selisih biaya penugasan [DeltaC (C-C1)] = -18


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 3 2 4 1 5 6
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 805
Selisih biaya penugasan [DeltaC (C-C1)] = -6


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 5 2 4 3 1 6
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 858
Selisih biaya penugasan [DeltaC (C-C1)] = -59


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 6 2 4 3 5 1
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 820
Selisih biaya penugasan [DeltaC (C-C1)] = -21


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 1 4 2 3 5 6
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 883
Selisih biaya penugasan [DeltaC (C-C1)] = -84


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 1 3 4 2 5 6
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 837
Selisih biaya penugasan [DeltaC (C-C1)] = -38


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 1 5 4 3 2 6
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 817
Selisih biaya penugasan [DeltaC (C-C1)] = -18


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 1 6 4 3 5 2
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 788
```

```
Selisih biaya penugasan [DeltaC (C-C1)] = 11

(DeltaC = 11) > (Maks =  0) maka Maks =11
* u = 2
* v = 6


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 1 2 3 4 5 6
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 849
Selisih biaya penugasan [DeltaC (C-C1)] = -50


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 1 2 5 3 4 6
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 784
Selisih biaya penugasan [DeltaC (C-C1)] = 15

(DeltaC = 15) > (Maks =  11) maka Maks =15
* u = 3
* v = 5


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 1 2 6 3 5 4
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 843
Selisih biaya penugasan [DeltaC (C-C1)] = -44


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 1 2 4 5 3 6
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 893
Selisih biaya penugasan [DeltaC (C-C1)] = -94


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 1 2 4 6 5 3
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 831
Selisih biaya penugasan [DeltaC (C-C1)] = -32


Posisi awal(a) = 1 2 4 3 5 6
posisi baru(a1)= 1 2 4 3 6 5
Biaya posisi awal(C) = 799
Biaya posisi baru(C1) = 802
Selisih biaya penugasan [DeltaC (C-C1)] = -3


Tukar departemen 3 dengan departemen 5
Posisi baru : a = 1 2 5 3 4 6
```

```
Iterasi : 3
-------------

Set Maks = 0

Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 2 1 5 3 4 6
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 777
Selisih biaya penugasan [DeltaC (C-C1)] = 7

(DeltaC = 7) > (Maks =  0) maka Maks =7
* u = 1
* v = 2


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 5 2 1 3 4 6
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 846
Selisih biaya penugasan [DeltaC (C-C1)] = -62


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 3 2 5 1 4 6
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 793
Selisih biaya penugasan [DeltaC (C-C1)] = -9


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 4 2 5 3 1 6
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 814
Selisih biaya penugasan [DeltaC (C-C1)] = -30


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 6 2 5 3 4 1
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 781
Selisih biaya penugasan [DeltaC (C-C1)] = 3


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 1 5 2 3 4 6
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 818
Selisih biaya penugasan [DeltaC (C-C1)] = -34


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 1 3 5 2 4 6
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 832
Selisih biaya penugasan [DeltaC (C-C1)] = -48
```

```
Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 1 4 5 3 2 6
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 867
Selisih biaya penugasan [DeltaC (C-C1)] = -83


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 1 6 5 3 4 2
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 779
Selisih biaya penugasan [DeltaC (C-C1)] = 5


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 1 2 3 5 4 6
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 865
Selisih biaya penugasan [DeltaC (C-C1)] = -81


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 1 2 4 3 5 6
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 799
Selisih biaya penugasan [DeltaC (C-C1)] = -15


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 1 2 6 3 4 5
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 798
Selisih biaya penugasan [DeltaC (C-C1)] = -14


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 1 2 5 4 3 6
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 862
Selisih biaya penugasan [DeltaC (C-C1)] = -78


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 1 2 5 6 4 3
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 809
Selisih biaya penugasan [DeltaC (C-C1)] = -25


Posisi awal(a) = 1 2 5 3 4 6
posisi baru(a1)= 1 2 5 3 6 4
Biaya posisi awal(C) = 784
Biaya posisi baru(C1) = 832
Selisih biaya penugasan [DeltaC (C-C1)] = -48
```

```
Jumlah Departemen : 6

  Matrik  Jarak Antar Lokasi(d) :
  00 02 06 06 04 06
  02 00 03 03 10 05
  06 03 00 10 07 06
  06 03 10 00 01 01
  04 10 07 01 00 05
  06 05 06 01 05 00


  Matrik Nilai Aliran Perpindahan Bahan antar Departemen(r) :
  00 05 01 02 05 07
  08 00 08 01 07 08
  05 02 00 03 10 07
  04 08 04 00 03 08
  07 07 10 07 00 09
  05 06 04 05 04 00


  Matrik Bobot Antar Depatemen(w) :
  00 13 06 06 12 12
  13 00 10 09 14 14
  06 10 00 07 20 11
  06 09 07 00 10 13
  12 14 20 10 00 13
  12 14 11 13 13 00

  Iterasi : 1
  -------------

  Dept. terpilih (random) = 2


  Posisi lokasi dept. terpilih (dept-2) = 1
  Dept. terpilih (random) = 1

  w(2,1)xd(2,1) = 26
  w(2,1)xd(3,1) = 78
  w(2,1)xd(4,1) = 78
  w(2,1)xd(5,1) = 52
  w(2,1)xd(6,1) = 78
  Posisi lokasi dept. terpilih (dept-1)= 2

  Dept. terpilih (random) = 6

  w(1,6)xd(3,2) + w(2,6)xd(3,1) = 120
  w(1,6)xd(4,2) + w(2,6)xd(4,1) = 120
  w(1,6)xd(5,2) + w(2,6)xd(5,1) = 176
  w(1,6)xd(6,2) + w(2,6)xd(6,1) = 144
  Posisi lokasi dept. terpilih (dept-6)= 3

  Dept. terpilih (random) = 5

  w(6,5)xd(4,3) + w(1,5)xd(4,2) + w(2,5)xd(4,1) = 250
  w(6,5)xd(5,3) + w(1,5)xd(5,2) + w(2,5)xd(5,1) = 267
  w(6,5)xd(6,3) + w(1,5)xd(6,2) + w(2,5)xd(6,1) = 222
  Posisi lokasi dept. terpilih (dept-5)= 6

  Dept. terpilih (random) = 3
```

```
w(5,3)xd(4,6) + w(6,3)xd(4,3) + w(1,3)xd(4,2) + w(2,3)xd(4,1) =
208
w(5,3)xd(5,6) + w(6,3)xd(5,3) + w(1,3)xd(5,2) + w(2,3)xd(5,1) =
277
Posisi lokasi dept. terpilih (dept-3)= 4

Dept. terpilih (random) = 4

w(3,4)xd(5,4) + w(5,4)xd(5,6) + w(6,4)xd(5,3) + w(1,4)xd(5,2) +
w(2,4)xd(5,1) = 244
Posisi lokasi dept. terpilih (dept-4)= 5

Total Biaya  = 820
Total Biaya =820 <  MinC = 1E38 maka MinC = 820
Alternatif Posisi Departemen (a) :
2 1 4 5 6 3

Iterasi : 2
-------------

Dept. terpilih (random) = 5

Posisi lokasi dept. terpilih (dept-5) = 1
Dept. terpilih (random) = 1

w(5,1)xd(2,1) = 24
w(5,1)xd(3,1) = 72
w(5,1)xd(4,1) = 72
w(5,1)xd(5,1) = 48
w(5,1)xd(6,1) = 72
Posisi lokasi dept. terpilih (dept-1)= 2

Dept. terpilih (random) = 3

w(1,3)xd(3,2) + w(5,3)xd(3,1) = 138
w(1,3)xd(4,2) + w(5,3)xd(4,1) = 138
w(1,3)xd(5,2) + w(5,3)xd(5,1) = 140
w(1,3)xd(6,2) + w(5,3)xd(6,1) = 150
Posisi lokasi dept. terpilih (dept-3)= 3

Dept. terpilih (random) = 4

w(3,4)xd(4,3) + w(1,4)xd(4,2) + w(5,4)xd(4,1) = 148
w(3,4)xd(5,3) + w(1,4)xd(5,2) + w(5,4)xd(5,1) = 149
w(3,4)xd(6,3) + w(1,4)xd(6,2) + w(5,4)xd(6,1) = 132
Posisi lokasi dept. terpilih (dept-4)= 6

Dept. terpilih (random) = 2

w(4,2)xd(4,6) + w(3,2)xd(4,3) + w(1,2)xd(4,2) + w(5,2)xd(4,1) =
232
w(4,2)xd(5,6) + w(3,2)xd(5,3) + w(1,2)xd(5,2) + w(5,2)xd(5,1) =
301
Posisi lokasi dept. terpilih (dept-2)= 4

Dept. terpilih (random) = 6
w(2,6)xd(5,4) + w(4,6)xd(5,6) + w(3,6)xd(5,3) + w(1,6)xd(5,2) +
w(5,6)xd(5,1) = 328
Posisi lokasi dept. terpilih (dept-6)= 5

Total Biaya  = 854
```

Hasil Akhir Perbandingan Metode Improvement & Construction Heuristics

---

Jumlah departemen : 6

Matrik Jarak Antar Lokasi(d) :
00 02 06 06 04 06
02 00 03 03 10 05
06 03 00 10 07 06
06 03 10 00 01 01
04 10 07 01 00 05
06 05 06 01 05 00

Matrik Nilai Aliran Perpindahan Bahan Antar Departemen(r) :
00 05 01 02 05 07
08 00 08 01 07 08
05 02 00 03 10 07
04 08 04 00 03 08
07 07 10 07 00 09
05 06 04 05 04 00

Matrik Bobot Antar Departemen(w) :
00 13 06 06 12 12
13 00 10 09 14 14
06 10 00 07 20 11
06 09 07 00 10 13
12 14 20 10 00 13
12 14 11 13 13 00

Metode Improvement Heuristics

Hasil Akhir

Posisi Departemen(a) =
2 6 5 3 4 1

Urutan Departemen =
6 | 1 | 4 | 5 | 3 | 2 |

Total Biaya (C) = 774

Metode Construction Heuristics

Hasil Akhir

Alternatif Posisi Departemen(a) =
2 6 5 3 4 1

Total Biaya (C) = 774

Urutan Departemen =
6 | 1 | 4 | 5 | 3 | 2 |