



**TUGAS AKHIR - KS141501**

**OPTIMASI PENJADWALAN KARYAWAN  
MENGUNAKAN METODE *GOAL PROGRAMMING*  
(STUDI KASUS PT ABC)**

***OPTIMIZATION FOR EMPLOYEES SCHEDULING  
USING GOAL PROGRAMMING APPROACH (CASE  
STUDY : PT ABC)***

**RAKA ANANTA ADHITYA WARMAN  
NRP 0521 1140000 110**

Dosen Pembimbing  
Wiwik Anggraeni, S.Si, M.Kom

DEPARTEMEN SISTEM INFORMASI  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018



**TUGAS AKHIR - KS141501**

**OPTIMASI PENJADWALAN KARYAWAN  
MENGUNAKAN METODE GOAL PROGRAMMING  
(STUDI KASUS PT ABC)**

**RAKA ANANTA ADHITYA WARMAN  
NRP 0521 1140000 110**

**Dosen Pembimbing  
Wiwik Anggraeni, S.Si, M.Kom**

**DEPARTEMEN SISTEM INFORMASI  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018**

**FINAL PROJECT - KS141501**

***OPTIMIZATION FOR EMPLOYEES SCHEDULING USING  
GOAL PROGRAMMING APPROACH (CASE STUDY : PT  
ABC)***

**RAKA ANANTA ADHITYA WARMAN  
NRP 0521 1140000 110**

**Supervisors  
Wiwik Anggraeni, S.Si, M.Kom**

**INFORMATION SYSTEMS DEPARTMENT  
Faculty of Information Technology and  
Communication  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018**

**LEMBAR PENGESAHAN**

**OPTIMASI PENJADWALAN KARYAWAN  
MENGUNAKAN METODE *GOAL PROGRAMMING*  
(STUDI KASUS PT ABC)**

**TUGAS AKHIR**

Disusun untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Departemen Sistem Informasi  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember

Oleh:

**RAKA ANANTA ADHITYA WARMAN**

**NRP. 0521 1140000 110**

Surabaya, Juli 2018

**KEPALA  
DEPARTEMEN SISTEM INFORMASI**

**Dr. Ir. Aris Tjahyanto, M.Kom.**  
**NIP.19650310 199102 1 001**

**LEMBAR PERSETUJUAN**  
**OPTIMASI PENJADWALAN KARYAWAN**  
**MENGGUNAKAN METODE *GOAL PROGRAMMING***  
**(STUDI KASUS PT ABC)**

**TUGAS AKHIR**

Disusun untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Departemen Sistem Informasi  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember

Oleh :

**RAKA ANANTA ADHITYA WARMAN**

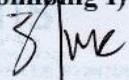
NRP. 0521 1140000 110

Disetujui Tim Penguji : Tanggal Ujian : 12 Juli 2018  
Periode Wisuda : September 2018

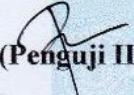
**Wiwik Anggraeni, S.Si, M.Kom**

  
(Pembimbing I)

**Edwin Riksakomara, S.Kom., MT.**

  
(Penguji I)

**Radityo Prasetyanto. W, S.Kom, M.Kom**

  
(Penguji II)



**OPTIMASI PENJADWALAN KARYAWAN  
MENGUNAKAN METODE *GOAL PROGRAMMING*  
(STUDI KASUS PT ABC)**

**Nama Mahasiswa** : RAKA ANANTA A W  
**NRP** : 0521 1140000 110  
**Departemen** : SISTEM INFORMASI FTIF-ITS  
**Dosen Pembimbing** : Wiwik Anggraeni, S.Si, M.Kom

**ABSTRAK**

*Penjadwalan karyawan merupakan salah satu permasalahan di organisasi yang sulit dipecahkan. Jumlah permintaan yang sulit dikendalikan, karakteristik organisasi, adanya absen dan permintaan pribadi untuk libur, serta kualifikasi dan spesialisasi karyawan itu sendiri menjadi beberapa faktor mengapa penjadwalan karyawan sulit dilakukan, termasuk pembuatan jadwal untuk setiap karyawan kedalam jam kerja yang berbeda-beda dalam jangka pendek. Banyaknya faktor yang berbeda menyebabkan masalah penjadwalan karyawan begitu luas dan berbeda-beda disetiap kasus. Permasalahan ini sendiri bukanlah hal baru. Diluar negeri, penelitian tentang masalah ini telah banyak dilakukan dengan bermacam-macam pemodelan. Di Indonesia sendiri, masalah penjadwalan karyawan juga mulai diteliti untuk berbagai macam organisasi walaupun jumlahnya masih bisa dikatakan sedikit.*

*Pada tugas akhir ini, akan dibahas tentang penggunaan Goal Programming (GP) untuk menyelesaikan masalah penjadwalan*

*karyawan di PT ABC khususnya subdivisi x. Metode Goal Programming merupakan metode yang mampu menyelesaikan permasalahan multi objective. Metode ini sesuai dengan kondisi permasalahan penjadwalan pada PT ABC.*

*Perencanaan optimasi penjadwalan ini bertujuan untuk meminimalkan jumlah kekurangan shift kerja serta hari libur setiap karyawan. Berdasarkan prioritas PT ABC, tujuan meminimalkan jumlah kekurangan shift kerja setiap karyawan memiliki prioritas lebih tinggi dibandingkan tujuan meminimalkan jumlah kekurangan hari libur setiap karyawan. Tugas akhir menghasilkan dua buah jadwal berdasarkan skenario 1 yang menggunakan metode preemptive serta skenario 2 yang menggunakan metode weights. Dua hasil jadwal tersebut telah memenuhi seluruh batasan yang diterapkan oleh perusahaan sehingga PT ABC dapat memilih salah satu hasil jadwal untuk diterapkan di subdivisi x.*

***Kata kunci: Optimasi, Penjadwalan Karyawan, Multi Objective, Goal Programming***

**OPTIMIZATION FOR EMPLOYEES SCHEDULING  
USING GOAL PROGRAMMING APPROACH (CASE  
STUDY : PT ABC)**

**Name** : RAKA ANANTA A W  
**NRP** : 0521 1140000 110  
**Departement** : SISTEM INFORMASI FTIF-ITS  
**Supervisor** : Wiwik Anggraeni, S.Si, M.Kom

**ABSTRACT**

*Employees scheduling is one of the most difficult issues in the organization. The number of uncontrollable requests, organizational characteristics, absences, personal requests for holidays, as well as qualifications and specializations themselves are some of the factors why employeeed scheduling is difficult. Including making schedules for each employee into different hours in the short run. The number of different factors causes employeses scheduling problems to be so wide-ranging and different in each case. This problem alone is not new. Research on this issue has been largely done abroad with a variety of modeling. In Indonesia, employees scheduling problems also began to be researched for various organizations although the number is still small.*

*In this final project, we will discuss about the use of Goal Programming (GP) to solve employees scheduling problem in PT ABC especially subdivision x. The method of Goal Programming is a method capable of solving multi objective*

*problems. This method is in accordance with the condition of scheduling problems in PT ABC.*

*This scheduling optimization planning aims to minimize the number of work shifts and the day off shortage for each employee. Based on PT ABC's corporate priorities, the goal of minimizing the number of work shifts for each employee has a higher priority than the goal of minimizing the number of days off for each employee.*

*The final project produced two schedules based on scenario 1 using preemptive method and scenario 2 using weights method. These two schedule have met all the constrains imposed by the company. PT ABC can choose one of the results to be applied in subdivision x.*

***Keywords: Optimization, Employees Scheduling, Multi Objective, Goal Programming***

## KATA PENGANTAR

Puji syukur penulis panjatkan atas kehadiran Tuhan Yang Maha Esa atas segala berkat dan rahmatnya, penulis dapat menyelesaikan buku tugas akhir dengan judul **“OPTIMASI PENJADWALAN KARYAWAN MENGGUNAKAN METODE GOAL PROGRAMMING (STUDI KASUS PT ABC)”** yang merupakan salah satu syarat kelulusan pada Departemen Sistem Informasi, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya.

Secara khusus penulis akan menyampaikan ucapan terima kasih yang sedalam-dalamnya kepada:

1. Ida Sang Hyang Widhi Wasa yang telah memberi segala rahmat dan pencerahan untuk dapat menyelesaikan tugas belajar selama menempuh ilmu di Departemen Sistem Informasi ITS dan telah memberikan kemudahan serta kesehatan selama pengerjaan tugas akhir ini.
2. Kedua orang tua serta saudara penulis yang selalu memberikan doa, dukungan, dan kekuatan dalam penyusunan tugas akhir ini.
3. Ibu Wiwik Anggraeni, S.Si, M.Kom selaku dosen pembimbing yang telah meluangkan waktu dan pikiran ditengah kesibukan beliau untuk membimbing dan mengarahkan penulis dalam mengerjakan tugas akhir ini sampai selesai.
4. Bapak Edwin Riksakomara, S.Kom., MT dan bapak Radityo Prasetyanto. W, S.Kom, M.Kom selaku dosen penguji yang telah memberikan masukan untuk perbaikan tugas akhir ini.
5. Seluruh dosen pengajar, staff, dan karyawan di Departemen Sistem Informasi ITS Surabaya yang telah memberikan ilmu dan bantuan kepada penulis selama ini.

6. Serta semua pihak yang telah membantu dalam pengerjaan Tugas Akhir ini yang belum mampu penulis sebutkan diatas.

Terima kasis atas segala bantuan, dukungan serta doanya. Semoga Tuhan senantiasa memberkati dan membalas kebaikan-kebaikan yang telah diberikan kepada penulis.

Penyusunan laporan ini masih terdapat ketidak sempurnaan dalam pengerjaan tugas akhir ini, sehingga kritik dan saran membangun akan bermanfaat bagi penulis. Semoga buku tugas akhir ini dapat memberikan manfaat bagi pembaca.

## DAFTAR ISI

LEMBAR PENGESAHAN.....	iv
LEMBAR PERSETUJUAN.....	v
ABSTRAK.....	vii
ABSTRACT.....	ix
KATA PENGANTAR .....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR .....	xv
DAFTAR SEGMENT KODE PROGRAM.....	xvii
DAFTAR TABEL.....	xix
BAB I PENDAHULUAN .....	1
1.1. Latar Belakang.....	1
1.2. Perumusan Masalah .....	3
1.3. Batasan Pengerjaan Tugas Akhir .....	3
1.4. Tujuan Tugas Akhir.....	4
1.5. Manfaat Tugas Akhir .....	4
1.6. Relevansi.....	5
BAB II TINJAUAN PUSTAKA.....	7
2.1. Penelitian Sebelumnya.....	7
2.2. Dasar Teori .....	10
2.2.1. Optimasi .....	10
2.2.2. Goal Programming .....	12
2.2.4.1 Fungsi Tujuan .....	14
2.2.4.2 Batasan .....	14
2.2.3. Lingo .....	16
BAB III METODOLOGI PENELITIAN.....	19
3.1. Tahapan Pelaksanaan Tugas Akhir.....	19
3.1.1. Identifikasi Masalah.....	19
3.1.2. Studi Literatur .....	20
3.1.3. Pengumpulan Data dan Informasi.....	20
3.1.4. Pembuatan Model dan Solusi.....	20
3.1.5. Verifikasi dan Validasi Model Optimasi.....	21
3.1.6. Analisa Hasil dan Penarikan Kesimpulan .....	21
3.1.7. Pembuatan Laporan Tugas Akhir.....	22
BAB IV PERANCANGAN .....	23

4.1.	Gambaran Umum Penjadwalan Karyawan PT ABC .....	23
4.2.	Rangkuman Batasan Penjadwalan .....	24
4.3.	Rangkuman Objektif atau <i>Goal</i> dari Penjadwalan ..	25
4.4.	Pemodelan <i>Linier Programming</i> .....	25
4.4.1.	Notasi dan Variabel .....	26
4.4.2.	Variabel Keputusan .....	26
4.4.3.	Perumusan Batasan .....	27
4.5.	Pembentukan Model <i>Goal Programming</i> .....	31
4.5.1.	Perumusan Batasan <i>Goal Programming</i> .....	31
4.5.2.	Perumusan <i>Goal</i> .....	32
4.6.	Perancangan Skenario .....	34
4.6.1.	Skenario 1 .....	34
4.6.2.	Skenario 2 .....	34
<b>BAB V IMPLEMENTASI .....</b>		<b>37</b>
5.1.	Implementasi Batasan dan Variabel Keputusan pada LINGO .....	37
5.2.	Implementasi Skenario pada LINGO .....	40
5.2.1.	Implementasi Skenario 1 .....	41
5.2.2.	Implementasi Skenario 2 .....	41
<b>BAB VI HASIL DAN PEMBAHASAN .....</b>		<b>43</b>
6.1.	Lingkungan Uji Coba .....	43
6.2.	Verifikasi Model .....	44
6.3.	Validasi Hasil Jadwal .....	46
6.4.	Hasil Optimasi <i>Goal Programming</i> .....	51
6.4.1.	Hasil Optimasi Skenario 1 .....	53
6.4.2.	Hasil Optimasi Skenario 2 .....	56
6.5.	Analisa Hasil Uji Coba Skenario .....	60
<b>BAB VII KESIMPULAN DAN SARAN .....</b>		<b>63</b>
7.1.	Kesimpulan .....	63
7.2.	Saran .....	63
<b>DAFTAR PUSTAKA .....</b>		<b>65</b>
<b>BIODATA PENULIS .....</b>		<b>67</b>
A.	LAMPIRAN A .....	A-1

## DAFTAR GAMBAR

Gambar 3.1 Metode Pengerjaan Penelitian .....	19
Gambar 6.1 Hasil Verifikasi Menggunakan Metode Preemptive .....	44
Gambar 6.2 Status Solver dengan Metode Preemptive .....	45
Gambar 6.3 Hasil Verifikasi Menggunakan Metode Weights	45
Gambar 6.4 Status Solver dengan Metode Weights.....	46
Gambar 6.5 Potongan Hasil Output LINGO Skenario 1 .....	52
Gambar 6.6 Potongan Hasil Output LINGO Skenario 2.....	52

*(Halaman ini sengaja dikosongkan)*

## **DAFTAR SEGMENT KODE PROGRAM**

Segment Kode 5.1 Kode Variabel Keputusan.....	37
Segment Kode 5.2 Kode Implementasi Batasan 1 .....	38
Segment Kode 5.3 Kode Implementasi Batasan 2 .....	38
Segment Kode 5.4 Kode Implementasi Batasan 3 .....	38
Segment Kode 5.5 Kode Implementasi Batasan 4 .....	39
Segment Kode 5.6 Kode Implementasi Batasan 5 .....	39
Segment Kode 5.7 Kode Implementasi Batasan 6 .....	39
Segment Kode 5.8 Kode Implementasi Batasan 8 .....	40
Segment Kode 5.9 Kode Implementasi Batasan 7 .....	40
Segment Kode 5.10 Kode Batasan Variabel Binary .....	40
Segment Kode 5.11 Kode Fungsi Tujuan Skenario 1 .....	41
Segment Kode 5.12 Kode Fungsi Tujuan Skenario 2 .....	41
Segment Kode A.1 Kode Program Skenario 1 .....	A-3
Segment Kode A.2 Kode Program Skenario 2.....	A-5

*(Halaman ini sengaja dikosongkan)*

## DAFTAR TABEL

Tabel 2.1 Penelitian Sebelumnya .....	7
Tabel 4.1 Rangkuman Kebutuhan Karyawan Tiap Shift .....	24
Tabel 6.1 Lingkungan Uji Coba Perangkat Keras.....	43
Tabel 6.2 Lingkungan Uji Coba Perangkat Lunak.....	43
Tabel 6.3 Jumlah Karyawan Setiap Shift Tanggal 1-10 Skenario 1.....	47
Tabel 6.4 Jumlah Karyawan Setiap Shift Tanggal 10-20 Skenario 1.....	47
Tabel 6.5 Jumlah Karyawan Setiap Shift Tanggal 21-31 Skenario 1.....	47
Tabel 6.6 Jumlah Karyawan Setiap Shift Tanggal 1-10 Skenario 2.....	47
Tabel 6.7 Jumlah Karyawan Setiap Shift Tanggal 11-20 Skenario 2.....	47
Tabel 6.8 Jumlah Karyawan Setiap Shift Tanggal 21-31 Skenario 2.....	48
Tabel 6.9 Jumlah Libur dan Shift Kerja Karyawan 1-10 Skenario 1.....	49
Tabel 6.10 Jumlah Libur dan Shift Kerja Karyawan 10-20 Skenario 1.....	50
Tabel 6.11 Jumlah Libur dan Shift Kerja Karyawan 21-31 Skenario 1.....	50
Tabel 6.12 Jumlah Libur dan Shift Kerja Karyawan 1-10 Skenario 2.....	50
Tabel 6.13 Jumlah Libur dan Shift Kerja Karyawan 11-20 Skenario 2.....	50
Tabel 6.14 Jumlah Libur dan Shift Kerja Karyawan 21-31 Skenario 2.....	50
Tabel 6.15 Jadwal Karyawan Skenario 1 Tanggal 1-10.....	53
Tabel 6.16 Jadwal Karyawan Skenario 1 Tanggal 11-20.....	54
Tabel 6.17 Jadwal Karyawan Skenario 1 Tanggal 21-31.....	55
Tabel 6.18 Jadwal Karyawan Skenario 2 Tanggal 1-10.....	57
Tabel 6.19 Jadwal Karyawan Skenario 2 Tanggal 11-20.....	58
Tabel 6.20 Jadwal Karyawan Skenario 2 Tanggal 21-31.....	59

*(Halaman ini sengaja dikosongkan)*

# **BAB I**

## **PENDAHULUAN**

Pada bab ini menjelaskan tentang pendahuluan pengerjaan tugas akhir ini, yang meliputi latar belakang, rumusan permasalahan, batasan masalah, tujuan penelitian hingga manfaat yang diambil dari hasil tugas akhir serta relevansi hasil tugas akhir terhadap bidang keilmuan laboratorium.

### **1.1. Latar Belakang**

Penjadwalan karyawan merupakan salah satu permasalahan di organisasi yang sulit dipecahkan. Jumlah permintaan yang sulit dikendalikan, karakteristik organisasi, adanya absen dan permintaan pribadi untuk libur, serta kualifikasi dan spesialisasi karyawan itu sendiri menjadi beberapa faktor mengapa penjadwalan karyawan sulit dilakukan, termasuk pembuatan jadwal untuk setiap karyawan ke dalam jam kerja yang berbeda-beda dalam jangka pendek. Dengan bermacam-macam faktor yang berbeda-beda tersebut menyebabkan masalah penjadwalan karyawan begitu luas dan berbeda-beda di setiap kasus.

Walaupun masalah penjadwalan karyawan bukan masalah baru dan di luar negeri telah banyak dilakukan penelitian tentang masalah ini, di Indonesia jumlah penelitian yang membahas masalah ini masih relatif kecil jumlahnya. Penjadwalan karyawan yang dilakukan menggunakan algoritma tertentu menjadi sangat penting karena melakukan penjadwalan karyawan yang dilakukan secara manual oleh kepala karyawan bisa menghabiskan waktu yang sangat lama dan karena pekerjaan ini memakan tenaga yang tidak sedikit, cenderung dilakukan tanpa mempedulikan faktor-faktor pribadi para karyawan. Tidak peduli dengan keinginan dan permintaan pribadi karyawan bisa menyebabkan menurunnya motivasi dan meningkatkan tingkat stres mereka. Hal ini jelas akan menurunkan performa kerja mereka dan berakibat tidak baik

dimana selanjutnya dapat mempengaruhi citra perusahaan di masyarakat.

Sebagai perusahaan yang memproduksi barang tertentu, PT ABC yang memiliki banyak pesanan tidak luput dari rantai permasalahan ini. PT ABC telah melakukan penjadwalan karyawan yang dilakukan secara manual. Para kepala karyawan akan berkumpul untuk menyusun jadwal semua karyawan di perusahaan.

Penjadwalan karyawan secara otomatis menjadi sangat penting mengingat begitu banyaknya jumlah karyawan dan kondisi yang harus dipenuhi dalam melakukan penjadwalan. Faktor pribadi karyawan pun tidak bisa dilepaskan dari permasalahan penjadwalan karyawan di perusahaan ini agar performa kerja kekaryawan tetap terjaga.

Beberapa model algoritma penjadwalan karyawan dibuat berdasarkan pemrograman linier, algoritma nilai-penalti dan mixed integer programming. Model algoritma optimasi lain yang digunakan adalah algoritma untuk penjadwalan karyawan yang tidak bersiklus. Yang termasuk algoritma tersebut adalah masalah penugasan, pemrograman integer, stochastic programming, pemrograman non-linier, dan lainnya [5].

Di dalam penulisan tugas akhir ini, akan diusulkan sebuah metode pemrograman matematis untuk menyelesaikan masalah penjadwalan karyawan di PT ABC, yang disebut dengan *Goal Programming* (GP). GP adalah salah satu cabang dari model optimasi multi-obyektif yang juga merupakan bagian dari analisis keputusan multi-kriteria [4].

GP telah banyak digunakan untuk menyelesaikan berbagai macam masalah optimasi, termasuk salah satunya adalah masalah penjadwalan karyawan. Penulis [1] menyelesaikan permasalahan penjadwalan tenaga kerja / *Manpower Scheduling Problem (MSP)* menggunakan pendekatan model *Fuzzy Goal Programming*. Penulis [3] mengembangkan model *hierarchical goal programming (HGP)* untuk penjadwalan kerja residen serta staff senior akademik terhadap *outpatient clinic (OC)* didepartemen *physical medicine and rehabilitation (PMR)*.

Banyaknya penggunaan model GP untuk masalah-masalah optimasi seperti penjadwalan karyawan ini dikarenakan kemampuannya untuk bisa mengolah penyelesaian yang memiliki banyak obyektif secara bersamaan [4]. Hal ini sesuai dengan kondisi permasalahan pada PT ABC. Untuk itu dalam tugas akhir ini akan dilakukan optimasi penjadwalan karyawan PT ABC khususnya pada subdivisi x menggunakan metode *Goal Programming*. Dengan dilakukan optimasi penjadwalan ini akan didapat hasilkan jadwal yang optimal sesuai dengan batasan-batasan serta harapan dari perusahaan. Tugas akhir ini juga dapat dijadikan sebagai patokan perusahaan dalam pembuatan jadwal karyawan menggunakan metode *Goal Programming*.

## **1.2. Perumusan Masalah**

Perumusan masalah dalam penelitian tugas akhir ini adalah sebagai berikut:

1. Pembuatan jadwal pada PT ABC masih menggunakan cara manual sehingga membutuhkan waktu yang cukup lama untuk menghasilkan jadwal tiap divisi.
2. Dikarenakan jumlah karyawan PT ABC yang banyak, kompleksitas pembuatan jadwal menjadi tinggi sehingga membutuhkan waktu dan tenaga yang cukup besar dalam pembuatannya.
3. Masih terdapat beberapa permasalahan serta pelanggaran aturan penjadwalan seperti: tidak meratanya jumlah libur karyawan.

## **1.3. Batasan Pengerjaan Tugas Akhir**

Agar tugas akhir lebih terarah dan topik yang dikerjakan tidak meluas, maka diperlukan batasan penelitian. Berikut ini merupakan batasan-batasan yang diterapkan dalam tugas akhir ini:

1. Optimasi penjadwalan yang dikerjakan pada tugas akhir ini hanya meliputi satu subdivisi saja, yaitu subdivisi x, PT ABC.

2. Data yang digunakan adalah data per januari 2018 yang diambil dari subdivisi x, PT ABC.
3. Pembuatan jadwal dibatasi untuk 1 bulan (31 hari) saja yang kemudian dapat digunakan sebagai acuan untuk bulan-bulan berikutnya.
4. Tugas akhirnya belum mencakup bulan dengan jumlah hari selain 31 hari (28, 29 dan 30 hari).
5. Pembuatan jadwal menggunakan *software* LINGO.

#### 1.4. Tujuan Tugas Akhir

Tujuan dilakukannya tugas akhir ini adalah:

1. Memperoleh model yang tepat berdasarkan aturan-aturan penjadwalan perusahaan serta ekspetasi karyawan PT ABC
2. Melakukan optimasi jadwal PT ABC berdasarkan model yang telah dibuat dengan menggunakan metode *Goal Programming* pada program LINGO.
3. Melakukan *verifikasi & validasi* model *Goal Programming* serta hasil jadwal yang telah diperoleh.
4. Meminimalkan pelanggaran terhadap aturan penjadwalan PT ABC serta mengoptimalkan preferensi karyawan.

#### 1.5. Manfaat Tugas Akhir

Berikut ini adalah manfaat yang bisa didapatkan oleh perusahaan dan mahasiswa dengan adanya tugas akhir ini:

1. **Manfaat Bagi Perusahaan:** Hasil optimasi jadwal PT ABC dapat dijadikan salah satu pilihan cara pembuatan jadwal yang lebih efektif dan efisien sehingga meminimalkan pelanggaran-pelanggaran aturan penjadwalan yang telah ditetapkan.
2. **Manfaat Bagi Mahasiswa:** Bagi mahasiswa, dengan adanya penelitian tugas akhir ini maka dapat dijadikan sebagai tambahan pengetahuan mengenai penyelesaian permasalahan penjadwalan karyawan menggunakan metode *Goal Programming*.

### **1.6. Relevansi**

Topik yang diangkat pada tugas akhir ini adalah mengenai optimasi penjadwalan karyawan PT ABC menggunakan metode *Goal Programming*. Jadwal yang dihasilkan diharapkan dapat memenuhi aturan-aturan perusahaan serta dapat mempertimbangkan preferensi karyawan sehingga pola kerja karyawan menjadi lebih baik serta kepuasan dari karyawan tetap terjaga. Topik tersebut berkaitan dengan mata kuliah Sistem Pendukung Keputusan (SPK) serta Riset Operasi Lanjut (ROL) yang merupakan salah satu topik yang diajarkan dalam matakuliah tersebut. Pada pohon penelitian laboratorium Rekayasa Data dan Intelejensi Bisnis (RDIB), topik ini termasuk pada kategori *Modeling & Optimization*.

*(Halaman ini sengaja dikosongkan)*

## BAB II TINJAUAN PUSTAKA

Pada bab ini akan menjelaskan mengenai penelitian sebelumnya dan dasar teori yang dijadikan acuan atau landasan dalam pengerjaan tugas akhir ini. Landasan teori akan memberikan gambaran secara umum dari landasan penjabaran tugas akhir ini.

### 2.1. Penelitian Sebelumnya

Beberapa penelitian sebelumnya yang dijadikan acuan dalam pengerjaan tugas akhir disajikan dalam tabel berikut. Tabel 2.1 adalah penjelasan mengenai penelitian sebelumnya.

**Tabel 2.1 Penelitian Sebelumnya**

No	Penelitian Sebelumnya	
1	Judul Paper	<i>Solving a multi-objective multi-skilled manpower scheduling model by a fuzzy goal programming approach.</i>
	Penulis	Parisa Shahnazari-Shahrezaei, Reza Tavakkoli-Moghaddam, Hamed Kazemipoor
	Tahun	2013
	Deskripsi Umum Penelitian	Penulis melakukan penelitian tentang penyelesaian permasalahan penjadwalan tenaga kerja / <i>Manpower Scheduling Problem (MSP)</i> menggunakan pendekatan model <i>Fuzzy Goal Programming</i> . MSP merupakan permasalahan yang sering ditemukan pada lingkungan perusahaan industri maupun servis. MSP bertujuan untuk membuat jadwal kerja yang sebisa mungkin memenuhi objektif perusahaan serta preferensi tenaga kerja. Model <i>Fuzzy</i> dipilih dikarenakan sering

No	Penelitian Sebelumnya	
		<p>ditemukannya informasi yang kurang jelas pada tiap tujuan perusahaan serta preferensi tenaga kerja. Pengembangan model fuzzy menggunakan pendekatan <i>Bellman</i> dan Zadeh <i>max–min operator</i> serta Li <i>two-phase approach</i>.            Penyelesaian permasalahan MSP dikerjakan berdasarkan data aturan serta regulasi dari pabrik Shabdiz Propylene. Hasil yang didapat menunjukkan bahwa metode <i>two-phase approach</i> menghasilkan solusi yang lebih baik dan efisien dibandingkan metode <i>max–min operator</i>.</p>
	Persamaan dengan Tugas Akhir	<p>Permasalahan pada tugas akhir serta paper membahas tentang penjadwalan tenaga kerja / <i>Manpower Scheduling Problem (MSP)</i>.            Menggunakan pendekatan metode <i>Goal Programming</i>.</p>
	Perbedaan dengan Tugas Akhir	<p>Paper menggunakan pendekatan <i>fuzzy</i> sebagai tambahan.</p>
2	Judul Paper	<p><i>A goal programming model for scheduling residents in an anesthesia and reanimation department</i></p>
	Penulis	<p>M. Güray Güler, Kadir İdi'n, Emine Yilmaz Güler</p>
	Tahun	<p>2013</p>
	Deskripsi umum penelitian	<p>Penulis mengembangkan model goal programming berdasarkan penjadwalan residen departemen <i>Anesthesia</i> dan <i>Reanimation</i> di <i>Bezmialem Vakif University Medical School (BUMS)</i>. Aturan <i>hard constraint</i> seperti jumlah residen dalam <i>shift</i>, harus terpenuhi.</p>

No	Penelitian Sebelumnya	
		<p>Sedangkan aturan <i>soft constraint</i> bisa dilanggar dengan ketentuan penalti yang didapatkan menggunakan <i>Analytical Hierarchy Process</i> (AHP).</p> <p>Model yang dihasilkan oleh penulis telah digunakan oleh masing-masing departemen. Hasil penerapan model ini telah menunjukkan peningkatan serta menghasilkan jadwal yang lebih baik dengan usaha yang minimal.</p>
	<p>Persamaan dengan Tugas Akhir</p>	<p>Menggunakan pendekatan metode <i>Goal Programming</i>.</p> <p>Permasalahan pada tugas akhir serta paper membahas tentang penjadwalan tenaga kerja.</p>
	<p>Perbedaan dengan Tugas Akhir</p>	<p>Perbedaan pada studi kasus dimana Tugas Akhir melakukan penerapan penjadwalan karyawan pada perusahaan PT ABC sedangkan paper menerapkan di universitas <i>Bezmialem Vakif University Medical School (BUMS)</i>.</p>
3	Judul Paper	<i>A hierarchical goal programming model for scheduling the outpatient clinics</i>
	Penulis	M. Güray Güler
	Tahun	2013
	Deskripsi umum penelitian	<p>Penulis Mengembangkan model <i>hierarchical goal programming (HGP)</i> untuk penjadwalan kerja residen serta staff senior akademik terhadap <i>outpatient clinic (OC)</i> didepartemen <i>physical medicine and rehabilitation (PMR)</i>. Saat ini jadwal masih dibuat secara manual. Departement masih merasa tidak puas dengan penjadwalan saat ini dikarenakan sering terjadinya perubahan jadwal serta</p>

No	Penelitian Sebelumnya	
		ketidak mampuan jadwal memenuhi preferensi dari residen. Dengan penerapan model yang dibuat oleh penulis paper, departemen dapat mengurangi frekuensi perubahan jadwal dalam satu bulan, memenuhi preferensi dari residen dan kebutuhan departemen serta mengatur jadwal residen sehingga dapat bekerja dengan spesialis yang berbeda hingga mereka lulus.
	Persamaan dengan Tugas Akhir	Menggunakan pendekatan metode <i>Goal Programming</i> . Permasalahan pada tugas akhir serta paper membahas tentang penjadwalan tenaga kerja.
	Perbedaan dengan Tugas Akhir	Paper menggunakan pendekatan AHP sebagai tambahan.

## 2.2. Dasar Teori

Konsep-konsep atau teori yang memiliki keterkaitan dengan tugas akhir meliputi optimasi, *goal programming* serta program LINGO.

### 2.2.1. Optimasi

Optimasi adalah tindakan untuk mendapatkan hasil yang terbaik. Tujuan optimasi ini dilakukan untuk meminimalkan upaya yang dibutuhkan atau untuk memaksimalkan hasil yang diinginkan. Dalam melakukan optimasi perlu menyatakan fungsi dari variabel keputusan, untuk dapat mendefinisikan proses tersebut mendapatkan nilai maksimum atau minimum dari suatu fungsi [9].

Tugas pengambilan keputusan memerlukan memilih antara berbagai alternatif. Pilihan ini diatur oleh keinginan kita untuk membuat keputusan "terbaik". Ukuran kebaikan alternatif digambarkan oleh fungsi tujuan atau indeks kinerja. teori

optimasi dan metode berurusan dengan memilih alternatif terbaik dalam arti fungsi tujuan yang diberikan.

Suatu permasalahan optimasi disebut nonlinear jika fungsi tujuan dan kendalanya mempunyai bentuk nonlinear pada salah satu atau keduanya. Optimasi merupakan masalah yang berhubungan dengan keputusan yang terbaik, maksimum, minimum dan memberikan cara penentuan solusi yang memuaskan.

Berikut ini akan dijelaskan langkah – langkah yang harus dilakukan untuk menyelesaikan suatu persoalan optimasi dengan pendekatan penyelidikan operasi [10].

- a) Langkah pertama, memahami persoalan nyata yang akan dioptimalkan, dalam hal ini disebut permasalahan optimasi.
- b) Langkah kedua, membuat model matematika yang merepresentasikan karakteristik dinamik permasalahan optimasi tersebut. Dalam membuat model matematika, langkah – langkah yang harus dilakukan adalah sebagai berikut:
  - 1) Mendefinisikan variabel – variabel dalam permasalahan. Variabel merupakan besaran yang nilainya dapat berubah – ubah.
  - 2) Mendefinisikan parameter – parameter dalam permasalahan. Parameter merupakan besaran yang nilainya tertentu dan tidak berubah – ubah pada kondisi yang standar.
  - 3) Mendefinisikan tujuan yang akan dicapai dalam permasalahan optimasi tersebut. Tujuan yang ingin dicapai dinyatakan dalam fungsi tujuan.
  - 4) Mendefinisikan kendala – kendala yang harus dipenuhi dalam mencapai tujuan. Kendala juga dinyatakan dalam fungsi kendala. Model matematika suatu permasalahan optimasi meliputi fungsi tujuan dan fungsi – fungsi kendala. .
- c) Langkah ketiga, menyesuaikan model matematika dengan menggunakan metode tertentu, yaitu menentukan nilai variabel – variabel dalam permasalahan optimasi yang mengoptimalkan fungsi tujuan.

- d) Langkah keempat, melakukan interpretasi penyelesaian berdasarkan hasil penyelesaian model.
- e) Langkah kelima, melakukan analisa sensitivitas, yaitu mempertimbangkan parameter – parameter apa saja yang berpengaruh terhadap hasil penyelesaian model. Analisa sensitivitas digunakan saat terjadi perubahan parameter, dimana perhitungan optimasinya tidak perlu mulai dari awal proses tetapi cukup menggunakan penyelesaian terakhir untuk memperoleh penyelesaian yang baru.

### 2.2.2. Goal Programming

*Goal programming* merupakan perluasan dari program linear (*linear programming*). *Goal programming* adalah salah satu metode matematis yang dipakai sebagai dasar mengambil keputusan untuk menganalisa dan mencari solusi optimal yang melibatkan banyak tujuan (*multi objektif*) sehingga diperoleh solusi yang optimal. Pendekatan dasar dari *goal programming* adalah untuk menetapkan suatu tujuan yang dinyatakan dengan angka tertentu untuk setiap tujuan, merumuskan suatu fungsi tujuan, dan kemudian mencari penyelesaian dengan meminimumkan jumlah (tertimbang) penyimpangan-penyimpangan dari fungsi tujuan [11]. Untuk melakukan metode *Goal Programming* terlebih dahulu menentukan variabel keputusan, fungsi tujuan dan batasan dari permasalahan yang diangkat [12].

*Goal Programming* (selanjutnya disingkat dengan GP) telah banyak digunakan untuk menyelesaikan berbagai macam masalah optimasi, termasuk salah satunya adalah masalah penjadwalan karyawan. Banyaknya penggunaan model ini untuk masalah-masalah optimasi seperti penjadwalan karyawan ini dikarenakan oleh kemampuannya untuk bisa mengolah penyelesaian yang memiliki banyak obyektif secara bersamaan. Penjadwalan karyawan merupakan masalah rumit dan kebanyakan memiliki banyak obyektif dan sering dikelompokkan dalam obyektif yang tidak sebanding. Obyektif yang tidak sebanding, dalam GP muncul saat variabel deviasi, yang diukur di unit yang berbeda-beda, dijumlahkan secara

langsung. GP menggunakan konstanta normalisasi untuk menyelesaikan masalah ini.

Pada awalnya, formulasi GP mengubah deviasi yang tidak diinginkan menjadi sejumlah tingkat prioritas, dengan minimasi deviasi di prioritas yang lebih tinggi menjadi lebih penting dari pada deviasi lain di tingkat prioritas yang lebih rendah. Yang perlu diperhatikan adalah deviasi yang diukur menggunakan unit yang berbeda-beda tidak bisa dijumlahkan secara langsung karena tidak sebanding. Karena itu, deviasi yang tidak diinginkan dikalikan terlebih dahulu menggunakan konstanta normalisasi agar bisa dibandingkan secara langsung. Solusi dari GP lebih mengarah ke solusi yang paling memuaskan daripada solusi yang optimal karena memenuhi tujuan-tujuan tertentu juga [16].

### **Konsep Dasar Goal Programming:**

Berikut adalah istilah-istilah yang sering digunakan dalam *Goal Programming*[13].

1. Variabel keputusan (*Decision variables*) yaitu seperangkat variabel yang tidak diketahui yang akan dicari nilainya.
2. Nilai sisi kanan (*Right Hand Side Values* atau RHS) yaitu nilai-nilai yang biasanya menunjukkan ketersediaan sumber daya yang akan ditentukan kekurangan atau kelebihan penggunaannya.
3. Tujuan (*Goal*) yaitu keinginan untuk meminimumkan angka penyimpangan dari suatu nilai RHS pada suatu goal constraint tertentu.
4. Kendala tujuan (*Goal Constraint*) merupakan sinonim dari istilah goal equation, yaitu suatu tujuan yang diekspresikan dalam persamaan matematika yang memasukkan variabel simpangan.
5. *Preemptive priority factor* yaitu suatu sistem urutan yang menunjukkan banyaknya tujuan dalam model yang memungkinkan tujuan-tujuan disusun secara ordinal dalam model *Linear Goal Programming*.
6. Variabel simpangan (*Deviational variables*) yaitu variabel-variabel yang menunjukkan kemungkinan

penyimpangan negatif atau positif dari suatu nilai RHS kendala tujuan. Variabel-variabel ini serupa dengan slack variabel dalam *Linear Programming*.

7. Bobot (*Differential Weight*) yaitu bobot yang diekspresikan dengan angka kardinal dan digunakan untuk membedakan variabel simpangan didalam suatu tingkat prioritas.

Untuk menentukan perumusan metode *Goal Programming*, menentukan formula yang digunakan terdiri dari fungsi tujuan, variable keputusan dan batasan permasalahan.

#### 2.2.4.1 Fungsi Tujuan

Fungsi tujuan merupakan perumusan tujuan secara matematis untuk mendapatkan hasil yang maksimal atau minimal sesuai dengan tujuan yang telah ditentukan sebelumnya. Dalam penulisan fungsi tujuan dapat dituliskan dengan formula sebagai berikut:

$$\textit{Maximize} = nX_1 + nX_2 + \dots + nX_n \quad (2.1)$$

$$\textit{Minimize} = nX_1 + nX_2 + \dots + nX_n \quad (2.2)$$

Perumusan *maximize* digunakan untuk tujuan yang fungsinya adalah meningkatkan tujuan, sedangkan perumusan *minimize* untuk tujuan yang fungsinya adalah meminimalkan tujuan.

Dimana:

$n$  = nilai positif dari variabel

$X_1, X_2, \dots, X_n$  = variabel keputusan yang digunakan untuk mencapai fungsi tujuan

#### 2.2.4.2 Batasan

Batasan merupakan variable yang menjadi pembatas dalam mencapai fungsi tujuan. Dalam penulisan batasan perlu

memperhatikan mengenai variabel yang menjadi batasan dalam mencapai fungsi tujuan. Penulisan batasan dapat ditulis secara matematis dengan formula seperti berikut :

$$\text{Batasan 1} = nX_1 + nX_2 + \dots + nX_n \geq p \quad (2.3)$$

$$\text{Batasan 2} = nX_1 + nX_2 + \dots + nX_n \leq q \quad (2.4)$$

Dimana:

Batasan 1&2 = variabel-variabel yang menjadi batasan dalam mencapai fungsi tujuan

n = nilai positif dari variabel

$X_1, X_2, \dots, X_n$  = variabel keputusan yang digunakan untuk mencapai fungsi tujuan

p & q = nilai konstanta yang menjadi pembatas pada masing-masing batasan

Fungsi tujuan dan batasan pada penjelasan di atas merupakan formula dalam bentuk metode *Linear Programming*. Langkah selanjutnya adalah mengubah formula tersebut kedalam bentuk *Goal Programming*. Dalam perubahan formula, terdapat penambahan variabel yang dinamakan variabel deviasi. Dengan adanya variabel deviasi digunakan untuk membuat fungsi tujuan baru dengan meminimalkan variabel deviasi yang sudah ditentukan [12]. Penulisan rumus fungsi tujuan dengan meminimalkan variabel deviasi adalah sebagai berikut:

$$\text{Minimize } Z = \sum_{i=0}^m w_i P_i (d_i^+ + d_i^-) \quad (2.5)$$

Dimana:

$P_i$  = level prioritas dari setiap tujuan

$w_i$  = konstanta dari non-negative untuk pembobotan

Dengan Batasan:

$$\sum_{j=i}^n a_{ij}x_{ij} + d_i^- - d_i^+ = b_i (i = 1, 2, \dots, m) \quad (2.6)$$

$$x_{ij}, d_i^-, d_i^+ \geq 0$$

$$i = 1, 2, \dots, m ; j = 1, 2, \dots, n$$

$$P_1 > P_2 > \dots > P_n$$

Dimana:

$d_i^-$  dan  $d_i^+$  = variabel deviasi untuk setiap j, pada tujuan  $b_i$

$x_{ij}$  = variabel keputusan

$a_{ij}$  = variabel keputusan yang koefisien

### 2.2.3. Lingo

Lingo adalah perangkat lunak yang digunakan untuk menyelesaikan masalah-masalah pemrograman linier, *non linier* dan *integer*. Lingo sudah banyak digunakan oleh perusahaan-perusahaan untuk membantu membuat perencanaan distribusi yang bertujuan untuk mendapatkan keuntungan yang optimum dan waktu yang minimum. Selain itu, Lingo juga digunakan dalam pengambilan keputusan dalam perencanaan produksi, transportasi, keuangan, alokasi saham, penjadwalan, inventarisasi, pengaturan model, alokasi daya dan lain-lain.

Sistem Lingo telah menjadi pilihan utama dalam penyelesaian yang cepat dan mudah, terutama dalam masalah optimasi persamaan matematika. [15]

Untuk menggunakan *software* Lingo ada beberapa tahapan yang perlu dilakukan, yaitu :

1. Merumuskan masalah dalam kerangka program linier.
2. Menuliskan dalam persamaan matematika.
3. Merumuskan rumusan ke dalam Lingo dan mengeksekusinya.
4. Interpretasi keluaran Lingo.

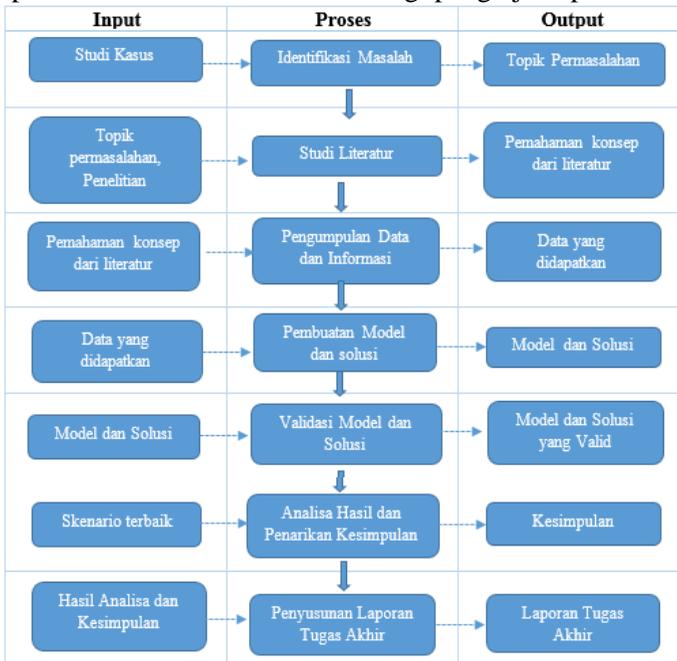
*(Halaman ini sengaja dikosongkan)*

## BAB III METODOLOGI PENELITIAN

Pada bab ini akan dijelaskan mengenai tahapan-tahapan dalam penyelesaian permasalahan tugas akhir ini. Metodologi digunakan sebagai panduan pengerjaan tugas akhir agar lebih sistematis, teratur dan terarah.

### 3.1. Tahapan Pelaksanaan Tugas Akhir

Tahapan Pelaksanaan Tugas Akhir meliputi input proses dan output. Gambar 3.1 adalah metodologi pengerjaan penelitian.



**Gambar 3.1 Metode Pengerjaan Penelitian**

#### 3.1.1. Identifikasi Masalah

Tahap ini dilakukan untuk menjabarkan permasalahan-permasalahan yang terdapat pada penjadwalan karyawan

perusahaan PT ABC saat ini. Identifikasi masalah dilakukan dengan memahami proses penjadwalan yang dilakukan oleh PT ABC. Beberapa permasalahan yang terdiksi yaitu, Jadwal karyawan perusahaan ABC masih dibuat secara manual, Kompleksitas pembuatan jadwal yang tinggi, serta masih sering ditemui preferensi dari karyawan yang dilanggar seperti jumlah libur antar karyawan yang tidak merata.

### **3.1.2. Studi Literatur**

Untuk menjawab serta memecahkan permasalahan yang dijabarkan sebelumnya, diperlukan proses peninjauan pustaka. Proses ini dilakukan untuk mencari sumber-sumber informasi serta dasar teori yang dapat digunakan untuk memecahkan masalah.

Berdasarkan tinjauan pustaka, pendekatan yang dapat digunakan untuk menyelesaikan permasalahan penjadwalan karyawan PT ABC adalah metode *Goal Programming*. Metode ini dipilih dikarenakan dapat digunakan untuk menyelesaikan permasalahan dengan fungsi objektif lebih dari satu [4] serta beberapa penelitian sebelumnya yang dibahas telah sukses mengimplementasikan varian *Goal Programming* untuk menyelesaikan permasalahan penjadwalan karyawan.

### **3.1.3. Pengumpulan Data dan Informasi**

Kemudian dilakukan proses pengumpulan data dan informasi yang dibutuhkan untuk menunjang pembuatan tugas akhir ini. Beberapa data serta informasi yang diperlukan adalah:

1. Data jadwal subdivisi x yang telah dibuat oleh PT ABC pada periode januari 2018.
2. Data aturan-aturan penjadwalan di PT ABC serta daftar preferansi karyawan yang dapat digunakan batasan dalam pembuatan model matematis.

### **3.1.4. Pembuatan Model dan Solusi**

Pada tahap ini akan dilakukan formulasi semua batasan dan parameter yang dibutuhkan dalam proses analisis masalah yang ada menjadi model matematis. Tujuan dari permodelan data ini

adalah untuk menentukan variabel keputusan, fungsi tujuan dan batasan dari permasalahan yang didapatkan. Model yang akan digunakan terlebih dahulu adalah model *Linear Programming* yang kemudian diubah menjadi model *Goal Programming*. Setelah didapatkan hasil model *Goal Programming*, langkah selanjutnya adalah mentranslasikan model tersebut kedalam kode program LINGO. Terdapat dua buah skenario yaitu skenario 1 dan 2 yang masing-masing menggunakan metode penyelesaian yang berbeda yaitu metode *preemptive* dan *weights*. Kedua skenario ditranslasikan kedalam kode program LINGO yang menghasilkan dua kode yang memiliki perbedaan pada fungsi tujuan. Setelah kedua kode program dijalankan, maka akan dihasilkan dua jadwal subdivisi x PT ABC berdasarkan skenario 1 dan 2.

### **3.1.5. Verifikasi dan Validasi Model Optimasi**

Verifikasi dilakukan untuk memastikan apakah program yang telah dibuat sudah bebas dari error melalui uji coba. Hal ini dilakukan dengan melihat pada program yang telah dibuat, apakah ada kesalahan yang ditandai dengan adanya tanda *error*. Verifikasi juga dapat dilakukan dengan menjalankan fungsi *solver* pada program LINGO. Jika tidak terdapat pesan *error* maka model *goal programming* dapat dikatakan telah terverifikasi. Validasi dilakukan dengan membandingkan hasil jadwal skenario 1 dan 2 dengan batasan-batasan yang ada. Jika semua batasan terpenuhi, maka hasil jadwal skenario 1 dan 2 telah *valid*.

### **3.1.6. Analisa Hasil dan Penarikan Kesimpulan**

Tahapan ini merupakan dilakukan setelah hasil jadwal dinyatakan valid. Proses yang dilakukan adalah menganalisa hasil keluaran yang didapatkan dan dibandingkan dengan hasil validasi. Hal ini dilakukan untuk mendapatkan kekurangan maupun kelebihan dari tugas akhir sehingga didapatkan kesimpulan yang tepat. Dari proses tersebut diharapkan dapat dihasilkan saran terhadap tugas akhir atau penelitian selanjutnya agar memberikan luaran yang jauh lebih baik.

### 3.1.7. Pembuatan Laporan Tugas Akhir

Tahap terakhir adalah penyusunan tugas akhir untuk melakukan dokumentasi terhadap proses pengerjaan tugas akhir. Seluruh pelaksanaan atau pengerjaan tugas akhir didokumentasikan dalam sebuah buku Tugas Akhir (TA) dengan mengikuti format yang telah ditetapkan oleh laboratorium Rekayasa Data dan Intelegensia Bisnis (RDIB) serta yang berlaku di Departemen Sistem Informasi ITS.

Di dalam laporan Tugas Akhir tersebut mencakup:

**a. Bab I Pendahuluan**

Dalam bab ini dijelaskan mengenai latar belakang, rumusan dan batasan masalah, tujuan dan manfaat pengerjaan tugas akhir.

**b. Bab II Tinjauan Pustaka**

Dijelaskan mengenai penelitian-penelitian serupa yang telah dilakukan serta teori – teori yang menunjang permasalahan yang dibahas pada tugas akhir ini.

**c. Bab III Metodologi**

Dalam bab ini dijelaskan mengenai tahapan – tahapan apa saja yang harus dilakukan dalam pengerjaan tugas akhir.

**d. Bab IV Perancangan**

Bab ini berisi tentang rancangan penelitian yang dilakukan, perancangan pemodelan yang digunakan dalam pengerjaan tugas akhir.

**e. Bab V Implementasi**

Bab yang berisi tentang setiap langkah yang dilakukan dalam implementasi metodologi yang digunakan dalam tugas akhir.

**f. Bab VI Hasil dan Pembahasan**

Bab yang berisi tentang hasil analisis dan pembahasan dalam penyelesaian permasalahan yang dibahas pada pengerjaan tugas akhir.

**g. Bab VII Kesimpulan dan Saran**

Bab yang berisi kesimpulan dan saran yang ditujukan untuk kelengkapan penyempurnaan tugas akhir ini

## **BAB IV PERANCANGAN**

Pada bab ini, akan dijelaskan mengenai rancangan bagaimana penelitian akan dilakukan. Terdiri dari penjelasan mengenai subjek dan objek penelitian, perumusan model *linier programming*, perumusan model *goal programming* serta pembuatan skenario.

### **4.1. Gambaran Umum Penjadwalan Karyawan PT ABC**

Di PT ABC, penjadwalan karyawan dilaksanakan oleh bagian kepegawaian. Penjadwalan dilakukan untuk mengatur jam kerja karyawan dilima divisi. Tiap divisi memiliki karyawan yang berbeda-beda, namun aturan penjadwalan pada setiap divisi sama. Hal ini dikarenakan penjadwalan dilakukan bersama-sama oleh para kepala subdivisi serta bagian administrasi kepegawai. Jadwal dibuat untuk jangka waktu satu bulan.

Jadwal setiap subdivisi memiliki pola yang sama karena ada aturan yang telah ditentukan oleh kepala kepegawaiaan. Karena setiap subdivisi memiliki kecenderungan pola yang sama, kali ini data diambil dari salah satu subdivisi saja yaitu subdivisi x. Pada subdivisi x, jumlah karyawan yang tersedia adalah 31 pekerja aktif. Penjadwalan pada subdivisi x menggunakan sistem *shift* dimana setiap harinya terdapat tiga buah *shift* yaitu pagi, siang dan malam. Untuk subdivisi x keseluruhan karyawan yang dibutuhkan pada pagi hari adalah 10 orang. Sedangkan kebutuhan pekerja pada siang serta sore hari adalah 7 dan 8 orang. Jam pagi dan siang memiliki jumlah waktu 7 jam setiap harinya yang mana secara berturut-turut dimulai pukul 07.00 – 14.00 dan pukul 14.00 – 21.00. Jam malam berjangka waktu 10 jam per harinya dimulai dari pukul 21.00 sampai dengan pukul 07.00.

Pada Perusahaan PT ABC, setiap harinya dihitung sebagai hari kerja. Setiap karyawan memiliki jadwal yang berbeda-beda. Masing-masing karyawan hanya boleh mengisi satu *shift* diantara *shift* pagi, siang, sore atau libur setiap harinya. Tabel

4.1 menunjukkan rangkuman peninjauan kebutuhan jam kerja karyawan di subdivisi x.

**Tabel 4.1 Rangkuman Kebutuhan Karyawan Tiap Shift**

<b>Jumlah karyawan</b>	31 Karyawan
<b>Kebutuhan <i>shift</i> pagi</b>	10 karyawan
<b>Kebutuhan <i>shift</i> siang</b>	7 karyawan
<b>Kebutuhan <i>shift</i> malam</b>	8 karyawan
<b>Jadwal jam pagi</b>	07.00 – 14.00 (7 jam)
<b>Jadwal jam sore</b>	14.00 – 21.00 (7 jam)
<b>Jadwal Jam malam</b>	21.00 – 07.00 (10 jam)

#### **4.2. Rangkuman Batasan Penjadwalan**

Rangkuman batasan ini merupakan interpretasi dari gambaran umum penjadwalan PT ABC serta batasan-batasan yang telah ditetapkan oleh perusahaan. Batasan-batasan yang digunakan perusahaan PT ABC dalam pembuatan jadwal adalah sebagai berikut:

- a. Kebutuhan karyawan pada setiap *shift* (pagi, siang dan malam) harus terpenuhi setiap harinya. Kebutuhan karyawan *shift* pagi minimal 10 pekerja, kebutuhan karyawan *shift* siang minimal 7 pekerja dan kebutuhan karyawan *shift* malam minimal 8 pekerja.
- b. Setiap karyawan hanya boleh mengisi satu *shift* (jam kerja) atau libur setiap harinya.
- c. Setiap karyawan harus memiliki minimal satu *shift* pagi, satu *shift* siang dan satu *shift* malam setiap bulannya.
- d. Jadwal setiap karyawan menghindari pola *shift* malam yang diikuti *shift* pagi dihari berikutnya.
- e. Perusahaan mengusahakan agar setiap karyawan mendapatkan minimal satu kali libur dan maksimal dua kali libur setiap minggunya.
- f. Perusahaan mengusahakan agar jadwal setiap karyawan tidak memiliki pola libur yang berderet (hari

libur yang diikuti kembali oleh libur pada hari berikutnya).

- g. Setiap karyawan memiliki kesempatan libur minimal empat kali setiap bulannya.
- h. Perusahaan menginginkan agar setiap karyawan memiliki *shift* kerja (pagi, siang atau malam) minimal 26 kali dalam satu bulan.

### **4.3. Rangkuman Objektif atau *Goal* dari Penjadwalan**

Perusahaan PT ABC memiliki dua buah objektif atau *goal* yang ingin dicapai dalam pembuatan jadwal karyawan ini.

- a. Perusahaan menginginkan agar setiap karyawan dapat bekerja secara maksimal. Untuk itu, perusahaan tidak menginginkan karyawan bekerja kurang dari 26 *shift* setiap bulannya.
- b. Untuk mengimbangi jadwal kerja masing-masing karyawan yang cukup ketat, perusahaan menginginkan agar setiap karyawan mendapatkan hari libur lebih dari 4 setiap bulannya.

Objektif atau *goal* a merupakan preferensi dari perusahaan sedangkan objektif atau *goal* b merupakan preferensi dari karyawan. Bagi perusahaan, objektif a memiliki prioritas yang lebih tinggi ketimbang objektif b dimana perusahaan menginginkan objektif jumlah *shift* kerja terpenuhi terlebih dahulu.

### **4.4. Pemodelan *Linier Programming***

Berdasarkan informasi yang didapat pada 4.1 dan 4.2, permasalahan penjadwalan perusahaan PT ABC akan ditranslasikan kedalam bentuk model matematis *Linier Programming* terlebih dahulu. Model *Linier Programming* akan dijabarkan menjadi dua bagian yaitu variabel keputusan serta batasan permasalahan. Untuk objektif/goal akan dibahas pada bagian subbab 4.5.

#### 4.4.1. Notasi dan Variabel

Notasi yang digunakan dalam membuat permodelan adalah sebagai berikut:

- $i$  = indeks tanggal ( $i = 1 \dots n$ )
- $j$  = indeks karyawan ( $j = 1 \dots m$ )
- $n$  = jumlah hari pada jadwal ( $n = 31$ )
- $m$  = jumlah karyawan yang tersedia ( $m = 31$ )

#### 4.4.2. Variabel Keputusan

Variabel keputusan yang digunakan dalam formulasi permodelan adalah sebagai berikut:

$$X_{(i,j)} = \begin{cases} 1, & \text{jika karyawan } j \text{ ditugaskan } \textit{shift} \text{ pagi pada tanggal } i \\ 0, & \text{jika karyawan } j \text{ tidak ditugaskan } \textit{shift} \text{ pagi pada tanggal } i \end{cases}$$

$$Y_{(i,j)} = \begin{cases} 1, & \text{jika karyawan } j \text{ ditugaskan } \textit{shift} \text{ siang pada tanggal } i \\ 0, & \text{jika karyawan } j \text{ tidak ditugaskan } \textit{shift} \text{ siang pada tanggal } i \end{cases}$$

$$Z_{(i,j)} = \begin{cases} 1, & \text{jika karyawan } j \text{ ditugaskan } \textit{shift} \text{ malam pada tanggal } i \\ 0, & \text{jika karyawan } j \text{ tidak ditugaskan } \textit{shift} \text{ malam pada tanggal } i \end{cases}$$

$$L_{(i,j)} = \begin{cases} 1, & \text{jika karyawan } j \text{ libur pada tanggal } i \\ 0, & \text{jika karyawan } j \text{ tidak libur pada tanggal } i \end{cases}$$

Variabel keputusan ini akan menunjukkan peran masing-masing karyawan setiap harinya. Peran tersebut terdiri dari variabel  $X$  yang melambangkan karyawan bekerja pada *shift* pagi, variabel yang melambangkan karyawan berkerja pada *shift* siang, variabel  $Z$  yang melambangkan karyawan berkerja pada *shift* malam serta  $L$  yang melambangkan karyawan yang libur. Masing-masing variabel memiliki dua indeks yaitu  $i$  dan  $j$ . Indeks  $i$  melambangkan tanggal kerja karyawan sedangkan indeks  $j$  melambangkan nomor karyawan. Masing-masing variabel keputusan, hanya memiliki satu nilai antara nol dan satu. Jika dilihat dari contoh variabel keputusan  $Z_{(2,1)} = 1$  maka dapat dibaca sebagai “Pada tanggal dua, karyawan nomer satu bekerja pada *shift* malam”. Sedangkan jika nilai variabel keputusan diganti menjadi nol ( $Z_{(2,1)} = 0$ ) maka dapat dibaca

sebagai “Pada tanggal dua, karyawan nomer satu tidak bekerja pada shift malam”.

#### 4.4.3. Perumusan Batasan

Pada proses ini, batasan-batasan penjadwalan yang telah dirangkum kedalam subbab 4.2 akan digunakan sebagai acuan dalam pembuatan batasan program linier. Batasan yang ada dalam model antara lain sebagai berikut.

**Batasan 1** : Kebutuhan karyawan pada setiap *shift* (pagi, siang dan malam) harus terpenuhi setiap harinya. Kebutuhan karyawan *shift* pagi minimal 10 pekerja, kebutuhan karyawan *shift* siang minimal 7 pekerja dan kebutuhan karyawan *shift* malam minimal 8 pekerja.

- a. Kebutuhan karyawan *shift* pagi terpenuhi (minimal 10 pekerja)

$$\sum_{j=1}^m X_{(i,j)} \geq 10, j = 1 \dots m \quad (4.1)$$

untuk semua  $i = 1 \dots n$

- b. Kebutuhan karyawan *shift* siang terpenuhi (minimal 7 pekerja)

$$\sum_{j=1}^m Y_{(i,j)} \geq 7, j = 1 \dots m \quad (4.2)$$

untuk semua  $i = 1 \dots n$

- c. Kebutuhan karyawan *shift* malam terpenuhi (minimal 8 pekerja)

$$\sum_{j=1}^m Z_{(i,j)} \geq 8, j = 1 \dots m \quad (4.3)$$

untuk semua  $i = 1 \dots n$

Dikarenakan subdivisi x memiliki jumlah minimal pekerja yang harus dialokasikan pada *shift* pagi, siang dan sore setiap harinya, maka kebutuhan ini dituangkan kedalam batasan 1. Setiap harinya, subdivisi x membutuhkan minimal 10 karyawan untuk bekerja pada *shift* pagi, 7 karyawan untuk bekerja pada *shift* siang, dan 8 karyawan untuk bekerja pada *shift* malam.

**Batasan 2** : Setiap karyawan hanya boleh mengisi satu *shift* (jam kerja) atau libur setiap harinya.

$$X_{(i,j)} + Y_{(i,j)} + Z_{(i,j)} + L_{(i,j)} = 1 \quad (4.4)$$

untuk semua  $i=1 \dots m$  dan  $j = 1 \dots n$

Batasan ini menggambarkan situasi dimana setiap harinya karyawan hanya boleh melakukan salah satu kegiatan diantara bekerja pada *shift* pagi, *shift* siang, *shift* malam atau libur. Karyawan tidak diperbolehkan bekerja lebih dari dua *shift* setiap harinya. Sebagai contoh, karyawan nomer satu tidak diperkenankan untuk bekerja pada *shift* pagi sekaligus *shift* siang dihari yang sama.

**Batasan 3** : Setiap karyawan harus memiliki minimal satu *shift* pagi, satu *shift* siang dan satu *shift* malam setiap bulannya.

- a. Setiap karyawan memiliki *shift* pagi setiap bulannya

$$\sum_{i=1}^n X_{(i,j)} \geq 1, i = 1 \dots n \quad (4.5)$$

untuk semua  $j = 1 \dots m$

- b. Setiap karyawan memiliki *shift* siang setiap bulannya

$$\sum_{i=1}^n Y_{(i,j)} \geq 1, i=1 \dots n \quad (4.6)$$

untuk semua  $j = 1 \dots m$

- c. Setiap karyawan memiliki *shift* malam setiap bulannya

$$\sum_{i=1}^n Z_{(i,j)} \geq 1, i=1 \dots n \quad (4.7)$$

untuk semua  $j = 1 \dots m$

Batasan ini menggambarkan setiap karyawan setidaknya harus memiliki satu *shift* pagi, satu *shift* siang dan satu *shift* malam setiap bulannya. Batasan ini digunakan untuk menghindari kasus dimana ada karyawan yang bekerja disatu atau dua macam shift saja setiap bulannya.

**Batasan 4** : Jadwal setiap karyawan menghindari pola *shift* malam yang diikuti *shift* pagi dihari berikutnya.

$$Z_{(i,j)} + X_{(i+1,j)} \leq 1, \quad (4.8)$$

untuk semua  $i = 1 \dots 30$  dan  $j = 1 \dots m$

$$Z_{(i,j)} + X_{(i-30,j)} \leq 1, \quad (4.9)$$

untuk semua  $i = 31$  dan  $j = 1 \dots m$

Batasan ini dimaksud untuk menghindari kasus dimana salah satu karyawan hari ini bekerja pada *shift* malam kemudian dilanjutkan bekerja pada *shift* pagi. Kasus ini dihindari dikarenakan tidak adanya waktu istirahat antara *shift* malam yang dilanjutkan dengan *shift* pagi. Karyawan akan bekerja dengan total waktu *shift* malam ditambah *shift* pagi yaitu 17 jam.

**Batasan 5 :** Perusahaan mengusahakan agar setiap karyawan mendapatkan minimal satu kali libur dan maksimal dua kali libur setiap minggunya.

$$L_{(i,j)} + L_{(i+1,j)} + L_{(i+2,j)} + L_{(i+3,j)} + L_{(i+4,j)} + L_{(i+5,j)} \\ + L_{(i+6,j)} \leq 2 \quad (4.10)$$

untuk semua  $i=1\dots m$  dan  $j = 1\dots n$

$$L_{(i,j)} + L_{(i+1,j)} + L_{(i+2,j)} + L_{(i+3,j)} + L_{(i+4,j)} + L_{(i+5,j)} \\ + L_{(i+6,j)} \geq 1 \quad (4.11)$$

untuk semua  $i=1\dots m$  dan  $j = 1\dots n$

Batasan ini dimaksudkan untuk mengatur pola libur karyawan. Batasan ini membuat karyawan setidaknya memiliki satu hari libur setiap tujuh hari atau bisa diterjemahkan menjadi minimal satu kali dan maksimal dua kali libur setiap minggunya. Hal ini dilakukan untuk menghindari kasus dimana libur karyawan tidak merata. Sebagai contoh kasus, libur karyawan bisa saja terkumpul pada minggu pertama sehingga karyawan tidak mendapatkan libur pada minggu berikutnya dalam satu bulan.

**Batasan 6 :** Perusahaan mengusahakan agar jadwal setiap karyawan tidak memiliki pola libur yang berderet (hari libur yang diikuti kembali oleh libur pada hari berikutnya).

$$L_{(i,j)} + L_{(i+1,j)} \leq 1, \quad (4.12)$$

untuk semua  $i = 1 \dots 30$  dan  $j = 1 \dots m$

$$L_{(i,j)} + L_{(i-30,j)} \leq 1, \quad (4.13)$$

untuk semua  $i = 31$  dan  $j = 1 \dots m$

Batasan ini dimaksudkan untuk menghindari kasus dimana karyawan memiliki hari libur yang berderet. Kasus ini dapat menyebabkan pola libur yang terpusat pada satu minggu saja. Sebagai contoh kasus, karyawan bisa saja memiliki hari libur pada tanggal 27 yang diikuti dengan libur pada tanggal 28 jika tidak terdapat batasan ini.

**Batasan 7 :** Setiap karyawan memiliki kesempatan libur minimal empat kali setiap bulannya.

$$\sum_{i=1}^n L_{(i,j)} \geq 4, i = 1 \dots n \quad (4.14)$$

untuk semua  $j = 1 \dots m$

Batasan ini dimaksudkan untuk memberikan kesempatan libur minimal empat kali kepada masing-masing karyawan setiap bulannya.

**Batasan 8 :** Perusahaan menginginkan agar setiap karyawan memiliki *shift* kerja (pagi, siang atau malam) minimal 26 kali dalam satu bulan.

$$\sum_{i=1}^n X_{(i,j)} + Y_{(i,j)} + Z_{(i,j)} \geq 26, i = 1 \dots n \quad (4.15)$$

untuk semua  $j = 1 \dots m$

Batasan dibuat karena perusahaan menginginkan setiap karyawan untuk bekerja sebanyak minimal 26 *shift* setiap bulannya.

## 4.5. Pembentukan Model *Goal Programming*

Setelah melakukan pemodelan umum dalam *linier programming*, selanjutnya dilakukan pembentukan permodelan *goal programming* dengan cara mengubah fungsi tujuan menjadi batasan sehingga didapatkan fungsi tujuan yang baru.

### 4.5.1. Perumusan Batasan *Goal Programming*

Langkah awal dalam dalam pembentukan model *goal programming* adalah mengubah batasan yang digunakan oleh fungsi tujuan *goal programming* kedalam bentuk baru dimana batasan diberikan nilai deviasi negatif dan deviasi positif.

**Batasan 1** : Batasan h pada subbab 4.2 berhubungan dengan objektif a pada subbab 4.3 Sehingga batasan h diubah kedalam bentuk batasan baru dengan menambahkan nilai deviasi. Batasan baru memastikan bahwa semua karyawan dijadwalkan untuk mempunyai 26 *shift* kerja sebisa mungkin dalam satu bulan. Notasi  $d1_{(j)}^-$  (secara berurutan  $d1_{(j)}^+$ ) adalah banyaknya deviasi negatif (positif) dari tujuan 1 untuk j karyawan.

$$\left( \sum_{i=1}^n X_{(i,j)} + Y_{(i,j)} + Z_{(i,j)} \right) + d1_{(j)}^- - d1_{(j)}^+ = 26, i = 1 \dots n \quad (4.16)$$

untuk semua  $j = 1 \dots m$

**Batasan 2** : Batasan g pada subbab 4.2 berhubungan dengan objektif b pada subbab 4.3 Sehingga batasan h diubah kedalam bentuk batasan baru dengan menambahkan nilai deviasi. Batasan baru memastikan bahwa semua karyawan sebisa mungkin memiliki 4 hari libur dalam satu bulan. Notasi  $d1_{(j)}^-$  (secara berurutan  $d1_{(j)}^+$ ) adalah banyaknya deviasi negatif (positif) dari tujuan 1 untuk j karyawan.

$$\left( \sum_{i=1}^n L_{(i,j)} \right) + d1_{(j)}^- - d1_{(j)}^+ = 4, i = 1 \dots n \quad (4.17)$$

untuk semua  $j = 1 \dots m$

#### 4.5.2. Perumusan Goal

Langkah selanjutnya adalah melakukan perumusan tujuan berdasarkan hasil formula subbab 4.5.1. Objektif atau *goal* ini nantinya akan digunakan dalam proses pembuatan fungsi tujuan akhir atau skenario.

Objektif atau *goal* ini terdiri dari variabel deviasi. Ada beberapa ketentuan dalam *Goal Programming* untuk menentukan *goal* yang baru, yaitu:

1. Jika formula awal yang ditambahkan variabel deviasi adalah  $Y \geq 0$ , maka fungsi tujuan yang baru adalah meminimalkan  $d_j^-$ .
2. Begitu pula sebaliknya, jika formula awal yang ditambahkan variabel deviasi adalah  $Y \leq 0$ , maka *goal* yang baru adalah meminimalkan  $d_j^+$ .
3. Jika formula awal yang ditambahkan variabel deviasi adalah  $Y = 0$ , maka *goal* yang baru adalah meminimalkan  $d_j^-$  dan  $d_j^+$ .

Berdasarkan ketentuan diatas, maka *goal* yang baru adalah sebagai berikut,

**Goal 1** : Berdasarkan batasan satu pada subbab 4.5.1 dan objektif a pada subbab 4.3 maka perusahaan menginginkan agar jadwal setiap karyawan mendekati jumlah *shift* minimal sebanyak 26. Pada kasus ini jumlah kelebihan *shift* seperti 27 dan 28 tidak dipermasalahakan sehingga *goal* satu menggunakan nilai  $d1_{(j)}^-$ .

$$\text{Min } Z = \sum_{j=1}^m d1_{(j)}^- \quad (4.18)$$

**Goal 2** : Berdasarkan batasan dua pada subbab 4.5.1 dan objektif b pada subbab 4.3 maka perusahaan menginginkan agar jadwal setiap karyawan mendekati jumlah hari libur minimal sebanyak 4. Pada kasus ini jumlah kelebihan *hari* libur seperti 5 dan 6 tidak dipermasalahakan sehingga *goal* dua menggunakan nilai  $d2_{(j)}^-$ .

$$\text{Min } Z = \sum_{j=1}^m d2_{(j)}^- \quad (4.19)$$

#### 4.6. Perancangan Skenario

Berdasarkan topik bahasan pada subbab 4.3, perusahaan PT ABC lebih memprioritaskan objektif a ketimbang objektif b dimana jumlah *shift* kerja masing-masing karyawan lebih penting dari pada jumlah libur masing-masing karyawan.

Pada penyelesaian model *goal programming* ini akan digunakan dua metode yaitu metode *weights* dan metode *preemptive*. Pada skenario satu digunakan metode *weights* dimana jumlah *shift* kerja karyawan memiliki bobot lebih besar ketimbang jumlah libur karyawan. Sedangkan pada skenario dua digunakan metode *preemptive* dimana jumlah *shift* kerja karyawan memiliki prioritas lebih besar ketimbang jumlah libur karyawan.

##### 4.6.1. Skenario 1

Pada skenario satu digunakan metode *preemptive* dimana jumlah *shift* kerja karyawan (*goal* satu) memiliki prioritas lebih besar ketimbang jumlah libur karyawan (*goal* dua). Persamaan (4.20) merupakan perwujudan dari fungsi tujuan untuk skenario 1.

$$\text{Min } P_1 \sum_{j=1}^m d1_{(j)}^-, P_2 \sum_{j=1}^m d2_{(j)}^- \quad (4.20)$$

Dimana  $P_1$  adalah prioritas pertama sedangkan  $P_2$  adalah prioritas kedua. Dikarenakan prioritas pertama adalah *goal* satu maka  $P_1$  dipasangkan dengan  $d1_{(j)}^-$  sedangkan  $P_2$  dipasangkan dengan  $d2_{(j)}^-$ .

##### 4.6.2. Skenario 2

Pada skenario dua digunakan metode *weights* dimana jumlah *shift* kerja karyawan (*goal* satu) memiliki bobot lebih besar ketimbang jumlah libur karyawan (*goal* dua). Persamaan (4.21) merupakan perwujudan dari fungsi tujuan untuk skenario 2.

$$\text{Min } Z = W_1 \sum_{j=1}^m d1_{(j)}^- + W_2 \sum_{j=1}^m d2_{(j)}^- \quad (4.21)$$

Dimana  $W_1$  adalah bobot untuk *goal* satu sedangkan  $W_2$  adalah bobot untuk *goal* dua. Untuk skenario 2 digunakan bobot  $W_1 = 0.6$  dan  $W_2 = 0.4$

*(Halaman ini sengaja dikosongkan)*

## BAB V IMPLEMENTASI

Pada bab ini akan dijelaskan mengenai hubungan antara baris kode program pada LINGO dengan model *Goal Programming* yang telah dirancang pada bab empat.

### 5.1. Implementasi Batasan dan Variabel Keputusan pada LINGO

Pada subbab ini dilakukan penjabaran batasan-batasan serta variabel keputusan pada *goal programming* yang telah dibahas pada bab empat yang kemudian dikaitkan dengan potongan kode program LINGO.

```
SETS:
```

```
set_i/1..31/;;  
set_j/1..31/: DN1, DP1, DN2, DP2;  
link(set_i,set_j): X, Y, Z, L;
```

```
ENDSETS
```

#### Segmen Kode 5.1 Kode Variabel Keputusan

Pada segmen kode 5.1 dapat dilihat bahwa kode program ini berhubungan dengan pembuatan variabel keputusan. Kode program menginisiasi pembuatan variabel X, Y, Z, dan L dengan indeks i dan j sedangkan variabel DN1, DP1, DN2 dan DP2 memiliki indeks j saja. Masing-masing indeks i dan j memiliki jarak nilai antara 1 sampai 31. Variabel X, Y, Z, dan L pada program lingo merepresentasikan variabel keputusan  $X_{(i,j)}$ ,  $Y_{(i,j)}$ ,  $Z_{(i,j)}$  dan  $L_{(i,j)}$  sedangkan variabel DN1, DP1, DN2 dan DP2 merepresentasikan variabel  $d1_{(j)}^-$ ,  $d1_{(j)}^+$ ,  $d2_{(j)}^-$  dan  $d2_{(j)}^+$ .

```

!Kebutuhan shift karyawan setiap harinya terpenuhi;
@FOR(set_i(i):
@SUM(set_j(j): X(i,j)) >= 10;
);
@FOR(set_i(i):
@SUM(set_j(j): Y(i,j)) >= 7;
);
@FOR(set_i(i):
@SUM(set_j(j): Z(i,j)) >= 8;
);

```

### Segmen Kode 5.2 Kode Implementasi Batasan 1

Segmen kode 5.2 merupakan perwujudan dari batasan satu pada subbab 4.4.3. Baris kode ini menggambarkan kebutuhan karyawan setiap harinya. *Shift* pagi membutuhkan karyawan 10, *shift* siang membutuhkan karyawan 7 dan *shift* malam membutuhkan karyawan 7.

```

!Setiap karyawan bekerja maksimal 1 giliran per hari;
@FOR(link(i,j):X(i,j)+Y(i,j)+Z(i,j)+L(i,j)=1
);

```

### Segmen Kode 5.3 Kode Implementasi Batasan 2

Segmen kode 5.3 merupakan perwujudan dari batasan dua pada subbab 4.4.3. Baris kode ini menggambarkan batasaan dimana setiap karyawan hanya boleh bekerja maksimal satu giliran *shift* (pagi, siang atau malam) atau libur setiap harinya.

```

!Setiap karyawan harus memiliki shift pagi, siang dan malam;
@FOR(set_j(j):
@SUM(set_i(i): X(i,j))>= 1;
);
@FOR(set_j(j):
@SUM(set_i(i): Y(i,j))>= 1;
);
@FOR(set_j(j):
@SUM(set_i(i): Z(i,j))>= 1;
);

```

### Segmen Kode 5.4 Kode Implementasi Batasan 3

Segmen kode 5.4 merupakan perwujudan dari batasan tiga pada subbab 4.4.3. Baris kode ini menggambarkan batasaan dimana

setiap karyawan harus memiliki minimal satu *shift* pagi, satu *shift* siang dan satu *shift* malam setiap bulannya.

```
!Menghindari jam malam yang diikuti jam pagi pada hari
@FOR(link(i,j)|i#LE#30: Z(i,j)+X(i+1,j)<=1);
@FOR(link(i,j)|i#EQ#31: Z(i,j)+X(i-30,j)<=1);
```

#### Segmen Kode 5.5 Kode Implementasi Batasan 4

Segmen kode 5.5 merupakan perwujudan dari batasan empat pada subbab 4.4.3. Baris kode ini menggambarkan batasan dimana jadwal setiap karyawan menghindari pola *shift* malam yang diikuti *shift* pagi dihari berikutnya.

```
!Setiap 7 hari memiliki libur minimal 1 maksimal 2;
@FOR(Link(i,j)|i#LE#24: L(i,j)+L(i+1,j)+L(i+2,j)
+L(i+3,j)+L(i+4,j)+L(i+5,j)+L(i+6,j)<=2);
@FOR(Link(i,j)|i#LE#24: L(i,j)+L(i+1,j)+L(i+2,j)
+L(i+3,j)+L(i+4,j)+L(i+5,j)+L(i+6,j)>=1);
```

#### Segmen Kode 5.6 Kode Implementasi Batasan 5

Segmen kode 5.6 merupakan perwujudan dari batasan lima pada subbab 4.4.3. Baris kode ini menggambarkan batasan dimana perusahaan mengusahakan agar setiap karyawan mendapatkan minimal satu kali libur setiap minggunya.

```
!Libur tidak boleh berderet;
@FOR(Link(i,j)|i#LE#30: L(i,j)+L(i+1,j)<=1);
@FOR(link(i,j)|i#EQ#31: L(i,j)+L(i-30,j)<=1);
```

#### Segmen Kode 5.7 Kode Implementasi Batasan 6

Segmen kode 5.7 merupakan perwujudan dari batasan enam pada subbab 4.4.3. Baris kode ini menggambarkan batasan dimana perusahaan mengusahakan agar jadwal setiap karyawan tidak memiliki pola libur yang berderet (hari libur yang diikuti kembali oleh libur pada hari berikutnya).

```
!Karyawan bekerja minimal 26 hari tiap bulannya;
@FOR(set_j(j):
@SUM(set_i(i): X(i,j)+Y(i,j)+Z(i,j)+DN1(j)-DP1(j))=26;
);
```

#### Segmen Kode 5.8 Kode Implementasi Batasan 8

Segmen kode 5.8 merupakan perwujudan dari batasan delapan pada subbab 4.4.3. Baris kode ini menggambarkan batasan dimana perusahaan menginginkan agar setiap karyawan memiliki *shift* kerja (pagi, siang atau malam) minimal 26 kali dalam satu bulan.

```
!Karyawan minimal mempunyai 4 hari libur tiap bulannya;
@FOR(set_j(j):
@SUM(set_i(i): L(i,j)+DN2(j)-DP2(j))=4; );
```

#### Segmen Kode 5.9 Kode Implementasi Batasan 7

Segmen kode 5.9 merupakan perwujudan dari batasan tujuh pada subbab 4.4.3. Baris kode ini menggambarkan batasan dimana setiap karyawan memiliki kesempatan libur minimal empat kali setiap bulannya.

```
!Nilai X, Y, Z dan L adalah Binari (1 atau 0);
@FOR(link(i,j): @BIN(X));
@FOR(link(i,j): @BIN(Y));
@FOR(link(i,j): @BIN(Z));
@FOR(link(i,j): @BIN(L));
```

#### Segmen Kode 5.10 Kode Batasan Variabel Binary

Segmen kode 5.10 merupakan perwujudan dari batasan dimana masing-masing variabel X, Y, Z, dan L memiliki nilai *binary* (satu atau nol).

## 5.2. Implementasi Skenario pada LINGO

Pada bagian ini, akan dilakukan uji coba dengan mencari solusi paling optimal menggunakan beberapa skenario.

### 5.2.1. Implementasi Skenario 1

```
!Fungsi Tujuan;
MIN = V1;
@SUM(set_j(j):DN1(j))=0;
@SUM(set_j(j):DN2(j))=V1;
```

#### Segmen Kode 5.11 Kode Fungsi Tujuan Skenario 1

Skenario 1 menggunakan metode *preemptive* dimana masing-masing goal memiliki prioritas tersendiri. Pada kasus ini, *goal* satu memiliki prioritas lebih tinggi dibandingkan *goal* dua. Dengan kata lain jumlah *shift* kerja karyawan memiliki prioritas lebih tinggi dibandingkan dengan jumlah hari libur karyawan. Segmen Kode Program 5.11 merupakan kode program yang digunakan untuk pengimplementasian fungsi tujuan skenario 1 pada subbab 4.6.1. Untuk mengimplementasikan metode *preemptive* kedalam kode program LINGO, maka goal dengan prioritas tertinggi yaitu goal satu ( $d1_{(j)}^-$ ) dijadikan batasan baru dengan nilai nol. Kemudian goal dua ( $d1_{(j)}^-/V1$ ) dengan nilai prioritas terkecil digunakan sebagai fungsi tujuan baru.

### 5.2.2. Implementasi Skenario 2

```
W1= 0.6;
W2= 0.4;
!Fungsi Tujuan;
MIN = @SUM(set_j(j):DN1(j)*W1)+@SUM(set_j(j):DN2(j)*W2);
```

#### Segmen Kode 5.12 Kode Fungsi Tujuan Skenario 2

Skenario 2 menggunakan metode *weights*. Penetapan prioritas pada goal dilakukan dengan memberikan bobot dimana bobot *goal* satu lebih besar dari pada bobot *goal* dua. Pada kasus skenario 1, bobot *goal* satu adalah 0.6 sedangkan bobot *goal* dua adalah 0.4. Segmen Kode Program 5.12 merupakan kode program yang digunakan untuk pengimplementasian fungsi tujuan skenario 2 pada subbab 4.6.2.

*(Halaman ini sengaja dikosongkan)*

## **BAB VI HASIL DAN PEMBAHASAN**

Pada bab ini akan dijelaskan verifikasi, validasi, dan analisis terhadap hasil yang diperoleh dari proses implementasi yang telah dibahas pada bab sebelumnya. Pada bagian ini juga terdapat pembahasan mengenai perbandingan kedua skenario yang telah dibuat.

### **6.1. Lingkungan Uji Coba**

Lingkungan uji coba merupakan kriteria perangkat pengujian yang digunakan dalam membangun, menguji coba, dan melakukan validasi model optimasi yang telah dibuat pada tugas akhir ini. Lingkungan uji coba terdiri dari perangkat keras dan perangkat lunak. Tabel 6.1 adalah perangkat keras yang digunakan dalam uji coba dan pembangunan model .

**Tabel 6.1 Lingkungan Uji Coba Perangkat Keras**

<b>Perangkat Keras</b>	<b>Spesifikasi</b>
Jenis	Notebook
Prosesor	Core (TM) i3
RAM	10 GB

Selain lingkungan uji coba perangkat keras, terdapat juga lingkungan uji coba perangkat lunak yang digunakan dalam membangun model optimasi. Tabel 6.2 merupakan rincian lingkungan uji coba perangkat lunak yang digunakan dalam tugas akhir ini.

**Tabel 6.2 Lingkungan Uji Coba Perangkat Lunak**

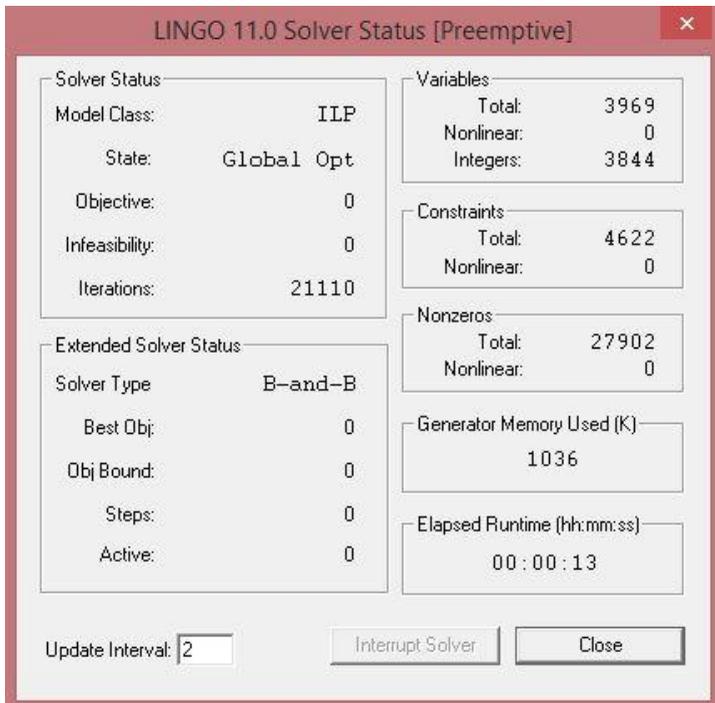
<b>Perangkat Keras</b>	<b>Fungsi</b>
Windows 8.1	Sistem Operasi
LINGO 11.0	Mengolah model dan validasi model
Microsoft Excel 2013	Mengolah data

## 6.2. Verifikasi Model

Verifikasi dilakukan untuk memastikan apakah kode program LINGO yang telah dibuat sebelumnya sudah benar dan bebas *error*. Hal ini dapat dilakukan dengan menggunakan fungsi *solver* pada program LINGO. Uji coba dapat dilakukan dengan cara menggunakan fungsi *solver* setiap selesai mengetikkan satu baris kode berupa fungsi atau ketika kode program telah lengkap. Jika setelah menggunakan fungsi *solver* tidak terdapat pesan *error* maka dapat disimpulkan bahwa kode program sudah benar dan bebas *error*. *Error* yang terjadi umumnya adalah salah ketik atau salah dalam memasukan input misalnya memasukkan desimal dengan koma dan menuliskan variabel dengan spasi. Aplikasi LINGO juga akan memberikan pesan jika model tidak memungkinkan untuk optimal yaitu dengan memberikan warning *No. Feasible Solution*. Terlihat pada gambar 6.1 dan 6.2, program LINGO dapat menemukan solusi *global optimal* dari model yang telah dibuat serta memiliki nilai *infeasibility* nol ketika dijalankan untuk menyelesaikan skenario satu yang menggunakan metode *preemptive*. Pada uji coba skenario dua digunakan menggunakan metode *weights*. Pada uji coba ini tidak terdapat pesan *error* serta ditemukannya *global optimal*. Terlihat pada gambar 6.3 dan 6.4, nilai *infeasibility* serta ditemukannya nilai *global optimal*.

```
Global optimal solution found.
Objective value:                0.000000
Objective bound:                0.000000
Infeasibilities:                0.000000
Extended solver steps:          0
Total solver iterations:        21110
```

**Gambar 6.1 Hasil Verifikasi Menggunakan Metode Preemptive**



**Gambar 6.2 Status Solver dengan Metode Preemptive**

```

Global optimal solution found.
Objective value:                0.000000
Objective bound:               -0.9769963E-13
Infeasibilities:                0.000000
Extended solver steps:         0
Total solver iterations:       24662
  
```

**Gambar 6.3 Hasil Verifikasi Menggunakan Metode Weights**



**Gambar 6.4 Status Solver dengan Metode Weights**

### 6.3. Validasi Hasil Jadwal

Validasi Jadwal dilakukan dengan cara membandingkan hasil penjadwalan yang didapatkan dari menjalankan program LINGO dengan batasan-batasan yang ada di subbab 4. Pada subbab ini akan dicek apakah setiap batasan pada *goal programming* telah terpenuhi.

**Batasan 1** : Kebutuhan karyawan pada setiap *shift* (pagi, siang dan malam) harus terpenuhi setiap harinya. Kebutuhan karyawan *shift* pagi minimal 10 pekerja, kebutuhan karyawan *shift* siang minimal 7 pekerja dan kebutuhan karyawan *shift* malam minimal 8 pekerja.



**Tabel 6.8 Jumlah Karyawan Setiap Shift Tanggal 21-31 Skenario 2**

	Tanggal/Hari										
	21	22	23	24	25	26	27	28	29	30	31
P	10	10	10	10	10	10	10	10	10	10	10
S	7	12	13	7	7	7	7	7	12	13	13
M	8	8	8	8	8	8	8	8	8	8	8

Dengan keterangan P adalah jumlah karyawan *shift* pagi, S adalah jumlah karyawan *shift* siang dan M adalah jumlah karyawan *shift* malam.

Tabel 6.3, 6.4 dan 6.5 merupakan penjabaran dari jumlah karyawan masing-masing *shift* setiap harinya yang dikerjakan menggunakan metode *preemptive*. Berdasarkan tabel tersebut, bisa dilihat jika batasan 1 terpenuhi karena jumlah minimum setiap karyawan dimana *shift* pagi, siang dan malam membutuhkan 10, 7 dan 8 karyawan terpenuhi.

Tabel 6.6, 6.7 dan 6.8 merupakan penjabaran dari jumlah karyawan masing-masing *shift* setiap harinya yang dikerjakan menggunakan metode *weights*. Berdasarkan tabel tersebut, bisa dilihat jika batasan 1 terpenuhi karena jumlah minimum karyawan setiap dimana *shift* pagi, siang dan malam membutuhkan 10, 7 dan 8 karyawan terpenuhi.

**Batasan 2** : Setiap karyawan hanya boleh mengisi satu *shift* (jam kerja) atau libur setiap harinya.

Berdasarkan hasil output dari LINGO, untuk skenario 1 dan 2 tidak ada karyawan yang memiliki *shift* atau libur lebih dari satu dalam satu harinya. Dengan demikian batasan 2 telah terpenuhi untuk skenario 1 dan 2.

**Batasan 3** : Setiap karyawan harus memiliki minimal satu *shift* pagi, satu *shift* siang dan satu *shift* malam setiap bulannya.

Dari tabel 6.15 sampai 6.20 dapat dilihat bahwa batasan 3 telah terpenuhi untuk skenario 1 dan skenario 2. Masing-masing karyawan telah bekerja minimal 1 kali *shift* pada setiap *shift* pagi, siang dan malam dalam 31 hari.



**Tabel 6.10 Jumlah Libur dan Shift Kerja Karyawan 10-20 Skenario 1**

	Nomor Karyawan									
	11	12	13	14	15	16	17	18	19	20
L	4	4	4	4	4	4	4	4	4	4
K	27	27	27	27	27	27	27	27	27	27

**Tabel 6.11 Jumlah Libur dan Shift Kerja Karyawan 21-31 Skenario 1**

	Nomor Karyawan										
	21	22	23	24	25	26	27	28	29	30	31
L	4	4	4	4	4	4	4	5	4	4	4
K	27	27	27	27	27	27	27	26	27	27	27

**Tabel 6.12 Jumlah Libur dan Shift Kerja Karyawan 1-10 Skenario 2**

	Nomor Karyawan									
	1	2	3	4	5	6	7	8	9	10
L	4	4	4	4	4	4	4	4	4	4
K	27	27	27	27	27	27	27	27	27	27

**Tabel 6.13 Jumlah Libur dan Shift Kerja Karyawan 11-20 Skenario 2**

	Nomor Karyawan									
	11	12	13	14	15	16	17	18	19	20
L	4	4	4	4	4	4	4	4	4	4
K	27	27	27	27	27	27	27	27	27	27

**Tabel 6.14 Jumlah Libur dan Shift Kerja Karyawan 21-31 Skenario 2**

	Nomor Karyawan										
	21	22	23	24	25	26	27	28	29	30	31
L	4	4	4	4	5	4	4	4	4	4	4
K	27	27	27	27	26	27	27	27	27	27	27

Dengan keterangan L adalah jumlah libur masing-masing karyawan dan K adalah jumlah *shift* kerja masing-masing karyawan.

Jika dilihat berdasarkan tabel 6.9, 6.10 dan 6.11, untuk kasus skenario 1, jumlah libur masing-masing karyawan memiliki rentang nilai antara empat sampai lima. Jumlah libur tersebut

telah melebihi batasan minimal empat kali libur dalam satu bulan. Maka dari itu, batasan 7 telah terpenuhi pada skenario 1. Jika dilihat berdasarkan tabel 6.12, 6.13 dan 6.14, untuk kasus skenario 2, jumlah libur masing-masing karyawan memiliki rentang nilai antara empat sampai lima. Jumlah libur tersebut telah melebihi batasan minimal empat kali libur dalam satu bulan. Maka dari itu, batasan 7 telah terpenuhi pada skenario 2.

**Batasan 8 :** Perusahaan menginginkan agar setiap karyawan memiliki *shift* kerja (pagi, siang atau malam) minimal 26 kali dalam satu bulan.

Jika dilihat berdasarkan tabel 6.9, 6.10 dan 6.11, untuk kasus skenario 1, jumlah *shift* kerja (pagi, siang dan malam) masing-masing karyawan memiliki rentang nilai antara 26 sampai 27. Jumlah *shift* kerja tersebut telah melebihi batasan minimal 26 *shift* kerja dalam satu bulan. Maka dari itu, batasan 8 telah terpenuhi pada skenario 1.

Jika dilihat berdasarkan tabel 6.12, 6.13 dan 6.14, untuk kasus skenario 2 jumlah *shift* kerja (pagi, siang dan malam) masing-masing karyawan memiliki rentang nilai antara 26 sampai 27. Jumlah *shift* kerja tersebut telah melebihi batasan minimal 26 *shift* kerja dalam satu bulan. Maka dari itu, batasan 8 telah terpenuhi pada skenario 2.

#### **6.4. Hasil Optimasi Goal Programming**

Model matematis untuk penjadwalan karyawan PT ABC yaitu batasan (4.1) - (4.17) dan fungsi tujuan (4.20) - (4.21) yang telah diimplementasikan kedalam kode LINGO selanjutnya akan dicari solusinya menggunakan perangkat lunak LINGO.

*Output* LINGO untuk penjadwalan perawat menghasilkan bilangan biner seperti pada gambar 6.5 dan 6.6 yang kemudian diproses menggunakan variabel keputusan sehingga membentuk sebuah jadwal jam kerja. Berikut ini adalah cara membaca hasil *output* LINGO.

Preemptive	
X( 6, 19)	0.000000
X( 6, 20)	0.000000
X( 6, 21)	1.000000
X( 6, 22)	0.000000
X( 6, 23)	1.000000
X( 6, 24)	0.000000
X( 6, 25)	1.000000
X( 6, 26)	0.000000
X( 6, 27)	1.000000
X( 6, 28)	1.000000
X( 6, 29)	0.000000
X( 6, 30)	0.000000
X( 6, 31)	1.000000
X( 7, 1)	0.000000
X( 7, 2)	0.000000
X( 7, 3)	0.000000
X( 7, 4)	0.000000
X( 7, 5)	1.000000

**Gambar 6.5 Potongan Hasil Output LINGO Skenario 1**

Weights	
DP2( 24)	0.000000
DP2( 25)	0.3225806E-01
DP2( 26)	0.000000
DP2( 27)	0.000000
DP2( 28)	0.000000
DP2( 29)	0.000000
DP2( 30)	0.000000
DP2( 31)	0.000000
X( 1, 1)	0.000000
X( 1, 2)	0.000000
X( 1, 3)	0.000000
X( 1, 4)	0.000000
X( 1, 5)	1.000000
X( 1, 6)	0.000000
X( 1, 7)	1.000000
X( 1, 8)	1.000000
X( 1, 9)	0.000000
X( 1, 10)	0.000000
X( 1, 11)	0.000000
X( 1, 12)	1.000000
X( 1, 13)	0.000000

**Gambar 6.6 Potongan Hasil Output LINGO Skenario 2**

Pada gambar 6.6 variabel X (1, 7) memiliki nilai satu sedangkan variabel X (1, 6) memiliki nilai nol. X merupakan lambang *shift* pagi, indeks 1 merupakan lambang dari tanggal/hari, indeks 6 dan 7 merupakan lambang dari nomor karyawan 6 dan 7 sedangkan nilai satu dan nol melambangkan kehadiran dari karyawan. Variabel X (1, 7) yang memiliki nilai satu dapat

dibaca sebagai “karyawan nomer 7 pada tanggal satu bekerja pada *shift* pagi”. Variabel X (1, 6) yang memiliki nilai nol dapat dibaca sebagai “karyawan nomer 6 pada tanggal satu tidak bekerja pada *shift* pagi”.

#### 6.4.1. Hasil Optimasi Skenario 1

Model matematika untuk penjadwalan karyawan PT ABC menggunakan metode *preemptive* dengan batasan yaitu persamaan (4.1) - (4.17) dan fungsi tujuan (4.20) selanjutnya akan dicari solusinya menggunakan perangkat lunak LINGO. Kode lengkap pemrograman LINGO untuk penjadwalan karyawan dapat ditemukan dilampiran A. *Output* LINGO akan diproses menggunakan variabel keputusan sehingga membentuk sebuah jadwal *shift* kerja masing-masing karyawan. Berikut ini adalah jadwal jam kerja yang dibentuk berdasarkan variabel keputusan yang bersumber pada hasil output LINGO untuk penjadwalan karyawan PT ABC menggunakan metode *preemptive* yang dikemas dalam tabel 6.15 sampai 6.17.

Tabel 6.15 Jadwal Karyawan Skenario 1 Tanggal 1-10

	Tanggal/Hari									
K	1	2	3	4	5	6	7	8	9	10
1	S	S	L	M	M	M	S	S	S	L
2	M	S	P	M	L	P	S	S	P	M
3	S	P	P	P	S	S	L	P	P	P
4	S	M	S	P	P	M	L	M	M	S
5	P	P	M	S	L	S	P	P	S	S
6	S	P	P	L	M	S	S	P	S	P
7	M	M	M	M	L	M	S	P	P	P
8	S	S	L	M	S	P	M	S	M	L
9	P	P	P	P	P	L	P	S	P	S
10	S	M	S	L	P	S	P	M	S	M
11	P	P	P	P	L	P	P	M	M	S
12	S	P	S	M	S	L	S	P	M	M
13	M	S	P	L	M	S	M	S	S	P
14	M	S	L	P	M	S	S	P	M	L

	Tanggal/Hari									
K	1	2	3	4	5	6	7	8	9	10
15	P	S	P	S	P	P	L	M	S	P
16	P	P	L	P	M	M	M	S	P	L
17	P	S	L	P	M	M	M	M	M	L
18	M	M	M	S	P	L	P	S	P	P
19	P	S	L	S	S	M	S	P	S	L
20	S	P	P	P	P	S	L	P	P	S
21	S	M	S	L	P	P	P	M	M	M
22	P	P	P	P	S	L	P	P	S	S
23	P	M	M	M	S	P	L	S	P	M
24	S	P	S	M	M	L	P	M	S	P
25	M	S	M	L	P	P	P	P	S	M
26	S	M	M	L	P	M	M	M	S	P
27	P	S	S	S	P	P	L	S	P	M
28	S	L	S	P	S	P	M	S	L	P
29	M	S	P	S	L	M	M	S	S	M
30	M	S	M	S	M	L	M	S	P	P
31	S	M	M	M	L	P	P	S	M	S

Tabel 6.16 Jadwal Karyawan Skenario 1 Tanggal 11-20

	Tanggal/Hari									
K	11	12	13	14	15	16	17	18	19	20
1	P	P	P	P	M	M	L	P	P	P
2	S	L	S	S	S	P	P	P	L	S
3	M	M	M	L	M	M	M	M	M	M
4	P	M	S	L	P	S	P	S	M	M
5	M	L	M	S	S	S	P	M	L	P
6	L	M	M	M	M	S	M	L	S	M
7	P	L	P	M	S	P	S	M	L	P
8	S	P	S	P	P	M	L	P	P	S
9	M	M	L	P	P	S	P	P	M	L
10	L	S	S	P	S	P	P	L	S	P
11	M	L	P	P	M	S	P	M	L	M
12	S	M	L	S	P	S	S	P	P	L
13	L	S	P	P	P	M	M	L	M	M

	Tanggal/Hari									
K	11	12	13	14	15	16	17	18	19	20
14	P	M	S	M	S	P	L	P	P	M
15	S	P	P	L	M	M	S	S	S	P
16	P	S	M	M	S	P	L	P	M	S
17	S	P	P	P	P	P	L	P	S	S
18	M	S	L	M	S	S	P	S	P	L
19	P	P	M	S	M	S	L	S	S	P
20	P	S	S	L	P	S	M	S	P	S
21	L	P	P	M	M	M	S	L	P	P
22	P	S	L	P	S	P	P	M	S	L
23	M	M	M	L	P	S	M	S	P	P
24	S	P	L	S	P	P	P	P	M	L
25	L	P	P	P	M	S	M	L	M	M
26	L	S	P	P	P	M	S	L	P	M
27	M	M	M	L	S	P	P	M	M	S
28	P	P	P	M	S	L	S	P	P	P
29	M	L	M	S	S	M	M	S	L	P
30	S	P	L	M	S	S	S	M	S	L
31	P	L	S	S	S	P	M	M	L	S

Tabel 6.17 Jadwal Karyawan Skenario 1 Tanggal 21-31

	Tanggal/Hari										
K	21	22	23	24	25	26	27	28	29	30	31
1	P	P	M	L	P	P	M	M	M	S	S
2	S	S	P	P	P	L	S	P	S	S	S
3	L	P	P	S	M	M	M	L	P	S	P
4	L	P	M	M	S	P	P	L	P	P	M
5	S	S	P	P	M	L	P	S	P	P	P
6	M	S	P	M	L	S	P	P	M	S	M
7	S	P	S	P	P	L	P	M	S	M	S
8	P	S	S	L	S	P	S	P	P	S	M
9	S	M	S	P	S	P	L	P	M	S	P
10	P	P	P	S	L	M	S	M	S	P	P
11	S	M	M	M	M	L	P	S	S	P	S
12	M	S	S	P	P	M	L	S	P	M	S

	Tanggal/Hari										
K	21	22	23	24	25	26	27	28	29	30	31
13	S	P	P	M	L	P	M	M	S	S	M
14	M	S	M	L	P	P	P	P	M	M	S
15	L	M	S	P	S	M	M	L	P	P	P
16	M	M	S	L	P	P	S	S	M	S	S
17	P	P	S	L	S	S	S	P	S	M	S
18	M	M	M	M	M	M	L	M	S	P	P
19	P	S	P	L	P	P	P	P	M	M	S
20	L	S	S	M	M	M	M	L	P	P	M
21	P	M	M	S	L	S	M	S	P	M	S
22	P	S	P	P	M	M	L	P	S	S	P
23	L	P	P	P	P	M	S	L	M	S	P
24	P	P	S	S	P	P	L	M	S	P	M
25	M	S	M	M	L	P	P	M	M	S	P
26	S	M	S	S	L	S	S	P	P	M	M
27	L	P	M	M	M	S	P	L	S	P	P
28	M	S	L	P	S	S	M	M	L	M	S
29	P	M	S	S	P	L	M	S	P	P	M
30	P	S	S	P	M	S	L	S	S	S	S
31	M	S	P	S	S	L	P	P	S	S	S

Dengan keterangan K adalah karyawan, P adalah *shift* pagi, S adalah *shift* siang, M adalah *shift* malam dan L adalah hari libur. Tabel dapat dibaca dari kiri ke kanan, yaitu karyawan satu pada tanggal satu, kemudian karyawan satu tanggal hari dua, dan seterusnya. Berdasarkan tabel 6.15, karyawan satu pada tanggal satu dan dua bekerja pada *shift* siang, lalu libur pada tanggal 3 dan bekerja lagi pada *shift* malam tanggal 4, 5 dan 6. Cara membaca ini juga berlaku untuk karyawan selanjutnya secara berurutan, dan pada tabel 6.15 sampai 6.20 selanjutnya dapat digunakan cara membaca yang sama.

#### 6.4.2. Hasil Optimasi Skenario 2

Model matematika untuk penjadwalan karyawan PT ABC menggunakan metode *weights* dengan batasan yaitu persamaan

(4.1) - (4.17) dan fungsi tujuan (4.21) selanjutnya akan dicari solusinya menggunakan perangkat lunak LINGO. Kode lengkap pemrograman LINGO untuk penjadwalan karyawan dapat ditemukan dilampiran A. *Output* LINGO akan diproses menggunakan variabel keputusan sehingga membentuk sebuah jadwal *shift* kerja masing-masing karyawan. Berikut ini adalah jadwal jam kerja yang dibentuk berdasarkan variabel keputusan yang bersumber pada hasil output LINGO untuk penjadwalan karyawan PT ABC menggunakan metode *weights* yang dikemas dalam tabel 6.18 sampai 6.20.

**Tabel 6.18 Jadwal Karyawan Skenario 2 Tanggal 1-10**

	Tanggal/Hari									
K	1	2	3	4	5	6	7	8	9	10
1	S	S	S	L	S	M	S	S	S	M
2	S	S	M	S	L	S	M	S	P	P
3	S	M	L	P	P	S	P	S	S	L
4	S	S	P	S	M	M	L	P	S	S
5	P	P	P	P	L	P	S	S	P	P
6	M	S	M	S	L	S	P	P	P	M
7	P	P	P	P	L	P	M	M	S	M
8	P	M	S	M	S	L	S	P	P	M
9	M	S	L	P	P	P	P	S	M	L
10	M	M	M	L	P	M	S	P	S	P
11	S	P	P	P	M	L	P	M	S	P
12	P	M	S	L	M	S	M	S	P	S
13	S	S	P	P	S	P	L	P	S	P
14	M	S	M	M	M	M	L	S	P	P
15	P	S	P	S	L	S	S	P	S	P
16	S	P	M	M	M	M	L	S	P	S
17	S	P	S	L	P	M	M	M	S	M
18	P	M	M	M	L	S	S	S	M	S
19	M	S	P	L	P	S	P	P	S	M
20	M	M	L	P	P	P	P	M	M	L
21	P	S	S	M	M	L	M	M	M	S
22	S	P	L	M	S	P	P	S	P	L

	Tanggal/Hari									
K	1	2	3	4	5	6	7	8	9	10
23	M	S	L	M	S	M	S	P	P	L
24	S	P	P	L	P	P	M	S	M	M
25	P	L	M	S	S	P	P	P	L	P
26	S	P	P	M	S	L	M	M	M	M
27	S	P	L	P	P	P	P	P	M	L
28	P	P	P	P	M	L	M	M	M	S
29	P	M	S	P	M	M	L	M	S	P
30	M	M	S	S	P	L	P	S	S	S
31	S	S	M	S	P	P	L	S	P	P

Tabel 6.19 Jadwal Karyawan Skenario 2 Tanggal 11-20

	Tanggal/Hari									
K	11	12	13	14	15	16	17	18	19	20
1	L	M	M	M	M	M	M	L	P	M
2	S	L	P	M	S	P	P	P	L	P
3	M	S	S	M	S	M	L	S	S	M
4	P	P	P	L	M	S	S	M	M	S
5	M	L	S	P	P	S	S	S	L	P
6	S	L	P	P	M	S	S	M	L	P
7	M	L	P	S	P	S	P	P	L	S
8	S	M	L	M	S	P	P	P	M	L
9	P	P	S	P	S	P	L	P	S	M
10	L	S	S	S	P	S	P	L	P	P
11	P	P	L	P	M	M	M	M	S	L
12	L	S	P	M	S	P	S	L	M	M
13	P	P	S	L	P	S	P	P	P	P
14	S	M	M	L	S	P	M	S	P	S
15	M	L	S	P	P	P	P	M	L	P
16	S	P	M	L	M	M	M	S	M	S
17	L	S	M	M	S	S	S	L	P	M
18	P	L	P	P	P	S	P	P	L	P
19	L	P	P	S	M	S	P	L	P	S
20	P	S	S	P	P	P	L	M	S	P
21	P	M	L	P	S	M	M	M	M	L

	Tanggal/Hari									
K	11	12	13	14	15	16	17	18	19	20
22	M	M	M	S	S	P	L	P	P	P
23	P	P	P	S	M	S	L	P	M	M
24	L	M	M	S	S	S	P	L	P	S
25	M	S	P	M	L	P	P	P	S	P
26	S	P	L	P	P	P	M	M	S	L
27	M	S	P	M	M	M	L	S	M	S
28	P	P	L	S	S	M	S	P	P	L
29	P	M	M	L	P	M	S	M	M	M
30	S	P	L	P	P	S	M	S	S	L
31	M	M	M	L	S	S	M	S	P	M

Tabel 6.20 Jadwal Karyawan Skenario 2 Tanggal 21-31

	Tanggal/Hari										
K	21	22	23	24	25	26	27	28	29	30	31
1	M	M	M	M	L	M	M	M	S	S	M
2	P	S	P	P	P	L	M	S	M	S	S
3	S	M	S	L	P	S	S	P	P	S	M
4	L	S	P	P	S	P	P	L	P	P	P
5	P	S	P	P	S	L	P	P	S	P	S
6	M	S	M	M	S	L	M	M	S	M	M
7	M	S	S	S	P	L	P	S	M	S	P
8	P	S	P	P	M	M	L	P	P	M	S
9	S	P	M	L	P	M	S	M	M	S	S
10	M	M	S	S	L	P	M	S	S	P	P
11	P	P	M	M	M	M	L	M	S	S	M
12	S	P	P	P	L	P	P	P	M	M	S
13	L	S	P	M	M	S	M	L	P	P	M
14	L	S	M	S	P	M	S	L	P	P	P
15	S	M	S	M	S	L	P	P	S	S	P
16	L	S	S	P	M	M	S	L	P	P	M
17	M	M	S	P	L	S	M	S	S	M	S
18	P	P	P	S	M	L	P	P	P	P	P
19	M	M	S	P	L	S	P	S	M	M	S
20	P	P	P	L	P	P	M	M	S	S	P

	Tanggal/Hari										
K	21	22	23	24	25	26	27	28	29	30	31
21	S	P	S	M	M	M	L	S	P	P	P
22	P	P	M	L	S	P	P	M	S	S	S
23	S	M	S	L	P	P	S	P	M	M	M
24	M	S	M	M	L	M	S	M	S	P	P
25	P	L	S	M	M	S	P	P	L	M	S
26	P	S	P	P	P	S	L	M	M	S	M
27	M	M	M	L	S	P	M	S	S	M	S
28	P	P	S	S	P	P	L	P	M	S	S
29	L	S	S	P	P	P	P	L	P	S	P
30	S	P	P	S	M	S	L	P	P	P	S
31	L	P	S	S	S	P	S	L	S	S	S

Dengan keterangan K adalah karyawan, P adalah *shift* pagi, S adalah *shift* siang, M adalah *shift* malam dan L adalah hari libur.

### 6.5. Analisa Hasil Uji Coba Skenario

Berdasarkan hasil validasi jadwal karyawan, jadwal karyawan skenario 1 dan 2 telah memenuhi semua batasan yang telah ditentukan pada subbab 4.4.3. Fungsi tujuan pada subbab 4.6.1 dan 4.6.2 dari masing-masing skenario dapat dikatakan terpenuhi dikarenakan nilai seluruh deviasi  $d1_{(j)}^-$  dan  $d2_{(j)}^-$  sama dengan nol. Nilai deviasi ini menggambarkan kedua jadwal yang dihasilkan telah meminimalkan kurangnya jumlah libur serta *shift* kerja karyawan. Dikarenakan nilai deviasi sama dengan nol maka kedua skenario telah memenuhi syarat *goal* minimal yaitu jumlah libur setiap karyawan lebih dari 4 dan jumlah *shift* kerja setiap karyawan lebih dari 26.

Ketika hasil jadwal skenario 1 dibandingkan dengan skenario 2, tidak ditemukan perbedaan yang signifikan. Masing-masing jadwal memiliki total jumlah libur sebanyak 125 hari dan total jumlah *shift* kerja sebanyak 836. Kedua jadwal juga telah memenuhi batasan-batasan yang ditetapkan oleh perusahaan. Perbedaan yang ditemukan pada hasil jadwal skenario 1 dan 2

adalah perbedaan letak *shift* kerja dan hari libur pada beberapa karyawan.

Dapat disimpulkan, bahwa perusahaan PT ABC dapat memilih salah satu jadwal yang dihasilkan oleh skenario 1 ataupun skenario 2. Kedua jadwal telah memenuhi batasan-batasan serta fungsi tujuan yang ada.

*(Halaman ini sengaja dikosongkan)*

## **BAB VII**

### **KESIMPULAN DAN SARAN**

Pada bab ini, dijelaskan mengenai kesimpulan dari semua proses yang telah dilakukan dan saran perbaikan yang dapat diberikan untuk penelitian selanjutnya.

#### **7.1. Kesimpulan**

Berdasarkan hasil penelitian pada tugas akhir ini, maka didapatkan kesimpulan sebagai berikut:

1. Metode *Goal Programming* mampu menjadi metode penyelesaian untuk kasus multi objektif, yang mana dalam tugas akhir ini terkait dengan optimasi penjadwalan karyawan PT ABC disubdivisi x.
2. Implementasi model skenario 1 dan 2 pada program LINGO yang menghasilkan jadwal karyawan subdivisi x, dapat memenuhi keseluruhan tujuan.
3. Hasil jadwal skenario 1 dan 2 tidak memiliki perbedaan yang signifikan sehingga PT ABC dapat memilih salah satu hasil jadwal untuk diterapkan pada subdivisi x.
4. Nilai deviasi  $d1_{(j)}$  dan  $d2_{(j)}$  pada skenario 1 dan 2 adalah nol sehingga *goal* 1 dan 2 yang telah ditentukan pada subbab 4.5.2 telah terpenuhi.
5. Hasil jadwal skenario 1 dan 2 dapat dikatakan optimal karena *report* pada program LINGO menunjukkan hasil jadwal yang didapat adalah solusi *global optimal*.

#### **7.2. Saran**

Dari pengerjaan tugas akhir ini, terdapat hal-hal yang dapat diperbaiki lagi. Untuk pengembangan yang lebih baik pada penelitian optimasi penjadwalan karyawan PT ABC maka ada beberapa saran yang diberikan untuk penelitian selanjutnya, diantaranya adalah:

1. Melakukan uji coba skenario lainya. Skenario bisa berupa kebalikan dari skenario 1 dan 2 dimana perusahaan lebih mementingkan jumlah libur karyawan ketimbang jumlah *shift* kerja.
2. Menyempurnakan model sehingga bisa mengalokasikan *shift* kerja atau hari libur berlebih untuk karyawan tertentu.

## DAFTAR PUSTAKA

- [1] Shahrezaei, P.S., Moghaddam, R.T., Kazemipoor, H., *Solving a multi-objective multi-skilled manpower scheduling model by a fuzzy goal programming approach*, **Applied Mathematical Modelling**, Vol. 37, pp. 5424–5443, 2013.
- [2] Güler, M. G., Idin, K., & Güler, E. Y., *A goal programming model for scheduling residents in ananesthesia and reanimation department*, **Expert Systems with Applications**, Vol. 40, pp. 2117–2126, 2013.
- [3] Güler, M. G., *A hierarchical goal programming model for scheduling the outpatient clinics*, **Expert Systems with Applications**, Vol. 40, pp. 4906-4914, 2013.
- [4] Taylor, Bernard W., III, **Introduction to Management Science**, 12th Ed, Pearson, 2012.
- [5] Hillier, F. S., & Lieberman, G. J., **Introduction to Operations Research**, 9th Ed, McGraw-Hill Companies, 2010.
- [6] Chen, J. G. and T. Yeung, *Hybrid expert system approach to Nurse scheduling*, **Computers in Nursing**, pp. 183-192, 1993.
- [7] Oldenkamp, J. H., *Quality in fives: On the analysis, operationalization and application of nursing schedule quality*, 1996.
- [8] Ignizio, J. P., **Linear Programming in Single and Multiple Objective System**, Pennsylvania State: Prentice Hall, Inc, 1982.
- [9] Jabidi, *Optimasi Distribusi LPG di Jakarta*, Tesis Fakultas Teknik Universitas Indonesia, 2012.
- [10] *Pengantar Konsep Optimasi*, Share ITS, [Online]. Available: [share.its.ac.id/mod/resource/view.php?id=1907](http://share.its.ac.id/mod/resource/view.php?id=1907). [Accessed 5 Juni 2018].

- [11] F. Hillier and G. Lieberman, **Introduction to stochastic models in operations research**, McGraw-Hill Companies, 1990.
- [12] A. Rio, *Penentuan Kapasitas Optimal Produksi CPO (Crude Palm Oil) Dipabrik Kelapa Sawit PT. Andira Argo Dengan Menggunakan Goal Programming*, 2006.
- [13] Siregar, Chairun Nissa;, *Pendekatan Fuzzy Goal Programming Untuk Optimasi Pola Distribusi*, 2012.
- [14] Siswanto, *Operations Research*, Bogor: Erlangga, 2007.
- [15] T. Harjiyanto, "*Aplikasi Model Goal Programming Untuk Optimasi Produksi Aksesoris (Studi Kasus : PT. Kosama Jaya Bangun Tapan Bantul)*," 2014. [Online]. Available: [eprints.uny.ac.id/12570/1/SKRIPSI.pdf](http://eprints.uny.ac.id/12570/1/SKRIPSI.pdf). [Accessed 27 06 2018].
- [16] A. Bhargava, S. Singh and D. Bansal, *A Fuzzy Goal Programming Approach for Food Product Distribution of Small and Medium Enterprises*, **Annals of Pure and Applied Mathematics**, vol. 9, pp. 157-166, 2015.
- [17] Pamungkas, Dimas, *Model Goal Programing untuk Optimasi Penjadwalan Perawat (Studi Kasus di Rumah Sakit Grhasia Yogyakarta)*, 2016. [Online]. <http://eprints.uny.ac.id/44512/>. [Accessed 27 06 2018].

## BIODATA PENULIS



Penulis lahir di Kudus, 11 Juli 1994, dengan nama lengkap Raka Ananta Adhitya Warman. Penulis merupakan anak kedua dari tiga bersaudara.

Riwayat pendidikan penulis yaitu TK Sleman Yogyakarta, SD Raj Yamuna Denpasar, SMP Negeri 1 Denpasar, SMA Negeri 1 Denpasar, dan akhirnya menjadi salah satu mahasiswa Departemen Sistem Informasi, Institut

Teknologi Sepuluh Nopember angkatan 2011 melalui jalur SNMPTN Tertulis dengan NRP lama 5211100110.

Ketertarikan penulis dalam bidang Peramalan, Optimasi, dan *Business Intelligent* menjadikan penulis untuk memilih laboratorium minat bakat Rekayasa Data dan Intelegensi Bisnis (RDIB) dalam mengambil topik dan tempat menyelesaikan Tugas Akhir dengan topik Optimasi. Penulis dapat dihubungi melalui alamat e-mail [raka.ananta.w@gmail.com](mailto:raka.ananta.w@gmail.com).

*(Halaman ini sengaja dikosongkan)*

## LAMPIRAN A

### Kode Program LINGO

```
MODEL:

SETS:

    set_i/1..31/;;
    set_j/1..31/: DN1, DP1, DN2, DP2;
    link(set_i,set_j): X, Y, Z, L;

ENDSETS

DATA:

    W1= 0.6;
    W2= 0.4;

ENDDATA

!Kebutuhan shift karyawan setiap harinya
terpenuhi;
@FOR(set_i(i):
@SUM(set_j(j): X(i,j)) >= 10;
);
@FOR(set_i(i):
@SUM(set_j(j): Y(i,j)) >= 7;
);
@FOR(set_i(i):
@SUM(set_j(j): Z(i,j)) >= 8;
);

!Setiap karyawan bekerja maksimal 1 giliran
per hari;
@FOR(link(i,j):X(i,j)+Y(i,j)+Z(i,j)+L(i,j)=1
);

!Setiap karyawan harus memiliki shift pagi,
siang dan malam;
@FOR(set_j(j):
@SUM(set_i(i): X(i,j))>= 1;
```

```
);
@FOR (set_j (j) :
@SUM (set_i (i) : Y (i, j)) >= 1;
);
@FOR (set_j (j) :
@SUM (set_i (i) : Z (i, j)) >= 1;
);

!Menghindari jam malam yang diikuti jam pagi
pada hari berikutnya;
@FOR (link (i, j) | i#LE#30: Z (i, j) + X (i+1, j) <= 1);
@FOR (link (i, j) | i#EQ#31: Z (i, j) + X (i-
30, j) <= 1);

!Setiap 7 hari memiliki libur minimal 1
maksimal 2;
@FOR (Link (i, j) | i#LE#24:
L (i, j) + L (i+1, j) + L (i+2, j)
+ L (i+3, j) + L (i+4, j) + L (i+5, j) + L (i+6, j) <= 2);
@FOR (Link (i, j) | i#LE#24:
L (i, j) + L (i+1, j) + L (i+2, j)
+ L (i+3, j) + L (i+4, j) + L (i+5, j) + L (i+6, j) >= 1);

!Libur tidak boleh berderet;
@FOR (Link (i, j) | i#LE#30: L (i, j) + L (i+1, j) <= 1);
@FOR (link (i, j) | i#EQ#31: L (i, j) + L (i-
30, j) <= 1);

!Karyawan bekerja minimal 26 hari tiap
bulannya;
@FOR (set_j (j) :
@SUM (set_i (i) : X (i, j) + Y (i, j) + Z (i, j) + DN1 (j) -
DP1 (j)) = 26;
);

!Karyawan minimal mempunyai 4 hari libur
tiap bulannya;
@FOR (set_j (j) :
@SUM (set_i (i) : L (i, j) + DN2 (j) - DP2 (j)) = 4; );

!Fungsi Tujuan;
```

```

MIN = V1;
@SUM(set_j(j):DN1(j))=0;
@SUM(set_j(j):DN2(j))=V1;

!Nilai X, Y, Z dan L adalah Binari (1 atau
0);
@FOR(link(i,j): @BIN(X));
@FOR(link(i,j): @BIN(Y));
@FOR(link(i,j): @BIN(Z));
@FOR(link(i,j): @BIN(L));

END

```

### Segmen Kode A.1 Kode Program Skenario 1

```

MODEL:

SETS:

    set_i/1..31/;;
    set_j/1..31/: DN1, DP1, DN2, DP2;
    link(set_i,set_j): X, Y, Z, L;

ENDSETS

DATA:

    W1= 0.6;
    W2= 0.4;

ENDDATA

!Kebutuhan shift karyawan setiap harinya
terpenuhi;
@FOR(set_i(i):
@SUM(set_j(j): X(i,j)) >= 10;
);
@FOR(set_i(i):
@SUM(set_j(j): Y(i,j)) >= 7;
);
@FOR(set_i(i):
@SUM(set_j(j): Z(i,j)) >= 8;
);

```

```
!Setiap karyawan bekerja maksimal 1 giliran
per hari;
@FOR(link(i,j):X(i,j)+Y(i,j)+Z(i,j)+L(i,j)=1
);

!Setiap karyawan harus memiliki shift pagi,
siang dan malam;
@FOR(set_j(j):
@SUM(set_i(i): X(i,j))>= 1;
);
@FOR(set_j(j):
@SUM(set_i(i): Y(i,j))>= 1;
);
@FOR(set_j(j):
@SUM(set_i(i): Z(i,j))>= 1;
);

!Menghindari jam malam yang diikuti jam pagi
pada hari berikutnya;
@FOR(link(i,j)|i#LE#30: Z(i,j)+X(i+1,j)<=1);
@FOR(link(i,j)|i#EQ#31: Z(i,j)+X(i-
30,j)<=1);

!Setiap 7 hari memiliki libur minimal 1
maksimal 2;
@FOR(Link(i,j)|i#LE#24:
L(i,j)+L(i+1,j)+L(i+2,j)+L(i+3,j)+L(i+4,j)+L
(i+5,j)+L(i+6,j)<=2);
@FOR(Link(i,j)|i#LE#24:
L(i,j)+L(i+1,j)+L(i+2,j)+L(i+3,j)+L(i+4,j)+L
(i+5,j)+L(i+6,j)>=1);

!Libur tidak boleh berderet;
@FOR(Link(i,j)|i#LE#30: L(i,j)+L(i+1,j)<=1);
@FOR(link(i,j)|i#EQ#31: L(i,j)+L(i-
30,j)<=1);

!Karyawan bekerja minimal 26 hari tiap
bulannya;
@FOR(set_j(j):
```

```

@SUM(set_i(i): X(i,j)+Y(i,j)+Z(i,j)+DN1(j)-
DP1(j))=26;
);

!Karyawan minimal mempunyai 4 hari libur
tiap bulannya;
@FOR(set_j(j):
@SUM(set_i(i): L(i,j)+DN2(j)-DP2(j))=4; );

!Fungsi Tujuan;
MIN =
@SUM(set_j(j):DN1(j)*W1)+@SUM(set_j(j):DN2(j)
)*W2);

!Nilai X, Y, Z dan L adalah binari (1 atau
0);
@FOR(link(i,j): @BIN(X));
@FOR(link(i,j): @BIN(Y));
@FOR(link(i,j): @BIN(Z));
@FOR(link(i,j): @BIN(L));

END

```

**Segmen Kode A.2 Kode Program Skenario 2**