STRATEGI PEMASARAN
McDONALD'S, KFC, DAN RICHEESE FACTORY
MENGGUNAKAN TEORI PERMAINAN
(Studi Kasus pada Mahasiswa Universitas Brawijaya)

SKRIPSI

oleh:

NASHRUL ANAS A.N

145090407111004

JURUSAN MATEMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS BRAWIJAYA

MALANG

2019

STRATEGI PEMASARAN
McDONALD'S, KFC, DAN RICHEESE FACTORY
MENGGUNAKAN TEORI PERMAINAN
(Studi Kasus pada Mahasiswa Universitas Brawijaya)

Sebagai salah satu syarat untuk memperoleh gelar Sarjana Matematika

SKRIPSI

oleh:
NASHRUL ANAS A.N
145090407111004

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2019

LEMBAR PENGESAHAN SKRIPSI

STRATEGI PEMASARAN
McDONALD'S, KFC, DAN RICHEESE FACTORY
MENGGUNAKAN TEORI PERMAINAN
(Studi Kasus pada Mahasiswa Universitas Brawijaya)

oleh:
NASHRUL ANAS A.N
145090407111004

Setelah dipertahankan di depan Majelis Penguji
pada tanggal 03 Juli 2019
dan dinyatakan memenuhi syarat untuk memperoleh gelar
Sarjana Matematika

Pembimbing

Dr. Sobri Abusini, M.T.
NIP. 196012071988021001

Mengetahui,
Ketua Jurusan Matematika
Fakultas MIPA Universitas Brawijaya

Ratno Bagus Edy Wibowo, S.Si.,M.Si, Ph.D.
NIP. 197509082000031003

LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama	**:**	**Nashrul Anas A.N**
NIM	**:**	**145090407111004**
Penulis Skripsi berjudul	**:**	**Strategi Pemasaran McDonald's, KFC, dan Richeese Factory Menggunakan Teori Permainan (Studi Kasus Pada Mahasiswa Universitas Brawijaya)**

dengan ini menyatakan bahwa:

1. **isi Skripsi yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain, nama-nama yang termasuk di isi dan tertulis di Daftar Pustaka dalam Skripsi ini hanya sebagai refrensi.**
2. **Apabila di kemudian hari ternyata Skripsi yang saya tulis terbukti hasil jiplakan, maka saya bersedia menanggung segala risiko yang akan saya terima. Demikian pernyataan ini dibuat dengan penuh kesadaran.**

Malang, 03 Juli 2019
yang menyatakan,

Nashrul Anas A.N
NIM 145090407111004

STRATEGI PEMASARAN
McDONALD’S, KFC, DAN RICHEESE FACTORY
MENGGUNAKAN TEORI PERMAINAN
(Studi Kasus pada Mahasiswa Universitas Brawijaya)

ABSTRAK

Masyarakat saat ini menyukai makanan cepat saji atau dikenal dengan istilah *fast food*. Tujuan penelitian ini untuk mengetahui *optimalisasi* strategi pemasaran yang optimal dari restoran *fast food.* Beberapa restoran *fast food* yaitu KFC, McDonald’s, dan Richeese Factory. Setiap restoran *fast food* memiliki keunggulan yang dapat dijadikan strategi pemasaran seperti harga, lokasi, promo, variasi produk, dan pelayanan. Pengoptimalan strategi pemasaran dapat diselesaikan dengan aplikasi teori permainan yaitu metode simpleks dan algoritma *brown* dengan menggunakan software matlab. Berdasarkan pengolahan data dengan software matlab didapat hasil yang paling diminati responden adalah perbandingan restoran *fast food* antara McDonald’s dan KFC. Setelah melakukan perhitungan dengan menggunakan metode simpleks dan algoritma *brown* dari matriks *payoff* McDonald’s didapat strategi optimal untuk McDonald’s adalah strategi harga sebesar 69,291 %. Strategi optimal untuk KFC adalah strategi harga 65,748 %. Sementara itu dari matriks *payoff* KFC didapat strategi opetimal untuk McDonald’s adalah strategi lokasi sebesar 53 %. Strategi optimal untuk KFC adalah strategi pelayanan sebesar 89,93 %. Nilai permainan yang didapat menggunakan metode simpleks berada pada interval algoritma *brown.*

Kata Kunci: Teori Permainan, simpleks, algoritma brown, fast food, strategi optimal.

MARKETING STRATEGY
McDONALD'S, KFC AND RICHEESE FACTORY
USING THE GAME THEORY
(Case Study of Brawijaya University Students)

ABSTRACT

People today love fast food or known as fast food. The purpose of this study is to determine the optimal optimization of marketing strategies from fast food restaurants. Some fast food restaurants are KFC, McDonald's, and Richeese Factory. Every fast food restaurant has advantages that can be used as marketing strategies such as price, location, promos, product variety, and service. Optimization of marketing strategies can be solved by the application of game theory, namely the simplex method and brown algorithm using the matlab software. Based on the processing of data with the Matlab software, the most desirable results of respondents were the comparison of fast food restaurants between McDonald 's and KFC. After calculating using the simplex and algorithmic brown method from the McDonald's payoff matrix, the optimal strategy for McDonald's is to obtain a pricing strategy of 69.291%. The optimal strategy for KFC is the price strategy of 65.748%. Meanwhile from the KFC payoff matrix, the optimal strategy for McDonald's is a location strategy of 53%. The optimal strategy for KFC is a service strategy of 89.93%. The game value obtained using the simplex method is at the brown algorithm interval.

Keywords: *Game theory, simplex, brown algorithm, fast food, optimal strategy*

KATA PENGANTAR

Puji syukur ke hadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul **STRATEGI PEMASARAN McDONALD'S, KFC DAN RICHEESE FACTORY MENGGUNAKAN TEORI PERMAINAN** dengan baik dan lancar. Shalawat serta salam selalu tercurahkan kepada Nabi Muhammad SAW sebagai suri teladan bagi penulis.

Skripsi ini tidak dapat diselesaikan dengan baik tanpa bantuan, bimbingan, dan motivasi dari berbagai pihak. Oleh karena itu, penulis menyampaikan terimakasih kepada:

1. Dr. Sobri Abusini, M.T. selaku dosen pembimbing atas segala bimbingan, motivasi, bantuan, saran, dan kesabaran yang telah diberikan selama penulisan skripsi ini.
2. Prof.Dr. Agus Widodo, M.Kes dan Kwardiniya Andawaningtyas, S.Si.,M.Si selaku dosen penguji atas segala kritik dan saran yang diberikan untuk perbaikan skripsi ini.
3. Prof.Dr. Marjono, M.Phil selaku dosen pembimbing akademik yang selalu memberikan motivasi dan bimbingan selama penulis menempuh kuliah.
4. Dr. Isnani Darti, S.Si., M.Si selaku Ketua Program Studi Matematika yang selalu memberikan motivasi kepada penulis untuk menyelesaikan skripsi ini tepat waktu.
5. Ratno Bagus Edy Wibowo, S.Si., M.Si., Ph. D. selaku Ketua Jurusan Matematika atas segala bantuan yang diberikan.
6. Segenap dosen dan staf Jurusan Matematika atas ilmu dan pengalaman yang sangat bermanfaat bagi penulis.
7. Orang tua atas segala doa, bantuan, dan dukungan yang tiada henti diberikan selama ini kepada penulis.

Semoga Allah SWT senantiasa memberikan rahmat dan anugrah-Nya kepada semua pihak yang telah membantu menyelesaikan skripsi ini. Penulis menyadari bahwa dalam penulisan skripsi ini masih terdapat kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang dapat disampaikan melalui email ke alamat nashrul.red@gmail.com. Semoga skripsi ini dapat bermanfaat bagi berbagai pihak, serta menjadi inspirasi bagi penulis skripsi selanjutnya.

Malang, 03 Juli 2019

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR LAMPIRAN

BAB I
PENDAHULUAN

1.1 Latar Belakang

Era globalisasi membawa kehidupan manusia ke dalam gerbang modernisasi yang membawa dampak dan perkembangan zaman dan teknologi yang pesat, sehingga menghasilkan *trend* atau gaya hidup baru. Perubahan gaya hidup masyarakat inilah yang mendasari perubahan pola makan. Sebagai contoh, gaya hidup masyarakat masa kini adalah senang mengkonsumsi makanan yang siap saji atau lebih memilih makanan instan yang dikenal dengan istilah *fast food.*

Bertram (1975) mendefinisikan *fast food* sebagai makanan yang dapat disiapkan dan dikonsumsi dalam waktu yang singkat. Menurut Irianto (2007), *fast food* memiliki beberapa kelebihan yaitu penyajiannya yang cepat sehingga tidak menghabiskan waktu yang lama dan dapat dihidangkan kapan pun dimana saja, higenis dan dianggap sebagai makanan bergengsi dan makanan gaul.

Beberapa restoran *fast food* yang menjadi tempat makan favorit masyarakat Indonesia adalah restoran yang menyajikan produk ayam goreng sebagai menu utamanya. Banyaknya bermunculan restoran *fast food* yang menawarkan produk utama berupa ayam goreng seperti KFC, McDonald's's dan Richeese Factory menjadi bukti bahwa restoran cepat saji disukai oleh masyarakat Indonesia. Namun, masuknya beragam merk-merk tersebut membuat persaingan semakin ketat di industri restoran *fast food* khususnya produk ayam goreng.

Dilihat dari persaingan antara McDonald's, KFC dan Richeese Factory tersebut akan memunculkan suatu strategi agar tidak kalah saing. Dalam ilmu matematika, salah satu cara untuk menganalisis suatu strategi adalah menggunakan teori permainan. Teori permainan adalah bagian dari ilmu matematika tentang analisis logis dari situasi konflik dan kerja sama (Straffin, 1993). Teori permainan merupakan suatu teori untuk menentukan strategi optimal dari beberapa pihak yang bersaing. Suatu permainan dikatakan optimal jika kedua pihak bersaing mencapai kesetimbangan. Didalam model teori permainan diklasifikasikan dengan sejumlah cara, salah satunya berdasarkan jumlah keuntungan dan kerugian. Jika jumlah keuntungan dan kerugian adalah nol maka disebut permainan berjumlah nol (*zero sum* *game*) atau permainan berjumlah konstan (*constant sum game*). Jika jumlah keuntungan dan kerugian tidak sama dengan nol maka disebut permainan berjumlah tak nol (*non-zero sum game*) (Subagyo dkk., 1990).

Ada beberapa metode yang dapat digunakan dalam penyelesaian teori permainan di antaranya metode Dominansi, keseimbangan Nash, Algoritma *Brown*, Aljabar Matriks, dan Pemrograman Linear. Pada penelitian sebelumnya, Wicaksana (2018) membahas tentang optimalisasi persaingan transportasi dengan menggunakan metode simpleks dan algoritma *brown.* Mengacu dari penelitian yang sudah ada, skripsi ini akan mengkolaborasikan algoritma brown dalam penyelesaian matriks berukuran *mxn* dan metode simpleks dengan menggunakan program matlab. Pada skripsi ini akan dibahas mengenai aplikasi teori permainan dengan mengacu pada artikel Ghadle dan Pawar (2014) dengan judul "*Game Theory Problems By An Alternative Simplex Method"*. Konsep ini akan diterapkan dalam persaingan pangsa pasar dalam bidang makanan *fastfood* antara McDonald's, KFC dan Richeese Factory dengan responden Mahasiswa Universitas Brawijaya.

1.2 Rumusan Masalah

1. Bagaimana aplikasi teori permainan pada optimalisasi strategi pemasaran McDonald's, KFC dan Richeese Factory dengan menggunakan metode simpleks dan algoritma *brown* ?
2. Bagaimana strategi optimal yang dibuat oleh McDonald's, KFC dan Richeese Factory Malang ?

1.3 Batasan Masalah

1. Jenis permainan merupakan permainan berjumlah nol.
2. Responden kuisioner adalah mahasiswa Universitas Brawijaya.
3. Kuisioner diisi oleh mahasiswa yang pernah mengonsumsi McDonald's berlokasi di jalan MT Haryono no 115 Malang, KFC berlokasi di jalan Veteran no 2 Malang dan Richeese Factory berlokasi di jalan Besar Ijen no 77A Malang.
4. Ukuran matriks *payoff* adalah $m \times n$

1.4 Tujuan

1. Untuk mengetahui aplikasi teori permainan pada optimalisasi strategi pemasaran dalam McDonald's, KFC dan Richeese Factory dengan menggunakan metode simpleks dan algoritma *brown.*
2. Untuk menentukan strategi optimal yang dibuat oleh McDonald's, KFC dan Richeese Factory Malang.

BAB II
TINJAUAN PUSTAKA

2.1 Matriks

Menurut Taha (1997), matriks adalah serangkaian elemen dalam bentuk persegi panjang. Elemen a_{ij} dari matriks A berada pada baris ke-*i* dan kolom ke-*j* dari rangkaian tersebut. Order (ukuran) matriks $m \times n$ jika matriks tersebut memiliki *m* baris dan *n* kolom. Bentuk umum matriks $m \times n$ sebagai berikut.

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ a_{31} & a_{32} & \cdots & a_{3n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = (a_{ij})_{m\times n}$$

di mana $i = 1, 2, \ldots, m$ dan $j = 1, 2, \ldots, n$.

2.2 Matriks *Payoff*

Menurut Kartono (1994), matriks *payoff* adalah suatu tabel berbentuk segi empat dengan elemen-elemennya yang merupakan besarnya nilai pembayaran yang bersesuaian dengan strategi yang digunakan oleh kedua pihak. Bentuk umum matriks *payoff* sebagai berikut.

Tabel 2.1 Bentuk umum matriks *payoff*

		Pemain Kedua (P_2)				
	i \ j	1	2	3	$\cdots$	n
Pemain Pertama (1)	1	a_{11}	a_{12}	a_{13}	$\cdots$	a_{1n}
	2	a_{21}	a_{22}	a_{23}	$\cdots$	a_{2n}
	3	a_{31}	a_{32}	a_{33}	$\cdots$	a_{3n}
	$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$
	m	a_{m1}	a_{m2}	a_{m3}	$\cdots$	a_{mn}

Keterangan :

m : banyaknya strategi yang dimiliki pemain P_1
n : banyaknya strategi yang dimiliki pemain P_2
a_{ij} : nilai pemabayaran didefinisikan secara numerik yang bersesuaian dengan strategi ke-i bagi pemain P_1 dan strategi ke- j bagi pemain P_2

2.3 Teori Permainan

Menurut Straffin (1993), teori permainan merupakan suatu pendekatan matematis untuk merumuskan situasi persaingan dan konflik antara berbagai kepentingan. Teori permainan melibatkan dua atau lebih pengambil keputusan atau yang disebut pemain. Setiap pemain dalam teori permainan mempunyai keinginan untuk menang. Masing-masing pemain memiliki sejumlah strategi yang akan diambil. Strategi yang diambil oleh masing-masing pemain menentukan hasil dari permainan. Hasil dari strategi-strategi yang dimainkan dinyatakan dengan angka-angka dalam matriks *payoff* atau biasa disebut matriks pembayaran.

Terdapat dua jenis strategi permainan yang dapat digunakan pada teori permainan, yaitu *pure strategy* (setiap pemain mempergunakan strategi tunggal) dan *mixed strategy* (setiap pemain menggunakan campuran dari berbagai strategi yang berbeda-beda). *Pure strategy* digunakan untuk jenis permainan yang hasil optimalnya mempunyai *saddle point* (semacam titik keseimbangan antara nilai permainan kedua pemain), sedangkan *mixed strategy* digunakan untuk mencari solusi optimal dari kasus teori permainan yang tidak mempunyai *saddle point*.

2.4 Unsur-Unsur Dasar Teori Permainan

Menurut Fatchiyah (2011), unsur-unsur dasar teori permainan sebagai berikut :

1. Jumlah Pemain

Permainan diklasifikasikan menurut jumlah kepentingan atau tujuan yang ada dalam permainan tersebut. Dalam hal ini perlu dipahami bahwa pengertian "jumlah pemain" tidak selalu sama artinya dengan "jumlah orang" yang terlibat dalam permainan. jumlah pemain disini berarti jumlah kelompok pemain berdasarkan masing-masing kepentingan atau tujuannya. Dengan demikian, dua orang atau lebih yang mempunyai kepentingan yang sama dapat diperhitungkan sebagai satu kelompok pemain.

2. Ganjaran / *Pay-Off*

Ganjaran / *pay-off* adalah hasil akhir yang terjadi pada akhir permainan berkenaan dengan ganjaran ini, permainan digolongkan menjadi 2 macam kategori, yaitu permainan jumlah-nol (*zero-sum games*) dan permainan jumlah-bukan-nol (*non-zero-sum games*). permainan jumlah-nol terjadi jika jumlah ganjaran dari seluruh pemain adalah nol, yaitu dengan memperhitungkan setiap keuntungan sebagai bilangan positif dan setiap kerugian sebagai bilangan negatif. Selain dari itu adalah permainan jumlah – bukan-nol. Dalam permainan jumlah-nol setiap kemenangan bagi suatu pihak pemain merupakan kekalahan bagi pihak pemain lain. letak arti penting dari perbedaan kedua kategori permainan berdasarkan ganjaran ini adalah bahwa permainan jumlah-nol adalah suatu sistem yang tertutup, sedangkan permainan jumlah-bukan-nol tidak demikian halnya. Hampir semua permainan pada dasarnya merupakan permainan jumlah-nol. Berbagai situasi dapat dianalisis sebagai permainan jumlah-nol.

3. Strategi Permainan

Strategi permainan dalam teori permainan adalah suatu siasat atau rencana tertentu dari seorang pemain, sebagai reaksi atas aksi yang mungkin dilakukan oleh pemain yang menjadi saingannya. Permainan diklasifikasikan menurut jumlah strategi yang tersedia bagi masing-masing pemain. Jika pemain pertama memiliki m kemungkinan strategi dan pemain kedua memiliki n kemungkinan strategi maka permainan tersebut dinamakan permainan $m \; x \; n$. Letak arti penting dari perbedaan jenis permainan berdasarkan jumlah strategi ini adalah bahwa permainan dibedakan menjadi permainan berhingga dan permainan tak berhingga. Permainan berhingga terjadi apabila jumlah terbesar dari strategi yang dimiliki oleh setiap pemain berhingga atau tertentu, sedangkan permainan tak berhingga terjadi jika setidak-tidaknya seorang pemain memiliki jumlah strategi yang tak berhingga atau tidak tertentu.

4. Matriks Permainan

Setiap permainan yang dianalisis dengan teori permainan selalu dapat disajikan dalam bentuk sebuah matriks permainan. Matriks permainan disebut juga matriks ganjaran yaitu sebuah matriks yang semua unsur berupa ganjaran dari para pemain yang terlibat dalam permainan tersebut. Baris-barisnya melambangkan strategi yang dimiliki pemain pertama, sedangkan kolom-kolomnya melambangkan strategi yang dimiliki pemain lain. Dengan demikian, permainan berstrategi $m\ x\ n$ dilambangkan dengan matriks permainan $m\ x\ n$.

Teori permainan berasumsi bahwa strategi yang tersedia bagi masing-masing pemain dapat dihitung dan ganjaran yang berkaitan dengannya dapat dinyatakan dalam unit, meskipun tidak selalu harus dalam unit moneter. Hal ini penting bagi penyelesaian permainan, yaitu untuk menentukan pilihan strategi yang akan dijalankan oleh masing-masing pemain, dengan menganggap bahwa masing masing pemain berusaha memaksimumkan keuntungannya yang minimum (maksimin) atau meminimumkan kerugiannya yang maksimum (minimaks).

Nilai dari suatu permainan adalah ganjaran rata-rata/ ganjaran yang diharapkan dari sepanjang rangkaian permainan, dengan menganggap kedua pemain selalu berusaha memainkan strateginya yang optimum. Secara konvensional, nilai permainan dilihat dari pihak pemain yang strategi - strateginya dilambangkan oleh baris-baris matriks ganjaran, dengan kata lain dilihat dari sudut pandang pemain tertentu. Pemain dikatakan adil (*fair*) apabila nilainya nol, dimana tak seorang pemain pun yang memperoleh keuntungan atau kemenangan dalam permainan yang tidak adil (*unfair*) seorang pemain akan memperoleh kemenangan atas pemain lain, yaitu jika nilai permainan tersebut bukan nol, dalam hal ini nilai pemain adalah positif jika pemain pertama (pemain baris) memperoleh kemenangan, sebaliknya nilai permainan negatif jika pemain lain (pemain kolom) memperoleh kemenangan.

5. Titik Pelana (*Saddle Point*)

Titik pelana adalah suatu unsur didalam matriks permainan yang sekaligus sebagai maksimin baris dan minimaks kolom. permainan dikatakan bersaing ketat (*Strictly determined*) jika matriksnya memiliki titik pelana. Strategi yang optimum bagi masing-masing pemain adalah strategi pada baris dan kolom yang mengandung titik pelana tersebut. Dalam hal ini baris yang mengandung titik pelana merupakan strategi optimum bagi pemain pertama, sedangkan kolom yang mengandung titik pelana merupakan strategi optimum bagi pemain lain. Langkah pertama penyelesaian sebuah matriks permainan adalah memeriksa ada atau tidaknya titik pelana. Bila terdapat titik pelana permainan dapat segera dianalisis untuk diselesaikan. Untuk menentukan titik pelana biasanya dilakukan dengan menuliskan nilai-nilai minimum dan maksimum masing-masing kolom, kemudian menentukan maksimun di antara minimum baris dan minimum di antara maksimum kolom. Jika unsur maksimum dari minimum baris sama dengan unsur minimum dari maksimum kolom, atau jika maksimin = minimaks, berarti unsur tersebut merupakan titik pelana.

2.5 Strategi Murni dan Strategi Campuran

Menurut Wijaya (2013), strategi murni digunakan apabila permainan mengandung titik pelana (*saddle point*) di mana masing-masing pemain menjalankan satu jenis strategi dengan menggunakan kriteria maksimin dan kriteria minimaks.

Menurut Subagyo, dkk (1990), strategi campuran digunakan apabila nilai maksimin tidak sama dengan nilai minimaks sehingga tidak ditemukan *saddle point* pada permainan tersebut. Penyelesaian permainan strategi campuran dapat dilakukan dengan beberapa metode, yaitu metode grafik, metode analitis, metode aljabar matriks, dan metode pemrograman linear.

2.6 Dominasi

Pada umumnya matriks *payoff* 2×2 dapat diselesaikan dengan mudah, tetapi pada permainan yang lebih besar biasanya membutuhkan teknik yang lebih rumit. Gillet (1976) menyarankan untuk mereduksi permainan yang berukuran $m \times n$ sebelum menyelesaikannya.

Secara sederhana dapat dikatakan bahwa baris i mendominasi baris k pada matriks *payoff* jika $p_{ij} \geq p_{kj}$ untuk $j = 1, 2, \ldots, n$. Pada kasus ini baris k dapat dieliminasi dari asumsi sebelumnya. Hal tersebut berarti bahwa baris k tidak akan menghasilkan suatu *payoff* yang lebih besar untuk pemain 1 daripada baris i, kemudian mengeliminasinya tanpa memperhatikan hal yang dilakukan oleh pemain 2. Dengan memperhatikan kolom, dikatakan bahwa kolom j mendominasi kolom k pada matriks *payoff* jika $p_{ij} \geq p_{ik}$ untuk $i = 1, 2, \ldots, m$. Pada kasus ini, kolom k dapat dieliminasi dari asumsi sebelumnya.

Hasil dominasi maksimal pada kenyataannya tidak selalu menghasilkan matriks *payoff* 2×2, sehingga diperlukan metode lain untuk menyelesaikan matriks *payoff*. Apabila matriks *payoff* tidak memenuhi ketentuan di atas, maka matriks *payoff* dapat diselesaikan dengan menggunakan pemrograman linear.

Contoh :

Diberikan matriks *payoff* 3×4 sebagai berikut. Ubahlah matriks berikut menjadi matriks *payoff* 2×2 dengan menggunakan metode dominasi.

$$\begin{array}{ccc} & & \text{Pemain 2} \\ & & \begin{array}{cccc} q_1 & q_2 & q_3 & q_4 \end{array} \\ \text{Pemain 1} & \begin{array}{c} p_1 \\ p_2 \\ p_3 \end{array} & \begin{bmatrix} 2 & 0 & 1 & 4 \\ 1 & 2 & 5 & 3 \\ 4 & 1 & 3 & 2 \end{bmatrix} \end{array}$$

Langkah-langkah penyelesaian matriks *payoff* di atas sebagai berikut.

1. Eliminasi kolom

Berdasarkan matriks di atas, eliminasi kolom untuk pemain 2 dapat dilakukan berdasarkan permainan strategi pemain 1. Jika pemain 1 memainkan strategi p_1 maka strategi q_2 optimal untuk pemain 2 dengan *payoff* terkecil dibanding yang lainnya yaitu 0. Jika pemain 1 memainkan strategi p_2 maka strategi q_1 optimal untuk pemain 2. Jika pemain 1 memainkan strategi p_3 maka strategi q_2 optimal untuk pemain 2. Strategi q_1 dan q_2 mendominasi strategi q_3 dan q_4 sehingga strategi q_3 atau q_4 dapat dieliminasi. Dalam hal ini dipilih strategi q_4 untuk dieliminasi.

$$\begin{array}{ccc} & & \text{Pemain 2} \\ & & \begin{array}{cccc} q_1 & q_2 & q_3 & q_4 \end{array} \\ \text{Pemain 1} & \begin{array}{c} p_1 \\ p_2 \\ p_3 \end{array} & \begin{bmatrix} 2 & 0 & 1 & \cancel{4} \\ 1 & 2 & 5 & 3 \\ 4 & 1 & 3 & \cancel{2} \end{bmatrix} \end{array}$$

2. Eliminasi baris

Eliminasi baris untuk pemain 1 dapat dilakukan berdasarkan permainan strategi pemain 2. Jika pemain 2 memainkan strategi q_1 maka strategi p_3 optimal untuk pemain 1 dengan *payoff* terbesar dibanding yang lainnya yaitu 4. Jika pemain 2 memainkan strategi q_2 maka strategi p_2 optimal untuk pemain 1. Jika pemain 2 memainkan strategi q_3 maka strategi p_2 optimal untuk pemain 1. Strategi p_2 dan p_3 mendominasi strategi p_1 sehingga strategi p_1 dapat dieliminasi.

$$\begin{array}{cc} & \text{Pemain 2} \\ & \begin{array}{ccc} q_1 & q_2 & q_3 \end{array} \\ \text{Pemain 1} \quad \begin{array}{c} \cancel{p_1} \\ p_2 \\ p_3 \end{array} & \begin{bmatrix} \cancel{2} & \cancel{0} & \cancel{1} \\ 1 & 2 & 5 \\ 4 & 1 & 3 \end{bmatrix} \end{array}$$

3. Eliminasi kolom

Eliminasi kolom selanjutnya dilakukan dengan cara yang sama seperti langkah 1. Jika pemain 1 memainkan strategi p_2 maka strategi q_1 optimal untuk pemain 2. Jika pemain 1 memainkan strategi p_3 maka strategi q_2 optimal untuk pemain 2. Strategi q_1 dan q_2 mendominasi strategi q_3 sehingga strategi q_3 dapat dieliminasi.

$$\begin{array}{cc} & \text{Pemain 2} \\ & \begin{array}{ccc} q_1 & q_2 & \cancel{q_3} \end{array} \\ \text{Pemain 1} \quad \begin{array}{c} p_2 \\ p_3 \end{array} & \begin{bmatrix} 1 & 2 & \cancel{5} \\ 4 & 1 & \cancel{3} \end{bmatrix} \end{array}$$

Jadi, matriks *payoff* 2×2 yang terbentuk adalah sebagai berikut.

$$\begin{array}{cc} & \text{Pemain 2} \\ & \begin{array}{cc} q_1 & q_2 \end{array} \\ \text{Pemain 1} \quad \begin{array}{c} p_2 \\ p_3 \end{array} & \begin{bmatrix} 1 & 2 \\ 4 & 1 \end{bmatrix} \end{array}$$

2.7 Pemerograman Linear

Menurut Mulyono (2007), pemrograman linear adalah suatu program untuk menyelesaikan permasalahn yang batas-batasannya berbentuk pertidaksamaan linear. Secara umum program linear terdiri dari dua bagian, yaitu fungsi kendala dan fungsi objektif. Fungsi kendala adalah batasan-batasan yang dipenuhi, sedangkan fungsi objektif adalah fungsi yang nilainya akan dioptimumkan (dimaksimumkan atau diminimumkan). Dalam program linear ini batasan-batasan yang terdapat didalam masalah program linear diterjemahkan terlebih dahulu ke dalam bentuk perumusan matematika, yang disebut model matematika. Model matematika adalah suatu bentuk interpretasi manusia dalam menerjemahkan atau merumuskan persoalan-persoalan yang ada ke bentuk matematika sehingga persoalan itu dapat diselesaikan secara matematis. Berikut adalah tabel matriks perolehan dalam pemrograman linear.

Tabel 2.2 Matriks perolehan dalam pemrogaman linear

I \ II	y_1	y_2	$\cdots$	y_n
x_1	a_{11}	a_{12}	$\cdots$	a_{1n}
x_2	a_{21}	a_{22}	$\cdots$	a_{2n}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
x_m	a_{m1}	a_{m2}	$\cdots$	a_{mn}

Masalah program linear untuk pemain I adalah :

$$\text{Min } Z_0 = x_1 + x_2 + x_3 + \cdots + x_m,$$

dengan batasan :

$$a_{11}x_1 + a_{21}x_2 + \cdots + a_{m1}x_m \geq 1$$
$$a_{12}x_1 + a_{22}x_2 + \cdots + a_{m2}x_m \geq 1$$
$$\vdots$$

$$a_{1n}x_1 + a_{2n}x_2 + \cdots + a_{mn}x_m \geq 1,$$
$$x_i \geq 0, i = 1,2,3, \ldots, m$$

dimana $F_0 = \frac{1}{v}$ dan $x_i* = \frac{x_i}{v}$.

Masalah program linear untuk pemain II adalah :

$$\text{Max } Z_0 = y + y_2 + y_3 + \cdots + y_n,$$

dengan batasan :

$$a_{11}y_1 + a_{12}y_2 + \cdots + a_{1n}y_n \leq 1$$
$$a_{21}y_1 + a_{22}y_2 + \cdots + a_{2n}y_n \leq 1$$
$$.$$
$$.$$
$$a_{n1}y_1 + a_{n2}y_2 + \cdots + a_{nm}y_n \leq 1,$$
$$y_j \geq 0, j = 1,2,3, \ldots, m$$

dimana $F_0 = \frac{1}{v}$ dan $y_j* = \frac{x_i}{v}$.

2.8 Metode Simpleks

Untuk menyelesaikan suatu permainan dengan hasil dominasi matriks payoff yang masih relatif besar ($m \times n$) maka dapat digunakan metode simpleks. Penyelesain menggunakan metode simpleks melalui beberapa tahap iterasi yang dilakukan berulang-ulang hingga diperoleh solusi optimal.

Menurut Soekartawi, (1992), penyelesaian dengan menggunakan metode simpleks melalui tahap iterasi yang dilakukan berulang-ulang hingga diperoleh solusi optimal. Langkah-langkah penyelesaian matriks $m \times n$ menggunakan metode simpleks sebagai berikut.

1. Mengubah masalah pemrograman linear menjadi bentuk standar pemrograman linear, yaitu dengan cara mengubah tanda "≤" menjadi "=", dan menambahkan variabel slack (*S*).
2. Melakukan iterasi dengan mencari kolom kunci, baris kunci, dan angka kunci (angka *pivot*).

Kolom kunci yang diambil (*entering variable*) berdasarkan nilai $Z_j - C_j$ paling negatif. Baris kunci yang diambil (*leaving variable*) berdasarkan rasio terkecil. Rumus untuk menentukan rasio, yaitu rasio = indeks masing-masing baris $\div$ masing-masing koefisien kolom kunci baru. Sementara itu, angka kunci yang diambil adalah angka pertemuan antara kolom kunci dan baris kunci.

3. Mengubah variabel keputusan pada baris kunci dengan variabel keputusan pada kolom kunci, lalu menentukan baris kunci baru, dan baris lainnya.
   Rumus untuk menentukan baris kunci baru, yaitu baris kunci baru = baris kunci lama $\div$ angka kunci.
   Koefisien kolom kunci baru = koefisien kolom kunci pada baris awal - (koefisien kolom kunci pada baris awal $\times$ angka kunci $\div$ angka kunci).
   Baris baru = baris awal – (baris kunci awal $\times$ koefisien kolom kunci pada baris awal $\div$ angka kunci awal).
4. Memastikan nilai $Z_j - C_j$ tidak bernilai negatif sehingga diperoleh solusi optimal. Jika nilai $Z_j - C_j$ masih bernilai negatif, maka dilakukan iterasi selanjutnya seperti pada langkah 2 dan 3.

Berikut adalah tabel simpleks awal untuk masalah maksimum dijelaskan pada tabel 2.3.

Tabel 2.3 Metode Simpleks Awal

C_b	C_j	C_1	C_2	..	C_n	0	0	..	0	Indeks	Rasio
	Basis	Y_1	Y_2	..	Y_n	S_1	S_2	..	S_m	(b_j)	
0	S_1	a_{11}	a_{12}	..	a_{1n}	1	0	..	0	b_1	-
0	S_2	a_{21}	a_{22}	..	a_{2n}	0	1	..	0	b_2	-
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$	$\vdots$
0	S_m	a_{m1}	a_{m2}	..	a_{mn}	0	0	..	1	b_m	-
$Z_j - C_j$		$\sum_{i=1}^{m} C_{bi}Y_{1i} - C_1$	$\sum_{i=1}^{m} C_{bi}Y_{2i} - C_2$	..	$\sum_{i=1}^{m} C_{bi}Y_{ni} - C_n$	$\sum_{i=1}^{m} C_{bi}S_{1i} - C_{n+1}$	$\sum_{i=1}^{m} C_{bi}S_{2i} - C_{n+2}$	..	$\sum_{i=1}^{m} C_{bi}S_{mi} - C_{n+m}$	$\sum_{i=1}^{m} C_{bi}b_{ji}$	

Keterangan:

Misalkan menentukan nilai $Z_1 - C_1$,

maka $Z_1 - C_1 = \sum_{i=1}^{m} C_{bi}Y_{1i} - C_1 \quad = (C_{b1}Y_{11} + C_{b2}Y_{12} + \ldots + C_{bm}Y_{1m}) - C_1$

2.9 Algoritma Brown

Misalkan diberikan suatu permainan pada tabel 2.4 berikut.

Tabel 2.4 Tabel permainan.

II / I	y_1	y_2	$\cdots$	y_n
x_1	a_{11}	a_{12}	$\cdots$	a_{1n}
x_2	a_{21}	a_{22}	$\cdots$	a_{2n}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
x_m	a_{m1}	a_{m2}	$\cdots$	a_{mn}

Menurut Siagian (1987), cara Brown menyelesaikan permainan ini ialah dengan melakukan beberapa langkah seperti dibawah ini :

1. Pemain I memilih sebuah baris untuk dimainkan dan pemain II kemudian memainkan kolom yang terkait dengan elemen minimum pada baris tersebut.
2. Selanjutnya pemain I memilih baris untuk dimainkan yang terkait terhadap elemen maksimum pada kolom yang dimainkan oleh pemain II pada langkah 1.
3. Pemain II menjumlahkan baris yang telah dimainkan pemain 1 sejauh itu dan memainkan kolom yang terkait terhadap jumlah elemen minimum.
4. Pemain I menjumlahkan kolom yang telah dimainkan pemain II sejauh itu dan memainkan baris yang terkait terhadap suatu jumlah elemen maksimum. Jika jumlah iterasi yang digunakan terpenuhi lanjut ke langkah 5. Sebaliknya, jika jumlah iterasi tidak terpenuhi kembali ke langkah 3.
5. Hitung batas atas $\overline{v}$ dan batas bawah $\underline{v}$ berturut-turut.

$$\overline{v} = \frac{jumlah\ elemen\ maksimum\ dari\ langkah\ ke\ 4}{banyaknya\ memainkan\ permainan}$$

$$\underline{v} = \frac{jumlah\ elemen\ maksimum\ dari\ langkah\ ke\ 3}{banyaknya\ memainkan\ permainan}$$

dimana $\underline{v} \leq v \leq \overline{v}$

6. Strategi untuk pemain I dan II dilakukan sebagai berikut,

$$X_i = \frac{jumlah\ baris\ i\ yang\ dimainkan}{n}, i = 1, \dots \dots m$$

$$Y_j = \frac{jumlah\ baris\ j\ yang\ dimainkan}{n}, j = 1, \dots \dots n$$

dimana n = jumlah langkah dalam permainan.

2.10 Sampel Penelitian

Menurut Riduwan (2005), Pengambilan sampel dalam penelitian diperlukan jika populasi yang diambil sangat besar. Pengambilan sampel dapat dilakukan dengan menggunakan Rumus Slovin sebagai berikut.

$$n = \frac{N}{1 + N(e)^2},$$

n : ukuran sampel,
N : ukuran populasi,
e : nilai kritis (batas ketelitian) yang diinginkan berupa persen kelonggaran ketidaktelitian karena kesalahan penarikan sampel.

BAB III
METODE PENELITIAN

3.1 Tempat dan Waktu Penelitian

Penelitian ini dilakukan di Universitas Brawijaya pada bulan November - Desember dengan studi kasus persaingan McDonald's, KFC, dan Richeese Factory.

3.2 Jenis dan Sumber Data

Menurut Agung (2012), data dapat dibedakan menjadi dua, yaitu data primer dan sekunder.

1. Data primer

Data primer adalah data yang diperoleh atau dikumpulkan oleh penelitian atau lembaga tertentu langsung dari sumbernya, dicatat dan diamati untuk pertama kalinya dan hasilnya digunakan langsung oleh peneliti atau lembaga itu sendiri untuk memecahkan persoalan yang akan dicari jawabannya.

2. Data sekunder

Data sekunder adalah data yang diperoleh atau dikumpulkan oleh orang lain atau lembaga tertentu.

Data yang digunakan dalam penelitian ini adalah data primer dengan cara melalui penyebaran kuisioner kepada mahasiswa Universitas Brawijaya. Tujuannya adalah untuk mengetahui minat mahasiswa terhadap McDonald's dan KFC dan Richeese Factory di kota malang.

3.3 Metode Pengumpulan Data

Menurut Sumarsono (2004), teknik pengumpulan data terbagi menjadi empat sebagai berikut.

1. Pengamatan langsung

Pengamatan langsung merupakan pengamatan yang dilakukan secara langsung terhadap objek riset.

2. Wawancara

Wawancara merupakan komunikasi dan interaksi dua arah antara peneliti dan objek riset untuk mendapatkan data.

3. Pengisian daftar pertanyaan

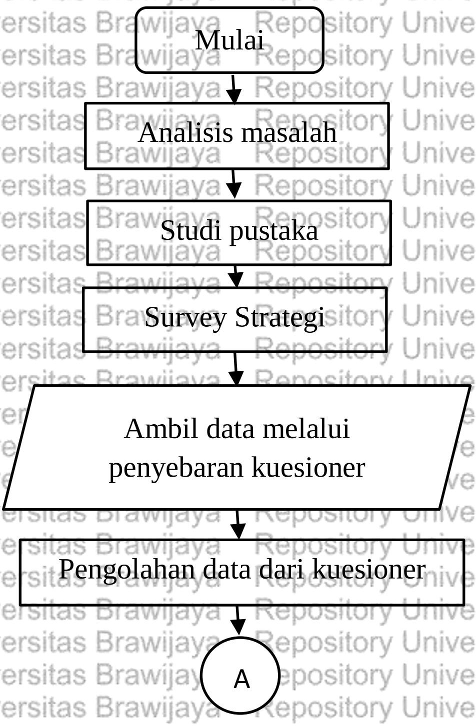
Pengisian daftar pertanyaan merupakan bentuk wawancara tidak langsung. Umumnya digunakan untuk responden yang berjumlah sangat banyak.

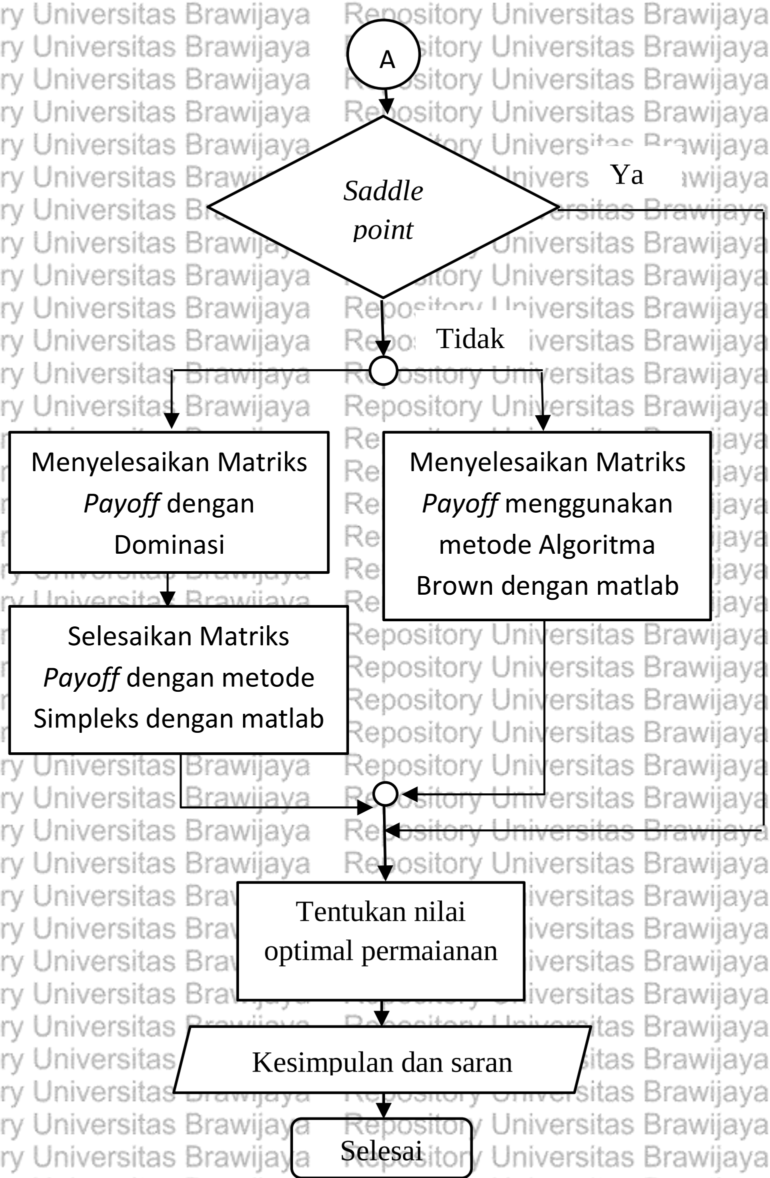
4. Studi pustaka

Studi pustaka merupakan teknik pengumpulan data dengan cara membaca buku untuk digunakan dalam kerangka riset yang berbeda.

Dalam penelitian ini menggunakan dua metode pengumpulan data, yaitu metode wawancara dan kuesioner. Berikut diagram alur untuk mempermudah penelitian.

3.4 Diagram Alir





Gambar 3.1 Diagram Alir

BAB IV
HASIL DAN PEMBAHASAN

4.1 Data Hasil Penelitian

Data yang digunakan dalam penelitian ini adalah data primer dengan cara melalui penyebaran kuisioner kepada mahasiswa Universitas Brawijaya. Berdasarkan hasil penyebaran kuisioner terhadap responden, yakni mahasiswa Universitas Brawijaya yang berjumlah 72.728 mahasiswa, ukuran sampel yang diambil (*n*) dari populasi tersebut dan persen kelonggaran (*e*) 10% adalah sebagai berikut.

$$n = \frac{72.728}{1 + 72.728(10\%)^2}$$

$$n = 99{,}86$$

$$n \cong 99{,}86.$$

Berdasarkan rumus Slovin, sampel yang diambil minimal berjumlah 99,86 responden. Pada penelitian ini, sampel yang diambil berjumlah 130 responden, dengan rincian 10 kuisioner dari 13 fakultas yang berada di area Universitas Brawijaya.
Alternatif restoran *fast food* yang menjadi pilihan sebanyak tiga pilihan, yaitu McDonald's, KFC, dan Richeese Factory yang ditentukan berdasarkan masing – masing strategi yang ditawarkan. Banyaknya alternatif strategi yang ditawarkan untuk masing-masing restoran *fast food*, mempunyai strategi seperti harga, lokasi, promo, variasi produk dan pelayanan.

Data diperoleh dari pengisian kuisioner oleh responden terhadap pertanyaan nomor 2, yaitu tentang pilihan restoran dalam Lampiran 1. Berdasarkan jumlah kuisioner yang terkumpul, yaitu 130 kuisioner diperoleh prosentase masing-masing restoran pilihan sebagai berikut.

1. Pilihan McDonald's

$$= \frac{78}{130} \times 100\ \% = 60\ \%$$

2. Pilihan KFC

$$= \frac{30}{130} \times 100\ \% = 23{,}08\%$$

3. Pilihan Richeese Factory

$$= \frac{22}{130} \times 100\ \% = 16{,}92\%$$

4.2 Teori Permainan Pada Optimalisasi Strategi McDonald's dan KFC

Berdasarkan Lampiran 3, pembentukan matriks McDonald's dan KFC diperoleh matriks berukuran $5x5$ sebagai berikut.

$$\begin{array}{cc} & \text{KFC} \\ \text{McDonald's} & \begin{bmatrix} 3{,}97; 3{,}64 & 3{,}88; 3{,}38 & 3{,}99; 3{,}33 & 3{,}87; 3{,}03 & 3{,}92; 3{,}24 \\ 3{,}86; 3{,}55 & 4{,}26; 3{,}49 & 4{,}08; 3{,}40 & 3{,}90; 3{,}15 & 4{,}31; 3{,}38 \\ 3{,}77; 3{,}33 & 3.91; 3{,}58 & 4{,}06; 3{,}49 & 3{,}88; 3{,}38 & 4{,}33; 3{,}21 \\ 3{,}87; 3{,}00 & 3{,}79; 3{,}19 & 3{,}87; 3{,}58 & 4{,}36; 3{,}45 & 4{,}24; 3{,}33 \\ 3{,}76; 3{,}24 & 4{,}10; 3{,}26 & 3{,}81; 3{,}45 & 3{,}88; 3{,}59 & 4{,}03; 3{,}33 \end{bmatrix} \end{array}$$

Cara untuk menentukan optimalisasi strategi antara McDonald's dan KFC dilakukan secara terpisah untuk pemain pertama dan pemain kedua, sehingga didapat matriks *payoff* McDonald's dan matriks *payoff* KFC.

4.2.1 Penyelesaian Matriks *PayOff* McDonald's

Langkah - langkah untuk menyelesaikan matriks *payoff* McDonald's adalah sebagai berikut.

4.2.1.1 Penyelesaian matriks *payoff* McDonald's dengan menggunakan metode simpleks.

Langkah - langkah untuk menyelesaikan matriks *payoff* McDonald's menggunakan metode simpleks adalah sebagai berikut.

Tabel 4.1 Penentuan *saddle point* pada matriks *payoff* McDonald's antara McDonald's dan KFC.

	q_1	q_2	q_3	q_4	q_5	Minimum Baris	Maksimin
p_1	3,97	3,88	3,99	3,87	3,92	3,87	3,87
p_2	3,86	4,26	4,08	3,90	4,31	3,86	
p_3	3,77	3,91	4,06	3,88	4,33	3,77	
p_4	3,87	3,79	3,87	4,36	4,24	3,79	
p_5	3,76	4,10	3,81	3,88	4,03	3,76	
Maksimum Kolom	3,97	4,26	4,08	4,36	4,33		
Minimaks	3,97						

1. Menentukan *saddle point*.

Penentuan *saddle point* pada matriks *payoff* McDonald's dapat dilihat pada Tabel 4.1. Karena nilai Maksimin ≠ Minimaks, sehingga dapat disimpulkan bahwa tidak terdapat *saddle point* pada matriks *payoff* McDonald's.

2. Menyelesaikan matriks *payoff* dengan menggunakan dominasi.

Dominasi dilakukan jika tidak terdapat *saddle point* pada matriks *payoff*. Penyelesaian matriks *payoff* dengan menggunakan dominasi, yaitu mengeliminasi matriks *payoff* berukuran besar menjadi matriks berukuran lebih kecil. Berikut adalah matriks *payoff* McDonald's.

$$\begin{array}{cc} & \text{KFC} \\ \text{McDonald's} & \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{array} \begin{array}{c} \begin{array}{ccccc} q_1 & q_2 & q_3 & q_4 & q_5 \end{array} \\ \begin{bmatrix} 3{,}97 & 3{,}88 & 3{,}99 & 3{,}87 & 3{,}92 \\ 3{,}86 & 4{,}26 & 4{,}08 & 3{,}90 & 4{,}31 \\ 3{,}77 & 3{,}91 & 4{,}06 & 3{,}88 & 4{,}33 \\ 3{,}87 & 3{,}79 & 3{,}87 & 4{,}36 & 4{,}24 \\ 3{,}76 & 4{,}10 & 3{,}81 & 3{,}88 & 4{,}03 \end{bmatrix} \end{array} \end{array}$$

Langkah-langkah dominasi matriks *payoff* McDonald's sebagai berikut.

i) Eliminasi baris

Cara mengeliminasi baris atau McDonald's sebagai pemain pertama, yaitu berdasarkan permainan strategi oleh pemain kolom atau KFC. Jika KFC memainkan strategi q_1, maka strategi p_1 optimal untuk McDonald's dengan *payoff* terbesar dibanding lainnya, yaitu 3,97. Jika KFC memainkan strategi q_2, maka strategi p_2 optimal untuk McDonald's.

Langkah tersebut diulang hingga strategi q_5. Strategi optimal yang dihasilkan McDonald's secara keseluruhan adalah strategi p_1 p_2 p_3 dan p_4. Dalam hal ini, dipilih strategi p_5 untuk dieliminasi.

$$
\text{McDonald's}\quad
\begin{array}{c}
\\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5
\end{array}
\begin{array}{c}
\text{KFC} \\
\begin{array}{ccccc} q_1 & q_2 & q_3 & q_4 & q_5 \end{array} \\
\begin{bmatrix}
3{,}97 & 3{,}88 & 3{,}99 & 3{,}87 & 3{,}92 \\
3{,}86 & 4{,}26 & 4{,}08 & 3{,}90 & 4{,}31 \\
3{,}77 & 3{,}91 & 4{,}06 & 3{,}88 & 4{,}33 \\
3{,}87 & 3{,}79 & 3{,}87 & 4{,}36 & 4{,}24 \\
\cancel{3{,}76} & \cancel{4{,}10} & \cancel{3{,}81} & \cancel{3{,}88} & \cancel{4{,}03}
\end{bmatrix}
\end{array}
$$

ii) Eliminasi kolom

Cara mengeliminasi kolom atau KFC sebagai pemain kedua, yaitu berdasarkan permainan strategi oleh pemain baris atau McDonald's. Jika McDonald's memainkan strategi p_1, maka strategi q_4 optimal untuk KFC. Jika McDOnald's memainkan strategi p_2, maka strategi q_1 optimal untuk KFC..

Langkah tersebut diulang hingga strategi q_5. Strategi optimal yang dihasilkan KFC secara keseluruhan adalah strategi q_1 q_3 dan q_4 sehingga dapat disimpulkan bahwa strategi q_1 q_3 dan q_4 mendominasi strategi q_2 dan q_5. Oleh karena itu, salah satu kolom q_2 atau q_5 dapat dieliminasi. Dalam hal ini, dipilih strategi q_5 untuk dieliminasi.

$$
\text{McDonald's}\quad
\begin{array}{c}
\\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5
\end{array}
\begin{array}{c}
\text{KFC} \\
\begin{array}{ccccc} q_1 & q_2 & q_3 & q_4 & q_5 \end{array} \\
\begin{bmatrix}
3{,}97 & 3{,}88 & 3{,}99 & 3{,}87 & \cancel{3{,}92} \\
3{,}86 & 4{,}26 & 4{,}08 & 3{,}90 & \cancel{4{,}31} \\
3{,}77 & 3{,}91 & 4{,}06 & 3{,}88 & \cancel{4{,}33} \\
3{,}87 & 3{,}79 & 3{,}87 & 4{,}36 & \cancel{4{,}24} \\
\cancel{3{,}76} & \cancel{4{,}10} & \cancel{3{,}81} & \cancel{3{,}88} & \cancel{4{,}03}
\end{bmatrix}
\end{array}
$$

iii) Eliminasi baris

Eliminasi baris selanjutnya dapat dilakukan dengan cara yang sama seperti langkah i). Jika KFC memainkan strategi q_1, maka strategi p_1 optimal untuk McDonald's. Jika KFC memainkan strategi q_2, maka strategi p_2 optimal untuk McDonald's.

Langkah tersebut diulang hingga strategi q_4. Strategi optimal yang dihasilkan KFC secara keseluruhan adalah strategi p_1 p_2 dan

p_4, sehingga dapat disimpulkan bahwa strategi strategi p_1 p_2 dan p_4 mendominasi strategi p_3. Dalam hal ini, dipilih strategi p_3 untuk dieliminasi.

$$
\begin{array}{cc}
 & \text{KFC} \\
\text{McDonald's} &
\begin{array}{c}
 \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5
\end{array}
\begin{array}{c}
\begin{array}{ccccc} q_1 & q_2 & q_3 & q_4 & q_5 \end{array} \\
\begin{bmatrix}
3{,}97 & 3{,}88 & 3{,}99 & 3{,}87 & \cancel{3{,}92} \\
3{,}86 & 4{,}26 & 4{,}08 & 3{,}90 & \cancel{4{,}31} \\
\cancel{3{,}77} & \cancel{3{,}91} & \cancel{4{,}06} & \cancel{3{,}88} & \cancel{4{,}33} \\
3{,}87 & 3{,}79 & 3{,}87 & 4{,}36 & \cancel{4{,}24} \\
\cancel{3{,}76} & \cancel{4{,}10} & \cancel{3{,}81} & \cancel{3{,}88} & \cancel{4{,}03}
\end{bmatrix}
\end{array}
\end{array}
$$

iv) Eliminasi kolom

Eliminasi kolom selanjutnya dapat dilakukan dengan cara yang sama seperti langkah ii). Jika McDonald's memainkan strategi p_1, maka strategi q_4 optimal untuk KFC. Jika McDonald's memainkan strategi p_2, maka strategi q_1 optimal untuk KFC.

Langkah tersebut diulang hingga strategi p_4. Strategi optimal yang dihasilkan McDonald's adalah strategi q_1 q_2 dan q_4, sehingga dapat disimpulkan bahwa strategi q_1 q_2 dan q_4 mendominasi strategi q_3. Dalam hal ini, dipilih strategi q_3 untuk dieliminasi.

$$
\begin{array}{cc}
 & \text{KFC} \\
\text{McDonald's} &
\begin{array}{c}
 \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5
\end{array}
\begin{array}{c}
\begin{array}{ccccc} q_1 & q_2 & q_3 & q_4 & q_5 \end{array} \\
\begin{bmatrix}
3{,}97 & 3{,}88 & \cancel{3{,}99} & 3{,}87 & \cancel{3{,}92} \\
3{,}86 & 4{,}26 & \cancel{4{,}08} & 3{,}90 & \cancel{4{,}31} \\
\cancel{3{,}77} & \cancel{3{,}91} & \cancel{4{,}06} & \cancel{3{,}88} & \cancel{4{,}33} \\
3{,}87 & 3{,}79 & \cancel{3{,}87} & 4{,}36 & \cancel{4{,}24} \\
\cancel{3{,}76} & \cancel{4{,}10} & \cancel{3{,}81} & \cancel{3{,}88} & \cancel{4{,}03}
\end{bmatrix}
\end{array}
\end{array}
$$

Diperoleh hasil dominasi maksimal untuk McDonald's adalah sebagai berikut.

$$
\begin{array}{cc}
 & \text{KFC} \\
\text{McDonald's} &
\begin{array}{c}
 \\ p_1 \\ p_2 \\ p_4
\end{array}
\begin{array}{c}
\begin{array}{ccc} q_1 & q_2 & q_4 \end{array} \\
\begin{bmatrix}
3{,}97 & 3{,}88 & 3{,}87 \\
3{,}86 & 4{,}26 & 3{,}90 \\
3{,}87 & 3{,}79 & 4{,}36
\end{bmatrix}
\end{array}
\end{array}
$$

3. Mengontruksi matriks *payoff* ke dalam pemrogaman linear.

Masalah pemrograman linear terdiri atas masalah minimum dan masalah maksimum. Untuk menyelesaikan matriks *payoff* McDonald's, maka digunakan masalah minimum karena McDonald's sebagai pemain pertama berusaha untuk meminimumkan kerugian yang maksimum.

Setiap entri matriks *payoff* dikalikan dengan $k = 100$ agar perhitungannya lebih mudah. Berikut adalah hasil kontruksi matriks *payoff* McDonald's setelah masing-masing entri dikalikan dengan $k = 100$.

KFC

$$\text{McDonald's} \quad \begin{array}{c} \\ p_1 \\ p_2 \\ p_4 \end{array} \begin{array}{c} \begin{array}{ccc} q_1 & q_2 & q_4 \end{array} \\ \begin{bmatrix} 397 & 388 & 387 \\ 386 & 426 & 390 \\ 387 & 379 & 436 \end{bmatrix} \end{array}$$

Minimum $Z = X_1 + X_2 + \mathrm{X}_3 = \frac{1}{v}$

dengan batasan

$$397\,X_1 + 386\,X_2 + 387\,\mathrm{X}_3 \geq 1$$
$$388\,X_1 + 426\,X_2 + 379\,\mathrm{X}_3 \geq 1$$
$$387\,X_1 + 390\,X_2 + 436\,\mathrm{X}_3 \geq 1\,,$$
$$X_1,\, X_2, \mathrm{X}_3 \geq 0$$

Keterangan :

$X_1 = p_1$, $X_2 = p_2$, $X_3 = p_4$

4. Menyelesaikan matriks *payoff* dengan menggunakan metode simpleks.

Metode simpleks digunakan untuk menyelesaikan masalah pemrograman linear dengan melalui tahap iterasi. Iterasi yang digunakan pada masalah maksimum berdasarkan metode dual simpleks. Masalah pada McDonald's dapat diselesaikan dengan menggunakan dual dari McDonald's sebagai berikut.

Maksimum $Z = Y_1 + Y_2 + \mathrm{Y}_3 = \frac{1}{v}$

dengan batasan

$$397\,Y_1 + 388\,Y_2 + 387\,\mathrm{Y}_3 \leq 1$$
$$386\,Y_1 + 426\,Y_2 + 390\,\mathrm{Y}_3 \leq 1$$
$$387\,Y_1 + 379\,Y_2 + 436\,\mathrm{Y}_3 \leq 1\,,$$
$$Y_1,\, Y_2, \mathrm{Y}_3 \geq 0$$

Berdasarkan masalah di atas, diperoleh bahwa masalah dual dari McDonald's merupakan masalah maksimum pada KFC. Oleh karena itu, masalah tersebut dapat diselesaikan dengan menggunakan metode simpleks. Langkah-langkah metode simpleks untuk McDonald's adalah sebagai berikut.

i) Mengubah masalah pemrograman linear menjadi bentuk standar pemrograman linear, yaitu dengan cara mengubah tanda "$\leq$" menjadi "$=$", dan menambahkan variabel slack (S).

Hasil bentuk standar masalah maksimum pemrograman linear tersebut adalah sebagai berikut.

Maksimum $Z = Y_1 + Y_2 + Y_3 = \frac{1}{V}$

dengan batasan

$$\begin{aligned} 397\,Y_1 + 388\,Y_2 + 387\,Y_3 + S_1 \qquad\quad &= 1 \\ 386\,Y_1 + 426\,Y_2 + 390\,Y_3 + \qquad S_2 \quad &= 1 \\ 370\,Y_1 + 379\,Y_2 + 436\,Y_3 + \qquad\quad S_3 &= 1\,, \\ Y_1, Y_2, Y_3 \geq 0 & \end{aligned}$$

Berikut adalah tabel simpleks awal McDonald's.

Tabel 4.2 Simpleks awal McDonald's.

C_b	C_j	1	1	1	0	0	0	Indeks	Rasio
	Basis	Y_1	Y_2	Y_3	S_1	S_2	S_3	(b_j)	
0	S_1	397	388	387	1	0	0	1	
0	S_2	386	426	390	0	1	0	1	
0	S_3	387	379	436	0	0	1	1	
$Z_j - C_j$		-1	-1	-1	0	0	0	$Z = 0$	

Nilai $Z_j - C_j$ diperoleh dari:

$$Z_1 - C_1 = \sum_{i=1}^{3} C_{bi} Y_{1i} - C_1 = (0 \times 397 + 0 \times 386 + 0 \times 387) - 1 = -1$$
$$Z_2 - C_2 = \sum_{i=1}^{3} C_{bi} Y_{2i} - C_2 = (0 \times 388 + 0 \times 426 + 0 \times 379) - 1 = -1$$
$$Z_3 - C_3 = \sum_{i=1}^{3} C_{bi} Y_{3i} - C_3 = (0 \times 387 + 0 \times 390 + 0 \times 436) - 1 = -1$$
$$Z_4 - C_4 = \sum_{i=1}^{3} C_{bi} S_{1i} - C_4 = (0 \times 1 + 0 \times 0 + 0 \times 0) - 0 = 0$$
$$Z_5 - C_5 = \sum_{i=1}^{3} C_{bi} S_{2i} - C_5 = (0 \times 0 + 0 \times 1 + 0 \times 0) - 0 = 0$$
$$Z_6 - C_6 = \sum_{i=1}^{3} C_{bi} S_{3i} - C_6 = (0 \times 0 + 0 \times 0 + 0 \times 1) - 0 = 0$$

Sementara itu, nilai Z diperoleh dari:

$$Z = \sum_{i=1}^{3} C_{bi} b_{ji} = 0 \times 1 + 0 \times 1 + 0 \times 1 = 0$$

ii) Melakukan iterasi dengan mencari kolom kunci, baris kunci, dan angka kunci (angka *pivot*).

Pada masalah maksimum, kolom kunci yang diambil berdasarkan nilai $Z_j - C_j$ paling negatif. Sementara itu, baris kunci yang diambil berdasarkan rasio terkecil.

Berdasarkan Tabel 4.2, kolom kunci yang di ambil (*entering variable*) adalah *Y*$_3$ karena mempunyai nilai $Z_j - C_j$ yang sama. Sementara itu, baris kunci yang diambil berdasarkan nilai rasio terkecil. Rasio diperoleh dengan menggunakan rumus sebagai berikut.

Rasio = indeks masing-masing baris $\div$ masing-masing koefisien kolom kunci baru

Berikut adalah perhitungan untuk rasio.

Rasio ke-1 $= \frac{1}{387} = 0{,}00258$

Rasio ke-2 $= \frac{1}{390} = 0{,}00256$

Rasio ke-3 $= \frac{1}{436} = 0{,}00229$

Dengan demikian, baris kunci yang diambil (*leaving variable*) adalah *S*$_3$, dan diperoleh angka kunci dengan nilai 436 yang merupakan pertemuan antara kolom kunci dan baris kunci. Berikut tabel simpleks hasil iterasi 1.

Tabel 4.3 Iterasi ke-1 McDonald's

C_b		C_j	1	1	1	0	0	0	Indeks (b_j)	Rasio
		Basis	Y_1	Y_2	Y_3	S_1	S_2	S_3		
0		S_1	397	388	387	1	0	0	1	0,00258
0		S_2	386	426	390	0	1	0	1	0,00256
0		S_3	387	379	436	0	0	1	1	0,00229
	$Z_j - C_j$		-1	-1	-1	0	0	0	$Z = 0$	

iii) Mengubah variabel keputusan pada baris kunci dengan variabel keputusan pada kolom kunci, lalu menentukan baris kunci baru, dan baris lainnya.

Variabel keputusan pada baris kunci, yaitu *S*$_3$ diubah menjadi *Y*$_3$ dan nilai *C*$_b$ menjadi 1. Selanjutnya menentukan baris kunci baru berdasarkan pada rumus sebagai berikut.

Baris kunci baru = baris kunci lama $\div$ angka kunci.

Baris kunci baru:

$$\begin{array}{ccccccc|l} 387 & 379 & 436 & 0 & 0 & 1 & 1 & \div 436 \\ \hline 0{,}88761 & 0{,}86926 & 1 & 0 & 0 & 0{,}00229 & 0{,}00229 & \end{array}$$

Sementara itu, untuk menentukan baris lainnya menggunakan rumus sebagai berikut.

Koefisien kolom kunci baru = koefisien kolom kunci pada baris awal – (koefisien kolom kunci pada baris awal $\times$ angka kunci $\div$ angka kunci)

Baris baru = baris awal – (baris kunci awal $\times$ koefisien kolom kunci pada baris awal $\div$ angka kunci awal).

Berikut adalah hasil perhitungan untuk baris baru lainnya.

Baris ke-1 :

$$\begin{array}{cccccc|c} 387 & 379 & 436 & 0 & 0 & 1 & 1 \\ 397 & 388 & 387 & 1 & 0 & 0 & 1 \\ \hline 53{,}49312 & 51{,}59404 & 0 & 1 & 0 & -0{,}88761 & 0{,}11239 \end{array}$$

Baris ke-2 :

$$\begin{array}{cccccc|c} 387 & 379 & 436 & 0 & 0 & 1 & 1 \\ 386 & 426 & 390 & 0 & 1 & 0 & 1 \\ \hline 39{,}83028 & 86{,}98624 & 0 & 0 & 1 & -0{,}89449 & 0{,}10551 \end{array}$$

Untuk menghitung nilai $Z_j - C_j$ dan Z dilakukan dengan cara yang sama seperti langkah i). Berikut adalah hasil perhitungan baris kunci baru dan baris lainnya dari iterasi ke-1.

Tabel 4.4 Hasil perhitungan baris kunci baru dan baris lainnya dari iterasi ke-1

C_b	C_j	1	1	1	0	0	0	Indeks (b_j)	Rasio
	Basis	Y_1	Y_2	Y_3	S_1	S_2	S_3		
0	S_1	53,49312	51,59404	0	1	0	-0,88761	0,11239	
0	S_2	39,83028	86,98624	0	0	1	-0,89449	0,10551	
1	Y_3	0,88761	0,86926	1	0	0	0,00229	0,00229	
$Z_j - C_j$		-0,11239	-0,13074	0	0	0	0,00229	$Z = 0{,}00229$	

iv) Memastikan nilai $Z_j - C_j$ tidak bernilai negatif sehingga diperoleh solusi optimal. Jika nilai $Z_j - C_j$ masih bernilai negatif, maka dilakukan iterasi selanjutnya seperti pada langkah ii) dan iii).

Berdasarkan Tabel 4.4, dapat disimpulkan bahwa nilai $Z_j - C_j$ masih ada yang bernilai negatif. Oleh karena itu, dilakukan iterasi selanjutnya seperti pada langkah ii) dan iii). Pada Tabel 4.7, kolom kunci yang diambil (*entering variable*) adalah Y_2. Sementara itu, baris kunci yang diambil berdasarkan nilai rasio terkecil. Untuk perhitungan rasio dapat dilakukan dengan cara yang sama pada langkah ii) sehingga baris kunci yang diambil (*leaving variable*) adalah S_2.

Setelah ditentukan kolom kunci, baris kunci, dan nilai rasio, maka diperoleh hasil iterasi ke-2 dengan nilai angka kunci, yaitu 20,87224 sebagai berikut.

Tabel 4.5 Iterasi ke-2 McDonald's

C_b	C_j	1	1	1	0	0	0	Indeks (b_j)	Rasio
	Basis	Y_1	Y_2	Y_3	S_1	S_2	S_3		
0	S_1	53,49312	51,59404	0	1	0	-0,88761	0,11239	0,00217
0	S_2	39,83028	86,98624	0	0	1	-0,89449	0,10551	0,00121
1	Y_3	0,88761	0,86926	1	0	0	0,00229	0,00229	0,00263
$Z_j - C_j$		-0,11239	-0,13074	0	0	0	0,00229	Z = 0,00229	

Berdasarkan Tabel 4.8, variabel keputusan pada baris kunci, yaitu S_2 diubah menjadi Y_2 dan nilai C_b menjadi 1. Selanjutnya menentukan nilai baris kunci baru, dan baris baru lainnya.

Baris kunci baru:

$$\begin{array}{cccccc|c} 39{,}83028 & 86{,}98624 & 0 & 0 & 1 & -0{,}89449 & 0{,}10551 \\ \hline 0{,}45789 & 1 & 0 & 0 & 0{,}01149 & 0{,}01028 & 0{,}00121 \end{array} \div 86{,}98624$$

Baris ke-1:

$$\begin{array}{cccccc|c} 39{,}83028 & 86{,}98624 & 0 & 0 & 1 & -0{,}89449 & 0{,}10551 \\ 53{,}49312 & 51{,}59404 & 0 & 1 & 0 & -0{,}88761 & 0{,}11239 \\ \hline 29{,}86864 & 0 & 0 & 1 & -0{,}59312 & -0{,}35707 & 0{,}04981 \end{array}$$

Baris ke-3 :

$$
\begin{array}{cccccc|c}
39{,}83028 & 86{,}98624 & 0 & 0 & 1 & -0{,}89449 & 0{,}10551 \\
0{,}88761 & 0{,}86926 & 1 & 0 & 0 & 0{,}00229 & 0{,}00229 \\
\hline
0{,}48959 & 0 & 1 & 0 & -0{,}00893 & 0{,}01122 & 0{,}00124
\end{array}
$$

Untuk menghitung nilai $Z_j - C_j$ dan Z dilakukan dengan cara yang sama seperti langkah i). Berikut adalah hasil perhitungan baris kunci baru dan baris lainnya dari iterasi ke-2.

Tabel 4.6 Hasil perhitungan baris kunci baru dan baris lainnya dari iterasi ke-2 McDonald's

C_b	C_j	1	1	1	0	0	0	Indeks (b_j)	Rasio
	Basis	Y_1	Y_2	Y_3	S_1	S_2	S_3		
0	S_1	29,86864	0	0	1	-0,59312	-0,35707	0,04981	0,00166
1	Y_2	0,45789	1	0	0	0,01149	-0,01028	0,00121	0,00264
1	Y_3	0,48959	0	1	0	-0,00893	0,01122	0,00124	0.00253
$Z_j - C_j$		-0,05252	0	0	0	-0,07781	-0,00086	Z = 0,00245	

Berdasarkan Tabel 4.6, dapat disimpulkan bahwa nilai $Z_j - C_j$ masih ada yang bernilai negatif sehingga dilakukan iterasi selanjutnya. Pada Tabel 4.6, kolom kunci yang diambil (*entering variable*) adalah Y_1. Sementara itu, baris kunci yang diambil berdasarkan nilai rasio positif terkecil. Untuk perhitungan rasio dapat dilakukan dengan cara yang sama pada langkah ii) sehingga baris kunci yang diambil (*leaving variable*) adalah S_1.

Tabel 4.7 Iterasi ke-3 McDonald's

C_b	C_j	1	1	1	0	0	0	Indeks (b_j)	Rasio
	Basis	Y_1	Y_2	Y_3	S_1	S_2	S_3		
0	S_1	29,86864	0	0	1	-0,59312	-0,35707	0,04981	0,00166
1	Y_2	0,45789	1	0	0	0,01149	-0,01028	0,00121	0,00264
1	Y_3	0,48959	0	1	0	-0,00893	0,01122	0,00124	0.00253
$Z_j - C_j$		-0,05252	0	0	0	-0,07781	-0,00086	Z = 0,00245	

Setelah ditentukan kolom kunci, baris kunci, dan nilai rasio, maka diperoleh hasil iterasi ke-3 dengan nilai angka kunci, yaitu 29,86864 seperti pada Tabel 4.7.

Berdasarkan Tabel 4.7, variabel keputusan pada baris kunci, yaitu S_1 diubah menjadi Y_1 dan nilai C_b menjadi 1. Selanjutnya menentukan nilai baris kunci baru, dan baris baru lainnya.

Baris kunci baru:

$$\begin{array}{cccccc|c} 29{,}86864 & 0 & 0 & 1 & -0{,}59312 & -0{,}35707 & 0{,}04981 \\ \hline 1 & 0 & 0 & 0{,}03347 & -0.01985 & -0{,}01195 & 0{,}00167 \end{array} \div 29{,}86864$$

Baris ke-2 :

$$\begin{array}{cccccc|c} 29{,}86864 & 0 & 0 & 1 & -0{,}59312 & -0{,}35707 & 0{,}04981 \\ 0{,}45789 & 1 & 0 & 0 & 0{,}01149 & -0{,}01028 & 0{,}00121 \\ \hline 0 & 1 & 0 & -0{,}01533 & 0{,}02058 & -0{,}00481 & 0{,}00045 \end{array}$$

Baris ke-3 :

$$\begin{array}{cccccc|c} 29{,}86864 & 0 & 0 & 1 & -0{,}59312 & -0{,}35707 & 0{,}04981 \\ 0{,}48959 & 0 & 1 & 0 & -0{,}00893 & 0{,}01122 & 0{,}00124 \\ \hline 0 & 0 & 1 & -0{,}01639 & -0{,}00027 & 0{,}01707 & 0.00042 \end{array}$$

Untuk menghitung nilai $Z_j - C_j$ dan Z dilakukan dengan cara yang 0sama seperti langkah 1. Berikut adalah hasil perhitungan baris kunci baru dan baris lainnya dari iterasi ke-3.

Tabel 4.8 Hasil perhitungan baris kunci baru dan baris lainnya dari iterasi ke-3 McDonald's

C_b	C_j	1	1	1	0	0	0	Indeks (b_j)	Rasio
	Basis	Y_1	Y_2	Y_3	S_1	S_2	S_3		
1	Y_1	1	0	0	0,03347	-0,01985	-0,01195	0,00167	
1	Y_2	0	1	0	-0,01533	0,02058	-0,00481	0,00045	
1	Y_3	0	0	1	-0,01639	-0,00027	0,01707	0,00042	
$Z_j - C_j$		0	0	0	0,00176	0,00046	0,00032	$Z = 0{,}00254$	

Karena nilai $Z_j - C_j$ sudah tidak ada yang bernilai negatif, maka tahap iterasi selesai dengan diperoleh hasil akhir iterasi simpleks, yaitu iterasi ke-4 McDonald's sebagai berikut.

Tabel 4.9 Iterasi ke-4 McDonald's

C_b	C_j	1	1	1	0	0	0	Indeks (b_j)	Rasio
	Basis	Y_1	Y_2	Y_3	S_1	S_2	S_3		
1	Y_1	1	0	0	0,03347	-0,01985	-0,01195	0,00167	-
1	Y_2	0	1	0	-0,01533	0,02058	-0,00481	0,00045	-
1	Y_3	0	0	1	-0,01639	-0,00027	0,01707	0,00042	-
$Z_j - C_j$		0	0	0	0,00176	0,00046	0,00032	$Z = 0{,}00254$	

5. Menentukan strategi optimal masing - masing pemain.

Berdasarkan Tabel 4.9, seluruh baris dan kolom dapat diabaikan sehingga strategi optimal untuk KFC, yaitu: $Y_1 = 0{,}00167$, $Y_2 = 0{,}00045$, $Y_3 = 0{,}00042$ dengan $Y_1 = q_1$, $Y_2 = q_2$, $Y_3 = q_4$, dan $Z = 0{,}00254$. Peluang strategi optimal untuk KFC adalah sebagai berikut.

Peluang $q_1 = \frac{Y_1}{Z} = \frac{0{,}00167}{0{,}00254} = 0{,}65748 = 65{,}748\%$

Peluang $q_2 = \frac{Y_2}{Z} = \frac{0{,}00045}{0{,}00254} = 0{,}17716 = 17{,}716\%$

Peluang $q_4 = \frac{Y_3}{Z} = \frac{0{,}00042}{0{,}00254} = 0{,}16535 = 16{,}535\%$

dengan nilai permainan $V = \frac{1}{Z} \div k = \frac{1}{0{,}00254} \div 100 = 3{,}9370$

Sementara itu, strategi optimal untuk McDonald's merupakan dual dari KFC, sehingga diperoleh $X_1 = 0{,}00176$, $X_2 = 0{,}00046$, $X_3 = 0{,}00032$, dan $Z = W = 0{,}00254$. Peluang strategi optimal untuk McDonald's adalah sebagai berikut.

Peluang $p_1 = \frac{X_1}{W} = \frac{0{,}00176}{0{,}00254} = 0{,}69291 = 69{,}291\%$

Peluang $p_2 = \frac{X_2}{W} = \frac{0{,}00046}{0{,}00254} = 0{,}18110 = 18{,}110\%$

Peluang $p_4 = \frac{X_3}{W} = \frac{0{,}00032}{0{,}00254} = 0{,}12598 = 12{,}598\%$

dengan nilai permainan $V = \frac{1}{W} \div k = \frac{1}{0{,}00254} \div 100 = 3{,}9370$.

4.2.1.2 Penyelesaian matriks *payoff* dengan menggunakan algoritma *brown*.

Langkah - langkah untuk menyelesaikan matriks *payoff* McDonald's menggunakan algoritma *brown* adalah sebagai berikut.

1. Menentukan *saddle point*.

Penentuan *saddle point* pada matriks *payoff* McDonald's terdapat pada Tabel 4.1.

2. Penyelesaian matriks *payoff* dengan menggunakan algoritma *brown*, karena tidak memiliki *saddle point*.

Berikut adalah penyelesaian matriks *payoff* McDonald's menggunakan algoritma *brown*.

$$\begin{array}{cc} & \text{KFC} \\ & \begin{array}{ccccc} q_1 & q_2 & q_3 & q_4 & q_5 \end{array} \\ \text{MCD} \begin{array}{c} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \end{array} & \begin{bmatrix} 3{,}97 & 3{,}88 & 3{,}99 & 3{,}87 & 3{,}92 \\ 3{,}86 & 4{,}26 & 4{,}08 & 3{,}90 & 4{,}31 \\ 3{,}77 & 3{,}91 & 4{,}06 & 3{,}88 & 4{,}33 \\ 3{,}87 & 3{,}79 & 3{,}87 & 4{,}36 & 4{,}24 \\ 3{,}76 & 4{,}10 & 3{,}81 & 3{,}88 & 4{,}03 \end{bmatrix} \end{array}$$

Algoritma *Brown* unntuk menyelesaikan permainan ini ialah dengan melakukan beberapa langkah seperti dibawah ini :

a) Pemain I atau McDonald's memilih baris p_5 untuk dimainkan, baris p_5 adalah strategi baris dengan nilai terkecil kemudian pemain II atau KFC memainkan kolom yang terkait dengan elemen minimum pada baris tersebut.

b) Selanjutnya pemain I atau McDonald's memilih baris untuk dimainkan yang terkait terhadap elemen maksimum pada kolom yang dimainkan oleh pemain II atau KFC pada langkah 1.

c) Pemain II atau KFC menjumlahkan baris yang telah dimainkan pemain I atau McDonald's sejauh itu dan memainkan kolom yang terkait terhadap jumlah elemen minimum.

d) Pemain I atau McDonald's menjumlahkan kolom yang telah dimainkan pemain II atau KFC sejauh itu dan memainkan baris yang terkait terhadap suatu jumlah elemen maksimum. Jika jumlah iterasi yang digunakan terpenuhi lanjut ke langkah 5. Sebaliknya, jika jumlah iterasi tidak terpenuhi kembali ke langkah 3.

e) Hitung batas atas $\overline{v}$ dan batas bawah $\underline{v}$ berturut-turut.

   i. $\overline{v} = \dfrac{jumlah\ elemen\ maksimum\ dari\ langkah\ ke\ 4}{banyaknya\ memainkan\ permainan}$

   ii. $\underline{v} = \dfrac{jumlah\ elemen\ maksimum\ dari\ langkah\ ke\ 3}{banyaknya\ memainkan\ permainan}$

Tabel 4.10 Penyelesaian menggunakan algoritma *brown* pada matriks *payoff* McDonald's antara McDonald's dan KFC

$$
\begin{array}{c}
 & \text{KFC} \\
\text{MCD} & \begin{bmatrix} 3{,}97 & 3{,}88 & 3{,}99 & 3{,}87 & 3{,}92 \\ 3{,}86 & 4{,}26 & 4{,}08 & 3{,}90 & 4{,}31 \\ 3{,}77 & 3.91 & 4{,}06 & 3{,}88 & 4{,}33 \\ 3{,}87 & 3{,}79 & 3{,}87 & 4{,}36 & 4{,}24 \\ 3{,}76 & 4{,}10 & 3{,}81 & 3{,}88 & 4{,}03 \end{bmatrix}
\end{array}
$$

KFC

1	2	3	4	5	6	7	8	9	10
3,97	7,94	11,81	15,78	19,66	23,54	27,51	31,48	35,45	39,42
3,86	7,72	11,62	15,48	19,74	24,00	27,86	31,72	35,58	39,44
3,77	7,54	11,42	15,19	19,10	23,01	26,78	30,55	34,32	38,09
3,87	7,74	12,10	15,97	19,76	23,55	27,42	31,29	35,16	39,03
3,76	7,52	11,40	15,16	19,26	23,36	27,12	30,88	34,64	38,40

1	3,76	4,10	3,81	3,88	4,03
2	7,73	7,98	7,80	7,75	7,95
3	11,70	11,86	11,79	11,62	11,87
4	15,57	15,65	15,66	15,98	16,11
5	19,44	19,44	19,53	20,34	20,35
6	23,31	23,23	23,40	24,70	24,59
7	27,17	27,49	27,48	28,60	28,90
8	31,03	31,75	31,56	32,50	33,21
9	34,89	36,01	35,64	36,40	37,52
10	38,75	40,27	39,72	40,30	41,83

Dalam pengerjaan algoritma *brown* digunakan sepuluh iterasi agar nilai permainan pada metode simpleks berada pada interval algoritma *brown*.

3. Menentukan strategi optimal masing-masing pemain.

Berdasarkan Tabel 4.10, didapat matriks p^* dan q^* sebagai berikut.

$$p^* = \begin{bmatrix} 0{,}2 \\ 0{,}5 \\ 0 \\ 0{,}3 \\ 0 \end{bmatrix} \; dan \; q^* = \begin{bmatrix} 0{,}7 & 0{,}2 & 0 & 0{,}1 & 0 \end{bmatrix}$$

Nilai batas atas matriks *payoff* McDonald's adalah :

$$\overline{v} = \frac{jumlah\ elemen\ maksimum\ dari\ langkah\ ke\ 4}{banyaknya\ memainkan\ permainan}$$

$$\overline{v} = \frac{39{,}44}{10} = 3{,}944$$

Nilai batas bawah matriks *payoff* McDonald's adalah :

$$\underline{v} = \frac{jumlah\ elemen\ maksimum\ dari\ langkah\ ke\ 3}{banyaknya\ memainkan\ permainan}$$

$$\underline{v} = \frac{38{,}75}{10} = 3{,}875$$

Atau bisa ditulis $3{,}875 \leq V \leq 3{,}944$.

Berdasarkan hasil perhitungan dengan metode simpleks dan algoritma *brown*, dapat disimpulkan bahwa nilai permainan untuk McDonald's dan KFC berdasarkan matriks *payoff* McDonald's menggunakan metode simpleks adalah $V = 3{,}9370$ berada pada interval algoritma brown yaitu $3{,}875 \leq V \leq 3{,}944$. Strategi optimal untuk KFC adalah q_1, q_2 dan q_4 atau strategi harga, strategi lokasi dan strategi variasi produk dengan peluang strategi harga sebesar 65,748 %, strategi lokasi sebesar 17,716 % dan strategi variasi produk sebesar 16,535 % untuk diprioritaskan dalam meminimumkan kerugiannya. Kerugian sebelumnya, yaitu 3,97 dapat diminimumkan menjadi 3,9370. Sementara itu, strategi optimal untuk McDonald's adalah p_1, p_2 dan p_4 atau strategi harga, strategi lokasi dan strategi variasi produk dengan peluang strategi harga sebesar 69,291 %, peluang strategi lokasi sebesar 18,110 % dan peluang strategi variasi produk sebesar 12,598 % untuk diprioritaskan dalam memaksimumkan keuntungannya. Keuntungan sebelumnya, yaitu 3,87 dapat dimaksimumkan menjadi 3,9370.

4.2.2 Penyelesaian matriks *PayOff* KFC dengan software MATLAB

Langkah - langkah untuk menyelesaikan matriks *payoff* KFC adalah sebagai berikut.

4.2.2.1 Penyelesaian matriks *payoff* dengan menggunakan metode simpleks

1. Masukan Matriks *payoff* KFC

Masukan matriks yang diperoleh dari kuisioner pada lampiran 3 seperti pada Gambar 4.1.

```
A =

    3.6400    3.3800    3.3300    3.0300    3.2400
    3.5500    3.4900    3.4000    3.1500    3.3800
    3.3300    3.5800    3.4900    3.3800    3.2100
    3.0000    3.1900    3.5800    3.4500    3.3300
    3.2400    3.2600    3.4500    3.5900    3.3300
```

Gambar 4.1 Matriks *payoff* KFC

2. Menentukan *saddle point*

Hasil *saddle point* dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.2.

```
minimaks =

    3.3800


maksimin =

    3.2400
```

Gambar 4.2 Nilai minimaks dan maksimin matriks *payoff* KFC

Berdasarkan Gambar 4.2 diperoleh nilai maksimin, 3,24 sebagai batas bawah permainan ($\underline{V}$). Sementara itu, nilai minimaks pada matriks *payoff* tersebut adalah 3,38 sebagai batas atas permainan ($\overline{V}$), karena ($\underline{V}$) ≠ ($\overline{V}$), maka dapat disimpulkan bahwa tidak terdapat *saddle point* pada matriks *payoff* KFC.

3. Penyelesaian matriks *payoff* menggunakan dominasi

Hasil dominasi dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.3.

```
Matriks_hasil_dominasi =

    3.5500    3.1500    3.3800
    3.3300    3.3800    3.2100
    3.2400    3.5900    3.3300
```

Gambar 4.3 Hasil dominasi matriks *payoff* KFC

Dari Gambar 4.3 didapat hasil bentuk standar masalah maksimum pemrograman linear tersebut adalah sebagai berikut.

Maksimum $Z = Y_1 + Y_2 + Y_3 = \frac{1}{V}$

dengan batasan

$$3{,}55\,Y_1 + 3{,}15\,Y_2 + 3{,}38\,Y_3 + S_1 = 1$$
$$3{,}33\,Y_1 + 3{,}38\,Y_2 + 3{,}21\,Y_3 + S_2 = 1$$
$$3{,}24\,Y_1 + 3{,}59\,Y_2 + 3{,}33\,Y_3 + S_3 = 1\,,$$
$$Y_1, Y_2, Y_3 \geq 0$$

4. Penyelesaian matriks *payoff* dengan metode simpleks

Untuk penyelesaian masalah maksimum pada KFC digunakan metode simpleks. Hasil iterasi metode simpleks menggunakan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.4 (a) – 4.4 (c).

```
ITERASI 0

---------------------------------------------------------------|------------------|
|       |  Cj   |    1       1       1       0       0       0       |          |
|  Cb   |-------|------------------------------------------------| Indeks  |  Rasio  |
|       | Basis |   Y1      Y2      Y3      S1      S2      S3      |          |
---------------------------------------------------------------|------------------|
|  0    |  S1   |   3.550  3.150  3.380  1.000  0.000  0.000    |  1.000  | 0.296  |
|  0    |  S2   |   3.330  3.380  3.210  0.000  1.000  0.000    |  1.000  | 0.312  |
|  0    |  S3   |   3.240  3.590  3.330  0.000  0.000  1.000    |  1.000  | 0.300  |
---------------------------------------------------------------|------------------|
|    Zj-cj      |   -1.000  -1.000  -1.000  -0.000  -0.000  -0.000    |     Z =0.000
---------------------------------------------------------------|------------------|
Zj-Cj Maksimum = 1.000
Rasio Minimum = 0.296
Kolom kunci    = 3
Baris kunci    = 1
Elemen Pivot   = 3.380
```

Gambar 4.4 (a) Iterasi ke-0 matriks *payoff* KFC

```
ITERASI 1

---------------------------------------------------------------|------------------|
|       |  Cj   |    1       1       1       0       0       0       |          |
|  Cb   |-------|------------------------------------------------| Indeks  |  Rasio  |
|       | Basis |   Y1      Y2      Y3      S1      S2      S3      |          |
---------------------------------------------------------------|------------------|
|  1    |  Y3   |   1.050  0.932  1.000  0.296  0.000  0.000    |  0.296  | 0.317  |
|  0    |  S2   |   -0.041  0.388  0.000  -0.950  1.000  0.000    |  0.050  | 0.129
|  0    |  S3   |   -0.257  0.487  0.000  -0.985  0.000  1.000    |  0.015  | 0.030
---------------------------------------------------------------|------------------|
|    Zj-cj      |   0.050  -0.068  -0.000  0.296  -0.000  -0.000    |     Z =0.296
---------------------------------------------------------------|------------------|
Zj-Cj Maksimum = 0.068
Rasio Minimum = 0.030
Kolom kunci    = 2
Baris kunci    = 3
Elemen Pivot   = 0.487
```

Gambar 4.4 (b) Iterasi ke-1 matriks *payoff* KFC

```
ITERASI 2

---------------------------------------------------------------|------------------|
|       |  Cj   |    1       1       1       0       0       0       |          |
|  Cb   |-------|------------------------------------------------| Indeks  |  Rasio  |
|       | Basis |   Y1      Y2      Y3      S1      S2      S3      |          |
---------------------------------------------------------------|------------------|
|  1    |  Y3   |   1.543  0.000  1.000  2.183  0.000  -1.915    |  0.268  | Inf  |
|  0    |  S2   |   0.164  0.000  0.000  -0.163  1.000  -0.798    |  0.038  | 0.038  |
|  1    |  Y2   |   -0.529  1.000  0.000  -2.025  0.000  2.055    |  0.030  | Inf  |
---------------------------------------------------------------|------------------|
|    Zj-cj      |   0.014  -0.000  -0.000  0.158  -0.000  0.140    |     Z =0.298
---------------------------------------------------------------|------------------|
Zj-Cj Maksimum = 0.000
Rasio Minimum = 0.038
Kolom kunci    = 5
Baris kunci    = 2
Elemen Pivot   = 1.000


Z Maksimum yaitu Z= 0.298
>>
```

Gambar 4.4 (c) Iterasi ke-2 matriks *payoff* KFC

Karena nilai $Z_j - C_j$ sudah tidak ada yang bernilai negatif, maka tahap itetasi selesai dengan diperoleh hasil akhir iterasi simpleks, yaitu iterasi ke-2 KFC.

5. Menentukan strategi optimal masing-masing pemain

Berdasarkan Gambar 4.4 (c), seluruh baris dan kolom dapat diabaikan sehingga strategi optimal untuk KFC, yaitu: $Y_1 = 0$, $Y_2 = 0{,}030$, $Y_3 = 0{,}268$ dengan $Y_1 = q_1$, $Y_2 = q_4$, $Y_3 = q_5$, dan $Z = 0{,}298$. Peluang strategi optimal untuk KFC adalah sebagai berikut.

Peluang $q_1 = \frac{Y_1}{Z} = \frac{0}{0{,}298} = 0 = 0\%$

Peluang $q_4 = \frac{Y_2}{Z} = \frac{0{,}030}{0{,}298} = 0{,}1006 = 10{,}06\%$

Peluang $q_5 = \frac{Y_3}{Z} = \frac{0{,}268}{0{,}298} = 0{,}8993 = 89{,}93\%$

dengan nilai permainan $V = \frac{1}{Z} = \frac{1}{0{,}298} = 3{,}355$

Sementara itu, strategi optimal untuk McDonald's, diperoleh $X_1 = 0{,}158$, $X_2 = 0$, $X_3 = 0{,}140$, dan $Z = W = 0{,}298$. Peluang strategi optimal untuk McDonald's adalah sebagai berikut.

Peluang $p_2 = \frac{X_1}{W} = \frac{0{,}158}{0{,}298} = 0{,}530 = 53{,}0\%$

Peluang $p_3 = \frac{X_2}{W} = \frac{0}{0{,}298} = 0 = 0\%$

Peluang $p_5 = \frac{X_3}{W} = \frac{0{,}140}{0{,}298} = 0{,}469 = 46{,}9\%$

dengan nilai permainan $V = \frac{1}{W} = \frac{1}{0{,}298} = 3{,}355$.

4.2.2.2 Penyelesaian matriks *payoff* dengan menggunakan algoritma *brown* dengan software MATLAB

Penyelesaian matriks *payoff* KFC menggunakan algoritma *brown* akan dilakukan dengan program Matlab. Berikut adalah langkah – langkah penyelesaian matriks *payoff* KFC.

1. Masukan Matriks

Masukan matriks yang diperoleh dari kuisioner pada lampiran 3 seperti pada Gambar 4.1.

2. Menentukan *saddle point*

Hasil *saddle point* dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.2.

3. Menyelesaikan matriks *payoff* dengan menggunakan algoritma *brown.*

Penyelesaian dengan menggunakan algoritma *brown* dapat dilihat pada Gambar 4.5.

```
baris_iterasi =

    3.0000    3.1900    3.5800    3.4500    3.3300
    6.6400    6.5700    6.9100    6.4800    6.5700
    9.8800    9.8300   10.3600   10.0700    9.9000
   13.2100   13.4100   13.8500   13.4500   13.1100
   16.7600   16.9000   17.2500   16.6000   16.4900
   20.3100   20.3900   20.6500   19.7500   19.8700
   23.5500   23.6500   24.1000   23.3400   23.2000
   26.7900   26.9100   27.5500   26.9300   26.5300
   30.0300   30.1700   31.0000   30.5200   29.8600
   33.2700   33.4300   34.4500   34.1100   33.1900


nilai_terkecil_dari_baris =

   33.1900


kolom_iterasi =

    3.6400    6.6700   10.0500   13.2900   16.5300   19.5600   22.8000   26.0400   29.2800   32.5200
    3.5500    6.7000   10.1900   13.5700   16.9500   20.1000   23.4800   26.8600   30.2400   33.6200
    3.3300    6.7100   10.2900   13.5000   16.7100   20.0900   23.3000   26.5100   29.7200   32.9300
    3.0000    6.4500    9.6400   12.9700   16.3000   19.7500   23.0800   26.4100   29.7400   33.0700
    3.2400    6.8300   10.0900   13.4200   16.7500   20.3400   23.6700   27.0000   30.3300   33.6600
```

Gambar 4.5 Iterasi algoritma *brown* matriks *payoff* KFC

Dari iterasi Gambar 4.5 akan didapat nilai batas atas ($\overline{V}$) dan batas bawah ($\underline{V}$), yaitu dimana ($\underline{V}$) ≤ V ≤ ($\overline{V}$), seperti Gambar 4.6

```
batasatas =

    3.3660


batasbawah =

    3.3190
```

Gambar 4.6 Hasil batas atas dan batas bawah matriks *payoff* KFC

Berdasarkan hasil perhitungan dengan menggunakan metode simpleks dan algoritma *brown*, dapat disimpulkan bahwa nilai permainan untuk McDonald's dan KFC berdasarkan matriks *payoff* KFC menggunakan metode simpleks adalah $V = 3{,}355$ berada pada interval algoritma brown yaitu $3{,}319 \leq V \leq 3{,}366$. Strategi optimal untuk KFC adalah $q_{1,}$ q_4 dan q_5 atau strategi harga, strategi variasi produk dan strategi pelayanan dengan peluang strategi harga sebesar 0 %, strategi variasi produk sebesar 10,06 % dan strategi pelayanan sebesar 89,93 % untuk diprioritaskan dalam meminimumkan kerugiannya. Kerugian sebelumnya, yaitu 3,38 dapat diminimumkan menjadi 3,355. Sementara itu, strategi optimal untuk McDonald's adalah p_2, p_3 dan p_5 atau strategi lokasi, strategi promo dan strategi pelayanan dengan peluang strategi lokasi sebesar 53,0 %, peluang strategi promo sebesar 0 % dan peluang strategi pelayanan sebesar 46,9 % untuk diprioritaskan dalam memaksimumkan keuntungannya. Keuntungan sebelumnya, yaitu 3,24 dapat dimaksimumkan menjadi 3,355.

4.3 Optimalisasi Strategi McDonald's dan Richeese Favtory

4.3.1 Penyelesaian matriks *payoff* McDonald's

Untuk memperoleh strategi optimal dari McDonald's dan Richeese Factory dengan menggunakan metode simpleks dan algoritma *brown* diselesaikan dengan cara yang sama menggunakan software MATLAB. Langkah-langkah pengerjaan dapat dilihat pada Lampiran 4. Nilai permainan untuk McDonald's dan Richeese Factory berdasarkan matriks *payoff* McDonald's menggunakan metode simpleks adalah $V = 4{,}081$ berada pada interval algoritma brown yaitu $4{,}0130 \leq V \leq 4{,}0930$. Strategi optimal untuk Richeese Factory adalah $q_{1,}$ q_2 dan q_4 atau strategi harga, strategi lokasi dan strategi variasi produk dengan peluang strategi harga sebesar 70,20 %, strategi lokasi sebesar 29,79 % dan strategi variasi produk sebesar 0 % untuk diprioritaskan dalam meminimumkan kerugiannya. Kerugian sebelumnya, yaitu 4,17 dapat diminimumkan menjadi 4,081. Sementara itu, strategi optimal untuk McDonald's adalah p_2, p_3 dan p_4 atau strategi lokasi, strategi promo dan strategi variasi produk dengan peluang strategi lokasi sebesar 40,40 %, peluang strategi promo sebesar 0 % dan peluang strategi variasi produk sebesar

59,59 % untuk diprioritaskan dalam memaksimumkan keuntungannya. Keuntungan sebelumnya, yaitu 3,97 dapat dimaksimumkan menjadi 4,081.

4.3.2 Penyelesaian matriks *Pay-Off* Richeese Factory

Untuk memperoleh strategi optimal dari McDonald's dan Richeese Factory dengan menggunakan metode simpleks dan algoritma *brown* diselesaikan dengan cara yang sama menggunakan software MATLAB. Langkah-langkah pengerjaan dapat dilihat pada Lampiran 5. Nilai permainan untuk McDonald's dan Richeese Factory berdasarkan matriks *payoff* Richeese Factory menggunakan metode simpleks adalah $V = 3{,}558$ berada pada interval algoritma brown yaitu $3{,}5290 \leq V \leq 3{,}6240$. Strategi optimal untuk Richeese Factory adalah $q_{1,}$ q_4 dan q_5 atau strategi harga, strategi variasi produk dan strategi pelayanan dengan peluang strategi harga sebesar 34,51 %, strategi variasi produk sebesar 65,48 % dan strategi pelayanan sebesar 0 % untuk diprioritaskan dalam meminimumkan kerugiannya. Kerugian sebelumnya, yaitu 3,77 dapat diminimumkan menjadi 3,558. Sementara itu, strategi optimal untuk McDonald's adalah p_2, p_3 dan p_5 atau strategi lokasi, strategi promo dan strategi pelayanan dengan peluang strategi lokasi sebesar 72,95 %, peluang strategi promo sebesar 0 % dan peluang strategi pelayanan sebesar 27,04 % untuk diprioritaskan dalam memaksimumkan keuntungannya. Keuntungan sebelumnya, yaitu 3,47 dapat dimaksimumkan menjadi 3,558.

4.4 Optimalisasi Strategi KFC, dan Richeese Favtory

4.4.1 Penyelesaian matriks *Pay-Off* KFC

Untuk memperoleh strategi optimal dari KFC dan Richeese Factory dengan menggunakan metode simpleks dan algoritma *brown* diselesaikan dengan cara yang sama menggunakan software MATLAB. Langkah-langkah pengerjaan dapat dilihat pada Lampiran 6. Nilai permainan untuk KFC dan Richeese Factory berdasarkan matriks *payoff* KFC menggunakan metode simpleks adalah $V = 4{,}016$ berada pada interval algoritma brown yaitu $3{,}9680 \leq V \leq 4{,}1270$. Strategi optimal untuk Richeese Factory adalah $q_{2,}$ q_3 dan q_4 atau strategi lokasi, strategi promo dan strategi variasi produk dengan peluang strategi lokasi sebesar 18,07%, strategi promo sebesar 81,92 % dan strategi variasi produk sebesar 0 % untuk diprioritaskan dalam meminimumkan kerugiannya. Kerugian sebelumnya, yaitu 4,09 dapat diminimumkan menjadi 4,016. Sementara itu, strategi optimal untuk KFC adalah p_2, p_3 dan p_5 atau strategi lokasi, strategi promo dan strategi pelayanan dengan peluang strategi lokasi sebesar 0 %, peluang strategi promo sebesar 18,07 % dan peluang strategi pelayanan sebesar 81,92 % untuk diprioritaskan dalam memaksimumkan keuntungannya. Keuntungan sebelumnya, yaitu 4 dapat dimaksimumkan menjadi 4,016.

4.4.2 Penyelesaian matriks *Pay-Off* Richeese Factory

Untuk memperoleh strategi optimal dari KFC dan Richeese Factory dengan menggunakan metode simpleks dan algoritma *brown* diselesaikan dengan cara yang sama menggunakan software MATLAB. Langkah-langkah pengerjaan dapat dilihat pada Lampiran 7. Nilai permainan untuk KFC dan Richeese Factory berdasarkan matriks *payoff* Richeese Factory menggunakan metode simpleks adalah $V = 3{,}584$ berada pada interval algoritma brown yaitu $3{,}5780 \leq V \leq 3{,}6740$. Strategi optimal untuk Richeese Factory adalah $q_{1,}$ q_3 dan q_4 atau strategi harga, strategi promo dan strategi variasi produk dengan peluang strategi harga sebesar 38,35 %, strategi promo sebesar 0 % dan strategi variasi produk sebesar 61,64 % untuk diprioritaskan dalam meminimumkan kerugiannya. Kerugian sebelumnya, yaitu 3,73 dapat diminimumkan menjadi 3,584. Sementara itu, strategi optimal untuk KFC adalah p_1, p_2 dan p_4 atau strategi harga, strategi lokasi dan strategi variasi produk dengan peluang strategi harga sebesar 0 %, peluang strategi lokasi sebesar 61,64 % dan peluang strategi variasi produk sebesar 38,35 % untuk diprioritaskan dalam memaksimumkan keuntungannya. Keuntungan sebelumnya, yaitu 3,32 dapat dimaksimumkan menjadi 3,584.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan tujuan pembahasan skripsi ini, dapat diambil kesimpulan sebagai berikut.

1. Metode simpleks dan algoritma *brown* dapat diterapkan pada optimalisasi strategi pemasaran McDonald's, KFC dan Richeese Factory.
2. Semua nilai permainan pada metode simpleks berada pada interval algoritma *brown*. Strategi optimal permainan yang yang paling diminati oleh responden adalah perbandingan McDonald's dan KFC. Setelah melakukan perhitungan dengan menggunakan metode simpleks dan algoritma *brown* dari matriks *payoff* McDonald's didapat strategi optimal untuk KFC adalah $q_{1,}$ q_2 dan q_4 atau strategi harga, strategi lokasi dan strategi variasi produk dengan peluang strategi harga sebesar 65,748 %, strategi lokasi sebesar 17,716 % dan strategi variasi produk sebesar 16,535 % untuk diprioritaskan dalam meminimumkan kerugiannya. Kerugian sebelumnya, yaitu 3,97 dapat diminimumkan menjadi 3,9370. Sementara itu, strategi optimal untuk McDonald's adalah p_1, p_2 dan p_4 atau strategi harga, strategi lokasi dan strategi variasi produk dengan peluang strategi harga sebesar 69,291 %, peluang strategi lokasi sebesar 18,110 % dan peluang strategi variasi produk sebesar 12,598 % untuk diprioritaskan dalam memaksimumkan keuntungannya. Keuntungan sebelumnya, yaitu 3,87 dapat dimaksimumkan menjadi 3,9370.

5.2 Saran

Skripsi ini membahas tentang aplikasi teori permainan yang diselesaikan dengan metode simpleks menggunakan program MATLAB. Pada penelitian selanjutnya disarankan untuk memaksimalkan strategi yang diperlukan dan dapat digunakan metode lain misalnya keseimbangan NASH menggunakan program MATLAB.

DAFTAR PUSTAKA

Agung, A. A. P. 2012. *Metodologi Penelitian Bisnis*. UB Press: Malang.

Bertram, P. 1975. *Fast Food Operations*. Berry and Jenkins: London.

Fatchiyah, N. 2011. *Aplikasi Matriks dalam Teori Permainan untuk Menentukan Strategi Pemasaran*, Skripsi. Universitas Islam Negeri Maulana Malik Ibrahim: Malang. 1990

Ghadle, K.P. dan T.S. Pawar. 2014. Game Theory Problems by An Alternative Simplex Method. *International Journal of Research in Engineering and Technology*. 3: 900-905.

Gillet, B.L. 1976. *Introduction to Operation Research: A Computer Oriented Algorithmic Approach*. New York: McGraw-Hill Inc

Irianto, K. 2007. *Gizi dan Pola Hidup Sehat*. Yrama Widya: Bandung

Kartono, 1994, *Teori Permainan*, Andi Offset: Yogyakarta.

Mulyono, S. 2007. *Riset Operasi*, Lembaga Penerbit Fakultas Ekonomi Universitas Indonesia, Jakarta.

Riduwan. 2005. *Belajar Mudah Penelitian untuk Guru, Karyawan, dan Peneliti Pemula*. Bandung: Alfabeta.

Siagian, P. 1987. *Penelitian Operasional, Teori dan Praktek*. Jakarta: Universitas Indonesia (UI-Press).

Soekartawi. 1992. *Linear Programming*. Jakarta: Rajawali Pers.

Straffin, P. D. 1993. *Game Theory and Strategy*. Washington : *The Mathematical Association of America*.

Subagyo, P. Marwan Asri dan T.H. Handoko. 1990. *Dasar-Dasar Operation Research.* Yogyakarta: BPFE.

Sumarsono, H.M.S. 2004. *Metode Riset Sumber Daya Manusia*. Yogyakarta: Graha Ilmu.

Taha, H. A. 1997. *Riset Operasi, Jilid 2*. Jakarta: Binarupa Aksara.

Wicaksana, S. P. 2018. *Optimalisasi Persaingan Transportasi Dengan Menggunakan Metode Simpleks Dan Algoritma Brown.* Skripsi Universitas Brawijaya Malang.

Wijaya, A. 2013. *Pengantar Riset Operasi, Edisi 3*. Jakarta: Mitra Wacana Media.

LAMPIRAN

Lampiran 1. Kuisioner

Kuisioner
Pilihan tempat kuliner mahasiswa Universitas Brawijaya

Identitas Responden
Fakultas :
Angkatan : 2015/ 2016/ 2017/ 2018 *)

Berilah tanda silang (X) pada jawaban sesuai pilihan Anda!

1. Apakah Anda berminat berkuliner ?
   a. Ya b. Tidak
2. Manakah yang Anda pilih di antara restoran berikut?
   a. McDonald's
   b. KFC
   c. Richeese Factory
3. Jika diberikan pilihan perbandingan restoran, alternatif manakah yang anda pilih ?
   a. McDonald's dan KFC
   b. McDonald's dan Richeese Factory
   c. KFC dan Richeese Factory
4. Ada 5 strategi pemasaran untuk restoran McDonald's, KFC, dan Richeese Factory, yaitu:
   - Harga [disingkat : H]
   - Lokasi [disingkat : L]
   - Promo [disingkat : PR]
   - Variasi Produk [disingkat : VP]
   - Pelayanan [disingkat : PL]

Jika masing-masing restoran mempunyai satu strategi unggulan, berikan skor sesuai urutan prioritas Anda untuk masing-masing pilihan **)

Catatan :
*)coret yang tidak perlu
**)skor prioritas yang diberikan sebagai berikut.
Tidak penting : 1
Kurang penting : 2
Netral : 3
Penting : 4
Sangat penting : 5

Keterangan
STR : Strategi
SK : Skor

Contoh Pengisian Tabel
Membandingkan Harga (H) pada McDonald's dengan Variasi Produk (VP) pada KFC. Karena H lebih penting daripada VP, maka dapat diisi dengan
i. H 5
ii. VP 4
Skor yang diberikan boleh sama jika menurut Anda kedua strategi sama-sama penting.

A) Berikut adalah tabel penilaian untuk McDonald's (i) dan KFC (ii), isi tabel di bawah ini.

No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK
1.	i. H		6.	i. L		11.	i. PR		16.	i. VP		21.	i. PL	
	ii. H			ii. H			ii. H			ii. H			ii. H	
2.	i. H		7.	i. L		12.	i. PR		17.	i. VP		22.	i. PL	
	ii. L			ii. L			ii. L			ii. L			ii. L	
3.	i. H		8.	i. L		13.	i. PR		18.	i. VP		23.	i. PL	
	ii. PR			ii. PR			ii. PR			ii. PR			ii. PR	
4.	i. H		9.	i. L		14.	i. PR		19.	i. VP		24.	i. PL	
	ii. VP			ii. VP			ii. VP			ii. VP			ii.VP	
5.	i. H		10.	i. L		15.	i. PR		20.	i. VP		25.	i. PL	
	ii. PL			ii. PL			ii. PL			ii. PL			ii. PL	

B) Berikut adalah tabel penilaian untuk McDonald's (i) dan Richeese Factory (ii), isi tabel di bawah ini.

No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK
1.	i. H		6.	i. L		11.	i. PR		16.	i. VP		21.	i. PL	
	ii. H			ii. H			ii. H			ii. H			ii. H	
2.	i. H		7.	i. L		12.	i. PR		17.	i. VP		22.	i. PL	
	ii. L			ii. L			ii. L			ii. L			ii. L	
3.	i. H		8.	i. L		13.	i. PR		18.	i. VP		23.	i. PL	
	ii. PR			ii. PR			ii. PR			ii. PR			ii. PR	
4.	i. H		9.	i. L		14.	i. PR		19.	i. VP		24.	i. PL	
	ii. VP			ii. VP			ii. VP			ii. VP			ii.VP	
5.	i. H		10.	i. L		15.	i. PR		20.	i. VP		25.	i. PL	
	ii. PL			ii. PL			ii. PL			ii. PL			ii. PL	

C) Berikut adalah tabel penilaian untuk KFC (i) dan Richeese Factory (ii), isi tabel di bawah ini.

No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK
1.	i. H		6.	i. L		11.	i. PR		16.	i. VP		21.	i. PL	
	ii. H			ii. H			ii. H			ii. H			ii. H	
2.	i. H		7.	i. L		12.	i. PR		17.	i. VP		22.	i. PL	
	ii. L			ii. L			ii. L			ii. L			ii. L	
3.	i. H		8.	i. L		13.	i. PR		18.	i. VP		23.	i. PL	
	ii. PR			ii. PR			ii. PR			ii. PR			ii. PR	
4.	i. H		9.	i. L		14.	i. PR		19.	i. VP		24.	i. PL	
	ii. VP			ii. VP			ii. VP			ii. VP			ii.VP	
5.	i. H		10.	i. L		15.	i. PR		20.	i. VP		25.	i. PL	
	ii. PL			ii. PL			ii. PL			ii. PL			ii. PL	

Lampiran 2. Data Hasil Dari Pengisian Kuisioner

A) Berikut adalah tabel penilaian untuk McDonald’s (i) dan KFC (ii), isi tabel di bawah ini.

No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK
1.	i. H	310	6.	i. L	303	11.	i. PR	311	16.	i. VP	302	21.	i. PL	306
	ii. H	284		ii. H	264		ii. H	260		ii. H	236		ii. H	253
2.	i. H	301	7.	i. L	332	12.	i. PR	311	17.	i. VP	304	22.	i. PL	336
	ii. L	277		ii. L	272		ii. L	265		ii. L	246		ii. L	264
3.	i. H	294	8.	i. L	305	13.	i. PR	317	18.	i. VP	306	23.	i. PL	338
	ii. PR	260		ii. PR	279		ii. PR	272		ii. PR	264		ii. PR	250
4.	i. H	302	9.	i. L	296	14.	i. PR	302	19.	i. VP	340	24.	i. PL	331
	ii. VP	234		ii. VP	249		ii. VP	279		ii. VP	269		ii.VP	260
5.	i. H	293	10.	i. L	320	15.	i. PR	297	20.	i. VP	336	25.	i. PL	314
	ii. PL	253		ii. PL	254		ii. PL	269		ii. PL	280		ii. PL	260

B) Berikut adalah tabel penilaian untuk McDonald's (i) dan Richeese Factory (ii), isi tabel di bawah ini.

No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK
1.	i. H	120	**6.**	i. L	114	**11.**	i. PR	121	**16.**	i. VP	120	**21.**	i. PL	115
	ii. H	120		ii. H	104		ii. H	101		ii. H	92		ii. H	99
2.	i. H	119	**7.**	i. L	131	**12.**	i. PR	120	**17.**	i. VP	119	**22.**	i. PL	133
	ii. L	112		ii. L	110		ii. L	107		ii. L	104		ii. L	113
3.	i. H	120	**8.**	i. L	115	**13.**	i. PR	125	**18.**	i. VP	122	**23.**	i. PL	132
	ii. PR	106		ii. PR	113		ii. PR	113		ii. PR	105		ii. PR	100
4.	i. H	125	**9.**	i. L	117	**14.**	i. PR	119	**19.**	i. VP	134	**24.**	i. PL	132
	ii. VP	88		ii. VP	98		ii. VP	113		ii. VP	114		ii.VP	98
5.	i. H	119	**10.**	i. L	126	**15.**	i. PR	122	**20.**	i. VP	133	**25.**	i. PL	120
	ii. PL	93		ii. PL	102		ii. PL	109		ii. PL	114		ii. PL	98

C) Berikut adalah tabel penilaian untuk KFC (i) dan Richeese Factory (ii), isi tabel di bawah ini.

No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK
1.	i. H	93	**6.**	i. L	85	**11.**	i. PR	87	**16.**	i. VP	85	**21.**	i. PL	82
	ii. H	91		ii. H	82		ii. H	73		ii. H	70		ii. H	77
2.	i. H	90	**7.**	i. L	95	**12.**	i. PR	86	**17.**	i. VP	83	**22.**	i. PL	99
	ii. L	88		ii. L	83		ii. L	84		ii. L	73		ii. L	82
3.	i. H	90	**8.**	i. L	81	**13.**	i. PR	90	**18.**	i. VP	89	**23.**	i. PL	98
	ii. PR	80		ii. PR	90		ii. PR	88		ii. PR	81		ii. PR	69
4.	i. H	95	**9.**	i. L	83	**14.**	i. PR	84	**19.**	i. VP	98	**24.**	i. PL	98
	ii. VP	64		ii. VP	75		ii. VP	90		ii. VP	88		ii.VP	76
5.	i. H	90	**10.**	i. L	90	**15.**	i. PR	88	**20.**	i. VP	99	**25.**	i. PL	90
	ii. PL	64		ii. PL	75		ii. PL	82		ii. PL	89		ii. PL	76

Lampiran 3. Pembentukan Matriks *Payoff*

A) Berikut adalah tabel penilaian untuk McDonald's (i) dan KFC (ii), isi tabel di bawah ini.

No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK
1.	i. H	3,97	**6.**	i. L	3,88	**11.**	i. PR	3,99	**16.**	i. VP	3,87	**21.**	i. PL	3,92
	ii. H	3,64		ii. H	3,38		ii. H	3,33		ii. H	3,03		ii. H	3,24
2.	i. H	3,86	**7.**	i. L	4,26	**12.**	i. PR	3,99	**17.**	i. VP	3,90	**22.**	i. PL	4,31
	ii. L	3,55		ii. L	3,49		ii. L	3,40		ii. L	3,15		ii. L	3,38
3.	i. H	3,77	**8.**	i. L	3,91	**13.**	i. PR	4,06	**18.**	i. VP	3,92	**23.**	i. PL	4,33
	ii. PR	3,33		ii. PR	3,58		ii. PR	3,49		ii. PR	3,38		ii. PR	3,21
4.	i. H	3,87	**9.**	i. L	3,79	**14.**	i. PR	3,87	**19.**	i. VP	4,36	**24.**	i. PL	4,24
	ii. VP	3,00		ii. VP	3,19		ii. VP	3,58		ii. VP	3,45		ii.VP	3,33
5.	i. H	3,76	**10.**	i. L	4,10	**15.**	i. PR	3,81	**20.**	i. VP	4,31	**25.**	i. PL	4,03
	ii. PL	3,24		ii. PL	3,26		ii. PL	3,45		ii. PL	3,59		ii. PL	3,33

B) Berikut adalah tabel penilaian untuk McDonald’s (i) dan Richeese Factory (ii), isi tabel di bawah ini.

No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK
1.	i. H	4,00	**6.**	i. L	3,80	**11.**	i. PR	4,03	**16.**	i. VP	4,00	**21.**	i. PL	3,83
	ii. H	4,00		ii. H	3,47		ii. H	3,37		ii. H	3,07		ii. H	3,30
2.	i. H	3,97	**7.**	i. L	4,37	**12.**	i. PR	4,00	**17.**	i. VP	3,97	**22.**	i. PL	4,43
	ii. L	3,73		ii. L	3,67		ii. L	3,57		ii. L	3,47		ii. L	3,77
3.	i. H	4,00	**8.**	i. L	3,83	**13.**	i. PR	4,17	**18.**	i. VP	4,07	**23.**	i. PL	4,40
	ii. PR	3,53		ii. PR	3,77		ii. PR	3,77		ii. PR	3,50		ii. PR	3,33
4.	i. H	4,17	**9.**	i. L	3,90	**14.**	i. PR	3,97	**19.**	i. VP	4,47	**24.**	i. PL	4,40
	ii. VP	2,93		ii. VP	3,27		ii. VP	3,77		ii. VP	3,80		ii.VP	3,27
5.	i. H	3,97	**10.**	i. L	4,20	**15.**	i. PR	4,07	**20.**	i. VP	4,43	**25.**	i. PL	4,00
	ii. PL	3,10		ii. PL	3,40		ii. PL	3,63		ii. PL	3,80		ii. PL	3,27

C) Berikut adalah tabel penilaian untuk KFC (i) dan Richeese Factory (ii), isi tabel di bawah ini.

No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK	No	STR	SK
1.	i. H	4,23	**6.**	i. L	3,86	**11.**	i. PR	3,95	**16.**	i. VP	3,86	**21.**	i. PL	3,73
	ii. H	4,14		ii. H	3,73		ii. H	3,32		ii. H	3,18		ii. H	3,50
2.	i. H	4,09	**7.**	i. L	4,32	**12.**	i. PR	3,91	**17.**	i. VP	3,77	**22.**	i. PL	4,50
	ii. L	4,00		ii. L	3,77		ii. L	3,82		ii. L	3,32		ii. L	3,73
3.	i. H	4,09	**8.**	i. L	3,68	**13.**	i. PR	4,09	**18.**	i. VP	4,05	**23.**	i. PL	4,45
	ii. PR	3,64		ii. PR	4,09		ii. PR	4,00		ii. PR	3,68		ii. PR	3,14
4.	i. H	4,32	**9.**	i. L	3,77	**14.**	i. PR	3,82	**19.**	i. VP	4,45	**24.**	i. PL	4,45
	ii. VP	2,91		ii. VP	3,41		ii. VP	4,09		ii. VP	4,00		ii.VP	3,45
5.	i. H	4,09	**10.**	i. L	4,09	**15.**	i. PR	4,00	**20.**	i. VP	4,50	**25.**	i. PL	4,09
	ii. PL	2,91		ii. PL	3,41		ii. PL	3,73		ii. PL	4,05		ii. PL	3,45

Lampiran 4. Pengerjaan Matriks *Payoff* McDonalds

1. Penyelesaian matriks *payoff* dengan menggunakan metode simpleks

Penyelesaian matriks *payoff* McDonald's menggunakan metode simpleks akan dilakukan dengan program Matlab. Berikut adalah langkah – langkah penyelesaian matriks *payoff* McDonald's.

1. Masukan Matriks *payoff* McDonald's

Masukan matriks yang diperoleh dari kuisioner pada lampiran 3 seperti pada gambar 4.7.

```
matriks_payoff =

    4.0000    3.8000    4.0300    4.0000    3.8300
    3.9700    4.3700    4.0000    3.9700    4.4300
    4.0000    3.8300    4.1700    4.0700    4.4000
    4.1700    3.9000    3.9700    4.4700    4.4000
    3.9700    4.2000    4.0700    4.4300    4.0000
```

Gambar 4.7 Matriks *payoff* McDonald's

2. Menentukan *saddle point*

Hasil *saddle point* dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.8

```
minimaks =

    4.1700


maksimin =

    3.9700
```

Gambar 4.8 Nilai minimaks dan maksimin matriks *payoff* McDonald's

Berdasarkan gambar 4.16 diperoleh nilai maksimin, 3,97 sebagai batas bawah permainan ($\underline{V}$). Sementara itu, nilai minimaks pada matriks *payoff* tersebut adalah 4,17 sebagai batas atas permainan ($\overline{V}$), karena ($\underline{V}$) $\neq$ ($\overline{V}$), maka dapat disimpulkan bahwa tidak terdapat *saddle point* pada matriks *payoff* McDonald's.

3. Penyelesaian matriks *payoff* menggunakan dominasi

Hasil dominasi dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.9.

```
Matriks_hasil_dominasi =


    3.9700    4.3700    3.9700
    4.0000    3.8300    4.0700
    4.1700    3.9000    4.4700
```

Gambar 4.9 Hasil dominasi matriks *payoff* McDonald's

Dari Gambar 4.9 hasil bentuk standar masalah maksimum pemrograman linear tersebut adalah sebagai berikut.

Maksimum $Z = Y_1 + Y_2 + Y_3 = \frac{1}{V}$

dengan batasan

$$\begin{aligned} 3{,}97\,Y_1 + 4{,}37\,Y_2 + 3{,}97\,Y_3 + S_1 \quad &= 1 \\ 4{,}00\,Y_1 + 3{,}83\,Y_2 + 4{,}07\,Y_3 + \quad S_2 \quad &= 1 \\ 4{,}17\,Y_1 + 3{,}90\,Y_2 + 4{,}47\,Y_3 + \quad\quad S_3 &= 1\,, \\ Y_1, Y_2, Y_3 \geq 0 & \end{aligned}$$

4. Penyelesaian matriks *payoff* dengan metode simpleks

Untuk penyelesaian masalah maksimum pada McDonald's digunakan metode simpleks. Berikut adalah hasil iterasi metode simpleks menggunakan program yang dibuat dengan sftware MATLAB. Dapat dilihat pada Gambar 4.10 (a) – 4.10 (d).

```
ITERASI 0

-----------------------------------------------------------------|-------------------|
|       |  Cj   |     1      1      1      0      0      0       |          |        |
|  Cb   |-------|------------------------------------------------| Indeks   | Rasio  |
|       | Basis |    Y1     Y2     Y3     S1     S2     S3       |          |        |
-----------------------------------------------------------------|-------------------|
|  0    |  S1   |   3.970  4.370  3.970  1.000  0.000  0.000     |  1.000   | 0.252  |
|  0    |  S2   |   4.000  3.830  4.070  0.000  1.000  0.000     |  1.000   | 0.246  |
|  0    |  S3   |   4.170  3.900  4.470  0.000  0.000  1.000     |  1.000   | 0.224  |
-----------------------------------------------------------------|-------------------|
|    Zj-cj      |   -1.000  -1.000  -1.000  -0.000  -0.000  -0.000     |      Z =0.000
-----------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 1.000
Rasio Minimum = 0.224
Kolom kunci   = 3
Baris kunci   = 3
Elemen Pivot  = 4.470
```

Gambar 4.10 (a) Iterasi ke-0 matriks *payoff* McDonald's

```
ITERASI 1

-----------------------------------------------------------------|-------------------|
|       |  Cj   |     1      1      1      0      0      0       |          |        |
|  Cb   |-------|------------------------------------------------| Indeks   | Rasio  |
|       | Basis |    Y1     Y2     Y3     S1     S2     S3       |          |        |
-----------------------------------------------------------------|-------------------|
|  0    |  S1   |   0.266  0.906  0.000  1.000  0.000  -0.888    |  0.112   | 0.123  |
|  0    |  S2   |   0.203  0.279  0.000  0.000  1.000  -0.911    |  0.089   | 0.321  |
|  1    |  Y3   |   0.933  0.872  1.000  0.000  0.000  0.224     |  0.224   | 0.256  |
-----------------------------------------------------------------|-------------------|
|    Zj-cj      |   -0.067  -0.128  -0.000  -0.000  -0.000  0.224     |      Z =0.224
-----------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 0.128
Rasio Minimum = 0.123
Kolom kunci   = 2
Baris kunci   = 1
Elemen Pivot  = 0.906
```

Gambar 4.10 (b) Iterasi ke-1 matriks *payoff* McDonald's

```
ITERASI 2

-----------------------------------------------------------------|-------------------|
|       |  Cj   |     1      1      1      0      0      0       |          |        |
|  Cb   |-------|------------------------------------------------| Indeks   | Rasio  |
|       | Basis |    Y1     Y2     Y3     S1     S2     S3       |          |        |
-----------------------------------------------------------------|-------------------|
|  1    |  Y2   |   0.294  1.000  0.000  1.103  0.000  -0.980    |  0.123   | 0.420  |
|  0    |  S2   |   0.121  0.000  0.000  -0.308  1.000  -0.637   |  0.055   | 0.454  |
|  1    |  Y3   |   0.676  0.000  1.000  -0.963  0.000  1.079    |  0.116   | 0.172  |
-----------------------------------------------------------------|-------------------|
|    Zj-cj      |   -0.030  -0.000  -0.000  0.141  -0.000  0.099     |      Z =0.239
-----------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 0.030
Rasio Minimum = 0.172
Kolom kunci   = 1
Baris kunci   = 3
Elemen Pivot  = 0.676
```

Gambar 4.10 (c) Iterasi ke-2 matriks *payoff* McDonald's

```
ITERASI 3

----------------------------------------------------------------|------------------|
|      |  Cj   |     1      1      1      0      0      0       |         |        |
|  Cb  |-------|------------------------------------------------| Indeks  | Rasio  |
|      | Basis |    Y1     Y2     Y3     S1     S2     S3       |         |        |
----------------------------------------------------------------|------------------|
|  1   |  Y2   |   0.000  1.000  -0.435  1.522  0.000  -1.449   |  0.073  | Inf    |
|  0   |  S2   |   0.000  0.000  -0.179  -0.135  1.000  -0.830  |  0.034  | 0.034  |
|  1   |  Y1   |   1.000  0.000  1.478  -1.423  0.000  1.595    |  0.172  | Inf    |
----------------------------------------------------------------|------------------|
|     Zj-cj    |   -0.000  -0.000  0.044  0.099  -0.000  0.146  |      Z =0.245    |
----------------------------------------------------------------|------------------|
Zj-Cj Maksimum = 0.000
Rasio Minimum = 0.034
Kolom kunci   = 5
Baris kunci   = 2
Elemen Pivot  = 1.000
```

Gambar 4.10 (d) Iterasi ke-4 matriks *payoff* McDonald's

Karena nilai $Z_j - C_j$ sudah tidak ada yang bernilai negatif, maka tahap itetasi selesai dengan diperoleh hasil akhir iterasi simpleks, yaitu iterasi ke-3 McDonald's.

5. Menentukan strategi optimal masing-masing pemain

Berdasarkan Gambar 4.10 (d), seluruh baris dan kolom dapat diabaikan sehingga strategi optimal untuk Richeese Factory, yaitu: $Y_1 = 0{,}172$ $Y_2 = 0{,}073$, $Y_3 = 0$ dengan $Y_1 = q_1$, $Y_2 = q_2$, $Y_3 = q_4$, dan $Z = 0{,}245$. Peluang strategi optimal untuk Richeese Factory adalah sebagai berikut.

Peluang $q_1 = \frac{Y_1}{Z} = \frac{0{,}172}{0{,}245} = 0{,}7020 = 70{,}20\%$

Peluang $q_2 = \frac{Y_2}{Z} = \frac{0{,}073}{0{,}245} = 0{,}2979 = 29{,}79\%$

Peluang $q_4 = \frac{Y_3}{Z} = \frac{0}{0{,}245} = 0 = 0\%$

dengan nilai permainan $V = \frac{1}{Z} = \frac{1}{0{,}245} = 4{,}081$

Sementara itu, strategi optimal untuk McDonald's, diperoleh $X_1 = 0{,}099$, $X_2 = 0$, $X_3 = 0{,}146$, dan $Z = W = 0{,}245$. Peluang strategi optimal untuk McDonald's adalah sebagai berikut.

Peluang $p_2 = \frac{X_1}{W} = \frac{0{,}099}{0{,}245} = 0{,}4040 = 40{,}40\%$

Peluang $p_3 = \frac{X_2}{W} = \frac{0}{0{,}245} = 0 = 0\%$

Peluang $p_4 = \frac{X_3}{W} = \frac{0,146}{0,245} = 0,5959 = 59,59\%$

dengan nilai permainan $V = \frac{1}{W} = \frac{1}{0,245} = 4,081$.

2. Penyelesaian matriks *payoff* dengan menggunakan algoritma *brown*

Penyelesaian matriks *payoff* KFC menggunakan algoritma *brown* akan dilakukan dengan program Matlab. Berikut adalah langkah – langkah penyelesaian matriks *payoff* KFC.

1. Masukan Matriks

Masukan matriks yang diperoleh dari kuisioner pada lampiran 3 seperti pada gambar 4.7.

2. Menentukan *saddle point*

Hasil *saddle point* dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.8

3. Menyelesaikan matriks *payoff* dengan menggunakan algoritma *brown*.

Penyelesaian dengan menggunakan algoritma *brown*, dapat dilihat pada Gambar 4.11.

```
baris_iterasi =

    4.0000    3.8000    4.0300    4.0000    3.8300
    7.9700    8.1700    8.0300    7.9700    8.2600
   11.9400   12.5400   12.0300   11.9400   12.6900
   15.9100   16.7400   16.1000   16.3700   16.6900
   20.0800   20.6400   20.0700   20.8400   21.0900
   24.2500   24.5400   24.0400   25.3100   25.4900
   28.2200   28.7400   28.1100   29.7400   29.4900
   32.1900   32.9400   32.1800   34.1700   33.4900
   36.1600   37.1400   36.2500   38.6000   37.4900
   40.1300   41.3400   40.3200   43.0300   41.4900
```

```
  Columns 1 through 5

    3.8000    7.8000   11.8000   15.8000   19.8300
    4.3700    8.3400   12.3100   16.2800   20.2800
    3.8300    7.8300   11.9000   15.9000   20.0700
    3.9000    8.0700   12.5400   16.7100   20.6800
    4.2000    8.1700   12.6000   16.5700   20.6400

  Columns 6 through 10

   23.8600   27.8900   31.9200   35.9200   39.9200
   24.2800   28.2800   32.2800   36.2500   40.2200
   24.2400   28.4100   32.5800   36.5800   40.5800
   24.6500   28.6200   32.5900   36.7600   40.9300
   24.7100   28.7800   32.8500   36.8200   40.7900
```

Gambar 4.11 Iterasi algoritma *brown* matriks *payoff* McDonald's

Dari iterasi gambar 4.11 akan didapat nilai batas atas ($\overline{V}$) dan batas bawah ($\underline{V}$), yaitu dimana ($\underline{V}$) ≤ V ≤ ($\overline{V}$), seperti gambar 4.12.

```
batasatas =

    4.0930


batasbawah =

    4.0130
```

Gambar 4.12 Hasil nilai permainan, batas atas, dan batas bawah matriks *payoff* McDonald's

Lampiran 5. Pengerjaan Matriks *Payoff* Richeese Factory

1. Penyelesaian matriks *payoff* dengan menggunakan metode simpleks

Penyelesaian matriks *payoff* Richeese Factory menggunakan metode simpleks akan dilakukan dengan program Matlab. Berikut adalah langkah – langkah penyelesaian matriks *payoff* Richeese Factory.

1. Masukan Matriks *payoff* Richeese Factory

Masukan matriks yang diperoleh dari kuisioner pada lampiran 3 seperti pada gambar 4.13.

```
Matriks_payoff =

    4.0000    3.4700    3.3700    3.0700    3.3000
    3.7300    3.6700    3.5700    3.4700    3.7700
    3.5300    3.7700    3.7700    3.5000    3.3300
    2.9300    3.2700    3.7700    3.8000    3.2700
    3.1000    3.4000    3.6300    3.8000    3.2700
```

Gambar 4.13 Matriks *payoff* Richeese Factory

2. Menentukan *saddle point*

Hasil *saddle point* dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.14.

```
minimaks =

    3.7700


maksimin =

    3.4700
```

Gambar 4.14 Nilai minimaks dan maksimin matriks payoff Richeese Factory

Berdasarkan gambar 4.14 diperoleh nilai maksimin, 3,47 sebagai batas bawah permainan ($\underline{V}$). Sementara itu, nilai minimaks pada matriks *payoff* tersebut adalah 3,77 sebagai batas atas permainan ($\overline{V}$), karena ($\underline{V}$) ≠ ($\overline{V}$), maka dapat disimpulkan bahwa tidak terdapat *saddle point* pada matriks *payoff* Richeese Factory.

3. Penyelesaian matriks *payoff* menggunakan dominasi

Hasil dominasi dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.15.

```
Matriks_hasil_dominasi =

    3.7300    3.4700    3.7700
    3.5300    3.5000    3.3300
    3.1000    3.8000    3.2700
```

Gambar 4.15 Hasil dominasi matriks *payoff* Richeese Factory

Hasil bentuk standar masalah maksimum pemrograman linear tersebut adalah sebagai berikut.

Maksimum $Z = Y_1 + Y_2 + \mathrm{Y}_3 = \frac{1}{V}$

dengan batasan

$$\begin{aligned} 3{,}73\,Y_1 + 3{,}47\,Y_2 + 3{,}77\,\mathrm{Y}_3 + S_1 \quad\quad &= 1 \\ 3{,}53\,Y_1 + 3{,}50\,Y_2 + 3{,}33\,\mathrm{Y}_3 + \quad S_2 \quad &= 1 \\ 3{,}10\,Y_1 + 3{,}80\,Y_2 + 3{,}27\,\mathrm{Y}_3 + \quad\quad S_3 &= 1\,, \\ Y_1, Y_2, \mathrm{Y}_3 \geq 0 \end{aligned}$$

4. Penyelesaian matriks *payoff* dengan metode simpleks

Untuk penyelesaian masalah maksimum pada Richeese Factory digunakan metode simpleks. Berikut adalah hasil iterasi metode simpleks menggunakan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.16 (a) – 4.16 (d).

```
ITERASI 0

---------------------------------------------------------------|-------------------|
|       |  Cj   |    1      1      1      0      0      0       |         |         |
|  Cb   |-------|-----------------------------------------------| Indeks  |  Rasio  |
|       | Basis |    Y1     Y2     Y3      S1     S2     S3     |         |         |
---------------------------------------------------------------|-------------------|
|  0    |  S1   |   3.730  3.470  3.770  1.000  0.000  0.000    |  1.000  | 0.265   |
|  0    |  S2   |   3.530  3.500  3.330  0.000  1.000  0.000    |  1.000  | 0.300   |
|  0    |  S3   |   3.100  3.800  3.270  0.000  0.000  1.000    |  1.000  | 0.306   |
---------------------------------------------------------------|-------------------|
|   Zj-cj       |   -1.000  -1.000  -1.000  -0.000  -0.000  -0.000   |      Z =0.000
---------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 1.000
Rasio Minimum = 0.265
Kolom kunci   = 3
Baris kunci   = 1
Elemen Pivot  = 3.770
```

Gambar 4.16 (a) Iterasi ke-0 matriks *payoff* Richeese Factory

```
ITERASI 1

---------------------------------------------------------------|-------------------|
|       |  Cj   |    1      1      1      0      0      0       |         |         |
|  Cb   |-------|-----------------------------------------------| Indeks  |  Rasio  |
|       | Basis |    Y1     Y2     Y3      S1     S2     S3     |         |         |
---------------------------------------------------------------|-------------------|
|  1    |  Y3   |   0.989  0.920  1.000  0.265  0.000  0.000   |  0.265  | 0.288   |
|  0    |  S2   |   0.235  0.435  0.000  -0.883  1.000  0.000   |  0.117  | 0.268   |
|  0    |  S3   |   -0.135  0.790  0.000  -0.867  0.000  1.000   |  0.133  | 0.168   |
---------------------------------------------------------------|-------------------|
|   Zj-cj       |   -0.011  -0.080  -0.000  0.265  -0.000  -0.000   |      Z =0.265
---------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 0.080
Rasio Minimum = 0.168
Kolom kunci   = 2
Baris kunci   = 3
Elemen Pivot  = 0.790
```

Gambar 4.16 (b) Iterasi ke-1 matriks *payoff* Richeese Factory

```
ITERASI 2

---------------------------------------------------------------|-------------------|
|       |  Cj   |    1      1      1      0      0      0       |         |         |
|  Cb   |-------|-----------------------------------------------| Indeks  |  Rasio  |
|       | Basis |    Y1     Y2     Y3      S1     S2     S3     |         |         |
---------------------------------------------------------------|-------------------|
|  1    |  Y3   |   1.147  0.000  1.000  1.276  0.000  -1.165   |  0.111  | 0.097
|  0    |  S2   |   0.310  0.000  0.000  -0.406  1.000  -0.550   |  0.044  | 0.141
|  1    |  Y2   |   -0.171  1.000  0.000  -1.098  0.000  1.265   |  0.168  | -0.980
---------------------------------------------------------------|-------------------|
|   Zj-cj       |   -0.024  -0.000  -0.000  0.178  -0.000  0.101   |      Z =0.279
---------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 0.024
Rasio Minimum = 0.097
Kolom kunci   = 1
Baris kunci   = 1
Elemen Pivot  = 1.147
```

Gambar 4.16 (c) Iterasi ke-2 matriks *payoff* Richeese Factory

```
ITERASI 3

--------------------------------------------------------------|-------------------|
|       |  Cj   |     1      1      1      0      0      0     |         |
|  Cb   |-------|-----------------------------------------------| Indeks  |  Rasio  |
|       | Basis |    Y1     Y2     Y3      S1     S2     S3     |         |         |
--------------------------------------------------------------|-------------------|
|  1    |  Y1   |    1.000  0.000  0.872  1.112  0.000  -1.016   |  0.097  | Inf   |
|  0    |  S2   |    0.000  0.000  -0.270  -0.750  1.000  -0.236   |  0.014  | 0.014
|  1    |  Y2   |    0.000  1.000  0.149  -0.907  0.000  1.092    |  0.184  | Inf   |
--------------------------------------------------------------|-------------------|
|    Zj-cj      |    -0.000  -0.000  0.021  0.205  -0.000  0.076    |      Z =0.281
--------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 0.000
Rasio Minimum = 0.014
Kolom kunci   = 5
Baris kunci   = 2
Elemen Pivot  = 1.000
```

Gambar 4.16 (d) Iterasi ke-3 matriks *payoff* Richeese Factory

Karena nilai $Z_j - C_j$ sudah tidak ada yang bernilai negatif, maka tahap itetasi selesai dengan diperoleh hasil akhir iterasi simpleks, yaitu iterasi ke-3 Richeese Factory.

5. Menentukan strategi optimal masing-masing pemain

Berdasarkan Tabel 4.11, seluruh baris dan kolom dapat diabaikan sehingga strategi optimal untuk Richeese Factory, yaitu: $Y_1 = 0{,}097$, $Y_2 = 0{,}184$, $Y_3 = 0$ dengan $Y_1 = q_1$, $Y_2 = q_4$, $Y_3 = q_5$, dan $Z = 0{,}281$. Peluang strategi optimal untuk Richeese Factory adalah sebagai berikut.

Peluang $q_1 = \frac{Y_1}{Z} = \frac{0{,}097}{0{,}281} = 0{,}3451 = 34{,}51\%$

Peluang $q_4 = \frac{Y_2}{Z} = \frac{0{,}184}{0{,}281} = 0{,}6548 = 65{,}48\%$

Peluang $q_5 = \frac{Y_3}{Z} = \frac{0}{0{,}281} = 0 = 0\%$

dengan nilai permainan $V = \frac{1}{Z} = \frac{1}{0{,}281} = 3{,}558$

Sementara itu, strategi optimal untuk McDonald's, diperoleh $X_1 = 0{,}205$, $X_2 = 0$, $X_3 = 0{,}076$, dan $Z = W = 0{,}281$. Peluang strategi optimal untuk McDonald's adalah sebagai berikut.

Peluang $p_2 = \frac{X_1}{W} = \frac{0{,}205}{0{,}281} = 0{,}7295 = 72{,}95\%$

Peluang $p_3 = \frac{X_2}{W} = \frac{0}{0{,}281} = 0 = 0\%$

Peluang $p_5 = \frac{X_3}{W} = \frac{0{,}076}{0{,}281} = 0{,}2704 = 27{,}04\%$

dengan nilai permainan $V = \frac{1}{W} = \frac{1}{0{,}281} = 3{,}558.$

2. Penyelesaian matriks *payoff* dengan menggunakan algoritma *brown*

Penyelesaian matriks *payoff* Richeese Factory menggunakan algoritma *brown* akan dilakukan dengan program Matlab. Berikut adalah langkah – langkah penyelesaian matriks *payoff* Richeese Factory.

1. Masukan Matriks

Masukan matriks yang diperoleh dari kuisioner pada lampiran 3 seperti pada gambar 4.13.

2. Menentukan *saddle point*

Hasil *saddle point* dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.14.

3. Menyelesaikan matriks *payoff* dengan menggunakan algoritma *brown.*

Penyelesaian dengan menggunakan algoritma *brown,* dapat dilihat seperti pada Gambar 4.17.

```
baris_iterasi =

    2.9300    3.2700    3.7700    3.8000    3.2700
    6.9300    6.7400    7.1400    6.8700    6.5700
   10.6600   10.4100   10.7100   10.3400   10.3400
   14.3900   14.0800   14.2800   13.8100   14.1100
   18.1200   17.7500   17.8500   17.2800   17.8800
   21.8500   21.4200   21.4200   20.7500   21.6500
   25.5800   25.0900   24.9900   24.2200   25.4200
   29.3100   28.7600   28.5600   27.6900   29.1900
   32.4100   32.1600   32.1900   31.4900   32.4600
   35.5100   35.5600   35.8200   35.2900   35.7300


nilai_terkecil_dari_baris =

   35.2900


kolom_iterasi =

    4.0000    7.3000   10.6000   13.6700   16.7400   19.8100   22.8800   25.9500   29.0200   32.0900
    3.7300    7.5000   11.2700   14.7400   18.2100   21.6800   25.1500   28.6200   32.0900   35.5600
    3.5300    6.8600   10.1900   13.6900   17.1900   20.6900   24.1900   27.6900   31.1900   34.6900
    2.9300    6.2000    9.4700   13.2700   17.0700   20.8700   24.6700   28.4700   32.2700   36.0700
    3.1000    6.3700    9.6400   13.4400   17.2400   21.0400   24.8400   28.6400   32.4400   36.2400
```

Gambar 4.17 Iterasi algoritma brown matriks payoff Richeese Factory

Dari iterasi gambar 4.17 akan didapat nilai batas atas ($\overline{V}$) dan batas bawah ($\underline{V}$), yaitu dimana ($\underline{V}$) ≤ V ≤ ($\overline{V}$), seperti Gambar 4.18.

```
batasatas =

    3.6240


batasbawah =

    3.5290
```

Gambar 4.18 Hasil batas atas dan batas bawah matriks payoff Richeese Factory

Lampiran 6. Pengerjaan Matriks *Payoff* KFC

1. Penyelesaian matriks *payoff* dengan menggunakan metode simpleks

Penyelesaian matriks *payoff* KFC menggunakan algoritma *brown* akan dilakukan dengan program Matlab. Berikut adalah langkah – langkah penyelesaian matriks *payoff* KFC.

1. Masukan Matriks

Masukan matriks yang diperoleh dari kuisioner pada lampiran 3 seperti pada Gambar 4.19.

```
matriks_payoff =

    4.2300    3.8600    3.9500    3.8600    3.7300
    4.0900    4.3200    3.9100    3.7700    4.5000
    4.0900    3.6800    4.0900    4.0500    4.4500
    4.3200    3.7700    3.8200    4.4500    4.4500
    4.0900    4.0900    4.0000    4.5000    4.0900
```

Gambar 4.19 Matriks *payoff* KFC

2. Menentukan *saddle point*

Hasil *saddle point* dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.20.

```
minimaks =

    4.0900


maksimin =

    4
```

Gambar 4.20 Nilai minimaks dan maksimin matriks *payoff* KFC

Berdasarkan gambar 4.20 diperoleh nilai maksimin, 4,09 sebagai batas bawah permainan ($\underline{V}$). Sementara itu, nilai minimaks pada matriks *payoff* tersebut adalah 4 sebagai batas atas permainan ($\overline{V}$), karena ($\underline{V}$) ≠ ($\overline{V}$), maka dapat disimpulkan bahwa tidak terdapat *saddle point* pada matriks *payoff* KFC.

3. Penyelesaian matriks *payoff* menggunakan dominasi

Hasil dominasi dengan program yang dibuat dengan software MATLAB. dapat dilihat pada Gambar 4.21.

```
Matriks_hasil_dominasi =

    4.3200    3.9100    3.7700
    3.6800    4.0900    4.0500
    4.0900    4.0000    4.5000
```

Gambar 4.21 Hasil dominasi matriks *payoff* KFC

Hasil bentuk standar masalah maksimum pemrograman linear tersebut adalah sebagai berikut.

Maksimum $Z = Y_1 + Y_2 + Y_3 = \frac{1}{V}$

dengan batasan

$$4{,}32\,Y_1 + 3{,}91\,Y_2 + 3{,}77\,Y_3 + S_1 = 1$$
$$3{,}68\,Y_1 + 4{,}09\,Y_2 + 4{,}05\,Y_3 + S_2 = 1$$
$$4{,}09\,Y_1 + 4{,}00\,Y_2 + 4{,}50\,Y_3 + S_3 = 1\,,$$
$$Y_1, Y_2, Y_3 \geq 0$$

4. Penyelesaian matriks *payoff* dengan metode simpleks

Untuk penyelesaian masalah maksimum pada KFC digunakan metode simpleks. Berikut adalah hasil iterasi metode simpleks menggunakan program yang dibuat dengan sftware MATLAB. Dapat dilihat pada Gambar 4.22 (a) – 4.22 (d).

```
ITERASI 0

---------------------------------------------------------------|-------------------|
|        |   Cj   |     1       1       1       0       0       0         |          |
|   Cb   |--------|-------------------------------------------------| Indeks  |  Rasio  |
|        | Basis  |    Y1      Y2      Y3      S1      S2      S3        |          |
---------------------------------------------------------------|-------------------|
|   0    |   S1   |   4.320   3.910   3.770   1.000   0.000   0.000      |  1.000  | 0.265   |
|   0    |   S2   |   3.680   4.090   4.050   0.000   1.000   0.000      |  1.000  | 0.247   |
|   0    |   S3   |   4.090   4.000   4.500   0.000   0.000   1.000      |  1.000  | 0.222   |
---------------------------------------------------------------|-------------------|
|     Zj-cj       |   -1.000  -1.000  -1.000  -0.000  -0.000  -0.000     |        Z =0.000
---------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 1.000
Rasio Minimum = 0.222
Kolom kunci   = 3
Baris kunci   = 3
Elemen Pivot  = 4.500
```

Gambar 4.20 (a) Iterasi ke-0 matriks *payoff* KFC

```
ITERASI 1

---------------------------------------------------------------|-------------------|
|        |   Cj   |     1       1       1       0       0       0         |          |         |
|   Cb   |--------|-------------------------------------------------| Indeks  |  Rasio  |
|        | Basis  |    Y1      Y2      Y3      S1      S2      S3        |          |
---------------------------------------------------------------|-------------------|
|   0    |   S1   |   0.893   0.559   0.000   1.000   0.000   -0.838     |  0.162  | 0.290   |
|   0    |   S2   |   -0.001   0.490   0.000   0.000   1.000   -0.900     |  0.100  | 0.204   |
|   1    |   Y3   |   0.909   0.889   1.000   0.000   0.000   0.222      |  0.222  | 0.250   |
---------------------------------------------------------------|-------------------|
|     Zj-cj       |   -0.091  -0.111  -0.000  -0.000  -0.000   0.222     |        Z =0.222
---------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 0.111
Rasio Minimum = 0.204
Kolom kunci   = 2
Baris kunci   = 2
Elemen Pivot  = 0.490
```

Gambar 4.20 (b) Iterasi ke-1 matriks *payoff* KFC

```
ITERASI 2

---------------------------------------------------------------|-------------------|
|        |   Cj   |     1       1       1       0       0       0         |          |         |
|   Cb   |--------|-------------------------------------------------| Indeks  |  Rasio  |
|        | Basis  |    Y1      Y2      Y3      S1      S2      S3        |          |
---------------------------------------------------------------|-------------------|
|   0    |   S1   |   0.895   0.000   0.000   1.000   -1.141   0.189     |  0.048  | 0.054   |
|   1    |   Y2   |   -0.002   1.000   0.000   0.000   2.041   -1.837     |  0.204  | -100.000
|   1    |   Y3   |   0.911   0.000   1.000   0.000   -1.814   1.855     |  0.041  | 0.045   |
---------------------------------------------------------------|-------------------|
|     Zj-cj       |   -0.091  -0.000  -0.000  -0.000   0.227   0.018     |        Z =0.245
---------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 0.091
Rasio Minimum = 0.045
Kolom kunci   = 1
Baris kunci   = 3
Elemen Pivot  = 0.911
```

Gambar 4.20 (c) Iterasi ke-2 matriks *payoff* KFC

```
ITERASI 3

------------------------------------------------------------------|-------------------|
|      |  Cj   |     1      1      1      0      0      0         |         |         |
|  Cb  |-------|--------------------------------------------------| Indeks  |  Rasio  |
|      | Basis |    Y1     Y2     Y3     S1     S2     S3         |         |         |
------------------------------------------------------------------|-------------------|
|  0   |  S1   |  0.000  0.000  -0.982  1.000  0.641  -1.633    |  0.008  | 0.008   |
|  1   |  Y2   |  0.000  1.000  0.002  0.000  2.037  -1.833     |  0.204  | Inf     |
|  1   |  Y1   |  1.000  0.000  1.098  0.000  -1.992  2.037     |  0.045  | Inf     |
------------------------------------------------------------------|-------------------|
|    Zj-cj     |  -0.000  -0.000  0.100  -0.000  0.045  0.204    |      Z =0.249       
------------------------------------------------------------------|-------------------|
Zj-Cj Maksimum = 0.000
Rasio Minimum = 0.008
Kolom kunci   = 4
Baris kunci   = 1
Elemen Pivot  = 1.000
```

Gambar 4.20 (d) Iterasi ke-3 matriks *payoff* KFC

5. Menentukan strategi optimal masing-masing pemain

Berdasarkan Gambar 4.20 (d), seluruh baris dan kolom dapat diabaikan sehingga strategi optimal untuk Richeese Factory, yaitu: $Y_1 = 0{,}045$ $Y_2 = 0{,}204$, $Y_3 = 0$ dengan $Y_1 = q_2$, $Y_2 = q_3$, $Y_3 = q_4$, dan $Z = 0{,}249$. Peluang strategi optimal untuk Richeese Factory adalah sebagai berikut.

Peluang $q_2 = \frac{Y_1}{Z} = \frac{0{,}045}{0{,}249} = 0{,}1807 = 18{,}07\%$

Peluang $q_3 = \frac{Y_2}{Z} = \frac{0{,}204}{0{,}249} = 0{,}8192 = 81{,}92\%$

Peluang $q_4 = \frac{Y_3}{Z} = \frac{0}{0{,}249} = 0 = 0\%$

dengan nilai permainan $V = \frac{1}{Z} = \frac{1}{0{,}249} = 4{,}016$

Sementara itu, strategi optimal untuk KFC, diperoleh $X_1 = 0$, $X_2 = 0{,}045$, $X_3 = 0{,}204$, dan $Z = W = 0{,}249$. Peluang strategi optimal untuk McDonald's adalah sebagai berikut.

Peluang $p_2 = \frac{X_1}{W} = \frac{0}{0{,}249} = 0 = 0\%$

Peluang $p_3 = \frac{X_2}{W} = \frac{0{,}045}{0{,}249} = 0{,}1807 = 18{,}07\%$

Peluang $p_5 = \frac{X_3}{W} = \frac{0{,}204}{0{,}249} = 0{,}8192 = 81{,}92\%$

dengan nilai permainan $V = \frac{1}{W} = \frac{1}{0{,}249} = 4{,}016$.

2. Penyelesaian matriks *payoff* dengan menggunakan algoritma *brown*

Penyelesaian matriks *payoff* KFC menggunakan algoritma *brown* akan dilakukan dengan program Matlab. Berikut adalah langkah – langkah penyelesaian matriks *payoff* KFC.

1. Masukan Matriks

Masukan matriks yang diperoleh dari kuisioner pada lampiran 3 seperti pada Gambar 4.19.

2. Menentukan *saddle point*

Hasil *saddle point* dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.20.

3. Menyelesaikan matriks *payoff* dengan menggunakan algoritma *brown.*

Penyelesaian dengan menggunakan algoritma *brown*, dapat dilihat pada Gambar 4.23.

```
baris_iterasi =

    4.2300    3.8600    3.9500    3.8600    3.7300
    8.3200    7.9500    7.9500    8.3600    7.8200
   12.4100   12.2700   11.8600   12.1300   12.3200
   16.5000   15.9500   15.9500   16.1800   16.7700
   20.5900   20.2700   19.8600   19.9500   21.2700
   24.6800   24.5900   23.7700   23.7200   25.7700
   28.7700   28.9100   27.6800   27.4900   30.2700
   32.8600   33.0000   31.6800   31.9900   34.3600
   36.9500   37.0900   35.6800   36.4900   38.4500
   41.0400   41.1800   39.6800   40.9900   42.5400
```

```
kolom_iterasi =

    3.7300    7.4600   11.4100   15.2700   19.2200   23.0800   26.9400   30.8900   34.8400   38.7900
    4.5000    9.0000   12.9100   17.2300   21.1400   24.9100   28.6800   32.5900   36.5000   40.4100
    4.4500    8.9000   12.9900   16.6700   20.7600   24.8100   28.8600   32.9500   37.0400   41.1300
    4.4500    8.9000   12.7200   16.4900   20.3100   24.7600   29.2100   33.0300   36.8500   40.6700
    4.0900    8.1800   12.1800   16.2700   20.2700   24.7700   29.2700   33.2700   37.2700   41.2700
```

Gambar 4.23 Iterasi algoritma *brown* matriks *payoff* KFC

Dari iterasi gambar 4.23 akan didapat nilai batas atas ($\overline{V}$) dan batas bawah ($\underline{V}$), yaitu dimana ($\underline{V}$) $\leq$ V $\leq$ ($\overline{V}$), seperti Gambar 4.24.

```
batasatas =

    4.1270


batasbawah =

    3.9680
```

Gambar 4.24 Hasil nilai permainan, batas atas, dan batas bawah matriks *payoff* KFC

Lampiran 7. Pengerjaan Matriks *Payoff* Richeese Factory

1. Penyelesaian matriks *payoff* dengan menggunakan metode simpleks

Penyelesaian matriks *payoff* Richeese Factory menggunakan metode simpleks akan dilakukan dengan program Matlab. Berikut adalah langkah – langkah penyelesaian matriks *payoff* Richeese Factory.

1. Masukan Matriks

Masukan matriks yang diperoleh dari kuisioner pada lampiran 3 seperti pada Gambar 4.25.

```
Matriks_payoff =

    4.1400    3.7300    3.3200    3.1800    3.5000
    4.0000    3.7700    3.8200    3.3200    3.7300
    3.6400    4.0900    4.0000    3.6800    3.1400
    2.9100    3.4100    4.0900    4.0000    3.4500
    2.9100    3.4100    3.7300    4.0500    3.4500
```

Gambar 4.25 Matriks *payoff* Richeese Factory

2. Menentukan *saddle point*

Hasil *saddle point* dengan program yang dibuat dengan software MATLAB. dapat dilihat pada Gambar 4.26.

```
minimaks =

    3.7300


maksimin =

    3.3200
```

Gambar 4.26 Nilai minimaks dan maksimin matriks *payoff* Richeese Factory

Berdasarkan gambar 4.26 diperoleh nilai maksimin, 3,32 sebagai batas bawah permainan ($\underline{V}$). Sementara itu, nilai minimaks pada matriks *payoff* tersebut adalah 3,73 sebagai batas atas permainan ($\overline{V}$), karena ($\underline{V}$) $\neq$ ($\overline{V}$), maka dapat disimpulkan bahwa tidak terdapat *saddle point* pada matriks *payoff* Richeese Factory.

3. Penyelesaian matriks *payoff* menggunakan dominasi

Hasil dominasi dengan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.27.

```
Matriks_hasil_dominasi =

    4.1400    3.3200    3.1800
    4.0000    3.8200    3.3200
    2.9100    4.0900    4.0000
```

Gambar 4.27 Hasil dominasi matriks *payoff* Richeese Factory

Hasil bentuk standar masalah maksimum pemrograman linear tersebut adalah sebagai berikut.

Maksimum $Z = Y_1 + Y_2 + Y_3 = \frac{1}{V}$

dengan batasan

$$\begin{aligned} 4{,}14\,Y_1 + 3{,}32\,Y_2 + 3{,}18\,Y_3 + S_1 \quad &= 1 \\ 4{,}00\,Y_1 + 3{,}82\,Y_2 + 3{,}32\,Y_3 + \quad S_2 \quad &= 1 \\ 2{,}91\,Y_1 + 4{,}09\,Y_2 + 4{,}00\,Y_3 + \quad\quad S_3 &= 1\,, \\ Y_1,\ Y_2, Y_3 \geq 0 \end{aligned}$$

4. Penyelesaian matriks *payoff* dengan metode simpleks

Untuk penyelesaian masalah maksimum pada Richeese Factory digunakan metode simpleks. Berikut adalah hasil iterasi metode simpleks menggunakan program yang dibuat dengan software MATLAB. Dapat dilihat pada Gambar 4.28 (a) – 4.28 (c).

```
ITERASI 0

--------------------------------------------------------------|------------------|
|      |  Cj  |    1      1      1      0      0      0       |         |        |
|  Cb  |------|-----------------------------------------------| Indeks  | Rasio  |
|      | Basis |   Y1     Y2     Y3     S1     S2     S3      |         |        |
--------------------------------------------------------------|------------------|
|  0   |  S1  |   4.140  3.320  3.180  1.000  0.000  0.000    |  1.000  | 0.314  |
|  0   |  S2  |   4.000  3.820  3.320  0.000  1.000  0.000    |  1.000  | 0.301  |
|  0   |  S3  |   2.910  4.090  4.000  0.000  0.000  1.000    |  1.000  | 0.250  |
--------------------------------------------------------------|------------------|
|    Zj-cj    |   -1.000  -1.000  -1.000  -0.000  -0.000  -0.000    |     Z =0.000
--------------------------------------------------------------|------------------|
Zj-Cj Maksimum = 1.000
Rasio Minimum = 0.250
Kolom kunci   = 3
Baris kunci   = 3
Elemen Pivot  = 4.000
```

Gambar 4.28 (a) Iterasi ke-0 matriks *payoff* Richeese Factory

```
ITERASI 1

--------------------------------------------------------------|------------------|
|      |  Cj  |    1      1      1      0      0      0       |         |        |
|  Cb  |------|-----------------------------------------------| Indeks  | Rasio  |
|      | Basis |   Y1     Y2     Y3     S1     S2     S3      |         |        |
--------------------------------------------------------------|------------------|
|  0   |  S1  |   1.827  0.068  0.000  1.000  0.000  -0.795   |  0.205  | 0.112  |
|  0   |  S2  |   1.585  0.425  0.000  0.000  1.000  -0.830   |  0.170  | 0.107  |
|  1   |  Y3  |   0.728  1.023  1.000  0.000  0.000  0.250    |  0.250  | 0.344  |
--------------------------------------------------------------|------------------|
|    Zj-cj    |   -0.272  0.022  -0.000  -0.000  -0.000  0.250    |     Z =0.250
--------------------------------------------------------------|------------------|
Zj-Cj Maksimum = 0.272
Rasio Minimum = 0.107
Kolom kunci   = 1
Baris kunci   = 2
Elemen Pivot  = 1.585
```

Gambar 4.28 (b) Iterasi ke-1 matriks *payoff* Richeese Factory

```
ITERASI 2

--------------------------------------------------------------|------------------|
|      |  Cj  |    1      1      1      0      0      0       |         |        |
|  Cb  |------|-----------------------------------------------| Indeks  | Rasio  |
|      | Basis |   Y1     Y2     Y3     S1     S2     S3      |         |        |
--------------------------------------------------------------|------------------|
|  0   |  S1  |   0.000  -0.422  0.000  1.000  -1.153  0.162  |  0.009  | 0.009  |
|  1   |  Y1  |   1.000  0.268  0.000  0.000  0.631  -0.524   |  0.107  | Inf    |
|  1   |  Y3  |   0.000  0.827  1.000  0.000  -0.459  0.631   |  0.172  | Inf    |
--------------------------------------------------------------|------------------|
|    Zj-cj    |   -0.000  0.096  -0.000  -0.000  0.172  0.107    |     Z =0.279
--------------------------------------------------------------|------------------|
Zj-Cj Maksimum = 0.000
Rasio Minimum = 0.009
Kolom kunci   = 4
Baris kunci   = 1
Elemen Pivot  = 1.000
```

Gambar 4.28 (c) Iterasi ke-2 matriks *payoff* Richeese Factory

Karena nilai $Z_j - C_j$ sudah tidak ada yang bernilai negatif, maka tahap itetasi selesai dengan diperoleh hasil akhir iterasi simpleks, yaitu iterasi ke-2 Richeese Factory.

5. Menentukan strategi optimal masing-masing pemain

Berdasarkan Tabel 4.11, seluruh baris dan kolom dapat diabaikan sehingga strategi optimal untuk Richeese Factory, yaitu: $Y_1 = 0{,}107$, $Y_2 = 0$, $Y_3 = 0{,}172$ dengan $Y_1 = q_1$, $Y_2 = q_3$, $Y_3 = q_4$, dan $Z = 0{,}279$. Peluang strategi optimal untuk Richeese Factory adalah sebagai berikut.

Peluang $q_1 = \frac{Y_1}{Z} = \frac{0{,}107}{0{,}279} = 0{,}3835 = 38{,}35\ \%$

Peluang $q_3 = \frac{Y_2}{Z} = \frac{0}{0{,}279} = 0 = 0\ \%$

Peluang $q_4 = \frac{Y_3}{Z} = \frac{0{,}172}{0{,}279} = 0{,}6164 = 61{,}64\ \%$

dengan nilai permainan $V = \frac{1}{Z} = \frac{1}{0{,}279} = 3{,}584$

Sementara itu, strategi optimal untuk KFC, diperoleh $X_1 = 0$, $X_2 = 0{,}172$, $X_3 = 0{,}107$, dan $Z = W = 0{,}279$. Peluang strategi optimal untuk McDonald's adalah sebagai berikut.

Peluang $p_1 = \frac{X_1}{W} = \frac{0}{0{,}279} = 0 = 0\ \%$

Peluang $p_2 = \frac{X_2}{W} = \frac{0{,}172}{0{,}279} = 0{,}6164 = 61{,}64\ \%$

Peluang $p_4 = \frac{X_3}{W} = \frac{0{,}107}{0{,}279} = 0{,}3835 = 38{,}35\%$

dengan nilai permainan $V = \frac{1}{W} = \frac{1}{0{,}279} = 3{,}584$.

2. Penyelesaian matriks *payoff* dengan menggunakan algoritma *brown*

Penyelesaian matriks *payoff* KFC menggunakan algoritma *brown* akan dilakukan dengan program Matlab. Berikut adalah langkah – langkah penyelesaian matriks *payoff* KFC.

1. Masukan Matriks

Masukan matriks yang diperoleh dari kuisioner pada lampiran 3 seperti pada Gambar 4.25.

2. Menentukan *saddle point*

Hasil *saddle point* dengan program yang dibuat dengan software MATLAB. dapat dilihat pada Gambar 4.26.

3. Menyelesaikan matriks *payoff* dengan menggunakan algoritma *brown.*

Penyelesaian dengan menggunakan algoritma *brown,* dapat dilihat pada Gambar 4.29.

baris_iterasi =

2.9100	3.4100	3.7300	4.0500	3.4500
7.0500	7.1400	7.0500	7.2300	6.9500
11.0500	10.9100	10.8700	10.5500	10.6800
15.0500	14.6800	14.6900	13.8700	14.4100
17.9600	18.0900	18.4200	17.9200	17.8600
21.9600	21.8600	22.2400	21.2400	21.5900
24.8700	25.2700	25.9700	25.2900	25.0400
28.8700	29.0400	29.7900	28.6100	28.7700
31.7800	32.4500	33.5200	32.6600	32.2200
35.7800	36.2200	37.3400	35.9800	35.9500

kolom_iterasi =

4.1400	7.6400	10.8200	14.0000	17.5000	20.6800	24.8200	28.0000	32.1400	36.2800
4.0000	7.7300	11.0500	14.3700	18.1000	21.4200	25.4200	28.7400	32.7400	36.7400
3.6400	6.7800	10.4600	14.1400	17.2800	20.9600	24.6000	28.2800	31.9200	35.5600
2.9100	6.3600	10.3600	14.3600	17.8100	21.8100	24.7200	28.7200	31.6300	34.5400
2.9100	6.3600	10.4100	14.4600	17.9100	21.9600	24.8700	28.9200	31.8300	34.7400

Gambar 4.29 Iterasi algoritma *brown* matriks *payoff* Richeese Factory

Dari iterasi gambar 4.29 akan didapat nilai batas atas $(\overline{V})$ dan batas bawah $(\underline{V})$, yaitu dimana $(\underline{V}) \leq V \leq (\overline{V})$, seperti Gambar 4.30.

batasatas =

3.6740

batasbawah =

3.5780

Gambar 4.30 Hasil nilai permainan, batas atas, dan batas bawah matriks *payoff* Richeese Factory

Lampiran 8. Source Code

```
clear all;
clc;
%A=[3.97 3.88 3.99 3.87 3.92;3.86 4.26 4.08 3.90
4.31;3.77 3.91 4.06 3.88 4.33;3.87 3.79 3.87 4.36
4.24;3.76 4.10 3.81 3.88 4.03]
%AA=[2.18 2.03 2.00 1.82 1.95;2.13 2.09 2.04 1.89
2.03;2.00 2.15 2.09 2.03 1.92;1.80 1.92 2.15 2.07
2.00;1.95 1.95 2.07 2.15 2.00]
%B=[4.00 3.80 4.03 4.00 3.83;3.97 4.37 4.00 3.97
4.43;4.00 3.83 4.17 4.07 4.40;4.17 3.90 3.97 4.47
4.40;3.97 4.20 4.07 4.43 4.00]
%BB=[4.00 3.47 3.37 3.07 3.30;3.73 3.67 3.57 3.47
3.77;3.53 3.77 3.77 3.50 3.33;2.93 3.27 3.77 3.80
3.27;3.10 3.40 3.63 3.80 3.27]
%C=[4.23 3.86 3.95 3.86 3.73;4.09 4.32 3.91 3.77
4.50;4.09 3.68 4.09 4.05 4.45;4.32 3.77 3.82 4.45
4.45;4.09 4.09 4.00 4.50 4.09]
%CC=[4.14 3.73 3.32 3.18 3.50;4.00 3.77 3.82 3.32
3.73;3.64 4.09 4.00 3.68 3.14;2.91 3.41 4.09 4.00
3.45;2.91 3.41 3.73 4.05 3.45]
fprintf('Pilih Metode Penyelesaian \n');
fprintf('1. Algoritma Brown \n');
fprintf('2. Metode Simpleks \n');
anas=input('Pilihan = ');
if anas==1
A=input('input matriks payoff =');
%iterasi=100
iterasi=input('jumlah iterasi =');
bar=length(A(1,:));
kol=length(A(:,1));
%% Pencarian Saddle Point
for j=1:bar
dumy1=0;
for i=1:kol
if dumy1<A(i,j)
   dumy1=A(i,j);
end
end
maks(j)=dumy1;
end
minimaks=min(maks);
```

```
for j=1:kol
dumy2=10;
for i=1:bar
if dumy2>A(j,i)
    dumy2=A(j,i);
end
end
mini(j)=dumy2;
end
maksimin=max(mini);
minimaks
maksimin
if minimaks~=maksimin
matriks_payoff=A
for j=1:length(A(:,1))
jumbarA(j)=sum(A(j,:));
end
jumlah_baris=jumbarA
minjumbarA=min(jumbarA);
jumlah_terkecil=minjumbarA
indeks1=find(jumbarA==minjumbarA);
B=A(indeks1,:);
baris_iterasi=B
minbarA=min(A(indeks1,:));
indeks2=find(A(indeks1,:)==minbarA);
aa=length(indeks2);
indeks3=randi(aa);
koldipilih=indeks2(indeks3);
kolom_dipilih=koldipilih;
C=A(:,koldipilih);
kolom_iterasi=C
for i=1:iterasi-1
aa=length(indeks2);
indeks3=randi(aa);
bardipilih=indeks2(indeks3);
B((i+1),:)=B(i,:)+A(bardipilih,:);
baris_iterasi=B
minbar=min(B(i+1,:));
nilai_terkecil_dari_baris=minbar
indeks2=find(B(i+1,:)==minbar);
aa=length(indeks2);
indeks3=randi(aa);
koldipilih=indeks2(indeks3);
```

```
C(:,i+1)=C(:,i)+A(:,koldipilih);
kolom_iterasi=C
maks1=max(C(:,i+1));
nilai_terbesar_dari_kolom=maks1
indeks2=find(C(:,i+1)==maks1);
end
%% cari X'
for j=1:length(C(1,:))
dummy(j)=0;
for i=1:length(C(:,j))
if C(i,j)>dummy(j)
    dummy(j)=C(i,j);
end
end
end
for j=1:length(C(1,:))
indeks6=[];
indeks6=find(C(:,j)==dummy(j));
aaa=length(indeks6);
indeks7=randi(aaa);
indeks8(j)=indeks6(indeks7);
end
for i=1:length(C(:,1))
indeks9=find(indeks8==i);
Xaks(i)=length(indeks9)/length(C(1,:));
end
Xaks=transpose(Xaks);
X_bintang=Xaks
%% cari Y'
for j=1:length(B(:,1))
dummy2(j)=1000000;
for i=1:length(B(j,:))
if B(j,i)<dummy2(j);
    dummy2(j)=B(j,i);
end
end
end
for j=1:length(B(:,1))
indeks10=[];
indeks10=find(B(j,:)==dummy2(j));
bbb=length(indeks10);
indeks11=randi(bbb);
indeks12(j)=indeks10(indeks11);
```

```
end
for i=1:length(B(1,:))
indeks13=find(indeks12==i);
Yaks(i)=length(indeks13)/length(B(:,1));
end
Y_bintang=Yaks
for i=1:length(Xaks)
for j=1:length(Yaks)
V(i,j)=Xaks(i)*Yaks(j)*A(i,j);
end
end
V=sum(V);
V=sum(V)
batasatas=dummy(iterasi)/iterasi
batasbawah=dummy2(iterasi)/iterasi
else
fprintf('ditemukan saddle point dengan nilai
permainan = %d\n',minimaks)
end
elseif anas==2
A=input('input matriks =')
%%A=B;
% A=CC;
% % A=C;
maxiter=200;
maxconv=20;
iter=1;
conv=0;
stopp=0;
Aasli=A;
display=1;
Matriks_payoff=A
while iter <= maxiter && conv <= maxconv &&
stopp~=1
fprintf('Iterasi: %d',iter);
A=Aasli;
b=length(A(1,:));
c=length(A(:,1));
%% Pencarian Saddle Point
for j=1:b
dummy1=0;
for i=1:c
if dummy1<A(i,j)
```

```
        dummy1=A(i,j);
    end
    end
    maks(j)=dummy1;
    end
    minimaks=min(maks);
    for j=1:c
    dummy2=10;
    for i=1:b
    if dummy2>A(j,i)
        dummy2=A(j,i);
    end
    end
    mini(j)=dummy2;
    end
    maksimin=max(mini);
    if minimaks~=maksimin
    %% Dominasi baris
    while b>3||c>3
    if c>3
    indeks1=[];
    for i=1:b
        dummy3=max(A(:,i));
        indeks111=find(A(:,i)==dummy3);
        indeks111=transpose(indeks111);
        indeks1=[indeks1 indeks111];
    end
    d=[1:c];
    for i=1:length(indeks1)
        d=d(d~=indeks1(i));
    end
    dsd=[d 1];
    if length(dsd)==1
        for i=1:c
            indeks22(i)=sum(A(i,:));
        end
        indeks23=min(indeks22);
        indeks24=find(indeks22==indeks23);
        A(indeks24,:)=[];
    else
        dd=randi(length(d));
        d=d(dd);
        A(d,:)=[];
```

```
end
end
if b>3
%% Dominasi kolom
b=length(A(1,:));
c=length(A(:,1));
indeks2=[];
for i=1:c
    dummy3=min(A(i,:));
    indeks112=find(A(i,:)==dummy3);
    indeks2=[indeks2 indeks112];
end
d2=[1:b];
for i=1:length(indeks2)
    d2=d2(d2~=indeks2(i));
end
dsd2=[d2 1];
if length(dsd2)==1
    for i=1:b
        indeks25(i)=sum(A(:,i));
    end
    indeks26=max(indeks25);
    indeks27=find(indeks25==indeks26);
    A(:,indeks27)=[];
else
    dd2=randi(length(d2));
    d2=d2(dd2);
    A(:,d2)=[];
end
end
b=length(A(1,:));
c=length(A(:,1));
end
%% Nyari Z NYA
%% Simpleks
pil=2;
n=3;
i=1;
%==================== INPUT DAN OUTPUT FUNGSI TUJUAN
=======================
if pil==1
c=-1*ones(1,n);
else
```

```
c=ones(1,n);
end
pil_lama=pil;
pil=2;
%===================== INPUT DAN OUTPUT FUNGSI
KENDALA ======================
m=3;
for i=1:m
for j=1:n                                    %
k(i,j)=A(i,j);   % input koefisien
end                                          % fungsi
kendala
if pil_lama==1
k(i,n+m+1)=-1;
else
k(i,n+m+1)=1;
end
end;
%===================================================
========================
%===================================================
========================

%======== DEKLARASI KOEFISIEN VARIABEL SLACK DARI
SUATU KENDALA ==========
for i=1:m
for j=1:m
if i==j
   k(i,j+n)=1;   %
else              % menyatakan elemen S tiap kendala
variabel slack
   k(i,j+n)=0;   %
end;
end;
end;
%%
===================================================
=======================

%====== DEKLARASI KOEFISIEN VARIABEL SLACK DARI
FUNGSI TUJUAN ============
for i=1:m        %
```

```
c(i+n)=0;   % menyatakan nilai koefisien S (nilai
koefisien slack)
end;
%=====================================================
========================
%==========DEKLARASI ELEMEN DAN VARIABEL BASIS PADA
TABEL SIMPLEK ==========
for i=1:m
cb(i)=c(n+i); % koefisien basis
var_cb(i)='S';
no_cb(i)=i;     % nomor variabel S1, S2,...,Sm
end;
%=====================================================
========================
%% ============== PERHITUNGAN NILAI Zj PADA TABEL
SIMPLEK ===================
for j=1:n+m+1
T(1)=0;
for v=1:m
    F(j,v)=(cb(v)*k(v,j));
    T(v+1)=T(v)+F(j,v);
end;
Z(j)=T(m+1);
end;
for j=1:n+m
ZC(j)= Z(j)-c(j); %selisih Zj-Cj
end;
%=====================================================
=======================
%proses ke iterasi nol sampai n (penampilan simplek
dengan disertai RATIO)
%=====================================================
========================
mulai=1;
if pil==1
ZC_m = max(ZC); %ZC_mini nilai maksimum ZC
else
ZC_m = min(ZC); %ZC_mini nilai minimum ZC  %
end
for i=1:m+n
if ZC(i)==ZC_m    %
kol_kun(1)=i;       % mencari letak kolom ZC_mi
ZC(minimum/maksimum)
```

```
end
end
for i=1:m
RHS(i,1)=k(i,m+n+1);
RATIO(i,1)=RHS(i,1)/k(i,kol_kun(1)); %mencari rasio
end;
RATIO_SAVE=RATIO; % MENYIMPAN NILAI RASIO
%====== PROSES MENCARI RASIO MINIMUM YANG LEBIH
BESAR SAMA DENGAN NOL =====
if m==1
RM=RATIO(1,1);
else
for i=1:m-1
if RATIO(i,1)>=0
for j=i+1:m
if RATIO(j,1)>=0
   if RATIO(i,1)<=RATIO(j,1)
     RM=RATIO(i,1);
     RATIO(j,1)=RM;
   else
     RM=RATIO(j,1);
     RATIO(j,1)=RM;
   end;
else
   RM=RATIO(i,1);
   RATIO(j,1)=RM;
end;

end;
else
for y=1:m
if RATIO(y,1)<0
  RM=min(RATIO(:,1));
end;
end;
end;
end;
end;
%===================================================
========================
RATIO=RATIO_SAVE; % MENNGEMBALIKAN NILAI RASIO DARI
TABEL
```

```
for i=1:m
if RATIO(i,1)==RM
bar_kun(1)=i; %mencari letak baris minimum rasio
terkecil
end;
end;
if pil==1
ZC_proses = -ZC_m;
else
ZC_proses = ZC_m;
end
while ZC_proses<0;
if RM<0  || RM== inf
fprintf('Tidak ditemukan solusi yang optimal');
ZC_m=1;
break;
else
for j=1:n+m+1
    T(1)=0;
    for  v=1:m
        F(j,v)=(cb(v)*k(v,j));
        T(v+1)=T(v)+F(j,v);
    end;
     Z(j)=T(m+1);
end;
for j=1:n+m
ZC(j)= Z(j)-c(j); %selisih Zj-Cj
end;
%====================proses minimasi atau
maksimasi==========================
if pil==1
ZC_m = max(ZC);
else
ZC_m = min(ZC); %ZC_mi nilai minimum ZC
end;
==============================================
=======================
for i=1:m+n
if ZC(i)==ZC_m
kol_kun(mulai)=i;
end; %mencari letak kolom ZC_mi
ZC(minimum/maksimum)
end;
```

```
for i=1:m
RHS(i,1)=k(i,m+n+1);
RATIO(i,1)=RHS(i,1)/k(i,kol_kun(mulai)); %mencari
rasio
end;
RATIO_SAVE=RATIO;%untuk save ratio sebelumnya
if m==1
RM=RATIO(1,1);
else
for i=1:m-1
if RATIO(i,1)>=0
for j=i+1:m
if RATIO(j,1)>=0
    if RATIO(i,1)<=RATIO(j,1)
      RM=RATIO(i,1);  RATIO(j,1)=RM;
    else
      RM=RATIO(j,1);  RATIO(j,1)=RM;
    end;
else
    RM=RATIO(i,1); RATIO(j,1)=RM;
end;
end;
else
for y=1:m
if RATIO(y,1)<0
  RM=min(RATIO(:,1));
end;
end;
end;
end;
end;
RATIO=RATIO_SAVE;

for i=1:m
if RATIO(i,1)==RM
bar_kun(mulai)=i; %mencari letak baris minimum
rasio terkecil
end;
end;
ev=k(bar_kun(mulai),kol_kun(mulai)); %elemen pivot
end
%matriks baru b(i,j) adalah proses iterasi
```

```
for i=1:m %baris
for j=1:m+n+1 %kolom
if i==bar_kun(mulai) && j==kol_kun(mulai)
b(bar_kun(mulai),kol_kun(mulai))=1;
elseif i==bar_kun(mulai) && j~=kol_kun(mulai)
b(bar_kun(mulai),j)=k(bar_kun(mulai),j)/ev;
elseif i~=bar_kun(mulai) && j==kol_kun(mulai)
b(i,kol_kun(mulai))=0;
else
    b(i,j)=k(i,j)-
(k(i,kol_kun(mulai))*k(bar_kun(mulai),j)/ev);
end;
end;
end;
for i=1:m
for j=1:mulai
if i==bar_kun(j)
cb(i)=c(kol_kun(j)); %c basis baru
var_cb(i)='Y';
no_cb(i)=kol_kun(j);
end;
end;
end;
k=b; %mengganti elemen matriks k menjadi elemen
matriks b
mulai=mulai+1;
if pil==1
ZC_proses = -ZC_m;
else
ZC_proses = ZC_m;
end
end;
disp(' ');
% Cek apa Zmin berubah
if iter == 1
Zmin = Z(n+m+1);
Amin = A;
end
if Z(n+m+1) < Zmin
Zmin = Z(n+m+1);
Amin = A;
conv = 0;
end
```

```
if Z(n+m+1) == Zmin
conv = conv+1;
end
iter=iter+1;
fprintf('Z Maksimum iterasi yaitu Z= %3.8f
\n',Z(n+m+1));
fprintf('Convergensi: %d\n',conv);
fprintf('Z minimum Z= %3.3f \n',Zmin);
disp(Amin);
elseif minimaks==maksimin
stopp=1;
disp('Saddle Point telah diperoleh dan Optimal');
display=0;
end
end
%% Perintah mencari Simpleks
%% Menampilkan A dengan Z minimum
if display==1 && minimaks~=maksimin
%%Menampilkan A hasil dominansi
minimaks
maksimin
Matriks_hasil_dominasi=A
pil=2;
%=============================================================
========================
n=3;
i=1;
%================== INPUT DAN OUTPUT FUNGSI TUJUAN
=======================
if pil==1
disp(' Minimasi Menggunakan Metode Dual ')
disp(' Sehingga minZ=-max(Z)')
c=-1*ones(1,n);
else
c=ones(1,n);
end
pil_lama=pil;
pil=2;
fprintf('\nFUNGSI TUJUAN\nZ= ');
if pil_lama==1
fprintf('%d*X%d ',c(1),1);
for i=1:n-1
%menentukan C(i)
```

```
if c(i+1)>=0
fprintf('+ %d*X%d ',c(i+1),i+1);   % menampilkan
fungsi tujuan
else
fprintf('- %d*X%d ',abs(c(i+1)),i+1);
end
end;
else
fprintf('%d*Y%d ',c(1),1);
for i=1:n-1
%menentukan C(i)
if c(i+1)>=0
fprintf('+ %d*Y%d ',c(i+1),i+1);   % menampilkan
fungsi tujuan
else
fprintf('- %d*Y%d ',abs(c(i+1)),i+1);
end
end;
end
fprintf('\n\n')                %garis baru
%===================================================
===========================
%==================== INPUT DAN OUTPUT FUNGSI
KENDALA =====================
m=3;
fprintf('FUNGSI KENDALA\n');
i=1;
while (i<n)                                          %
if pil_lama==1
fprintf('k%d*X%d + ',i,i);
else
fprintf('k%d*Y%d + ',i,i);
end% menampilkan contoh
i=i+1;                              % fungsi
kedala
end;
if pil_lama==1
fprintf('k%d*X%d <=  %d\n',n,n,-1);%
else
fprintf('k%d*Y%d <=  %d\n',n,n,1);%
end
for i=1:m
fprintf('\nKendala %d\n',i);
```

```
for j=1:n                                              %
fprintf('k%d',j); k(i,j)=A(i,j);  %    input
koefisien
end;                                                   %
fungsi kendala
if pil_lama==1
k(i,n+m+1)=-1;
else
k(i,n+m+1)=1;
end
end;
%====================================================
========================
%clc;
fprintf('\n================== PROSES SIMPLEK
========================\n');
%================== OUTPUT FUNGSI TUJUAN DAN KENDALA
=======================
fprintf('\nFUNGSI TUJUAN\nZ= ');
%-----------------------------------
if pil_lama==1
fprintf('%d*X%d ',c(1),1);
for i=1:n-1
%menentukan C(i)
if c(i+1)>=0
fprintf('+ %d*X%d ',c(i+1),i+1); % menampilkan
fungsi tujuan
else
fprintf('- %d*X%d ',abs(c(i+1)),i+1);
end
end;
else
fprintf('%d*Y%d ',c(1),1);
for i=1:n-1
%menentukan C(i)
if c(i+1)>=0
fprintf('+ %d*Y%d ',c(i+1),i+1); % menampilkan
fungsi tujuan
else
fprintf('- %d*Y%d ',abs(c(i+1)),i+1);
end
end;
end
```

```
fprintf('\n\n')
fprintf('\n ')
fprintf('\nFUNGSI KENDALA');
for i=1:m
fprintf('\nKendala %d\n',i);
%-----------------------------------------------------
if pil_lama==1
fprintf('%d*X%d ',k(i,1),1);
for j=1:n-1                             % menampilkan
fungsi kendala
if k(i,j)>=0
fprintf(' + %d*X%d ',k(i,j+1),j+1);      %
else
   fprintf(' - %d*X%d ',abs(k(i,j+1)),j+1);
end
end;
else
fprintf('%d*Y%d ',k(i,1),1);
for j=1:n-1                             % menampilkan
fungsi kendala
if k(i,j)>=0
fprintf(' + %d*Y%d ',k(i,j+1),j+1);      %
else
   fprintf(' - %d*Y%d ',abs(k(i,j+1)),j+1);
end
end;
end
fprintf(' <=  %d',k(i,n+m+1));
end;
%=========================================================
=======================
%========= DEKLARASI KOEFISIEN VARIABEL SLACK DARI
SUATU KENDALA ===========
for i=1:m
for j=1:m
if i==j
   k(i,j+n)=1;   %
else                   % menyatakan elemen S tiap kendala
variabel slack
   k(i,j+n)=0;   %
end;
end;
end;
```

```
%%
====================================================================
========================
%====== DEKLARASI KOEFISIEN VARIABEL SLACK DARI
FUNGSI TUJUAN ============
for i=1:m
c(i+n)=0; % menyatakan nilai koefisien S (nilai
koefisien slack)
end;
%===================================================================
========================
%=========DEKLARASI ELEMEN DAN VARIABEL BASIS PADA
TABEL SIMPLEK =========
for i=1:m
cb(i)=c(n+i);   %#ok % koefisien basis
var_cb(i)='S'; %#ok
no_cb(i)=i;     % nomor variabel S1, S2,...,Sm
end;
%===================================================================
========================
%% ============== PERHITUNGAN NILAI Zj PADA TABEL
SIMPLEK ================
for j=1:n+m+1
T(1)=0;
for v=1:m
    F(j,v)=(cb(v)*k(v,j));
    T(v+1)=T(v)+F(j,v);
end;
Z(j)=T(m+1);
end;
for j=1:n+m
ZC(j)= Z(j)-c(j); %selisih Zj-Cj
end;
%===================================================================
========================
%% ============== TABEL SIMPLEK AWAL DARI INPUTAN
===========================
fprintf('\n\nTabel Simplek\n\n ')
fprintf('---------------'); for i=1:m+n ,fprintf('-
---');end; fprintf('------------|\n');
fprintf('|        |    Cj   |');for i=1:n+m
,fprintf('%4d',c(i)); end; fprintf('      |
|\n');
```

```
fprintf('|  Cb  |-------|');for i=1:m+n ,fprintf('----');end; fprintf('----|Indeks |\n');
fprintf('|      | Basis |');
if pil_lama==1
for i=1:n   ,fprintf('   X%d',i); end;
else
for i=1:n   ,fprintf('   Y%d',i); end;
end;
for j=1:m ,fprintf('   S%d',j);end;fprintf('     |\n');
fprintf('----------------'); for i=1:m+n ,fprintf('----');end; fprintf('-------------|\n');
for i=1:m
fprintf('|%3d   |    %s%d  |',cb(i),var_cb(i),i);
for j=1:n+m ,fprintf('%4d',k(i,j)); end; fprintf('  |%4d     |\n',k(i,n+m+1));
end;
fprintf('----------------');for i=1:m+n  ,fprintf('----');end; fprintf('-------------|\n');
fprintf('|      Zj-Cj      |');for i=1:n+m ,fprintf('%4d',ZC(i)); end;fprintf('  | Z =%2d |\n',Z(n+m+1));
fprintf('----------------');for i=1:m+n  ,fprintf('----');end; fprintf('-------------|\n');
%=====================================================================================
%proses ke iterasi nol sampai n (penampilan simplek dengan disertai RATIO)
%=====================================================================================
mulai=1;
if pil==1
ZC_m = max(ZC); %ZC_mini nilai maksimum ZC
else
ZC_m = min(ZC); %ZC_mini nilai minimum ZC
end;
for i=1:m+n
if ZC(i)==ZC_m  %
kol_kun(1)=i;       % mencari letak kolom ZC_mi ZC(minimum/maksimum)
end;
end;
for  i=1:m
```

```
RHS(i,1)=k(i,m+n+1);
RATIO(i,1)=RHS(i,1)/k(i,kol_kun(1)); %mencari rasio
end;
RATIO_SAVE=RATIO; % MENYIMPAN NILAI RASIO
%====== PROSES MENCARI RASIO MINIMUM YANG LEBIH
BESAR SAMA DENGAN NOL =====
if m==1
RM=RATIO(1,1);
else
for i=1:m-1
if RATIO(i,1)>=0
for j=i+1:m
if RATIO(j,1)>=0
   if RATIO(i,1)<=RATIO(j,1)
     RM=RATIO(i,1);
     RATIO(j,1)=RM;
   else
     RM=RATIO(j,1);
     RATIO(j,1)=RM;
   end;
else
   RM=RATIO(i,1);
   RATIO(j,1)=RM;
end;
end;
else
for y=1:m
if RATIO(y,1)<0
    RM=min(RATIO(:,1));
end;
end;
end;
end;
end;
fprintf('Ratio Minimum = %3.3f',RM);
%==================================================
=========================
RATIO=RATIO_SAVE; % MENNGEMBALIKAN NILAI RASIO DARI
TABEL
for i=1:m
if RATIO(i,1)==RM
bar_kun(1)=i; %mencari letak baris minimum rasio
terkecil
```

```
end;
end;
if pil==1
ZC_proses = -ZC_m;
else
ZC_proses = ZC_m;
end
while ZC_proses<0;
if RM<0 || RM== inf
fprintf('Tidak ditemukan solusi yang optimal');
ZC_m=1;
break;
else
for j=1:n+m+1
T(1)=0;
for v=1:m
    F(j,v)=(cb(v)*k(v,j));
    T(v+1)=T(v)+F(j,v);
end;
Z(j)=T(m+1);
end;
for j=1:n+m
ZC(j)= Z(j)-c(j); %selisih Zj-Cj
end;
%====================proses minimasi atau
maksimasi==========================
if pil==1
ZC_m = max(ZC);
else
ZC_m = min(ZC); %ZC_mi nilai minimum ZC
end;
%==================================================
========================
for i=1:m+n
if ZC(i)==ZC_m
kol_kun(mulai)=i;
end; %mencari letak kolom ZC_mi
ZC(minimum/maksimum)
end;
for i=1:m
RHS(i,1)=k(i,m+n+1);
RATIO(i,1)=RHS(i,1)/k(i,kol_kun(mulai)); %mencari
rasio
```

```
end;
RATIO_SAVE=RATIO;%untuk save ratio sebelumnya
if m==1
RM=RATIO(1,1);
else
for i=1:m-1
if RATIO(i,1)>=0
for j=i+1:m
if RATIO(j,1)>=0
   if RATIO(i,1)<=RATIO(j,1)
     RM=RATIO(i,1);  RATIO(j,1)=RM;
   else
     RM=RATIO(j,1);  RATIO(j,1)=RM;
   end;
else
   RM=RATIO(i,1);  RATIO(j,1)=RM;
end;
end;
else
for y=1:m
if RATIO(y,1)<0
  RM=min(RATIO(:,1));
end;
end;
end;
end;
end;
RATIO=RATIO_SAVE;
for i=1:m
if RATIO(i,1)==RM
bar_kun(mulai)=i; %mencari letak baris minimum
rasio terkecil
end;
end;
ev=k(bar_kun(mulai),kol_kun(mulai)); %elemen pivot
%====================== MENAMPILKAN PROSES SIMPLEK
=========================

fprintf('\n\nITERASI %d \n\n',mulai-1)
fprintf('-----------------'); for i=1:m+n
,fprintf('---------');end; fprintf('|----------------
----|\n');
```

```
fprintf('|          |    Cj     |');for i=1:n+m ,
fprintf('%6d ',c(i));end; fprintf('          |
|            |\n');
fprintf('|  Cb   |--------|');for i=1:m+n ,
fprintf('---------');end; fprintf('| Indeks    |
Rasio  |\n');
fprintf('|           | Basis |');
if pil_lama==1
for i=1:n  ,fprintf('     X%d ',i); end;
else
for i=1:n   ,fprintf('     Y%d ',i); end;
end;
for j=1:m ,fprintf('     S%d',j);end;fprintf('
|           |            |\n');
fprintf('----------------'); for i=1:m+n
,fprintf('---------');end; fprintf('|----------------
----|\n');
for i=1:m ,fprintf('|%3d    |   %s%d    |
',cb(i),var_cb(i),no_cb(i)); for j=1:n+m
,fprintf('  %5.3f ',k(i,j)); end; fprintf('  |
%3.3f  | %3.3f  |\n',k(i,n+m+1),RATIO(i,1));end;
fprintf('-------------------'); for i=1:m+n
,fprintf('---------');end; fprintf('|----------------
-----|\n');
fprintf('|        Zj-cj        |   ');
for i=1:n+m
if ZC(i)==0
fprintf('  %5.3f',-ZC(i));
else
fprintf('  %5.3f',ZC(i));
end;
end
fprintf('     |     Z =%3.3f      |\n',Z(n+m+1));
end
fprintf('-------------------'); for i=1:m+n
,fprintf('---------');end; fprintf('|----------------
-----|\n');
%===================================================
=========================
if pil==1
if ZC_m==0
fprintf('Zj-Cj Minimum = %3.3f\n',ZC_m);
else
```

```
fprintf('Zj-Cj Minimum = %3.3f\n',-ZC_m);
end
else
if ZC_m==0
fprintf('Zj-Cj Maksimum = %3.3f\n',ZC_m);
else
fprintf('Zj-Cj Maksimum = %3.3f\n',-ZC_m);
end
end;
fprintf('Rasio Minimum = %3.3f\n',RM);
fprintf('Kolom kunci    = %d\n',kol_kun(mulai));
fprintf('Baris kunci    = %d\n',bar_kun(mulai));
fprintf('Elemen Pivot   = %3.3f\n',ev);
%===================================================
=========================
%matriks baru b(i,j) adalah proses iterasi
for i=1:m %baris
for j=1:m+n+1 %kolom
if      i==bar_kun(mulai) && j==kol_kun(mulai)
b(bar_kun(mulai),kol_kun(mulai))=1;
elseif i==bar_kun(mulai) && j~=kol_kun(mulai)
b(bar_kun(mulai),j)=k(bar_kun(mulai),j)/ev;
elseif i~=bar_kun(mulai) && j==kol_kun(mulai)
b(i,kol_kun(mulai))=0;
else
      b(i,j)=k(i,j)-
(k(i,kol_kun(mulai))*k(bar_kun(mulai),j)/ev);
end;
end;
end;
for i=1:m
for j=1:mulai
if i==bar_kun(j)
cb(i)=c(kol_kun(j)); %c basis baru
var_cb(i)='Y';
no_cb(i)=kol_kun(j);
end;
end;
end;
k=b; %mengganti elemen matriks k menjadi elemen
matriks b
mulai=mulai+1;
if pil==1
```

```
ZC_proses = -ZC_m;
else
ZC_proses = ZC_m;
end

end;
if pil_lama==1
fprintf('\n\nJadi Z Maksimum dari hasil Dual yaitu
Z= %3.3f',Z(n+m+1));
if Z(n+m+1)==0
fprintf('\nJadi Z Minimumnya adalah Z=
%3.3f',Z(n+m+1));
else
fprintf('\nJadi Z Minimumnya adalah Z= %3.3f',-
Z(n+m+1));
end
else
fprintf('\n\nZ Maksimum yaitu Z=
%3.3f\n',Z(n+m+1));
end;
end
else
fprintf('Tidak ada pilihan\n');
end
```