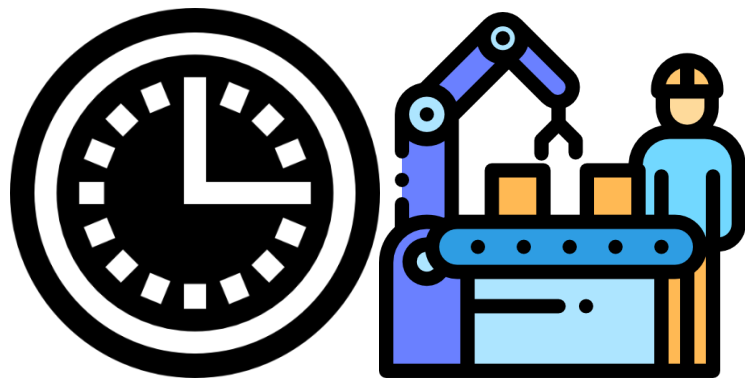




MODUL PRAKTIKUM PENELITIAN OPERASIONAL II



Disusun oleh:
**Tim Dosen dan Asisten Praktikum Laboratorium
Laboratorium Teknologi Industri**
Jurusan Teknik Industri - Universitas
Mulawarman
2022

**MODUL PRAKTIKUM
PENELITIAN OPERASIONAL II
PROGRAM STUDI TEKNIK INDUSTRI
FAKULTAS TEKNIK
UNIVERSITAS MULWARMAN**

Telah dibahas dan dinyatakan telah memenuhi syarat untuk menjadi Modul Praktikum
Penelitian Operasional II di lingkungan Teknik Industri Fakultas Teknik Universitas
Mulawarman

Samarinda, 16 Oktober 2022

Dosen Penanggung Jawab,

Ir. Lina Dianati Fathimahhayati, S.T., M.Sc., IPM. Asean Eng.

Dr. Ir. Wahyuda, S.T., M.T

Asistensi Praktikum,

- | | |
|----------------------------|-----------------|
| 1. Jenny Noviani | NIM. 1909036004 |
| 2. Karine Nathania Lumenta | NIM. 1909036030 |
| 3. M. Harsa Wardana | NIM. 1909036031 |
| 4. Azura Dahlia | NIM. 1909036041 |

Mengetahui dan Menyetujui,

Dosen Penanggung Jawab I Praktikum Penelitian Operasional II <u>Dr. Ir. Wahyuda, S.T., M.T</u> NIP. 19750424 200312 1 001	Dosen Penanggung Jawab II Praktikum Penelitian Operasional II <u>Ir. Lina Dianati Fathimahhayati, S.T., M.Sc., IPM. Asean Eng.</u> NIP. 19870306 201404 2 001
--	--

KATA PENGANTAR

Puji Syukur kami panjatkan kepada Tuhan Yang Maha Esa atas terselesaikannya modul Praktikum Penelitian Operasional II ini. Modul ini merupakan panduan dalam pelaksanaan praktikum terintegrasi dari Penelitian Operasional. Modul ini berisi 5 modul, yaitu: *Linear Goal Programming*, Sistem Antrian, *Markov Chain*, *Game Theory*, dan *Dynamic Programming*.

Tujuan Pembelajaran dari Praktikum Penelitian Operasional II ini adalah mahasiswa diharapkan: mampu memahami integrasi mata kuliah Penelitian Operasional II, mampu menerapkan prinsip-prinsip agar mendapatkan aktivitas produksi yang optimal, mampu mencari nilai maksimal dari profit, hasil produksi, lini perakitan, dan lain sebagainya), dan meminimumkan ketidaktercapaian suatu produksi.

Akhir kata, semoga modul ini dapat berguna bagi kegiatan pembelajaran di Jurusan Teknik Industri, Universitas Mulawarman dan khususnya di Laboratorium Teknologi Industri.

Samarinda, 16 Oktober 2022

Tim Penyusun

DAFTAR ISI

Kata Pengantar.....	ii
Daftar Isi	iii
Pendahuluan.....	1
Petunjuk Umum	3
Modul 1. <i>Linear Goal Programming</i>	6
Modul 2. Teori Antrian.....	11
Modul 3. <i>Markov Chain</i>	22
Modul 4. <i>Game Theory</i>	33
Modul 5. <i>Dynamic Programming</i>	44

PENDAHULUAN

I. LATAR BELAKANG

Kemampuan mahasiswa untuk memahami teori dalam suatu bidang ilmu diharapkan dapat ditunjang oleh kegiatan praktek sehingga mahasiswa dapat mengaplikasikan teori yang diperoleh sebelumnya dalam sebuah kasus yang nyata. Kegiatan Praktikum Penelitian Operasional II memberikan sarana praktek agar kompetensi mahasiswa di bidang ilmu Penelitian Operasional dapat meningkat dan pemahaman mahasiswa dapat terintegrasi dengan baik.

Dalam beberapa kondisi, terkadang mahasiswa menemukan kesulitan beradaptasi dengan dunia kerja dan mencoba menjembatani situasi yang dihadapi dengan teori yang didapatkan di perkuliahan. Dengan adanya kegiatan Praktikum Penelitian Operasional II, permasalahan yang mungkin di hadapi mahasiswa terutama pada ruang lingkup analisa kerja dan ergonomi dapat diminimalkan. Pemahaman seorang mahasiswa akan suatu teori akan teruji saat melakukan praktek di kegiatan praktikum.

II. TUJUAN PEMBELAJARAN

Tujuan Pembelajaran praktikum Penelitian Operasional II ini adalah mahasiswa diharapkan:

- Mampu memahami integrasi mata kuliah Penelitian Operasional,
- Mampu menerapkan prinsip-prinsip agar mendapatkan aktivitas produksi yang optimal,
- Mampu mencari nilai maksimal dari profit, hasil produksi, lini perakitan, dan lain sebagainya), dan
- Mampu meminimumkan ketidaktercapaian suatu produksi.

III. MANFAAT

Dengan adanya kegiatan Praktikum Penelitian Operasional II, diharapkan dapat memberikan manfaat bagi mahasiswa, Program Studi, dan pihak Laboratorium :

1. Mahasiswa dapat memahami ruang lingkup Penelitian Operasional secara komprehensif dan terintegrasi.
2. Mahasiswa dapat menyelaraskan teori dengan aplikasinya.
3. Program Studi dapat mengevaluasi kurikulum dan sistem pembelajaran yang saling menunjang dan komprehensif.
4. Pihak Laboratorium dapat mengevaluasi kebutuhan alat dan peralatan penunjang kegiatan praktikum.

IV. MODUL PRAKTIKUM

Praktikum Industri II ini terdapat 5 (lima) modul, yaitu sebagai berikut:

1. *Linear Goal Programming*,
2. Sistem Antrian,
3. *Markov Chain*,
4. *Game Theory*, dan
5. *Dynamic Programming*

PETUNJUK UMUM

Berikut merupakan petunjuk umum berupa tata tertib yang di gunakan selama kegiatan praktikum berlangsung :

I. TATA TERTIB UMUM

1. Praktikan diwajibkan hadir di ruang praktikum sesuai dengan waktu yang telah ditentukan dengan batas keterlambatan maksimal 10 menit (bagi yang tidak ada konfirmasi). Batas banyaknya keterlambatan ialah maksimal 2 kali.
2. Hasil pengerjaan latihan soal dikerjakan setelah penjelasan modul dengan waktu pengumpulan hingga batas yang ditentukan yaitu pukul 16.45 WITA yang berupa file pdf pengerjaan dan software LINGO, POM QM, WINQSB file word (berisi langkah pengerjaan dan analisa singkat hasil pengerjaan) yang dikumpulkan melalui email praktikumor2@gmail.com
3. Praktikan dapat mengikuti kegiatan praktikum apabila laporan praktikum sebelumnya telah disetujui oleh asisten praktikum.
4. Praktikan dianggap inhal jika:
 - a. Terlambat mengikuti kegiatan praktikum dari batas waktu yang telah ditentukan.
 - b. Tidak mengumpulkan lampiran secara lengkap pada laporan modul bersangkutan.
 - c. Tidak dapat menyelesaikan laporan modul sebelumnya hingga batas waktu yang telah ditentukan.
 - d. Terindikasi melakukan kecurangan dalam pembuatan laporan.
 - e. Terlambat dalam mendapatkan ACC perihal soal yang dibuat lebih dari waktu yang telah ditentukan.
 - f. Tidak memakai seragam PDH HMTI saat praktikum.
5. Jika dalam satu kelompok ada salah satu praktikan yang inhal, maka inhal berlaku bagi kelompok yang bersangkutan.
6. Inhall berupa pengurangan nilai.

7. Praktikan dianggap gugur (tidak lulus) jika:
 - a. Telah inhall lebih dari dua kali inhall asistensi soal, dua kali inhall laporan modul, dan dua kali inhall laporan resmi.
 - b. Tidak menyelesaikan laporan akhir praktikum.
 - c. Tidak mengikuti *final test* praktikum.
 - d. Telah melakukan keterlambatan lebih dari batas waktu yang telah ditentukan

A. TATA TERTIB BERKENAAN DENGAN RESPONSI

1. Responsi berupa latihan soal yang berkenaan dengan modul yang akan dipraktikkan dan diperhitungkan sebagai nilai kelompok. Hal ini bertujuan agar praktikan mengerti prosedur dan materi praktikum.
2. Waktu pengerjaan responsi adalah 1 jam dan allowance waktu pengumpulan adalah 5 menit hanya bagi yang mengalami kendala jaringan dan telah menginformasikan ke aslab disertai bukti yang kuat dan relevan, bagi yang terlambat mengumpulkan responsi akan diberikan pengurangan 2 poin setiap menitnya (waktu dengan sistem pembulatan kebawah).

B. TATA TERTIB BERKENAAN DENGAN ASISTENSI

1. Selain kegiatan praktikum, terdapat **minimal dua kali** asistensi laporan secara offline (tatap muka) untuk penulisan laporan setiap modul sebelum laporan diterima oleh asisten. Waktu pengumpulan laporan dikumpulkan sampai batas pukul 09.00 WITA.
2. Setiap kelompok WAJIB menyertakan catatan pengerjaan yang ditulis pada bagian cover laporan.
3. Untuk waktu pelaksanaan asistensi soal harus sudah mendapat persetujuan aslab hingga batas jam 23.00 WITA dihari yang sama dengan hari praktikum modul yang bersangkutan, asistensi melalui personal chat kepada masing-masing aslab dengan mencantumkan soal (studi kasus) dan screenshot hasil pengerjaan software.

C. TATA TERTIB BERKENAAN DENGAN LAPORAN

1. Asistensi soal merupakan prasyarat pengerjaan laporan.
2. Laporan tiap modul diketik dengan rapi dengan disertai Cover Laporan.
3. Laporan dikumpulkan setiap hari Jumat, Sabtu, dan Minggu dengan batas pengumpulan pukul 09.00 WITA dalam bentuk hard file, PDF dan Word beserta

lampiran yaitu Jurnal ISSN 5 tahun terakhir dan di *highlight*/ditandai (*online*), *file* LINGO, POM QM dan WIN-QSB (*online*).

II. KOMPOSISI NILAI PRAKTIKUM

Nilai praktikum didapat dari hasil penilaian yang meliputi sebagai berikut:

1. Kehadiran (5%)
2. Keaktifan (10%)
3. Responsi (15%)
4. Proses Asistensi (20%)
5. Tanya jawab (10%)
6. Kemampuan Praktek (30%)
7. Laporan Akhir (10%)

MODUL I *LINEAR GOAL PROGRAMMING*

1.1. Tujuan Praktikum

Adapun tujuan dari praktikum mengenai *linear goal programming* antara lain sebagai berikut:

1. Praktikan mampu untuk menyelesaikan persoalan *linear goal programming* dengan komputerisasi.
2. Praktikan mampu untuk menganalisis hasil (*output*) dari persoalan *linear goal programming*.

1.2. Dasar Teori

Linear goal programming merupakan pengembangan *linear programming*. *Goal programming* diperkenalkan oleh Charnes dan Cooper. *Linear goal programming* merupakan modifikasi atau variasi khusus dari *linear goal programming* yang memiliki banyak tujuan. *Linear goal programming* bertujuan untuk meminimumkan penyimpangan-penyimpangan dari tujuan-tujuan tertentu dengan mempertimbangkan hierarki prioritas. Model *linear goal programming* banyak diterapkan dalam beberapa situasi pengambilan keputusan, seperti bidang *finance*, produksi, *marketing*, dan akuntansi. Unsur-unsur *linear goal programming* adalah fungsi tujuan, kendala-kendala, kendala *non-negative*, dan kendala *structural* (Widya, 2014).

1.3. Penyelesaian *Linear Goal Programming*

Toei *Company* merupakan perusahaan dibidang industri per filman, dalam satu hari Toei *Company* membuat produk *Figure*, *Costume*, dan DVD. Untuk memproduksi *Figure* memakan waktu (12) jam proses pembuatan, (5) jam pemeriksaan, dan (4) jam proses *finishing*, dalam memproduksi *Costume* memakan waktu (5) jam proses pembuatan, (8) jam pemeriksaan dan (8) jam proses *finishing*, dan memproduksi DVD memakan waktu (9) jam proses pembuatan, (9) jam pemeriksaan dan (5) jam proses *finishing*. Adapun untuk proses produksinya secara keseluruhan membutuhkan waktu (130) jam untuk pembuatan, (110) jam pemeriksaan, dan (85) jam untuk *finishing*.

Toei *Company* menentukan 4 tujuan sebagai berikut:

1. Meminimumkan ketidak tercapaian total produksi sebesar (25) perminggu,
2. Meminimumkan ketidak tercapaian total produksi sebesar (22) figure perminggu,
3. Meminimumkan ketidak tercapaian total produksi sebesar (28) costume perminggu,
4. Meminimumkan ketidak tercapaian total produksi sebesar (32) DVD perminggu,

1.4. Penyelesaian Masalah *Linear Goal Programming*

Berdasarkan studi kasus diatas, diketahui bahwa Toei *Company* memiliki 4 tujuan yang harus dicapai, kemudian diformulasikan model GP untuk menyelesaikan permasalahan di atas menggunakan *software* LINGO. Adapun prosedur penggunaan untuk menyelesaikan persoalan *linear goal programming* dengan *software* LINGO dapat dijabarkan sebagai berikut:

1. Membuka *file* LINGO dengan cara klik dua kali pada ikon LINGO.
2. Pada layar akan muncul *untitled* baru yang siap untuk tempat mengetikkan formulasi.
3. Berdasarkan persoalan diatas, maka terlebih dahulu ditentukan variabel keputusan yang kemudian diterjemahkan ke dalam model matematis seperti berikut:

Variabel keputusan :

- | | | |
|-------------|---|---|
| X1 | : | Jumlah produksi <i>Figure</i> yang harus diselesaikan (unit) |
| X2 | : | Jumlah produksi <i>Costume</i> yang harus diselesaikan (unit) |
| X3 | : | Jumlah produksi DVD yang harus diselesaikan (unit) |
| Tujuan | : | |
| Prioritas 1 | : | Meminimumkan ketidak tercapaian total produksi sebesar (25) perminggu (<i>goal</i> 1) |
| Prioritas 2 | : | Meminimumkan ketidak tercapaian total produksi sebesar (22) <i>figure</i> perminggu (<i>goal</i> 2) |
| Prioritas 3 | : | Meminimumkan ketidak tercapaian total produksi sebesar (28) <i>costume</i> perminggu (<i>goal</i> 3) |
| Prioritas 4 | : | Meminimumkan ketidak tercapaian total produksi sebesar (32) DVD perminggu (<i>goal</i> 4) |

Formulasikan ke dalam model matematis:

Fungsi tujuan: $\text{Min } P_1d_1^- + P_2d_2^- + P_3d_3^-$

Batasan:

$$12X_1 + 5X_2 + 9X_3 \leq 130$$

$$5X_1 + 8X_2 + 9X_3 \leq 110$$

$$4X_1 + 8X_2 + 5X_3 \leq 85$$

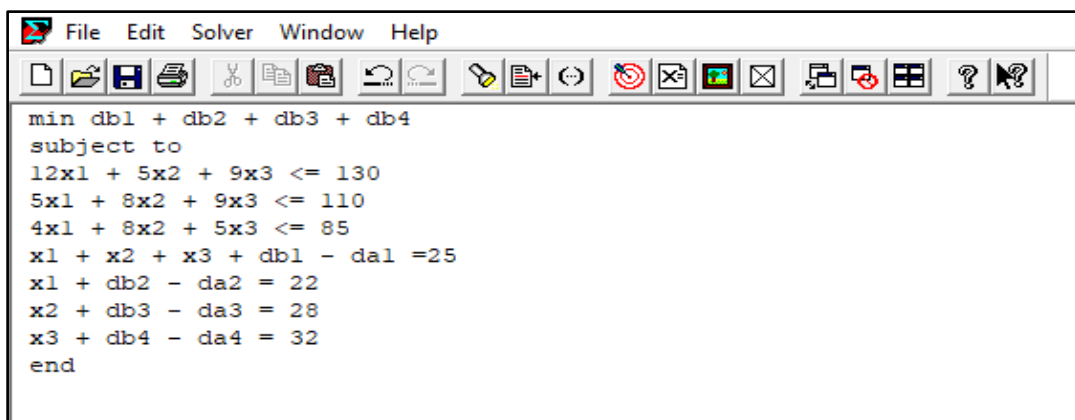
$$X_1 + X_2 + X_3 + d_1^- - d_1^+ = 25$$

$$X_1 + db_2^- - da_2^+ = 22$$

$$X_2 + db_3^- - da_3^+ = 28$$

$$X_3 + db_4^- - da_4^+ = 32$$

4. Diketik formulasi model matematis yang telah dibuat diatas pada layar LINGO seperti pada gambar di bawah ini:



```

min db1 + db2 + db3 + db4
subject to
12x1 + 5x2 + 9x3 <= 130
5x1 + 8x2 + 9x3 <= 110
4x1 + 8x2 + 5x3 <= 85
x1 + x2 + x3 + db1 - da1 =25
x1 + db2 - da2 = 22
x2 + db3 - da3 = 28
x3 + db4 - da4 = 32
end

```

Gambar 1.1 Formulasi model matematis *software* LINGO

5. Diklik *Solver* lebih dari *Solve* pada toolbar untuk melihat solusi dari perumusan masalah tersebut.
6. Maka akan muncul kotak dialog *Solution Report – Lingo1* dari permasalahan di atas seperti pada gambar dibawah ini:

```

Global optimal solution found.
Objective value:                77.00000
Infeasibilities:                0.000000
Total solver iterations:        7
Elapsed runtime seconds:        0.06

Model Class:                    LP

Total variables:                11
Nonlinear variables:            0
Integer variables:              0

Total constraints:              8
Nonlinear constraints:          0

Total nonzeros:                27
Nonlinear nonzeros:            0

Variable      Value      Reduced Cost
DB1           10.00000    0.000000
DB2           17.00000    0.000000
DB3           23.00000    0.000000
DB4           27.00000    0.000000
X1            5.000000    0.000000
X2            5.000000    0.000000
X3            5.000000    0.000000
DA1           0.000000    1.000000
DA2           0.000000    1.000000
DA3           0.000000    1.000000
DA4           0.000000    1.000000

Row    Slack or Surplus    Dual Price
1      77.00000            -1.000000
2      0.000000            0.1011673
3      0.000000            0.3891051E-01
4      0.000000            0.1478599
5      0.000000            -1.000000
6      0.000000            -1.000000
7      0.000000            -1.000000
8      0.000000            -1.000000

```

Gambar 1.2 Output LINGO

1.5. Analisis Output Linear Goal Programming

Untuk menyelesaikan permasalahan *Linear Goal Programming* dengan menggunakan *software* akan didapatkan *output* hasil dari penyelesaian masalah tersebut. Analisis *output* dengan menggunakan *software* LINGO dapat dilihat dibawah ini. Hasil dari permasalahan Toei Company yang menginginkan produksi *Figure*, *Costume*, dan DVD yang sesuai dengan prioritas-prioritas yang diinginkan.

1. Objective Function Value

Objective Function Value merupakan nilai objektif dari fungsi tujuan yaitu $\text{Min } P_1d_1^- + P_2d_2^- + P_3d_3^-$ dengan nilai 77. Nilai 77 didapatkan karena jumlah kekurangan dari prioritas-prioritas yang telah diberikan. Dengan penjelasan *Goal 1* kekurangan 10 unit, *Goal 2* kekurangan 17 unit, *Goal 3* kekurangan 23 unit dan *Goal 4* kekurangan 27 unit.

2. Value

Nilai *value* dari permasalahan diatas, yaitu:

- a. *Variable X1* yaitu jumlah produksi *figure* yang harus dikerjakan sebanyak 5 unit.

- b. *Variable X2* yaitu jumlah produksi *costume* yang harus dikerjakan 5 unit.
 - c. *Variable X2* yaitu jumlah produksi DVD yang harus dikerjakan 5 unit.
 - d. *Variable DB1* yaitu 10 yang berarti *goal 1* kekurangan 10 unit.
 - e. *Variable DB2* yaitu 17 yang berarti *goal 2* kekurangan 17 unit.
 - f. *Variable DB3* yaitu 23 yang berarti *goal 3* kekurangan 23 unit.
 - g. *Variable DB4* yaitu 27 yang berarti *goal 4* kekurangan 27 unit.
 - h. *Variable DA1, DA2, DA3, dan DA4* yaitu 0 yang berarti tidak terdapat penyimpangan pada variabel-variabel ini atau semua *goal* pada variabel DA1, DA2, DA3 dan D4 telah terpenuhi.
3. *Reduced Cost*
- Nilai *reduced cost* dari permasalahan diatas, yaitu:
- a. *Variable X1, X2, dan X3* yaitu 0 yang berarti jumlah pesanan baju yang dikerjakan sudah optimal dengan pesanan secara berurut sebanyak 5 unit, 5 unit, dan 5 unit.
 - b. *Variable DB1, DB2, DB3, dan DB4* yaitu 0 yang berarti tidak ada pengurangan keuntungan atau sudah optimal.
 - c. *Variable DA1, DA2, DA3* yaitu 1, 1, 1, 1 berarti tidak optimal karena masih terdapat pengurangan keuntungan.
4. *Slack or Surplus*
- Nilai *slack* dari permasalahan diatas untuk *row 2* hingga 8 yaitu 0 yang berarti bahwa dari S1, S2, S3, S4, S5, S6 dan S7 sebagai *variable non basis*.
5. *Dual Price*
- Nilai *dual price* pada permasalahan diatas untuk *row 2* hingga *row 8* berturut-turut sebesar 0,1; 0,3; 0,1 ; -1; -1; -1; dan -1. Maka pada kendala 1 yaitu 0,1 yang berarti apabila produk tersebut ditambah maka akan menambah 0,1 unit disetiap kendalanya. Pada Kendala 2 dan 3 yaitu 0,3 dan 0,1 yang berarti apabila produk tersebut ditambah maka akan menambah 0,3 dan 0,1 unit disetiap kendalanya. Pada kendala 4 sampai kendala 7 yaitu -1 yang berarti apabila produk tersebut ditambah maka akan mengurangi 1 unit disetiap kendalanya.

MODUL II TEORI ANTRIAN

2.1 Tujuan Praktikum

Adapun tujuan dari praktikum mengenai teori antrian antara lain sebagai berikut:

1. Praktikan mampu untuk menyelesaikan persoalan sistem antrian dengan komputerisasi.
2. Praktikan mampu untuk menganalisis hasil (*output*) dari persoalan sistem antrian.

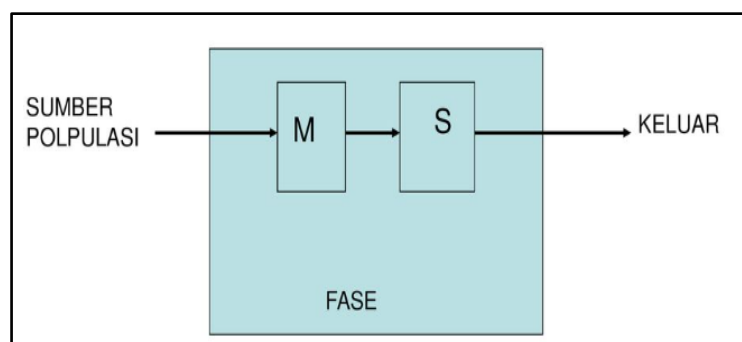
2.2 Dasar Teori

Teori antrian adalah teori yang menyangkut studi matematis dari antrian-antrian atau baris-baris penungguan (Dimiyati, 1992). Pelopor penyusunan teori antrian adalah seorang ahli matematika dari Denmark, Agner Kramp Erlang (1878-1929). Penulisan model antrian pada umumnya mengikuti notasi Kendall-Lee bentuk $a/b/c/d/e$. Suatu antrian adalah garis tunggu dari pelanggan yang memerlukan layanan dari satu atau lebih pelayanan. Studimatematika dari garis tunggu ini disebut teori antrian.

Tujuan dasar model-model antrian adalah untuk meminimumkan total biaya langsung penyediaan fasilitas pelayanan dan biaya tidak langsung karena para individu harus menunggu untuk dilayani.

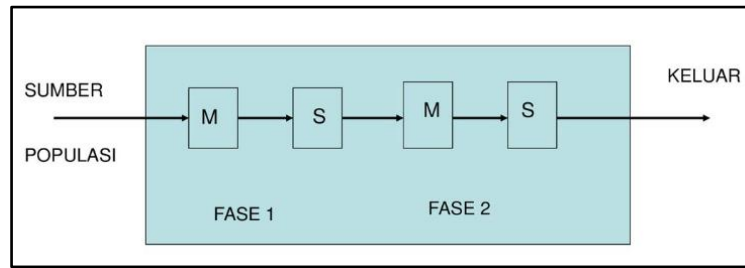
Model struktur antrian secara umum adalah sebagai berikut:

1. *Single channel-single phase*, menunjukkan bahwa dalam struktur hanya ada satu jalur untuk memasuki sistem pelayanan atau ada satu fasilitas pelayanan.



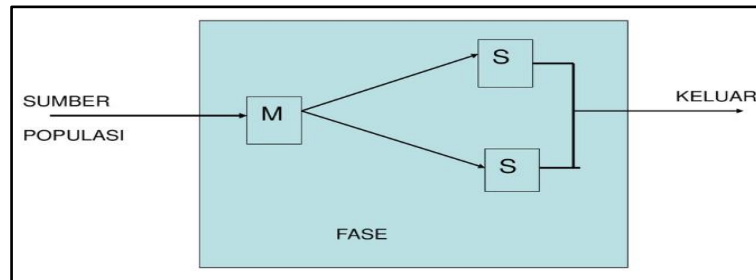
Gambar 2.1 Struktur Antrian *Single Channel-Single Phase*

2. *Single channel-multiphase*, menunjukkan bahwa ada dua atau lebih pelayanan yang dilaksanakan secara berurutan.



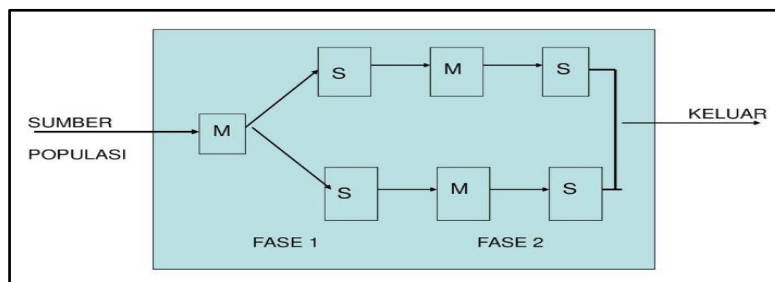
Gambar 2.2 Struktur Antrian *Single Channel-Multiphase*

3. *Multichannel - single phase*, menunjukkan bahwa terdapat dua atau lebih fasilitas pelayanan dialiri oleh antrian tunggal.



Gambar 2.3 Struktur Antrian *Multichannel-single phase*

4. *Multichannel-multiphase*, menunjukkan bahwa terdapat dua atau lebih fasilitas pelayanan dan pelayanan yang dilaksanakan.



Gambar 2.4 Struktur Antrian *Multichannel-multiphase*

2.3 Penyelesaian Teori Antrian

Berikut adalah cara penyelesaian masalah sistem antrian dengan komputerisasi:

Contohkasus:

Kafe Cetro yang berlokasi di Jalan Siradj Salman memiliki 1 unit *server* antrian untuk pemesanan minuman dan makanan ringan. Dalam sebuah penelitian di kafe ini, didapatkan waktu rata-rata untuk melayani pelanggan adalah 60 pelanggan/jam. Pelanggan yang datang rata-rata adalah 85 pelanggan/jam. Gambar antrian di Kafe Cetro dapat dilihat pada Gambar 2.5. Tentukan:

1. Tingkat rata-rata kesibukan sistem,
2. Rata-rata jumlah pelanggan dalam garis tunggu,
3. Jumlah rata-rata pelanggan di dalam sistem,
4. Waktu rata-rata pelanggan di garis tunggu, dan
5. Waktu rata-rata pelanggan berada dalam sistem.



Gambar 2.5 Antrian Kafe Cetro

2.4 Penyelesaian Masalah Teori Antrian

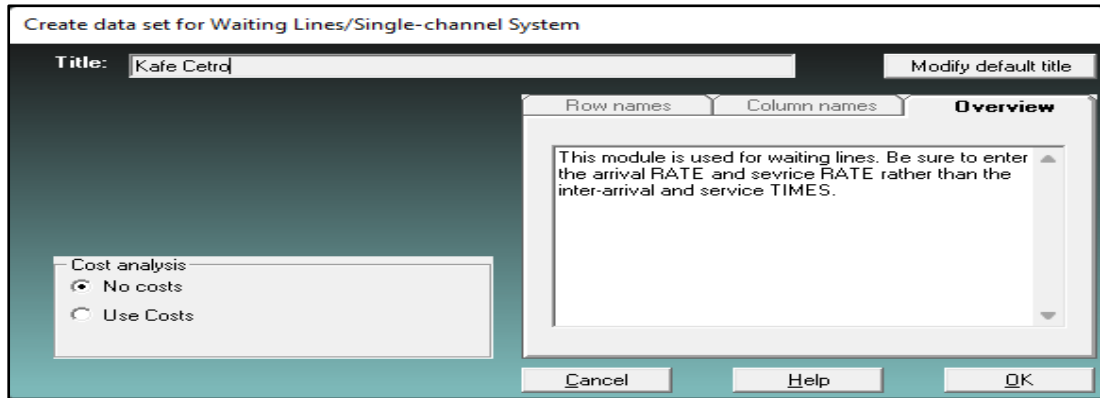
Permasalahan teori antrian di atas akan diselesaikan secara komputerisasi dengan *software* POM-QM *For Windows*. Di bawah ini adalah langkah-langkah perhitungan untuk permasalahan yang ada di atas dengan menggunakan *software* POM-QM *For Windows*.

1. *Single Channel- Single Phase*

Langkah-langkah untuk menyelesaikan persoalan teori antrian dengan *software* POM-QM *For Windows* adalah sebagai berikut:

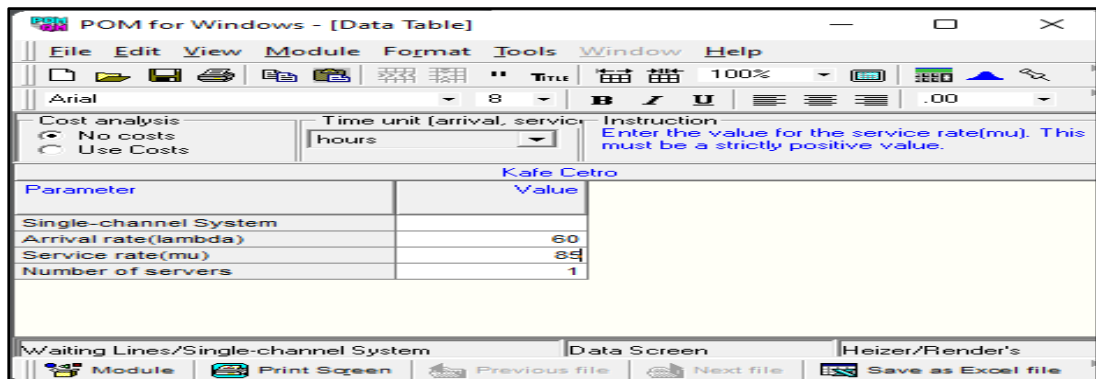
- a. Klik dua kali pada *icon* POM-QM *For Windows* untuk membuka aplikasi POM-QM *For Windows*.
- b. Klik *Module*, kemudian pilih *waiting line* pada *toolbar*.
- c. Layar kosong akan muncul, kemudian klik *File*. Lalu Klik *New*, kemudian pilih *Single-channel System* pada *toolbar*.

- d. Kotak dialog *Create data set for Waiting Lines/Single-channel System* akan muncul. Ketik judul “Kafe Cetro” pada *Title* dan pilih *no cost* pada *Cost Analysis* seperti yang dapat dilihat pada Gambar 2.6.



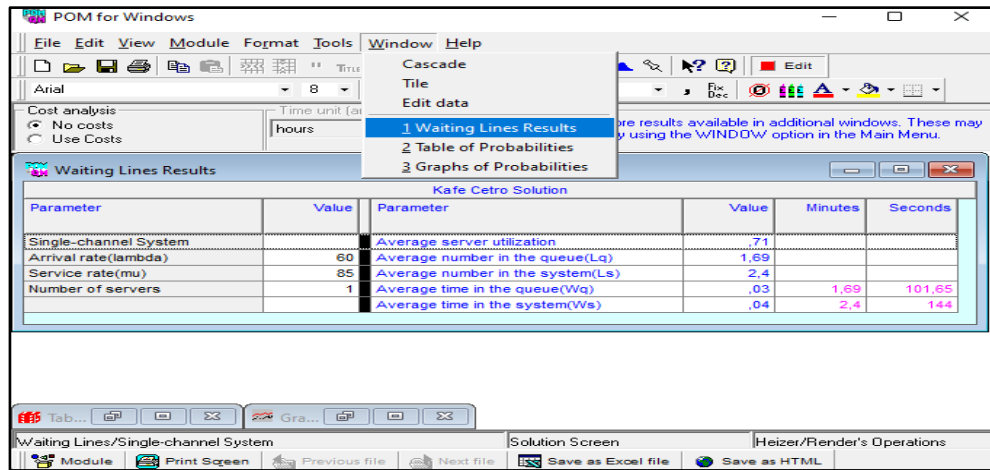
Gambar 2.6 Kotak dialog *Create data set for Waiting Lines*

- e. Klik OK, akan muncul *Data Table* yang akan diisi sesuai dengan data yang ada pada permasalahan di atas yaitu: *arrival rate (lambda) = 23*, *service rate (mu) = 30*, dan *number of service = 1*, seperti yang dapat dilihat pada Gambar 2.7.



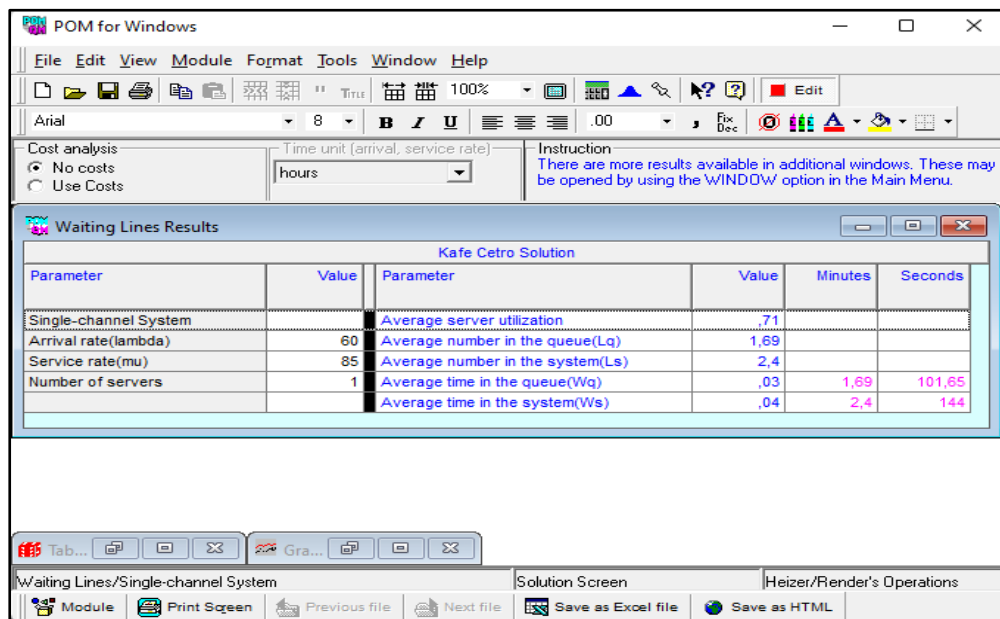
Gambar 2.7 *Data Table*

- f. Klik *Solve* pada *toolbar*, lalu akan muncul kotak dialog *output* pada *toolbar windows* yang akan keluar seperti pada Gambar 2.8.



Gambar 2.8 Kotak dialog output

- g. Untuk melihat hasil yang akan diperoleh maka pilih *Waiting Lines Result* sehingga menghasilkan *output* seperti pada Gambar 2.9.



Gambar 2.9 *Waiting lines result*

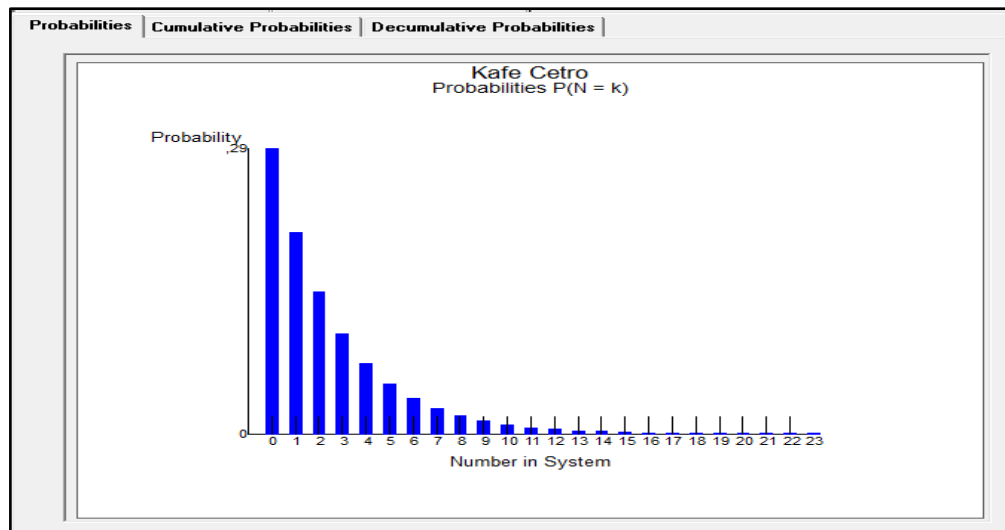
- h. Untuk melihat hasil tabel probabilitas maka pada *toolbar Window* pilih *Table of Probabilities* sehingga menghasilkan *output* seperti pada Gambar 2.10.

The screenshot shows the 'Table of Probabilities' window in POM for Windows. The window title is 'POM for Windows - [Table of Probabilities]'. The menu bar includes File, Edit, View, Module, Format, Tools, Window, and Help. The toolbar contains various icons for file operations and editing. The main area displays a table with the following data:

k	Prob (num in sys = k)	Prob (num in sys <= k)	Prob (num in sys >k)
0	.29	.29	.71
1	.21	.5	.5
2	.15	.65	.35
3	.1	.75	.25
4	.07	.82	.18
5	.05	.88	.12
6	.04	.91	.09
7	.03	.94	.06
8	.02	.96	.04
9	.01	.97	.03
10	0	.98	.02
11	0	.98	.02
12	0	.99	.01
13	0	1	0
14	0	1	0
15	0	1	0
16	0	1	0
17	0	1	0
18	0	1	0
19	0	1	0
20	0	1	0
21	0	1	0
22	0	1	0
23	0	1	0

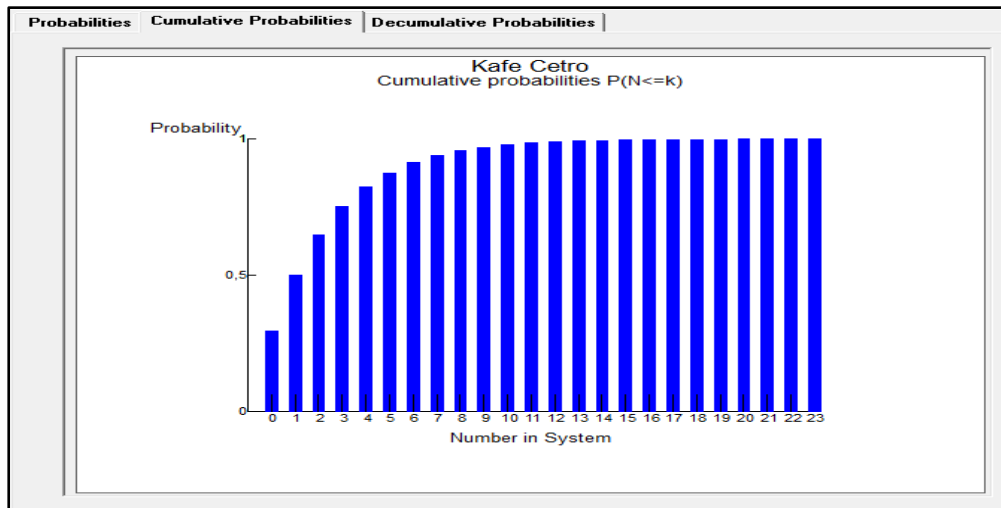
Gambar 2.10 Table of probabilities

- i. Untuk melihat hasil grafik probabilitas maka pada *toolbar Window* pilih *Graphs of Probabilities* sehingga menghasilkan *output* seperti pada Gambar 2.11.



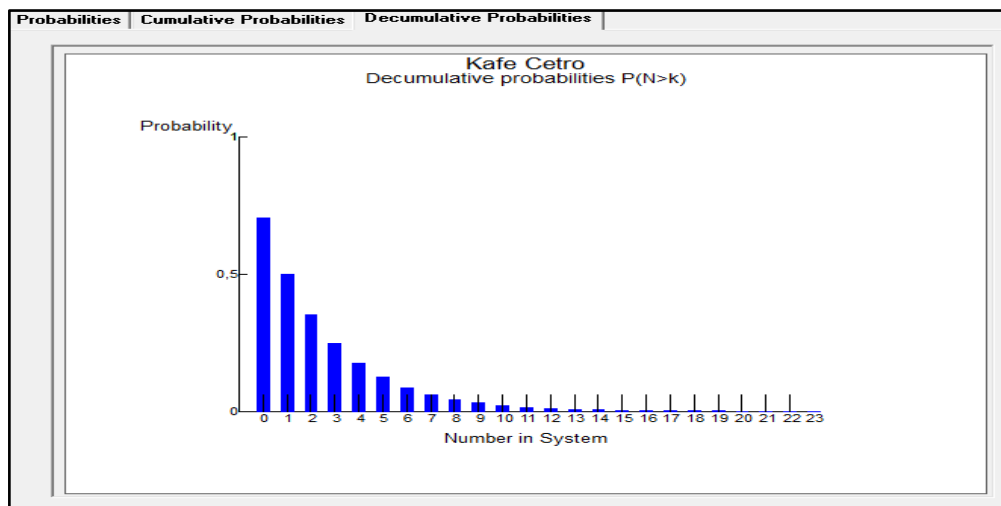
Gambar 2.11 Graph of probabilities

- j. Klik *Cumulative Probabilities*, maka akan keluar kotak dialog *Graph of Cumulative Probabilities* pada *toolbar window* yang akan keluar seperti pada Gambar 2.12.



Gambar 2.12 *Graph of Cumulative Probabilities*

- k. Klik *Decumulative Probabilities*, maka akan keluar kotak dialog *Graph of Decumulative Probabilities* pada *toolbar window* yang akan keluar seperti pada Gambar 2.13.



Gambar 2.13 *Graph of Decumulative Probabilities*

- l. Setelah *output* didapatkan, maka berikutnya dilakukan analisis *output* atau interpretasi hasil.

2.5 Analisis Output Teori Antrian

Penyelesaian masalah teori antrian menggunakan *software* POM-QM For Windows akan mengeluarkan *output* berupa hasil dari penyelesaian masalah. Berikut merupakan analisis dari *output* teori antrian:

1. *Single Channel- Single Phase*

Berikut merupakan analisis *output* teori antrian dari permasalahan di Kafe Cetro yang merupakan permasalahan *Single Channel- Single Phase*:

a. *Waiting Line Result*

1) *Average server utilitation*

Output dari hasil perhitungan mengenai permasalahan di atas menggunakan *software POM-QM For Windows* menunjukkan nilai *average server* yaitu 0,71 yang menunjukkan bahwa tingkat kesibukan sistem dalam antrian Kafe Cetro adalah 0,71 atau 71%.

2) *Average number in the queue (Lq)*

Output dari hasil perhitungan mengenai permasalahan di atas menggunakan *software POM-QM For Windows* menunjukkan nilai *average number in the queue (Lq)* yaitu 1,69 yang menunjukkan bahwa rata-rata jumlah pelanggan dalam garis tunggu atau antrian Kafe Cetro adalah 1,69 orang/jam.

3) *Average number in the system (Ls)*

Output dari hasil perhitungan mengenai permasalahan di atas menggunakan *software POM-QM For Windows* menunjukkan nilai *average number in the system (Ls)* yaitu 2,4 yang menunjukkan bahwa jumlah rata-rata pelanggan dalam sistem adalah 2,4 orang/jam.

4) *Average time in the queue (Wq)*

Output dari hasil perhitungan mengenai permasalahan di atas menggunakan *software POM-QM For Windows* menunjukkan nilai *average time in the queue (Wq)* yaitu 0,03 yang menunjukkan bahwa waktu rata-rata pelanggan di garis tunggu 0,03 jam atau 1,8 menit.

5) *Average time in the system (Ws)*

Output dari hasil perhitungan mengenai permasalahan di atas menggunakan *software POM-QM For Windows* menunjukkan nilai *average time in the system (Ws)* yaitu 0,04 yang menunjukkan bahwa waktu rata-rata pelanggan berada dalam sistem adalah 0,04 jam atau 2,4 menit.

b. *Table of Probabilities*

Output Table of Probabilities menunjukkan tabel mengenai probabilitas saat ada sebanyak k pelanggan dalam sebuah sistem. Dalam tabel tersebut dapat terlihat probabilitas saat jumlah pelanggan sama dengan k , lebih kecil dibandingkan k atau lebih besar dibandingkan k .

1) *Probabilities (in system = k)*

Pada saat tidak ada antrian maka probabilitas kejadiannya adalah 0,29. Saat ada 1 pelanggan probabilitas kejadiannya adalah 0,21. Saat ada 2 pelanggan probabilitas kejadiannya adalah 0,15. Saat ada 3 pelanggan probabilitas kejadiannya adalah 0,1. Saat ada 4 pelanggan probabilitas kejadiannya adalah 0,07. Saat ada 5 pelanggan probabilitas kejadiannya adalah 0,05. Saat ada 6 pelanggan probabilitas kejadiannya adalah 0,04. Saat ada 7 pelanggan probabilitas kejadiannya adalah 0,03. Saat ada 8 pelanggan probabilitas kejadiannya adalah 0,02. Saat ada 9 pelanggan probabilitas kejadiannya adalah 0,01. Saat ada 10 sampai dengan 23 pelanggan maka probabilitas kejadiannya adalah 0. Sehingga dapat diketahui bahwa probabilitas kejadian terbesar terjadi ketika jumlah pelanggan dalam sistem adalah 0 pelanggan yaitu probabilitasnya sebesar 0,29 Dan probabilitas kejadian terkecil adalah saat ada lebih dari 10 pelanggan yaitu sebesar 0.

2) *Probabilities (in system \leq k)*

Pada saat tidak ada antrian maka probabilitas kejadiannya adalah 0,29. Saat antrian yang terjadi \leq 1 pelanggan probabilitas kejadiannya adalah 0,50. Saat antrian yang terjadi \leq 2 pelanggan probabilitas kejadiannya adalah 0,65. Saat antrian yang terjadi \leq 3 pelanggan probabilitas kejadiannya adalah 0,75. Saat antrian yang terjadi \leq 4 pelanggan probabilitas kejadiannya adalah 0,82. Saat antrian yang terjadi \leq 5 pelanggan probabilitas kejadiannya adalah 0,88. Saat antrian yang terjadi \leq 6 pelanggan probabilitas kejadiannya adalah 0,91. Saat antrian yang terjadi \leq 7 pelanggan probabilitas kejadiannya adalah 0,94. Saat antrian yang terjadi \leq 8 pelanggan probabilitas kejadiannya adalah 0,96. Saat antrian yang terjadi \leq 9 pelanggan probabilitas kejadiannya adalah 0,97. Saat antrian yang terjadi \leq 10 pelanggan probabilitas kejadiannya adalah 0,98. Saat antrian yang terjadi \leq 11 pelanggan probabilitas kejadiannya adalah 0,98. Saat antrian yang terjadi \leq 12 pelanggan probabilitas kejadiannya adalah 0,99. Saat antrian yang terjadi \leq 13 hingga, 22 pelanggan probabilitas kejadiannya adalah 1. Sehingga dapat dilihat bahwa probabilitas antrian yang terjadi akan lebih besar jika kurang dari sama dengan 13 pelanggan karena nilai probabilitasnya yang semakin besar.

3) *Probabilities (in system $>$ k)*

Pada saat tidak ada antrian maka probabilitas kejadiannya adalah 0,71. Saat antrian yang terjadi $>$ 1 pelanggan probabilitas kejadiannya adalah 0,50. Saat antrian yang terjadi $>$ 2 pelanggan probabilitas kejadiannya adalah 0,35. Saat antrian yang terjadi $>$ 3 pelanggan probabilitas kejadiannya adalah 0,25. Saat antrian yang terjadi $>$ 4

pelanggan probabilitas kejadiannya adalah 0,18. Saat antrian yang terjadi > 5 pelanggan probabilitas kejadiannya adalah 0,12. Saat antrian yang terjadi > 6 pelanggan probabilitas kejadiannya adalah 0,09. Saat antrian yang terjadi > 7 pelanggan probabilitas kejadiannya adalah 0,06. Saat antrian yang terjadi > 8 pelanggan probabilitas kejadiannya adalah 0,04. Saat antrian yang terjadi > 9 pelanggan probabilitas kejadiannya adalah 0,03. Saat antrian yang terjadi > 10 pelanggan probabilitas kejadiannya adalah 0,02. Saat antrian yang terjadi > 11 pelanggan probabilitas kejadiannya adalah 0,02. Saat antrian yang terjadi > 12 pelanggan probabilitas kejadiannya adalah 0,01. Saat antrian yang terjadi > 13 hingga, 21 pelanggan maka probabilitas kejadiannya adalah 0. Sehingga dapat dikatakan bahwa probabilitas kejadian saat jumlah antrian pelanggan lebih dari 13 pelanggan sangat kecil nilainya.

c. *Graphs of Probabilities*

Dilakukan analisis untuk *Graph of Probabilities* pada penyelesaian diatas sebagai berikut ini.

1) *Graph of Probabilities (N = K)*

Graph of Probabilities merupakan gambaran dari hasil Table of

Probabilities yang digambarkan kedalam bentuk grafik. Pada saat antrian 0 orang dengan probabilitas kemungkinan terjadi yaitu 0,29, dan pada saat antrian 1 orang dengan probabilitas kemungkinan terjadi adalah 0,20

sehingga terjadi penurunan pada grafik, dan pada saat antrian 2 orang probabilitas kemungkinan terjadi adalah 0,14 sehingga grafik terus mengalami penurunan, pada saat antrian 3 orang probabilitas kemungkinan terjadi adalah 0,1 sehingga grafik tetap mengalami penurunan, dan pada saat antrian 4 orang probabilitas kemungkinan terjadi adalah 0,07 sehingga grafik, pada saat antrian 5 orang probabilitas kemungkinan terjadi adalah 0,05 grafik tetap mengalami penurunan hingga pada saat antrian 6 orang probabilitas kemungkinan terjadi adalah 0,03, saat antrian 7 orang probabilitas kemungkinan terjadi adalah 0,02, pada saat antrian 8 orang probabilitas kemungkinan terjadi adalah 0,01, dan pada saat antrian 9 orang sampai dengan antrian 23 orang nilai probabilitas kemungkinan yang terjadi adalah 0 sehingga grafik berada pada titik 0 secara konstan.

2) *Graph of Cumulative Probabilities (N < K)*

Pada saat antrian 0 orang probabilitas kemungkinan terjadi adalah 0,29, pada saat antrian 1 orang probabilitas kemungkinan terjadi adalah 0,50, pada saat antrian 2

orang probabilitas kemungkinan terjadi adalah 0,64, pada saat antrian 3 orang probabilitas kemungkinan terjadi adalah 0,76, pada saat antrian 4 orang probabilitas kemungkinan terjadi yaitu 0,83, pada saat antrian 5 orang probabilitas kemungkinan terjadi adalah 0,87, pada saat antrian 6 orang probabilitas kemungkinan terjadi adalah 0,91 sehingga grafik tetap mengalami kenaikan, pada saat antrian 7 orang probabilitas kemungkinan terjadi adalah 0,93, pada saat antrian 8 orang probabilitas kemungkinan terjadi adalah 0,95, pada saat antrian 9 orang probabilitas kemungkinan terjadi adalah 0,97 sehingga grafik terus mengalami kenaikan, pada saat antrian 10 orang probabilitas kemungkinan terjadi adalah 0,99, dan grafik tetap mengalami kenaikan konstan pada saat antrian 11 orang sampai dengan 23 orang dengan probabilitas kemungkinan terjadi adalah 1.

3) *Graph of Decumulative Probabilities ($N > K$)*

Pada saat antrian 0 orang dengan probabilitas kemungkinan terjadi yaitu 0,70, dan pada saat antrian 1 orang dengan probabilitas kemungkinan terjadi adalah 0,50 sehingga terjadi penurunan pada grafik, dan pada saat antrian 2 orang probabilitas kemungkinan terjadi adalah 0,35 sehingga grafik terus mengalami penurunan, pada saat antrian 3 orang probabilitas kemungkinan terjadi adalah 0,24 sehingga grafik tetap mengalami penurunan, dan pada saat antrian 4 orang probabilitas kemungkinan terjadi adalah 0,18 sehingga grafik, pada saat antrian 5 orang probabilitas kemungkinan terjadi adalah 0,12 grafik tetap mengalami penurunan hingga pada saat antrian 6 orang probabilitas kemungkinan terjadi adalah 0,08, saat antrian 7 orang probabilitas kemungkinan terjadi adalah 0,07, pada saat antrian 8 orang probabilitas kemungkinan terjadi adalah 0,04, pada saat antrian 9 orang probabilitas kemungkinan terjadi adalah 0,02, pada saat antrian 10 orang probabilitas kemungkinan terjadi adalah 0,01, dan pada saat antrian 11 orang sampai dengan 23 orang probabilitas kemungkinan yang terjadi adalah 0 sehingga grafik berada pada titik 0 secara konstan.

MODUL III *MARKOV CHAIN*

3.1 Tujuan Praktikum

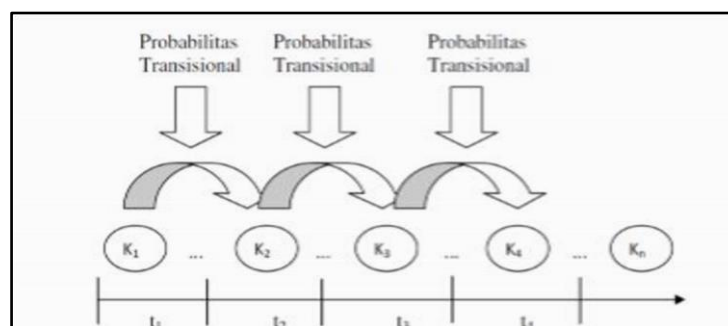
Adapun tujuan dari praktikum mengenai *markov chain* antara lain sebagai berikut:

1. Praktikan mampu untuk menyelesaikan persoalan *markov chain* dengan komputerisasi, dan
2. Praktikan mampu untuk menganalisis hasil (*output*) dari persoalan *markov chain* dengan komputerisasi.

3.2 Dasar Teori

Rantai Markov (*Markov Chains*) adalah suatu teknik matematika yang biasa digunakan untuk melakukan pemodelan (*modelling*) bermacam-macam sistem dan proses bisnis. Teknik ini dapat digunakan untuk memperkirakan perubahan-perubahan di waktu yang akan datang dalam variabel-variabel dinamis atas dasar perubahan-perubahan dari variabel-variabel dinamis tersebut di waktu yang lalu. Teknik ini dapat digunakan juga untuk menganalisis kejadian-kejadian pada waktu-waktu mendatang secara matematis (Andry, 2015).

Gambaran mengenai rantai Markov ini kemudian dituangkan dalam Gambar 2.1 Peristiwa Dalam Rantai Markov, dimana gerakan-gerakan dari beberapa variabel di masa yang akan datang bisa diprediksi berdasarkan gerakan-gerakan variabel tersebut pada masa lalu. K_t4 dipengaruhi oleh kejadian K_t3 , K_t3 dipengaruhi oleh kejadian K_t2 dan demikian seterusnya dimana perubahan ini terjadi karena peranan probabilitas transisi (*transition probability*) (Andry, 2015).



Gambar 3.1 Peristiwa Dalam Rantai Markov

Rantai markov ini dikenalkan oleh Andrei A. Markov, ahli matematika dari Rusia yang lahir tahun 1856 (Ching, W., dan Ng, M. 2006). Analisis markov menghasilkan suatu informasi probabilistik yang dapat digunakan untuk membantu pembuatan keputusan, jadi analisis ini bukan suatu teknik optimisasi melainkan suatu teknik deskriptif. Analisis markov merupakan suatu bentuk khusus dari model probabilistik yang lebih umum dinamakan dengan *Stochastic process*, yaitu proses perubahan perubahan probabilistik yang terjadi terus –menerus (Putra, 2018).

Konsep dasar analisis markov adalah *state* dari sistem atau *state* transisi, sifat dari proses ini adalah apabila diketahui proses berada dalam suatu keadaan tertentu, maka peluang berkembangnya proses di masa mendatang hanya tergantung pada keadaan saat ini dan tidak tergantung pada keadaan sebelumnya, atau dengan kata lain rantai Markov adalah rangkaian proses kejadian dimana peluang bersyarat kejadian yang akan datang tergantung pada kejadian sekarang (Putra, 2018).

Menurut Basjaraduin (2016), beberapa syarat yang harus dipenuhi pada analisis Markov adalah sebagai berikut:

1. Jumlah probabilitas transisi keadaan awal dari sistem sama dengan 1,
2. Probabilitas tersebut berlaku untuk semua partisipan dalam sistem,
3. Probabilitas transisi konstan sepanjang waktu, dan
4. Kondisi merupakan kondisi yang independen sepanjang waktu.

3.3 Penyelesaian Rantai Markov

Penyelesaian pada persoalan masalah *markov chain* dapat diselesaikan dengan secara komputerisasi menggunakan *software POM-QM For Windows* dan *WinQSB*.

Terdapat 3 toko *handphone* yang terdapat di Kota Samarinda yaitu Surya Phone, Erafone dan Carin Cell. Jumlah seluruh pelanggan dari ke 3 toko *handphone* tersebut sebanyak 530. Berikut adalah hasil dari penelitian ke 3 toko *handphone* tersebut.

Tabel 3.1 Jumlah Pelanggan Toko Handphone

Toko	Banyaknya pelanggan	
	Bulan pertama	Bulan Kedua
Surya Phone	132	175
Erafone	178	173
Carin Cell	215	177
Jumlah	525	525

Dari data diatas dapat dilihat adanya perubahan dari bulan pertama ke bulan kedua. Pada Surya Phone bertambah 43, Erafone berkurang 5 dan Carin Cell berkurang 38. Berikut merupakan data detail perubahan tersebut.

Tabel 3.2 Detail Perubahan Jumlah Pelanggan

Bulan Pertama	Bulan kedua			Jumlah
	Surya Phone	Erafone	Carin Cell	
Surya Phone	45	45	42	132
Erafone	55	68	55	178
Carin Cell	75	60	80	215
Jumlah	175	173	177	525

1. Buatlah tabel probabilitas transisi dan matriks transisi!
2. Hitunglah presentase untuk ketiga toko *handphone* tersebut dalam 2 periode mendatang?
3. Jika harga rata-rata :

Surya Phone = Rp 1.200.000

Erafone = Rp 1.500.000

Carin Cell = Rp 1.000.000

3.4 Penyelesaian Masalah Rantai Markov

1. Software POM-QM

Penyelesaian pada persoalan masalah *markov* pada toko *handphone* di atas dapat diselesaikan secara komputerisasi menggunakan *software POM-QM For Windows*. Berikut dijabarkan prosedur untuk menyelesaikan permasalahan di atas dengan menggunakan *software POM-QM For Windows*.

- a. Pertama, buka aplikasi *POM-QM For Windows* dengan cara klik dua kali pada *icon POM-QM For Windows*.
- b. Selanjutnya, pada layar akan memunculkan layar baru, kemudian klik *Module* kemudian *Markov Analysis* pada *toolbar*.

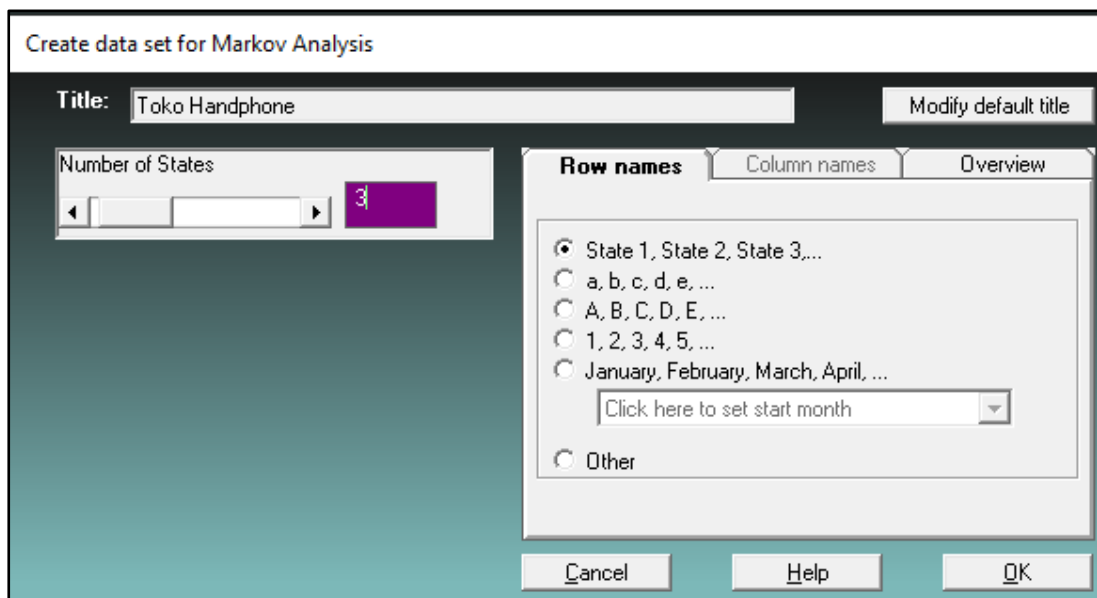
- c. Setelah itu akan muncul layar kosong, kemudian klik *File* selanjutnya klik *New* lalu *Multichannel System* pada *toolbar*.
- d. Kemudian muncul kotak dialog *Create data set for Markov Analysis*, kemudian pada kotak dialog *title* masukkan seperti pada Gambar 3.1.

Masukan pada *title* judul dengan nama *Toko Handphone*.

Isi *Number of States* sebanyak 3.

Pilih salah satu pilihan pada *Row names*, pilih *state 1, state 2, state 3,...*

Detail tampilan pengisian kotak dialog *Create data set for Markov Analysis* adalah seperti pada Gambar 3.1



Gambar 3.1 *Create data set for Markov Analysis*

- e. Pada kotak dialog diklik *OK*, tampilan *Data Table* diisi sesuai dengan *number of transition* dengan 2 untuk mengetahui data 2 periode setelahnya, dan model matematis yaitu dimasukkan nilai probabilitas dari keempat restoran. Berikut adalah model matematis yang dimasukkan seperti pada Gambar 3.2.

	Initial	Surya Phone	Erafone	Carin Cell
Surya Phone	132	.34	.34	.32
Erafone	178	.31	.38	.31
Carin Cell	215	.35	.28	0.37

Gambar 3.2 *Formulasi Model Matematis POM QM*

- f. Klik *Solve* pada *toolbar*, maka akan muncul kotak dialog *output* pada *toolbar window* yang akan keluar hasilnya seperti Gambar 3.3.

There are more results available in additional window option in the Main Menu.

Markov Analysis Results

Toko Handphone 2 step transition matr

	Surya Phone	Erafone	Carin Cell
Surya Phone	.333	.3344	.3326
Erafone	.3317	.3366	.3317
Carin Cell	.3353	.329	.3357
Ending number (given initial)	175.0881	174.7906	175.1213
Steady State probability	.3333	.3333	.3333

Gambar 3.3 *Markov analysis results*

- g. Diklik *Window* kemudian diklik *Multiplications* untuk memperoleh hasil seperti pada Gambar 3.4.

Multiplications

Toko Handphone Solution

	State 1	State 2	State 3
End of Period 1			
Surya Phone	.34	.34	.32
Erafone	.31	.38	.31
Carin Cell	.35	.28	.37
End Number (given initial)	175.31	172.72	176.97
End of Period 2			
Surya Phone	.333	.3344	.3326
Erafone	.3317	.3366	.3317
Carin Cell	.3353	.329	.3357
End Number (given initial)	175.0881	174.7906	175.1213

Gambar 3.4 *Multiplications*

- h. Selanjutnya untuk lebih memperjelas hasil *output* bisa ditambahkan dengan pilih *State Analysis*. *Outputnya State* tersebut akan muncul seperti pada Gambar 3.5.

Toko Handphone Solution		
State	Type	Class number
Surya Phone	Recurrent	1
Erafone	Recurrent	1
Carin Cell	Recurrent	1

Gambar 3.5 *State analysis*

- i. Setelah diperoleh hasil seperti *output* dari hasil formulasi dan grafik seperti pada penjelasan diatas, maka selanjutnya di lakukan analisis atau interprestasi hasil.

2. Software WinQSB

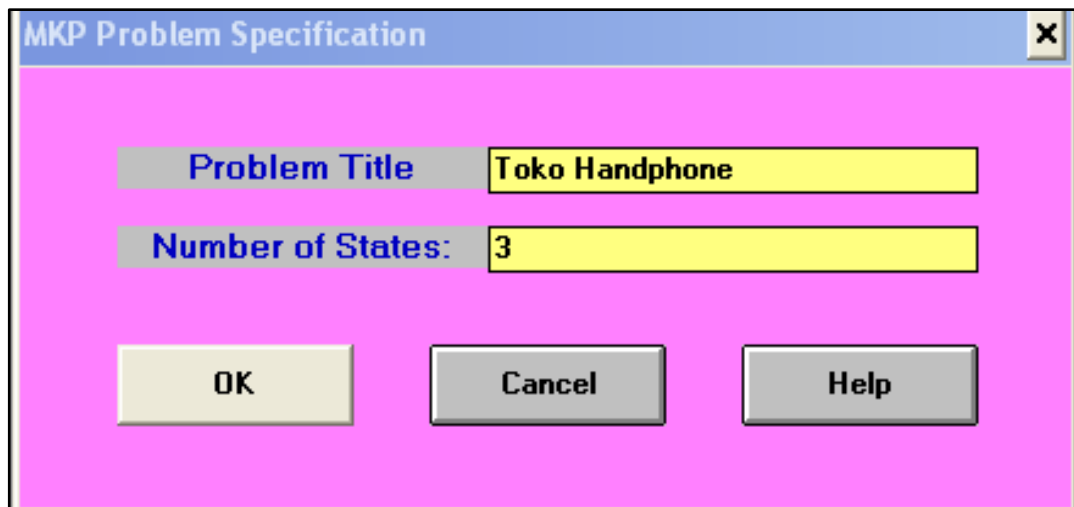
Berikut dijabarkan prosedur untuk menyelesaikan permasalahan sistem antrian dengan menggunakan *software WinQSB*.

- Klik start lalu membuka *WinQSB* dan pilih *Markov Process*
- Klik *File* lalu pilih *new problem*
- Input sesuai persoalan yang akan diselesaikan:

Problem Title : Toko Handphone

Number of States : 3

Tampilannya bisa dilihat pada Gambar 3.6 sebagai berikut.



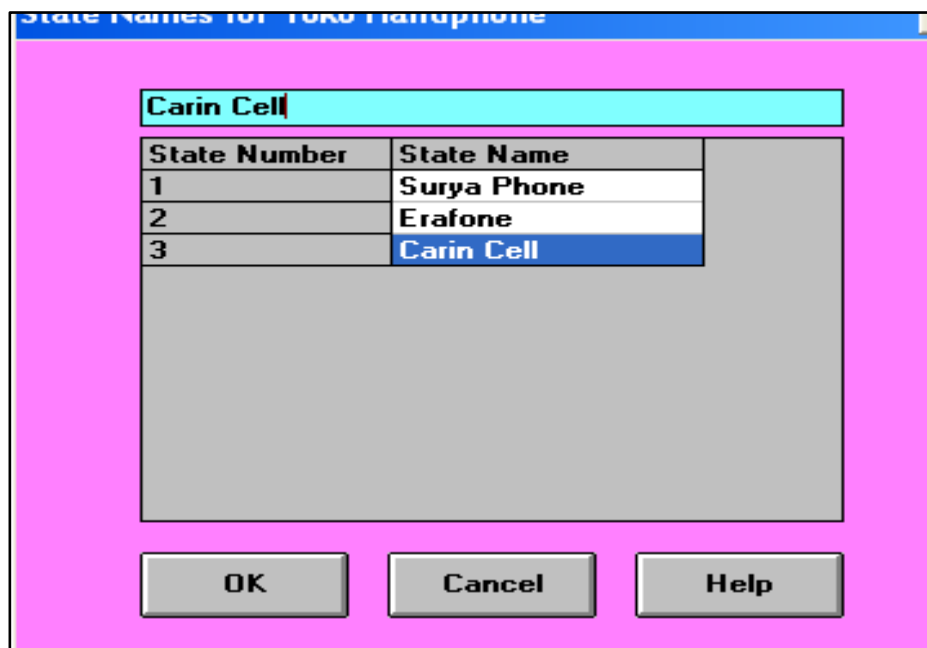
Gambar 3.6 MKP *Problem specification*

- d. Klik *Edit* lalu pilih *state name* untuk mengganti nama variable, kemudian ubah variable sesuai dengan yang ditentukan :

State 1 : Surya Phone

State 2 : Erafone

State 3 : Carin Cell



Gambar 3.7 *State names*

e. Masukkan data sesuai persoalan :

From \ To	Surya Phone	Erafone	Carin Cell
Surya Phone	0.34	0.34	0.32
Erafone	0.31	0.38	0.31
Carin Cell	0.35	0.28	0.37
Initial Prob.	132	178	215
State Cost	1200000	1500000	1000000

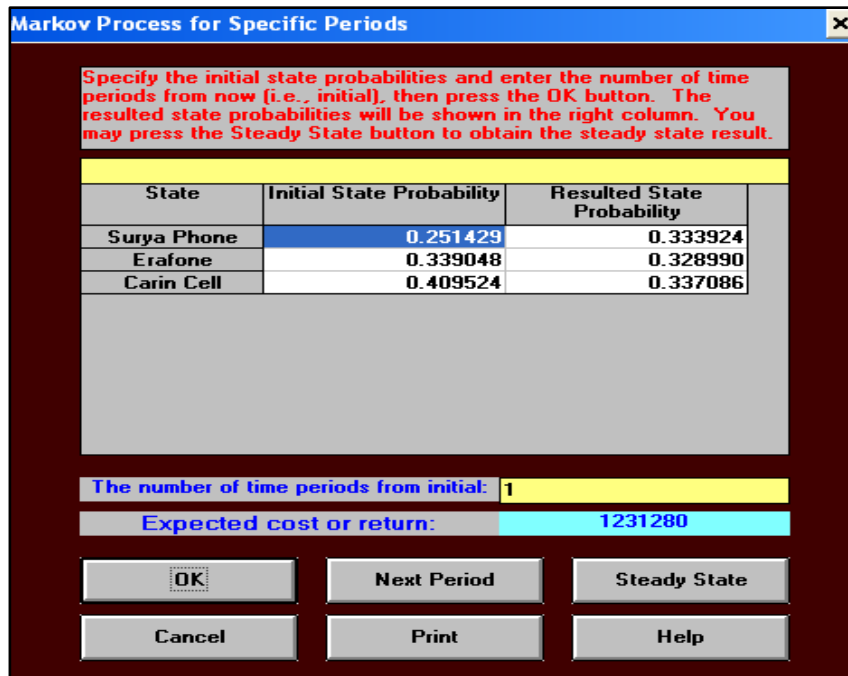
Gambar 3.8 *Input data*

f. Klik *solve and analyze* lalu pilih *Solve steady state*

11-15-2021	State Name	State Probability	Recurrence Time
1	Surya Phone	0.3333	3.0000
2	Erafone	0.3333	3
3	Carin Cell	0.3333	3
	Expected	Cost/Return =	1233333.0000

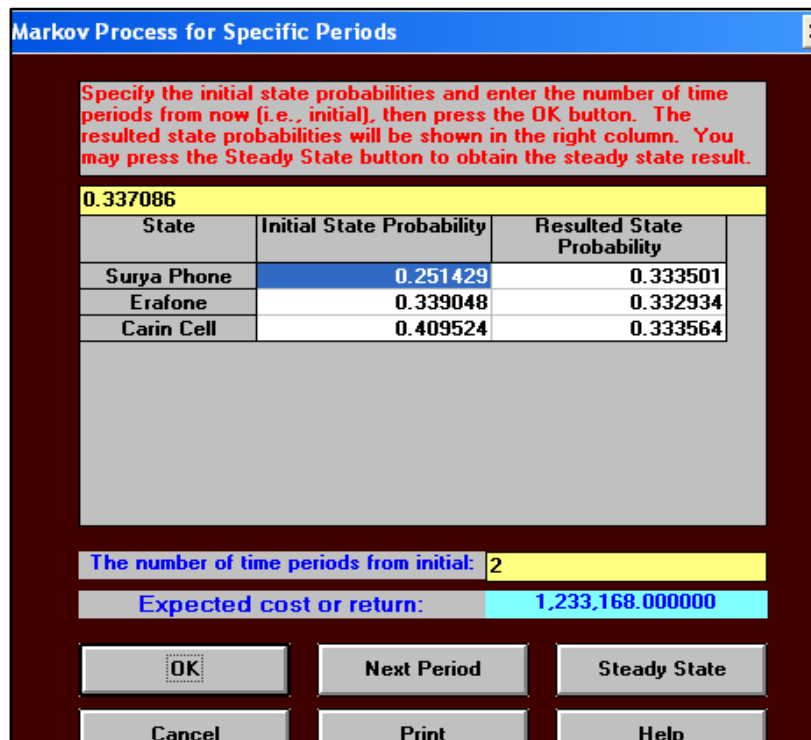
Gambar 3.9 *Steady state*

g. Untuk mengetahui probabilitas pada periode tertentu, klik *solve and analyze* lalu pilih *Markov Process Step*. Untuk mengetahui periode 1 bulan yang akan datang, ketik angka 1 pada *the number of time periods from initial* lalu klik OK.



Gambar 3.10 Markov process steps periode 1

- h. Untuk mengetahui periode 2 bulan yang akan datang, ketik angka 2 pada *the number of time periods from initial* lalu klik OK. Maka akan muncul *Resulted State Probability* dan *Expected Cost or Return*.



Gambar 3.11 Markov process steps periode 2

3.5 Analisis *Output* Rantai Markov

1. *Software POM-QM*

Berdasarkan langkah-langkah komputerisasi menggunakan *software* POM-QM *ForWindows*, dapat dilakukan analisa *output* penyelesaian permasalahan rantai markov (*markov chain*) sesuai studi kasus di atas. Analisis interpretasi hasil dari permasalahan warung makan berdasarkan *output* pada POM-QM *for windows* adalah sebagai berikut:

a. *Markov analysis results*

Output dari *Markov analysis result* menunjukkan angka yang berbeda pada 3 Toko *Handphone*, maka pada studi kasus ini belum bisa dikatakan seimbang. *Ending number* pada *output* ini menunjukkan banyak pelanggan pada tiap toko *handphone* pada periode dua bulan berikutnya. Pada Surya Phone terdapat $175,0881 \approx 176$ pelanggan, Erafone $174,7906 \approx 175$ pelanggan, dan Carin Cell $175,1213 \approx 176$ pelanggan. Untuk *steady state probability* yang didapatkan adalah dari keseluruhan pelanggan yaitu 525 pelanggan, diketahui sebanyak 0,3333 atau 33% pindah ke Surya Phone, 0,3333 atau 33% pindah ke Erafone dan 0,3333 atau 33% pindah ke Carin Cell.

b. *Multiplications*

Menunjukkan probabilitas pengunjung pada setiap periodenya dan menunjukkan jumlah pengunjung pada setiap *state* akhir periode. Pada bulan selanjutnya yaitu bulan ketiga, toko *handphone* surya phone sebanyak $175,31 \approx 176$ pelanggan, toko *handphone* erafone sebanyak $172,72 \approx 173$ pelanggan, dan toko *handphone* carin cell sebanyak $176,97 \approx 177$ pelanggan. Pada bulan selanjutnya yaitu bulan keempat, toko *handphone* surya phone sebanyak $175,0881 \approx 176$ pelanggan, toko *handphone* erafone sebanyak $174,7906 \approx 175$ pelanggan, dan toko *handphone* carin cell sebanyak $175,7906 \approx 176$ pelanggan.

c. *Output state analysis*

Hasil dari *Output state analysis* dapat ditinjau dari kolom *type* yang menunjukkan bahwa pada Surya Phone, Erafone dan Carin Cell adalah *recurrent* atau bersifat berulang. *Recurrent* menunjukkan bahwa tipe pada perhitungan yang selesai untuk ketiga toko

2. *Software WinQSB*

Penyelesaian masalah *markov chain* menggunakan *software* akan mengeluarkan *output* berupa hasil dari penyelesaian masalah. Analisis interpretasi hasil dengan menggunakan Win-QSB adalah sebagai berikut:

a. *State analyze*

Berdasarkan data yang dihasilkan *output* nilai *state analyze* untuk Suryaphone adalah sebesar 0,3333 atau 33% dengan kondisi pelanggan akan berpaling atau berpindah ke toko lain setelah 3 bulan. Untuk toko Erafone adalah sebesar 0,3333 atau 33% dengan kondisi pelanggan akan berpaling atau berpindah ke toko lain setelah 3 bulan. Untuk toko Carin Cell adalah sebesar 0,3333 atau 33% dengan kondisi pelanggan akan berpaling atau berpindah ke toko lain setelah 3 bulan..

b. *Markov process step*

Di dalam *Markov process step* menjelaskan probabilitas pendapatan yang diperoleh oleh objek selama periode mendatang dan juga menjelaskan *expected cost or Return*.

1) Periode 1

Probabilitas pendapatan untuk ke 3 toko pada satu bulan mendatang yaitu untuk surya phone 0,333924 atau 33,33%, erafone 0,328990 atau 32,89%, dan carin cell 0,337086 atau 33,7%. Dengan *expected cost or return* periode ketiga yaitu Rp 1.231.280, dan

2) Periode 2

Probabilitas pendapatan untuk ke 3 toko pada dua bulan mendatang yaitu untuk surya phone 0,333501 atau 33,35%, erafone 0,332934 atau 33,29%, dan carin cell 0,333564 atau 33,35%. Dengan *expected cost or return* periode ketiga yaitu Rp 1.233.168.

MODUL IV

TEORI PERMAINAN (*GAME THEORY*)

4.1 Tujuan Praktikum

Adapun tujuan dari praktikum mengenai Modul IV Teori Permainan antara lain sebagai berikut:

1. Praktikan mampu untuk menyelesaikan persoalan teori permainan dengan komputerisasi.
2. Praktikan mampu untuk menganalisis hasil (*output*) dari persoalan teori permainan.

4.2 Dasar Teori

Teori permainan (*game theory*) merupakan ilmu pengetahuan yang berhubungan dengan pembuatan keputusan ketika ada dua pihak atau lebih berada dalam kondisi persaingan atau konflik. Pihak-pihak yang bersaing ini disumsikan bersifat rasional dan cerdas, artinya masing-masing pihak akan melakukan strategi tindakan yang rasional untuk memenangkan persaingan itu, dan masing-masing pihak juga mengetahui strategi pihak lawannya, selanjutnya pihak ini disebut pemain. Menurut J Von Neumann dan O. Morgenstern (1944) dalam hal teori permainan, tidak dibiarkan pemain lain diam, tetapi melakukan aksi-reaksi atas apa yang terjadi dalam interaksi. Pemahaman tentang teori permainan:

1. *One-shot*, contohnya jika dua orang bertemu dan tahu bahwa mereka hanya akan sekali saja bertemu, terjadi kasus dimana mereka tidak menjaga reputasi mereka. Mereka berbuat buruk karena tahu bahwa tindakan mereka tidak akan menimbulkan tindakan lain di kemudian hari.
2. *Repeated, finite*; terjadi misalkan ketika orang tahu bahwa 10 hari lagi dia akan mati. Maka segala ekspektasi tersebut mengubah *behavior*-nya untuk berpikir mengenai hari akhir ke-10. Ini membuat orang menjadi *desperate* dan bertindak di luar dugaan atau disebut juga efek akhir.
3. *Repeated, infinite*: terjadi dimana seseorang menjaga reputasinya karena ia yakin akan berinteraksi dengan pihak lain *in the near future and more*.

4.2.1 Ketentuan Umum

Dapat dijelaskan beberapa ketentuan dasar yang terpenting dalam teori permainan yaitu:

1. Setiap pemain bermain rasional, dengan asumsi memiliki intelegensi yang sama, dan tujuan sama, yaitu memaksimumkan *payoff*, dengan kriteria *maximin* dan *minimax*.
2. Terdiri dari 2 pemain, keuntungan bagi salah satu pemain merupakan kerugian bagi pemain lain.
3. Tabel yang disusun menunjukkan keuntungan pemain baris, dan kerugian pemain kolom.
4. Permainan dikatakan adil jika hasil akhir menghasilkan nilai nol (0), tidak ada yang menang/kalah.
5. Tujuan dari teori permainan ini adalah mengidentifikasi strategi yang paling optimal

4.2.2 Strategi

Berikut beberapa strategi yang digunakan dalam *game theory* adalah sebagai berikut:

1. Strategi Murni

Penyelesaian dilakukan dengan menggunakan konsep *maximin* untuk pemain baris dan *minimax* untuk pemain kolom. *Maximin* adalah memaksimalkan keuntungan minimum sedangkan *Minimax* meminimalkan kerugian maksimum, dalam strategi ini pemain akan menggunakan satu strategi tunggal untuk mendapat hasil optimal *saddle point* yang sama

2. Strategi Campuran

Strategi ini dilakukan bila strategi murni belum memberi penyelesaian optimal. Sehingga perlu dilakukan tindak lanjut untuk mendapat titik optimal, dengan usaha mendapatkan *saddle point* yang sama.

4.3 Penyelesaian Teori Antrian

Berikut adalah cara penyelesaian masalah teori permainan dengan komputerisasi menggunakan software POM-QM *for windows*:

1. Strategi Murni

Dua toko roti yaitu Roti Boy dan Roti'o adalah 2 toko roti mentega yang cukup terkenal dikalangan masyarakat kota Samarinda. Keduanya selalu bersaing untuk mendapatkan hasil yang lebih optimal yaitu dengan mendapatkan keuntungan yang maksimum dan meminimalkan kerugian. Produk yang dijual pada kedua toko roti tersebut relatif sama. Kedua toko roti ini ingin menentukan strategi yang dapat menghasilkan keuntungan yang optimal. Hal yang dilakukan Roti Boy untuk menarik para konsumen untuk membeli produk mereka yaitu dengan menggunakan 3 strategi yaitu dengan

menerapkan Harga Normal, Harga Bundle dan Harga Diskon dan Roti'o juga menggunakan 3 strategi yang sama dalam menarik minat konsumen. Berikut strategi dari masing-masing toko roti dapat dilihat sebagai berikut:

Tabel 4.1 Data strategi murni toko roti

Roti Boy \ Roti'O	Harga Normal (S1)	Harga Bundle (S2)	Harga Diskon (S3)
Harga Normal(S1)	5	3	5
Harga Bundle (S2)	6	7	4
Harga Diskon (S3)	6	8	9

2. Strategi Campuran

Karena terjadi perubahan pasar dan tingkat penghasilan dari masyarakat, maka Roti Boy yang pada sebelumnya melakukan strategi dilihat dari harga yang diberikan kembali melakukan mencoba melakukan strategi ulang dengan menambahkan satu variabel strategi yakni Harga VIP yang disesuaikan dengan tingkat penghasilan masyarakat. Strategi ini dilihat dari persentase keuntungan yang telah diakumulasi dari berbagai faktor yang ada pada strategi sebelumnya yakni harga Normal, harga Bundle dan harga normal. Berikut ini merupakan tabel dari strategi dari masing-masing toko roti tersebut.

Tabel 4.2 Data strategi campuran toko roti

Roti Boy \ Roti'O	Harga Normal (S1)	Harga Bundle (S2)	Harga Diskon (S3)
Harga Normal(S1)	5	6	4
Harga Bundle (S2)	6	4	5
Harga Diskon (S3)	3	8	3
Harga VIP (S4)	7	9	4

4.4 Penyelesaian Masalah Teori Antrian

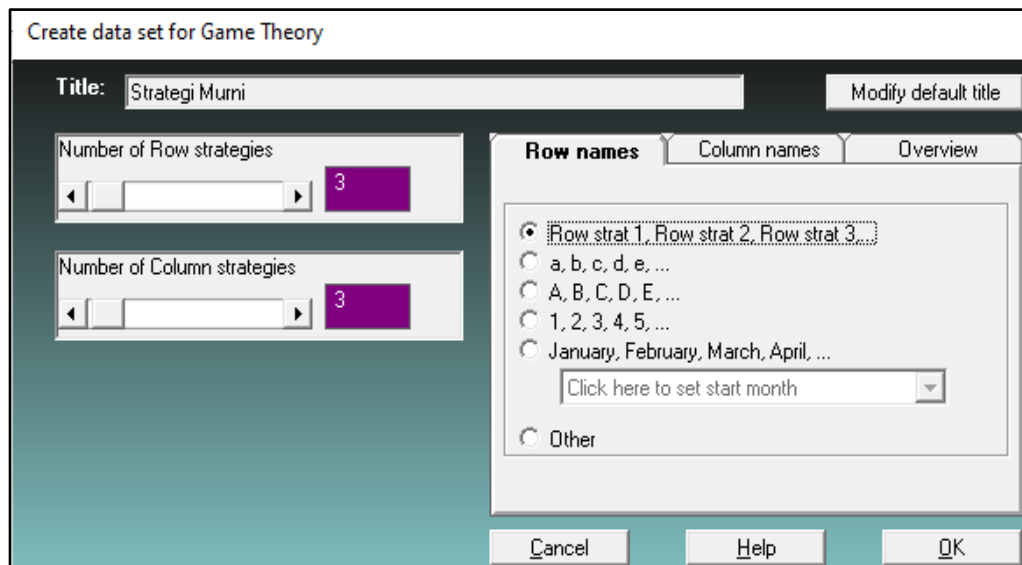
4.4.1 Software POM-QM

Pada penyelesaian masalah studi kasus Roti Boy dan Roti'o dengan menggunakan metode *game theory* terdapat 2 penyelesaian masalah, yaitu strategi murni dan strategi campuran. Berikut merupakan penyelesaian masalah strategi murni dan campuran dengan menggunakan *software* POM-QM:

1. Strategi Murni

Pada penyelesaian masalah studi kasus Roti Boy dan Roti'o dengan menggunakan metode *game theory* pada strategi murni, masalah tersebut dapat diselesaikan dengan menggunakan *software* POM-QM *For Windows* dengan tujuan mencari *saddle point* atau titik pelana untuk kedua toko roti tersebut yang dijabarkan sebagai berikut:

- Membuka *software* POM-QM *For Windows* dengan cara klik pada *icon* POM-QM *For Windows*,
- Kemudian, setelah *software* terbuka, pilih pada bagian *Module* kemudian klik *Game Theory*,
- Setelah itu, akan muncul layar kosong, kemudian klik *File*, lalu klik *New*,
- Selanjutnya, akan muncul kotak dialog *Create data set for Game Theory*, kemudian pada bagian kotak dialog *title* masukkan judul “Strategi Murni”,
- Kemudian, pilih angka 3 pada kolom *Number of Row strategies* dan pada kolom *Number of Column strategies* dan klik OK, untuk lebih jelasnya dapat dilihat seperti pada Gambar 4.1 berikut,



Gambar 4.1 *Create data set for game theory*

- Selanjutnya, klik OK, maka akan muncul *Data Table* dan isi tabel tersebut dengan data-data strategi pada *Column strat 1, 2, 3* dan pada *Row strat 1, 2 dan 3* seperti yang terlihat pada Gambar 4.2,

POM for Windows - [Data Table]

File Edit View Module Format Tools Window Help

Instruction
Enter the name for this column strategy. Almost any character is permissible.

Strategi Murni

	Normal	Bundle	Diskon
Normal	5	3	5
Bundle	6	7	4
Diskon	6	8	9

Gambar 4.2 Data table

- g. Selanjutnya, klik *Solve* pada *toolbar*, maka akan diperoleh kotak dialog *output* seperti pada Gambar 4.3,

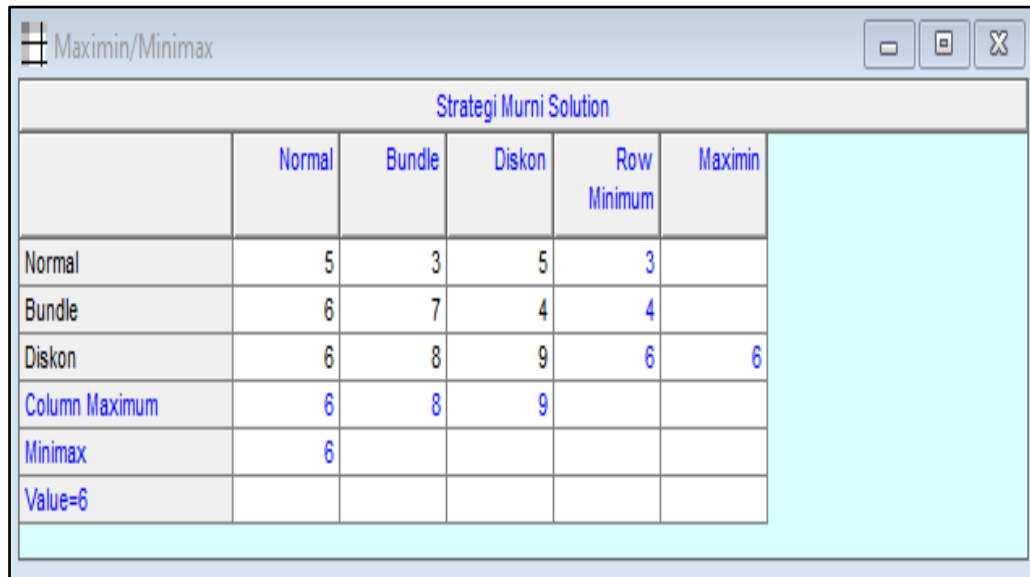
Game Theory Results

Strategi Murni Solution

	Normal	Bundle	Diskon	Row Mix
Normal	5	3	5	0
Bundle	6	7	4	.6
Diskon	6	8	9	.4
Column Mix-->	1	0	0	
Value of game (to row)	6			

Gambar 4.3 Kotak dialog output game theory results

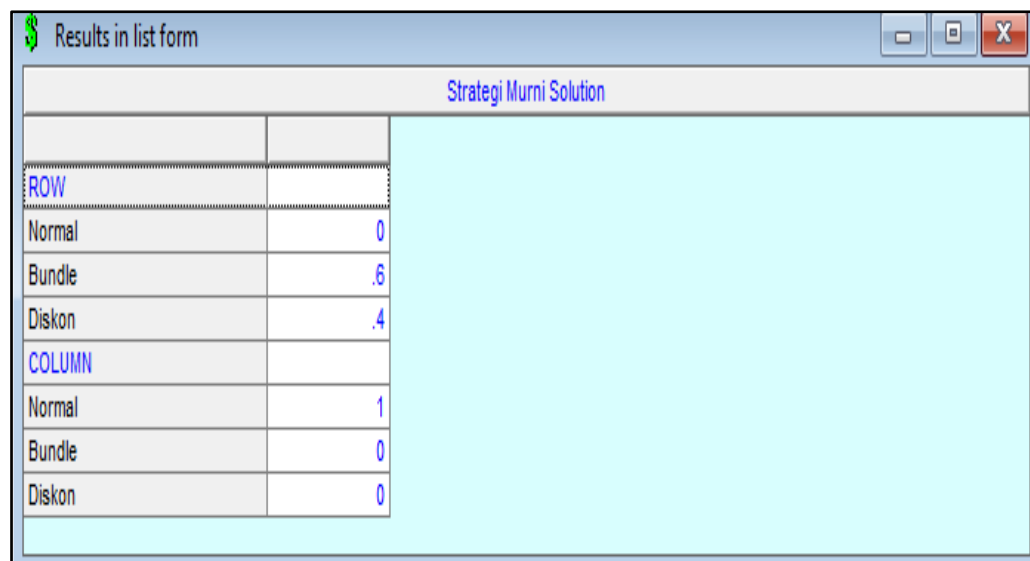
- h. Kemudian, klik *window* pada *toolbar*, kemudian pilih *maximin* dan *minimax* pada *windows* kemudian akan muncul layar seperti pada Gambar 4.4,



Strategi Murni Solution					
	Normal	Bundle	Diskon	Row Minimum	Maximin
Normal	5	3	5	3	
Bundle	6	7	4	4	
Diskon	6	8	9	6	6
Column Maximum	6	8	9		
Minimax	6				
Value=6					

Gambar 4.4 Kotak dialog *output maximin/minimax*

- i. Kemudian, klik *window* pada *toolbar*, kemudian pilih *results in list form* pada *windows*, maka layar akan terlihat seperti pada Gambar 4.5, dan



Strategi Murni Solution	
ROW	
Normal	0
Bundle	.6
Diskon	.4
COLUMN	
Normal	1
Bundle	0
Diskon	0

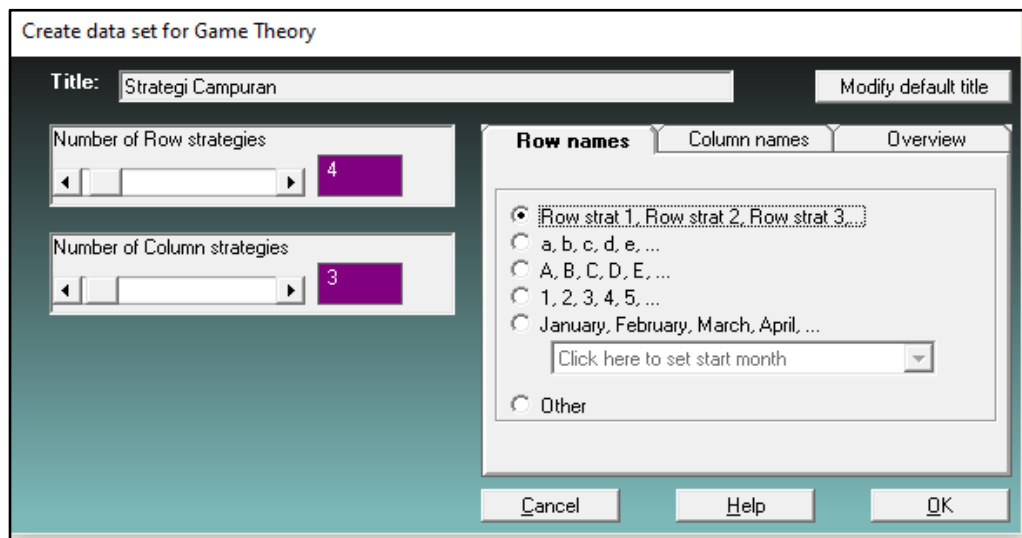
Gambar 4.5 Kotak dialog *output results in list form*

- j. Kemudian, langkah selanjutnya dilakukan analisis atau interpretasi hasil terhadap masalah yang telah diselesaikan.

2. Strategi Campuran

Pada penyelesaian masalah studi kasus Roti Boy dan Roti'o dengan menggunakan metode *game theory* pada strategi campuran, masalah tersebut dapat diselesaikan dengan menggunakan *software POM-QM For Windows* dengan tujuan mencari *saddle point* atau titik pelana untuk kedua toko roti tersebut yang dijabarkan sebagai berikut:

- Membuka *software POM-QM For Windows* dengan cara klik pada *icon POM-QM For Windows*,
- Kemudian, setelah *software* terbuka, pilih pada bagian *Module* kemudian klik *Game Theory*,
- Setelah itu, akan muncul layar kosong, kemudian klik *File*, lalu klik *New*,
- Selanjutnya, akan muncul kotak dialog *Create data set for Game Theory*, kemudian pada bagian kotak dialog *title* masukkan judul "Strategi Campuran",
- Kemudian, pilih angka 4 pada kolom *Number of Row strategies* dan pada kolom *Number of Column strategies* dan klik OK, untuk lebih jelasnya dapat dilihat seperti pada Gambar 4.6 berikut,



Gambar 4.6 *Create data set for game theory*

- Selanjutnya, klik OK, maka akan muncul *Data Table* dan isi tabel tersebut dengan data-data strategi pada *Column strat 1, 2, 3* dan pada *Row strat 1, 2 dan 3* seperti yang terlihat pada Gambar 4.7,

The screenshot shows the POM for Windows interface with a data table titled "Strategi Campuran". The table has four columns: "Normal", "Bundle", "Diskon", and an unlabeled column. The rows are labeled "Normal", "Bundle", "Diskon", and "VIP".

	Normal	Bundle	Diskon	
Normal	5	6	4	
Bundle	6	4	5	
Diskon	3	8	3	
VIP	7	9	4	

Gambar 4.7 Data table

- g. Selanjutnya, klik *Solve* pada *toolbar*, maka akan diperoleh kotak dialog *output* seperti pada Gambar 4.8,

The screenshot shows the "Game Theory Results" dialog box in POM for Windows. It displays the "Strategi Campuran Solution" table, which includes the original data table plus a "Row Mix" column and a "Value of game (to row)" row.

	Normal	Bundle	Diskon	Row Mix
Normal	5	6	4	0
Bundle	6	4	5	.83
Diskon	3	8	3	0
VIP	7	9	4	.17
Column Mix-->	0	.17	.83	
Value of game (to row)	4.83			

Gambar 4.8 Kotak dialog *output game theory results*

- h. Kemudian, klik *window* pada *toolbar*, kemudian pilih *maximin* dan *minimax* pada *windows* kemudian akan muncul layar seperti pada Gambar 4.9,

Strategi Campuran Solution					
	Normal	Bundle	Diskon	Row Minimum	Maximin
Normal	5	6	4	4	4
Bundle	6	4	5	4	4
Diskon	3	8	3	3	
VIP	7	9	4	4	4
Column Maximum	7	9	5		
Minimax			5		
4 ≤ value ≤ 5					

Gambar 4.9 Kotak dialog *output maximin/minimax*

- i. Kemudian, klik *window* pada *toolbar*, kemudian pilih *results in list form* pada *windows*, maka layar akan terlihat seperti pada Gambar 4.10, dan

Strategi Campuran Solution	
ROW	
Normal	0
Bundle	.83
Diskon	0
VIP	.17
COLUMN	
Normal	0
Bundle	.17
Diskon	.83

Gambar 4.10 Kotak dialog *output results in list form*

- j. Kemudian, langkah selanjutnya dilakukan analisis atau interpretasi hasil terhadap masalah yang telah diselesaikan.

4.5 Analisis *Output* Teori Antrian

Analisis *output* berdasarkan penyelesaian masalah teori permainan (*Game Theory*) dengan menggunakan *software* POM-QM *For Windows* akan menghasilkan *output* berupa hasil dari penyelesaian masalah. Berikut ini adalah analisis interpretasi hasil atau *output* dari permasalahan strategi murni dan permasalahan strategi campuran pada teori permainan dengan menggunakan *software* POM-QM *For Windows*.

1. Strategi Murni

Berikut adalah analisis interpretasi hasil dari permasalahan strategi murni dari Roti Boy dan Roti'o untuk mencari *saddle point* berdasarkan *output* yang diperoleh dari *software* POM-QM *For Windows*.

a. *Game theory Result*

Berdasarkan *output* pada *Game Theory result* didapatkan hasil untuk nilai *pay off* sebesar 6. Hal ini dapat diartikan bila Rotiboy menginginkan keuntungan lebih besar, namun hanya akan mendapat maksimal 6 dan pada Roti'o meskipun menginginkan kerugian sekecil mungkin, tetapi tetap akan mengalami kerugian maksimum sebesar 6.

b. *Maximin dan Minimax*

Berdasarkan tabel *maximin* dan *minimax*, didapatkan nilai *saddle point* atau titik pelana sebesar 6. Hal ini dapat diartikan bahwa toko Rotiboy mendapatkan keuntungan maksimum sebesar 6 yang dapat diambil dari nilai terkecil dari nilai maksimal pada harga diskon, dan Roti'o mendapatkan kerugian minimum 6 diambil dari nilai terbesar dari nilai minimum pada harga normal.

c. *Results in list form*

Berdasarkan *result in list form*, maka didapatkan persentase dari tiap strategi yang ada di tiap toko roti, untuk Rotiboy pada strategi harga normal sebesar 0%, strategi harga bundle sebesar 60%, dan strategi harga diskon sebesar 40%. Pada Roti'o untuk strategi harga normal sebesar 100%, strategi harga bundle sebesar 0%, dan strategi harga diskon sebesar 0%.

2. Strategi Campuran

Berikut merupakan analisis strategi campuran pada permasalahan toko roti menggunakan *software* POM-QM *For Windows*, maka diperoleh hasil *saddle point* atau titik pelana pada permasalahan kedua toko roti beserta analisis dari *output* yang diperoleh sebagai berikut:

a. *Game theory Result*

Berdasarkan *output* pada *Game Theory result* didapatkan hasil untuk nilai *pay off* sebesar 4,83. Hal ini dapat diartikan bila Rotiboy menginginkan keuntungan lebih besar, namun hanya akan mendapat maksimal 4,83 dan pada Roti'o meskipun menginginkan kerugian sekecil mungkin, tetapi tetap akan mengalami kerugian maksimum sebesar 4,83.

b. *Maximin dan Minimax*

Berdasarkan tabel pada *output maximin* dan *minimax* diperoleh nilai *maximin* sebesar 4 dan nilai *minimax* sebesar 5. Sehingga value yang didapatkan adalah nilai antara 4 dan 5 dengan menggunakan strategi harga normal, bundle, VIP pada *maximin* dan harga diskon pada *minimax*.

c. *Results in list form*

Berdasarkan hasil dari *result in list form* maka didapatkan persentase dari tiap strategi yang ada, untuk Rotiboy pada strategi harga normal sebesar 0%, harga bundle sebesar 83%, harga diskon sebesar 0% dan pada strategi harga VIP sebesar 17%. Roti'o pada strategi harga normal sebesar 0%, strategi harga bundle sebesar 17%, dan strategi harga diskon sebesar 83%.

MODUL V *DYNAMIC PROGRAMMING*

5.1 Tujuan Praktikum

Adapun tujuan dari praktikum ini adalah:

1. Praktikan dapat mengetahui jarak terdekat yang akan digunakan dalam pendistribusian.
2. Praktikan dapat mencari solusi optimal dalam suatu persoalan menggunakan Metode Pemrograman Dinamis.

5.2 Dasar Teori

1. Definisi Pemrograman Dinamis

Program Dinamis (*dynamic programming*) adalah metode pemecahan masalah dengan cara menguraikan solusi menjadi sekumpulan langkah (*step*) atau tahapan (*stage*) sedemikian sehingga solusi dari persoalan dapat dipandang dari serangkaian keputusan yang saling berkaitan (Nurhayati, 2010).

Pada penyelesaian persoalan dengan metode ini:

- a. Terdapat sejumlah berhingga pilihan yang mungkin,
 - b. Solusi pada setiap tahap dibangun dari hasil solusi tahap sebelumnya,
 - c. Menggunakan persyaratan optimasi dan,
 - d. Kendala untuk membatasi sejumlah pilihan yang harus dipertimbangkan pada suatu tahap.
- #### 2. Bentuk – Bentuk Pemrograman Dinamis
- Pemrograman dinamis dibedakan menjadi pemrograman dinamis masalah deterministik dan *probabilistic* (Puspitasari, Rita. 2014).
- a. Pemrograman dinamis *deterministic* dicirikan dimana keadaan pada tahap berikutnya ditentukan sepenuhnya oleh keadaan dan keputusan pada tahap sekarang. Masalah deterministik dapat dibedakan antara kasus maksimum dan minimum.
 - b. Pemrograman dinamis *probabalistic*, dimana keadaan berikutnya memiliki suatu distribusi probabilitas tertentu.

3. Istilah yang Digunakan dalam Pemrograman Dinamis

Istilah yang biasa digunakan antara lain (Puspitasari, Rita. 2014):

- a. *Stage* (tahap) adalah bagian persoalan yang mengandung *decision variable*.
- b. *Alternatif*, pada setiap *stage* terdapat *decision variable* dan fungsi tujuan yang menentukan besarnya nilai setiap *alternative*.
- c. *State*, *state* menunjukkan kaitan satu *stage* dengan *stage* lainnya, sedemikian serupa sehingga setiap *stage* dapat dioptimisasikan secara terpisah sehingga hasil optimasi layak untuk seluruh persoalan.

4. Karakteristik Pemrograman Dinamis

Persoalan pada program dinamis dirancang untuk memberikan suatu interpretasi secara fisik dari struktur yang abstrak dari persoalan program dinamis (Sutrisnoadityo. 2013). Salah satu cara untuk mengelal situasi yang dapat diformulasikan sebagai persoalan program dinamis ini ialah dengan memperhatikan karakteristik yang ada pada program dinamis. Berikut ini merupakan gambaran dasar atau karakteristik program dinamis:

- a. Persoalan dapat dibagi menjadi beberapa tahap (*stage*), yang pada masing-masing stage diperlukan adanya satu keputusan.
- b. Masing-masing tahap terdiri dari sejumlah status (*state*) yang berhubungan dengan tahap tersebut. Secara umum, status merupakan bermacam kemungkinan masukan yang ada pada tahap tersebut.
- c. Hasil dari keputusan yang diambil pada setiap tahap ditransformasikan dari status yang bersangkutan ke status berikutnya pada tahap berikutnya.
- d. Ongkos (*cost*) pada suatu tahap meningkat secara teratur (*steadily*) dengan bertambahnya jumlah tahapan.
- e. Ongkos pada suatu tahap bergantung pada ongkos tahap-tahap yang sudah berjalan dan ongkos pada tahap tersebut.
- f. Keputusan terbaik pada suatu tahap bersifat independen terhadap keputusan yang dilakukan pada tahap sebelumnya.
- g. Adanya hubungan rekursif yang mengidentifikasi keputusan terbaik untuk setiap status pada tahap k memberikan keputusan terbaik untuk setiap status pada tahap $k + 1$. Prinsip optimalitas berlaku pada persoalan tersebut.

5.3 Penyelesaian *Dynamic Programming*

Berikut adalah cara penyelesaian masalah *dynamic programming* dengan komputerisasi:

Contoh kasus:

Sebuah usaha travel dan pengantaran barang akan mengirimkan barang dari kota Wonogiri (1) ke kota Jakarta (stage 10). Untuk mengirimkan barang ke kota Jakarta terdapat banyak rute perjalanan yang bisa diambil/dipilih. Buatlah rute penjualan travel tersebut untuk menentukan rute perjalanan dengan jarak yang optimal untuk sampai ketujuan dengan keterangan jarak antar wilayah yang dilalui sebagai berikut (satuan kilometer);

Table 5.1 Data Jarak

Dari	Ke	Jarak (KM)
1	3	12
1	2	13
2	4	8
2	5	10
3	4	13
3	5	9
3	6	8
4	7	9
4	8	8
5	7	11
5	8	9
5	9	10
6	8	13
6	9	9
7	10	13
8	10	12
9	10	13

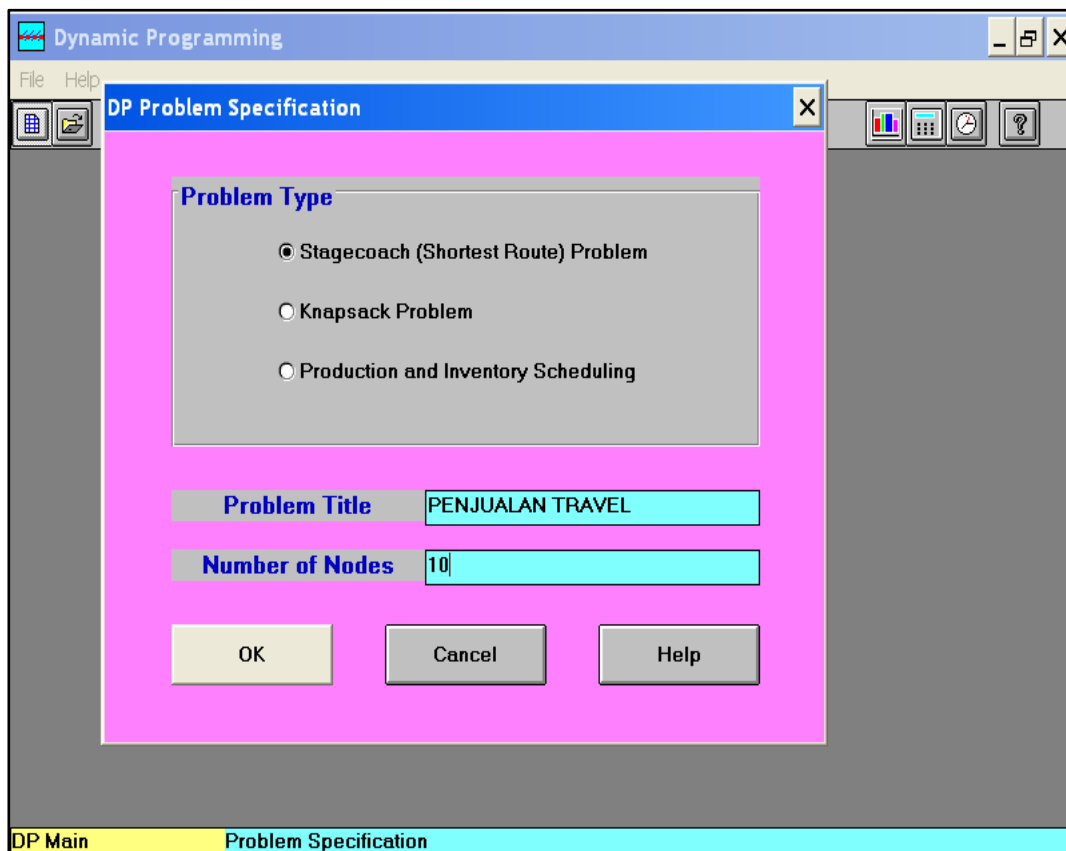
Sumber: Data Praktikum PO II, 2020

Buatlah node dari transportasinya, berapa jarak optimal rute pengiriman dari barang dari travel tersebut

5.4 Penyelesaian Masalah *Dynamic Programming*

Berikut tahapan penyelesaian persoalan pemrograman dinamis/*dynamic programming* dengan menggunakan *software* WinQSB

1. Membuka *ORACLE VM Virtual Box*
2. Selanjutnya setelah windows XP terbuka, klik Start lalu pilih All programs dan pada folder WinQSB pilih modul *dynamic programming*
3. Pada saat tampilan dari modul telah terbuka, isi data seperti dibawah ini.



Gambar 5.1 *Problem Specification*

4. Kemudian, isi data jarak yang ada pada soal. Dari node 1 hingga node ke 10 sesuai dengan data yang telah ada.

Dynamic Programming
 File Edit Format Solve and Analyze Results Utilities Window WinQSB Help

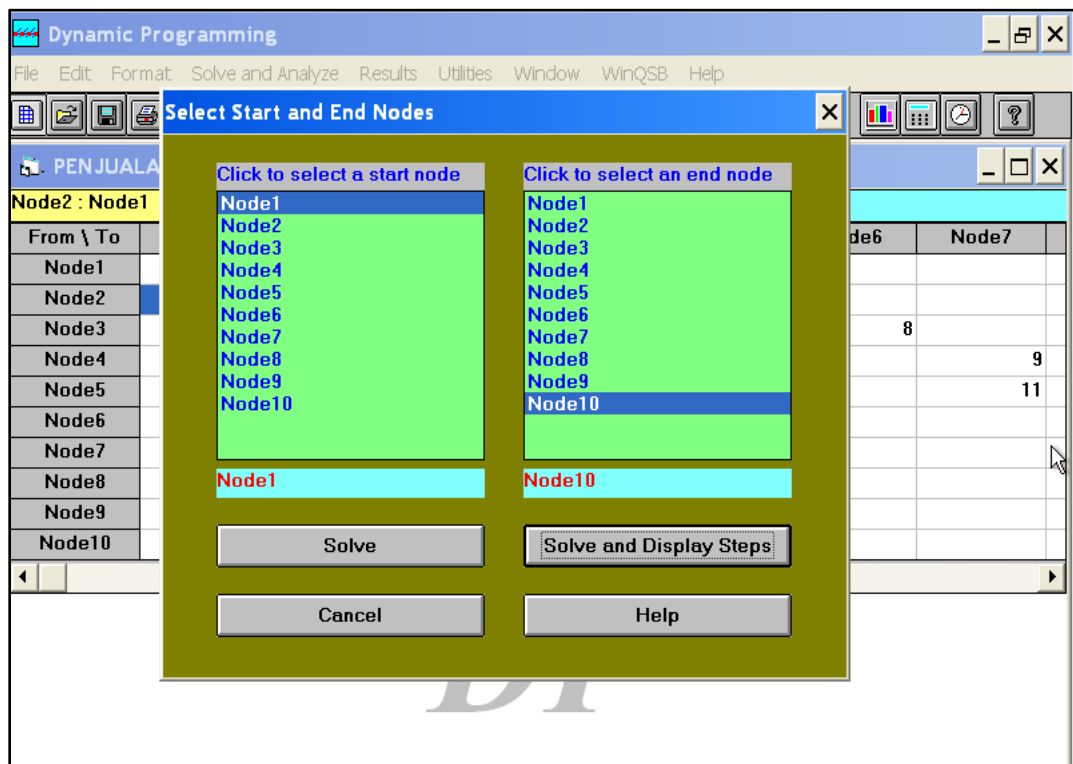
PENJUALAN TRAVEL: Stagecoach-Shortest Route Problem

Node10 : Node10

From \ To	Node4	Node5	Node6	Node7	Node8	Node9	Node10
Node1							
Node2	8	10					
Node3	13	9	8				
Node4				9	9		
Node5				11	9	10	
Node6					13	9	
Node7							13
Node8							12
Node9							13
Node10							

Gambar 5.2 *Dynamic Programming*

- Selanjutnya, klik *solve and analyze* lalu pilih *solve and display steps* dan isi data *start* dan *end node* kemudian pilih *solve and display steps*.



Gambar 5.3 *Start and End Nodes*

6. Berikut hasil dari *solution steps*.

The screenshot shows a software window titled 'Dynamic Programming' with a menu bar (File, Format, Results, Utilities, Window, Help) and a toolbar. The main content area displays a table of solution steps for a stagecoach shortest route problem. The table includes columns for Stage, From Input State, To Output State, Distance, Distance to Node10, and Status. The steps are numbered 1 through 9, with the final summary row indicating a minimum distance of 42 from Node1 to Node10.

10-02-2022 06:10:43	Stage	From Input State	To Output State	Distance	Distance to Node10	Status	
1	1	Node1	Node2	13	42	Optimal	
2	2	Node2	Node4	8	29	Optimal	
3	2	Node3	Node5	9	30		
4	3	Node4	Node8	9	21	Optimal	
5	3	Node5	Node8	9	21		
6	3	Node6	Node9	9	22		
7	4	Node7	Node10	13	13		
8	4	Node8	Node10	12	12	Optimal	
9	4	Node9	Node10	13	13		
		From Node1	To Node10	Minimum	Distance =	42	CPU = 0

Gambar 5.4 *Solution Steps*

7. Selanjutnya pilih *results* lalu klik show solution summary untuk melihat hasil penyelesaian lainnya.

The screenshot shows the same software window, but now displaying the 'Solution for PENJUALAN TRAVEL: Stagecoach-Shortest Route Problem'. The table shows the cumulative distance for each stage and the final minimum distance of 42. The 'Results' tab is active at the bottom, and the system tray shows the time as 6:11 AM.

10-02-2022 Stage	From Input State	To Output State	Distance	Cumulative Distance	Distance to Node10	
1	Node1	Node2	13	13	42	
2	Node2	Node4	8	21	29	
3	Node4	Node8	9	30	21	
4	Node8	Node10	12	42	12	
		From Node1	To Node10	Min. Distance	= 42	CPU = 0

Gambar 5.5 *Shortest Route Problem*

5.5 Analisis *Output Dynamic Programming*

Berdasarkan hasil penyelesaian persoalan pemrograman dinamis dengan *software* WinQSB, didapatkan beberapa hasil analisis sebagai berikut.

1. *Distance to Node 10* dari masing-masing *stage*

Pada *stage* 1 terdapat 1 jalur yaitu node 1 ke node 2 dengan jarak 13 km, dan jarak ke node 10 sebesar 42 km. Pada *stage* 2 terdapat 2 jalur yakni node 2 ke node 4 dengan jarak 8 km dan jarak ke node 10 sebesar 29 km, serta node 3 ke node 5 dengan jarak 9 km, dan jarak ke node 10 sebesar 30 km. Pada *stage* 3 terdapat 3 jalur yakni node 4 ke node 8 dengan jarak 9 km, dan jarak node 10 sebesar 21 km, node 5 ke node 8 dengan jarak 9 km dan jarak ke node 10 sebesar 21 km, node 6 ke node 9 dengan jarak 9 km dan jarak ke node 10 sebesar 22 km. Pada *stage* 4 terdapat 3 jalur yakni node 7 ke node 10 dengan jarak 13 km dan jarak ke node 10 sebesar 13 km, node 8 ke no 10 dengan jarak 12 km, dan jarak ke node 10 sebesar 12 km, dan node 9 ke node 10 dengan jarak 13 km dan jarak ke node 10 sebesar 13 km.

2. *Solution steps*

Berdasarkan hasil penyelesaian didapatkan beberapa rute perjalanan penjualan travel yang optimal untuk dilalui yakni pada *stage* 1 ialah node 1 ke node 2 dengan jarak 13 km dan jarak ke node 10 sebesar 42 km, pada *stage* 2 node 2 ke node 4 dengan jarak 8 km dan jarak ke node 10 sebesar 29 km, pada *stage* 3 node 4 ke node 8 dengan jarak 9 km dan jarak ke node 10 sebesar 21 km, dan pada *stage* 4 node 8 ke node 10 dengan jarak 12 km serta jarak ke node 10 sebesar 12 km. Pada persoalan ini didapatkan minimum distance dari node 1 ke node 10 sebesar 42 km.

