

**PENENTUAN KOMPOSISI PAKAN TERNAK UNTUK MEMENUHI
KEBUTUHAN NUTRISI AYAM PETELUR DENGAN BIAYA MINIMUM
MENGUNAKAN ALGORITMA GENETIKA**

SKRIPSI

Laboratorium Komputasi Cerdas dan Visualisasi

Untuk memenuhi sebagian persyaratan mencapai gelar Sarjana Komputer



Disusun oleh

ERVINA MARGININGTYAS

NIM. 115060800111085

**PROGRAM STUDI INFORMATIKA/ ILMU KOMPUTER
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA**

MALANG

2015

LEMBAR PERSETUJUAN

**PENENTUAN KOMPOSISI PAKAN TERNAK UNTUK MEMENUHI
KEBUTUHAN NUTRISI AYAM PETELUR DENGAN BIAYA MINIMUM
MENGUNAKAN ALGORITMA GENETIKA**

**SKRIPSI
KONSENTRASI KOMPUTASI CERDAS DAN VISUALISASI**

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer



Disusun oleh :
ERVINA MARGININGTYAS
NIM. 115060800111085

Telah diperiksa dan disetujui oleh :

Dosen Pembimbing I

Dosen Pembimbing II

Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D.
NIP. 19720919 199702 001

Indriati, ST., M.Kom
NIK. 831013 06120035

LEMBAR PENGESAHAN

**PENENTUAN KOMPOSISI PAKAN TERNAK UNTUK MEMENUHI
KEBUTUHAN NUTRISI AYAM PETELUR DENGAN BIAYA MINIMUM
MENGUNAKAN ALGORITMA GENETIKA**

SKRIPSI

KONSENTRASI KOMPUTASI CERDAS DAN VISUALISASI

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer

Disusun oleh :

Ervina Marginingtyas

NIM. 115060800111085

Skripsi ini telah diuji dan dinyatakan lulus pada tanggal 5 Mei 2015

Penguji I

Penguji II

Drs. Achmad Ridok, M.Kom

NIP. 19680825 199403 1 002

Suprpto, S.T., M.T

NIP. 19710727 199603 1 001

Penguji III

Dian Eka Ratnawati, S.Si., M.Kom

NIP. 19730619 200212 2 001

Mengetahui

Ketua Program Studi Informatika/Illmu Komputer

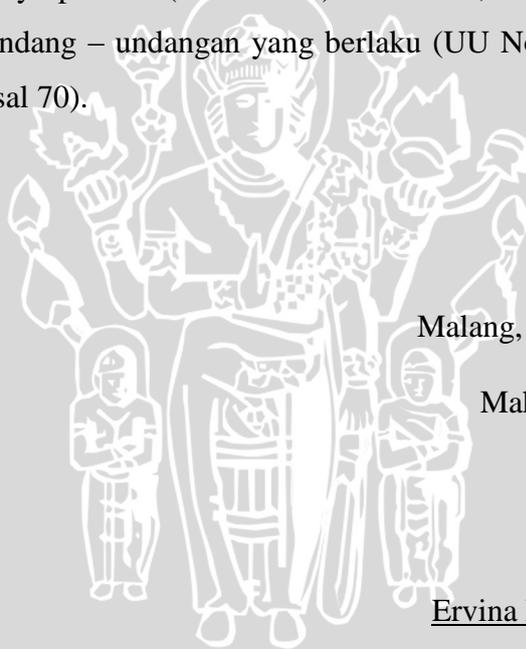
Drs. Marji, M.T.

NIP. 19670801 199203 1 001

Pernyataan Orisinalitas Skripsi

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah SKRIPSI ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur – unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang – undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).



Malang, 11 Mei 2015

Mahasiswa,

Ervina Marginingtyas

NIM. 115060800111085

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT, yang telah melimpahkan rahmat, taufik dan hidayah-Nya sehingga pada akhirnya penulis dapat menyelesaikan laporan skripsi dengan judul “Penentuan Komposisi Pakan Ternak Untuk Memenuhi Kebutuhan Nutrisi Ayam Petelur Dengan Biaya Minimum Menggunakan Algoritma Genetika”.

Pada kesempatan ini penulis menyampaikan ucapan terimakasih yang setulus-tulusnya kepada:

1. Bapak Wayan Firdaus Mahmudy, S.Si., MT.Ph.D selaku dosen pembimbing utama yang telah memberikan arahan, bimbingan, dan saran hingga selesainya laporan ini,
2. Ibu Indriati, ST., M.Kom selaku dosen pembimbing pendamping yang telah memberikan bimbingan selama pelaksanaan penelitian ini,
3. Bapak R. Arief Setyawan, S.T., M.T. selaku dosen pembimbing akademik yang telah banyak memberikan saran dan motivasi selama masa perkuliahan,
4. Kedua orang tua saya, Bapak Zainuddin dan Ibu Saropah yang telah memberi motivasi, kasih sayang serta dukungan moril serta materil. Kedua kakak saya, Mismawati dan Dewi Mistyaningsih, serta seluruh keluarga besar yang telah memberikan semangat dari awal sampai akhir pengerjaan skripsi ini,
5. Yasmin Ghassani Panhares, Nurul Khoirina, Elliya Lestari, Dina Rizky Anggraeni, dan Adinda Setyowulan atas bantuan, dukungan, kerja sama dan do'a selama menempuh studi di Informatika/Ilmu Komputer Universitas Brawijaya,
6. Ike Trio Setyawan yang telah memberikan dukungan berupa moril dan tenaga serta do'a dari awal pengerjaan hingga terselesaikannya tugas akhir ini,
7. Seluruh Civitas Akademika Informatika/Ilmu Komputer Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penulis menempuh studi di Teknik Informatika Universitas Brawijaya dan selama penyelesaian skripsi ini,

8. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun tidak langsung demi terselesaikannya tugas akhir ini.

Menyadari adanya keterbatasan pengetahuan, referensi dan pengalaman, penulis mengharapkan masukan, saran, dan kritik yang membangun untuk lebih memperbaiki laporan ini. Akhirnya, semoga laporan ini dapat bermanfaat bagi penulis ataupun bagi pihak yang membutuhkan.

Malang, Mei 2015

Penulis

UNIVERSITAS BRAWIJAYA



Abstrak

Ervina Marginingtyas. 2015 : Penentuan Komposisi Pakan Ternak Untuk Memenuhi Kebutuhan Nutrisi Ayam Petelur Dengan Biaya Minimum Menggunakan Algoritma Genetika. Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya, Malang.

Dosen Pembimbing : Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D dan Indriati, ST., M.Kom

Pemberian pakan sesuai kebutuhan nutrisi kepada ayam petelur sangat dianjurkan karena mempengaruhi kualitas ayam dalam bertelur. Terdapat enam nutrisi dasar yang harus terpenuhi dalam pakan ayam petelur diantaranya yaitu protein, ME, lemak, serat kasar, kalsium, dan fosfor. Dalam proses pemenuhan kebutuhan nutrisi ini para peternak membutuhkan biaya yang cukup besar setiap harinya. Sehingga peternak harus membuat kombinasi pakan yang sesuai agar memperoleh biaya yang minimum namun dengan nutrisi yang tercukupi. Untuk mendapatkan kombinasi pakan yang sesuai maka dilakukan penelitian dengan menggunakan metode algoritma genetika. Algoritma genetika merupakan salah satu metode optimasi yang dapat menyelesaikan permasalahan kombinasi pakan untuk memperoleh biaya yang minimum dengan nutrisi yang tercukupi. Penelitian ini menggunakan representasi real code dimana setiap kromosomnya memiliki panjang yang sejumlah dengan data bahan pakan yang digunakan yaitu 40. Setiap gen dalam sebuah kromosom mewakili bobot dari bahan pakan. Dari hasil pengujian didapatkan ukuran populasi terbaik adalah 160, banyaknya generasi optimal adalah 500, serta kombinasi Cr sebesar 0,9 dan Mr sebesar 0,1. Hasil akhir yang didapatkan adalah berupa kombinasi bahan pakan terbaik dengan nutrisi terpenuhi dan biaya minimum.

Kata kunci : Algoritma genetika, optimasi pakan ayam petelur.

Abstract

Ervina Marginingtyas. 2015 : *The Determination of Livestock Feed Composition to Fulfill The Needs of Laying Hens Nutrition With a Minimum Cost Using Genetic Algorithms. Information Technology and Computer Science Program, Brawijaya University, Malang.*

Advisor : Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D and Indriati, ST., M.Kom

Feeding of laying hens that appropriate to the nutritional needs is highly recommended because it affects the quality of the egg-laying chickens. There are six basic nutrients that must be fulfilled in the feed of laying hens among which the protein, ME, fat, crude fiber, calcium, and phosphorus. In process to fulfill the nutritional needs, chicken farmers need significant costs every day. So chicken farmers must make an appropriate combination of feeds in order to obtain the minimum cost with adequate nutrition. To obtain the appropriate feed combination research is conducted using a genetic algorithm. Genetic algorithm is one of the optimization methods that can solve the problems of feed combinations to obtain minimum cost with adequate nutrition. This research uses a real representation of code where each chromosomes have long number with the data feed material used is 40. Each gene in a chromosome represents the weight of the feed material. From the test results obtained the best population size is 160, the number of optimal generation is 500, and a combination of Cr are 0,9 and Mr are 0,1. The final result is a combination of the best feed ingredients with nutritional met and minimum cost.

Keywords : Genetic algorithms, optimization layer chicken feed.

DAFTAR ISI

LEMBAR PERSETUJUAN	i
LEMBAR PENGESAHAN	ii
Pernyataan Orisinalitas Skripsi	iii
KATA PENGANTAR	iv
Abstrak	vi
Abstract	vii
DAFTAR ISI	viii
Daftar Gambar	xii
Daftar Tabel	xiii
Daftar Persamaan	xiv
Daftar Lampiran	xv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Metodologi Penelitian	4
1.7 Sistematika Penulisan	4
BAB II DASAR TEORI	6
2.1 Kajian Pustaka	6
2.2 Ayam Petelur	7
2.3 Ransum	7
2.4 Nutrisi Pakan Ternak	8



2.4.1	Protein	9
2.4.2	Lemak	10
2.4.3	Serat Kasar	11
2.4.4	Energi Metabolis	11
2.4.5	Kalsium	12
2.4.6	Fosfor	13
2.5	Algoritma Genetika	14
2.4	Implementasi Algoritma Genetika	15
2.4.1	Representasi Kromosom	16
2.4.2	<i>Fitness</i>	16
2.4.3	Inisialisasi Populasi Awal	17
2.4.4	Operator Algoritma Genetika	17
2.4.4.1	Crossover	17
2.4.4.2	Mutasi	18
2.4.4.3	Seleksi	19
BAB III METODOLOGI PENELITIAN		20
3.1	Tahapan Penelitian	20
3.2	Analisa Kebutuhan Sistem	21
3.2.1	Deskripsi Umum Sistem	21
3.2.2	Daftar Kebutuhan Sistem	22
3.2.3	Data Yang Digunakan	22
3.3	Formulasi Permasalahan	23
3.4	Siklus Penyelesaian Masalah	23
3.4.1	Representasi Kromosom Dan Perhitungan <i>Fitness</i>	25
3.4.2	Inisialisasi Populasi Awal	31
3.4.3	<i>Crossover</i>	33

3.4.4	Mutasi	34
3.4.5	Evaluasi	36
3.4.6	Seleksi	38
BAB IV PERANCANGAN		40
4.1	Perancangan Database	40
4.2	Perancangan <i>User Interface</i>	41
4.3	Perancangan Uji Coba dan Evaluasi	43
4.3.1	Uji Coba Ukuran Populasi	43
4.3.2	Uji Coba Banyak Generasi	44
4.3.3	Uji Coba Kombinasi Crossover Rate dan Mutation Rate	45
BAB V IMPLEMENTASI		47
5.1	Lingkungan Implementasi	47
5.1.1	Lingkungan Perangkat Keras	47
5.1.2	Lingkungan Perangkat Lunak	47
5.2	Implementasi Program	47
5.2.1	Inisialisasi Populasi Awal	48
5.2.2	Perhitungan <i>Fitness</i>	48
5.2.3	Proses <i>Crossover</i> dan Mutasi	50
5.2.4	Proses Evaluasi	54
5.2.5	Proses Seleksi <i>Elitis</i>	54
5.3	Implementasi <i>User Interface</i>	55
BAB VI PENGUJIAN DAN ANALISA		58
6.1	Pengujian dan Analisa Ukuran Populasi	58
6.2	Pengujian dan Analisa Banyak Generasi	59
6.3	Pengujian dan Analisa Kombinasi <i>Crossover Rate</i> dan <i>Mutation Rate</i>	61



6.4 Analisa Perbandingan Kebutuhan Nutrisi Harian Ayam Petelur dengan Nutrisi yang Dihasilkan Kromosom Terbaik dari Hasil Optimasi 63

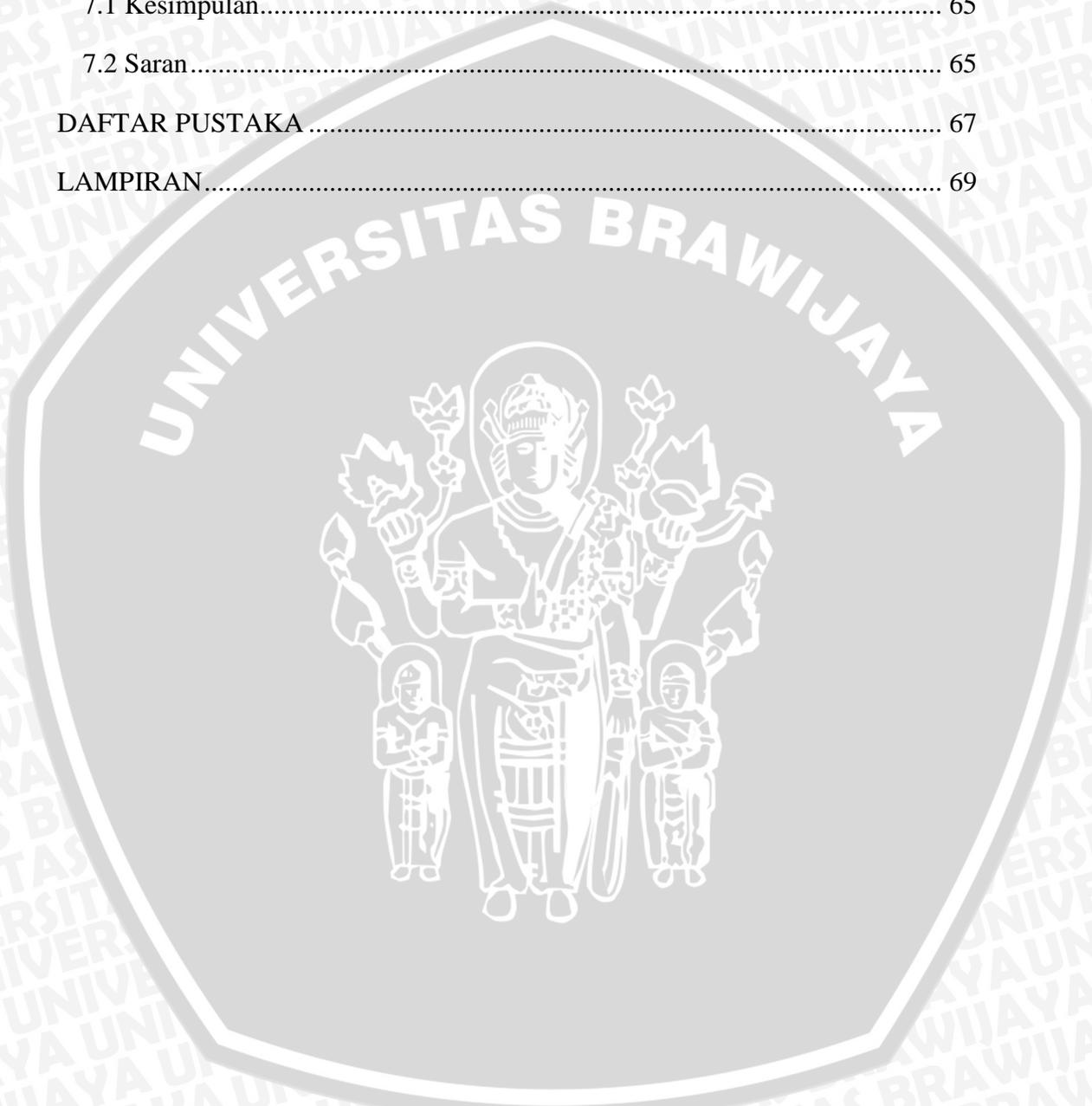
BAB VII KESIMPULAN DAN SARAN..... 65

7.1 Kesimpulan..... 65

7.2 Saran..... 65

DAFTAR PUSTAKA..... 67

LAMPIRAN..... 69



Daftar Gambar

Gambar 2.1 Metode <i>One Cut Point</i>	18
Gambar 2.2 Metode <i>Reciprocal Exchange</i>	19
Gambar 3.1 Diagram tahap penelitian	20
Gambar 3.2 <i>Flowchart</i> Algoritma Genetika	24
Gambar 3.3 Representasi kromosom	25
Gambar 3.4 Hasil <i>Sorting</i> Bobot Bahan Makanan	26
Gambar 3.5 <i>Flowchart</i> membangun generasi awal	31
Gambar 3.6 <i>Flowchart</i> proses <i>crossover</i>	33
Gambar 3.7 Proses <i>crossover</i>	34
Gambar 3.8 <i>Flowchart</i> proses mutasi	35
Gambar 3.9 Proses mutasi	36
Gambar 3.10 <i>Flowchart</i> proses hitung <i>fitness</i>	36
Gambar 3.11 <i>Flowchart</i> proses seleksi	38
Gambar 4.1 Tabel database bahan pakan	40
Gambar 4.2 Rancangan Halaman Utama	41
Gambar 4.3 Rancangan Hasil Proses GA's	42
Gambar 4.4 Rancangan Detail Hasil Optimasi	43
Gambar 5.1 Implementasi halaman utama	56
Gambar 5.2 Implementasi hasil proses GA's	57
Gambar 5.3 Implementasi detail hasil optimasi	57
Gambar 6.1 Grafik uji coba ukuran populasi	59
Gambar 6.2 Grafik uji coba banyak generasi	61
Gambar 6.3 Grafik uji coba kombinasi <i>crossover</i> dan mutasi	63

Daftar Tabel

Tabel 2.1 Kebutuhan nutrisi di dalam ransum 8

Tabel 3.1 Hasil perhitungan nutrisi 29

Tabel 3.2 Inisialisasi Populasi Awal 32

Tabel 3.3 Hasil Perhitungan *Fitness* 37

Tabel 3.4 Hasil Seleksi 39

Tabel 4.1 Rancangan uji coba ukuran populasi 44

Tabel 4.2 Rancangan uji coba banyak generasi 45

Tabel 4.3 Rancangan uji coba kombinasi *Cr* dan *Mr*. 46

Tabel 6.1 Hasil uji coba ukuran populasi 58

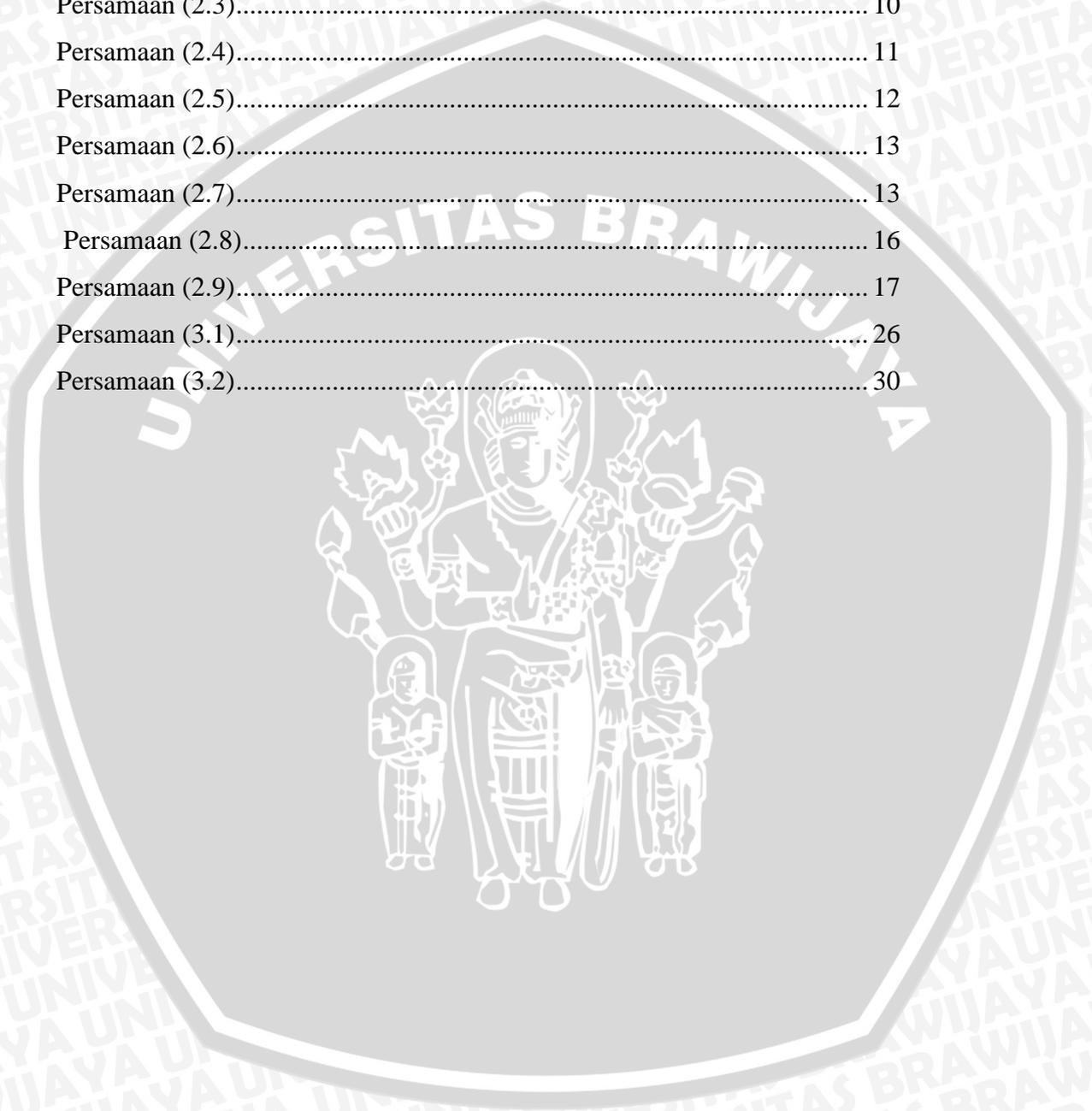
Tabel 6.2 Hasil uji coba banyak generasi 60

Tabel 6.3 Hasil uji coba kombinasi *crossover rate* dan *mutation rate* 62



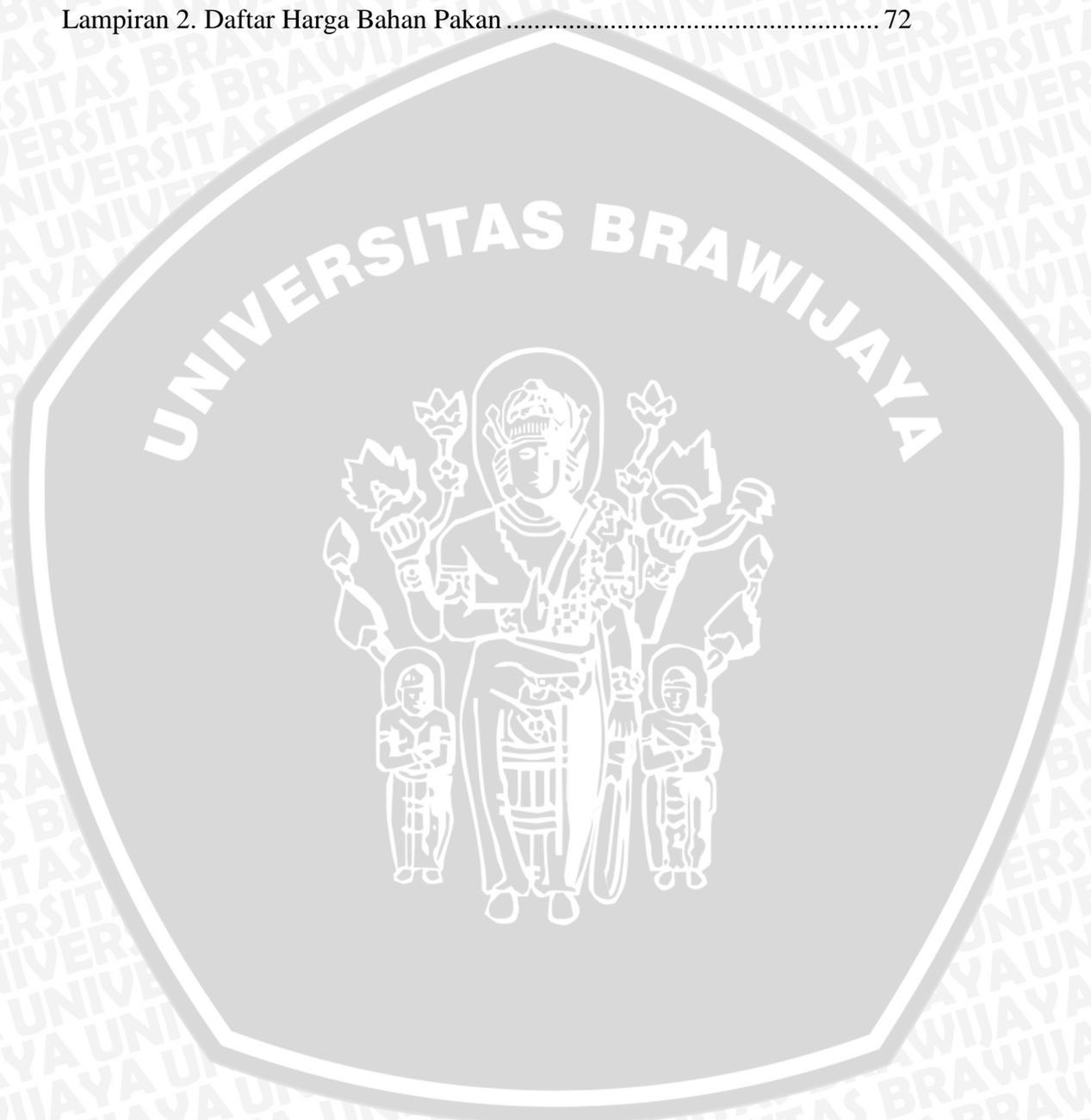
Daftar Persamaan

Persamaan (2.1).....	8
Persamaan (2.2).....	9
Persamaan (2.3).....	10
Persamaan (2.4).....	11
Persamaan (2.5).....	12
Persamaan (2.6).....	13
Persamaan (2.7).....	13
Persamaan (2.8).....	16
Persamaan (2.9).....	17
Persamaan (3.1).....	26
Persamaan (3.2).....	30



Daftar Lampiran

Lampiran 1. Daftar nutrisi bahan makanan.....	69
Lampiran 2. Daftar Harga Bahan Pakan.....	72



BAB 1

PENDAHULUAN

1.1 Latar Belakang

Pakan merupakan sumber energi utama bagi ternak. Semua hewan ternak (terutama ayam ras petelur yang menjadi pembahasan utama dalam penelitian ini) pasti membutuhkan pakan bernutrisi seimbang untuk dapat mempertahankan hidup dan menghasilkan telur dengan kualitas baik. Pemberian pakan ternak tanpa memperhatikan kualitas dan kuantitas akan mengakibatkan pertumbuhan maupun produktivitas dari ternak tersebut tidak maksimal.

Menurut Nugroho (2014), jenis pakan ayam petelur dibedakan ke dalam 4 kategori, dimana setiap kategorinya tergantung pada kandungan nutrisi yang dibutuhkan pada setiap masa pertumbuhan ayam. Fase pertumbuhan ayam dimulai dari fase *pre-starter*, fase *starter*, fase *grower*, dan fase *layer*. Setiap fase pertumbuhan ini memiliki kebutuhan nutrisi yang berbeda-beda. Sehingga pemberian pakan terhadap ayam petelur harus disesuaikan terhadap umur atau fase pertumbuhannya.

Kandungan nutrisi utama yang dibutuhkan dalam pakan ternak ayam petelur adalah berupa protein, ME, lemak, serat kasar, kalsium dan fosfor (Aribowo, 2008). Umumnya ayam petelur diberi makan berupa ransum. Ransum sendiri merupakan kombinasi dari bahan-bahan makanan unggas. Dalam proses pemberian ransum ini, terkadang peternak terjegal masalah biaya pemenuhan kebutuhan pakan. Untuk itu diperlukan cara untuk mengoptimalkan pemenuhan kebutuhan nutrisi ternak dengan biaya yang minimum.

Pengoptimalan biaya ransum dapat dilakukan dengan cara mengkombinasikan beberapa komposisi bahan pakan yang memenuhi kebutuhan nutrisi ayam petelur namun memiliki biaya yang murah. Terdapat banyak kemungkinan yang bisa diperoleh dalam mengkombinasikan pakan yang bisa digunakan untuk meningkatkan jumlah produksi telur dengan biaya yang minimum. Namun tidak semua kombinasi ini bisa memberikan solusi yang terbaik.

Atas dasar masalah tersebut, perlu dibangun sebuah aplikasi yang dapat memudahkan para peternak dalam menentukan kombinasi pakan yang optimal. Implementasi aplikasi inilah yang nantinya akan menggunakan algoritma genetika sebagai dasar logikanya.

Banyak penelitian tentang optimasi yang menggunakan metode algoritma genetika. Misalkan dalam permasalahan pencarian kombinasi bahan makanan bergizi untuk mendapatkan biaya minimum yang diteliti oleh Pratiwi (2014). Dalam penelitiannya, kromosom terbaik yang didapatkan sudah bisa memenuhi kebutuhan gizi ideal, namun dalam data bahan makanannya masih belum terdapat variasi berat bahan makanan yang dikonsumsi. Sari (2014) juga melakukan penelitian serupa namun dengan objek yang berbeda yaitu pada pencarian makanan bergizi untuk ibu hamil. Hasil yang didapatkan oleh Sari (2014) menunjukkan bahwa hasil optimasi terbaik masih belum bisa memenuhi kebutuhan gizi dari ibu hamil. Hal ini dimungkinkan karena tidak adanya variasi berat bahan makanan dalam data bahan makanan, selain itu juga adanya pengaruh dari rumus *fitness* yang digunakan.

Algoritma genetika merupakan algoritma pengembangan dari algoritma evolusi yang paling populer dan paling banyak digunakan pada masalah-masalah optimasi yang kompleks. Masalah optimasi memang banyak diselesaikan dengan menggunakan metode algoritma genetika karena algoritma ini memiliki kemampuan untuk menghasilkan solusi terbaik untuk masalah-masalah rumit (Mahmudy, 2014). Pemanfaatan algoritma genetika yang mampu menghasilkan output optimal diharapkan bisa menyelesaikan permasalahan dalam penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum menggunakan algoritma genetika.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, maka dapat dibuat rumusan masalah penelitian sebagai berikut :

1. Bagaimana representasi kromosom yang efisien untuk menyelesaikan masalah optimasi pakan ayam petelur?

2. Bagaimana menentukan parameter algoritma genetika yang tepat dalam menyelesaikan permasalahan optimasi pakan ayam petelur?
3. Bagaimana mengukur tingkat kualitas dari solusi yang telah dihasilkan?

1.3 Batasan Masalah

Batasan masalah dalam penelitian ini ditujukan guna menghindari pelebaran masalah yang akan dibahas, berikut adalah batasan - batasannya :

1. Acuan nutrisi yang digunakan adalah berupa protein, lemak, serat kasar, metabolis energi (ME), kalsium, dan fosfor.
2. Ayam yang digunakan adalah ayam petelur tipe *layer*, yaitu ayam petelur pada periode peneluran hingga afkir.
3. Harga yang digunakan mengacu pada informasi yang didapatkan dari Dinas Peternakan Jawa Timur (*update* januari 2015)

1.4 Tujuan Penelitian

Adapun yang menjadi tujuan penulisan dalam penyusunan skripsi adalah sebagai berikut :

1. Membuat sebuah aplikasi penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum menggunakan algoritma genetika.
2. Untuk mengetahui nilai dari parameter algoritma genetika yang efektif dalam penyelesaian masalah aplikasi penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah untuk mendapatkan solusi dari permasalahan optimasi biaya dalam pemenuhan nutrisi ayam petelur, sehingga dapat membantu peternak dalam menentukan pakan ternak bernutrisi seimbang dengan biaya seminimum mungkin

1.6 Metodologi Penelitian

- a. Studi Literatur
Mempelajari teori yang berhubungan dengan optimasi biaya, pakan ternak, dan algoritma genetika.
- b. Analisis Kebutuhan Sistem
Menganalisa permasalahan yang ada untuk mendapat unsur-unsur yang dibutuhkan dalam pembangunan system.
- c. Perancangan Sistem
Membuat permodelan system dan menyusun atau mengembangkannya sesuai dengan kebutuhan pengguna .
- d. Implementasi Sistem
Proses implementasi model yang telah dibangun menjadi sebuah perangkat lunak yang siap digunakan oleh pengguna.
- e. Analisis Dan Pengujian
Menguji coba sistem yang telah dihasilkan dan menganalisis hasil dari implementasi apakah telah sesuai dengan tujuan yang dirumuskan sebelumnya atau tidak untuk dapat dievaluasi.

1.7 Sistematika Penulisan

Gambaran secara garis besar pembahasan dari keseluruhan isi laporan penelitian untuk setiap bab adalah sebagai berikut :

BAB I : Pendahuluan

Bab pendahuluan ini berisikan latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II : Tinjauan Teoritis

Bab ini menguraikan tentang dasar teori tentang segala hal yang berkaitan dengan Penentuan Komposisi Pakan Ternak Untuk Memenuhi Kebutuhan Nutrisi Ayam Petelur Dengan Biaya Minimum Menggunakan Algoritma Genetika.

BAB III : Metodologi Penelitian

Membahas metode yang digunakan dalam penelitian serta langkah-langkah kerja yang diterapkan dalam proses perancangan dan implementasi pada Penentuan Komposisi Pakan Ternak Untuk Memenuhi Kebutuhan Nutrisi Ayam Petelur Dengan Biaya Minimum Menggunakan Algoritma Genetika

BAB IV : Perancangan

Bab ini menguraikan tentang perancangan dalam membangun system. Diantaranya terdapat perancangan database, perancangan *user interface*, dan perancangan uji coba dan evaluasi.

BAB V : Implementasi

Menjelaskan implementasi dari perangkat lunak mengenai Penentuan Komposisi Pakan Ternak Untuk Memenuhi Kebutuhan Nutrisi Ayam Petelur Dengan Biaya Minimum Menggunakan Algoritma Genetika yang sesuai dengan perancangan sistem yang telah dibangun

BAB VI : Pengujian Dan Analisis

Menjelaskan proses dan hasil dari pengujian terhadap sistem yang telah dibangun dan memastikan bahwa program telah sesuai dengan perancangan dan disertai analisis.

BAB VII : Penutup

Memuat kesimpulan dari keseluruhan uraian bab-bab sebelumnya, serta saran-saran dari hasil yang diperoleh.

BAB II

DASAR TEORI

2.1 Kajian Pustaka

Pada penulisan tugas akhir ini, penulis menggunakan tiga kajian pustaka dari penelitian yang pernah dibuat sebelumnya dengan metode yang sama yaitu metode algoritma genetika.

Pratiwi (2013) meneliti tentang permasalahan pencarian kombinasi bahan makanan bergizi untuk mendapatkan biaya minimum. Dari hasil pengujian yang dilakukan, didapat sebuah hasil bahwa ukuran populasi optimal yang dicapai adalah 40 dengan rata-rata *fitness* sebesar 3890. Sedangkan generasi optimalnya terletak pada generasi ke 500 dengan rata-rata *fitness* 38820. Parameter lain yang berpengaruh dalam penelitian ini yaitu *crossover* dan *mutation*. Pada hasil yang telah diperoleh dari uji coba kombinasi *crossover rate* dan *mutation rate* adalah sebesar 0,4 (*crossover rate*) dan 0,6 (*mutation rate*). Kromosom terbaik yang didapatkan sudah bisa memenuhi kebutuhan gizi ideal, namun dalam data bahan makanannya masih belum terdapat variasi berat bahan makanan yang dikonsumsi.

Sari (2014) juga melakukan penelitian serupa namun dengan objek yang berbeda yaitu pada pencarian makanan bergizi untuk ibu hamil. Hasil yang didapatkan oleh Sari (2014) menunjukkan bahwa hasil optimasi terbaik terletak pada generasi 1500 dengan nilai rata-rata *fitness* 142670, ukuran populasi sebanyak 150 dengan rata-rata nilai *fitness* 143050, serta dengan kombinasi probabilitas mutasi dan probabilitas *crossover* sebesar 0,6 dan 0,4 dengan rata-rata nilai *fitness* sebesar 142725. Namun dalam penelitiannya, hasil optimasi terbaik masih belum bisa memenuhi kebutuhan gizi dari ibu hamil. Hal ini dimungkinkan karena tidak adanya variasi berat bahan makanan dalam data bahan makanan, selain itu juga adanya pengaruh dari rumus *fitness* yang digunakan.

Sedangkan Aribowo (2008), meneliti tentang pencarian kombinasi bahan pakan untuk ayam petelur. Hasil dari penelitian ini menunjukkan bahwa nilai *fitness* optimum dicapai pada ukuran populasi 1000, probabilitas *crossover* 0.5, probabilitas mutasi 0.1, dan jumlah generasi 4000. Walaupun hasil yang

didapatkan sudah bisa dikatakan optimal, namun pada penelitian ini tidak tercantum variasi harga bahan pakan.

2.2 Ayam Petelur

Ayam ras petelur merupakan salah satu jenis unggas yang banyak dikembangkan oleh masyarakat, baik dalam skala kecil, sekelompok peternak, maupun industri peternakan dalam skala yang cukup besar. Banyaknya peminat dikarenakan ayam ras petelur ini dapat menghasilkan telur melebihi ayam lainnya. Sifat genetis ayam, manajemen pemeliharaan, makanan, dan kondisi pasar sangat mempengaruhi keberhasilan peternak dalam mengelola usaha ayam ras petelur (Amrullah, 2003).

Menurut Banong (2012), ayam ras petelur dibedakan atas tiga fase, yaitu fase *starter* (umur 1 hari - umur 6 minggu), fase *grower* (umur 6 minggu – 18 minggu), fase *layer* (umur 18 minggu – afkir).

Menurut Cahyono (1995), jenis ayam petelur di bagi menjadi dua tipe, yaitu :

1. Tipe Ayam Petelur Ringan.

Ayam pada tipe ini lebih sering disebut sebagai ayam petelur putih. Ayam petelur putih memiliki ukuran badan yang lebih kecil, sehingga ayam ini hanya dimanfaatkan sebagai ayam petelur murni karena tidak memiliki nilai jual.

2. Tipe Ayam Petelur Medium.

Ukuran ayam petelur tipe medium lebih besar bila dibandingkan dengan ayam petelur putih, bobot ayam petelur medium juga lebih berat. Telur dan daging yang dihasilkan cukup banyak, sehingga ayam ini disebut juga dengan ayam tipe dwiguna.

2.3 Ransum

Ransum merupakan kombinasi bahan makanan yang diberikan kepada hewan ternak. Ransum dibuat dari beberapa bahan makanan ternak yang disusun berdasarkan metode tertentu. Untuk menentukan jumlah pakan yang harus diberikan pada ayam, peneliti menggunakan persamaan *winter and funk* (Rasyaf,

2003). Persamaan jumlah kebutuhan pakan/hari ditunjukkan pada Persamaan 2.1 berikut ini.

$$\frac{8,3 + 2,2 \times \frac{\text{Bobot ayam}(gr)}{454} + 0,1 \times \text{produksi telur}(\%)}{100} \times 454 \dots (2.1)$$

Dalam proses penyusunan ransum ini kandungan nutrisinya harus disesuaikan dengan kebutuhan hewan ternak, karena kebutuhan masing-masing ternak tidaklah sama walaupun termasuk dalam satu ras. Oleh sebab itu, daftar kandungan nutrisi pada bahan makanan yang akan diberikan harus diketahui. Daftar nutrisi bahan makanan ini disajikan pada Lampiran 1.

2.4 Nutrisi Pakan Ternak

Ayam ras petelur membutuhkan sejumlah nutrisi yang lengkap untuk menunjang kebutuhan pokok seperti bernapas, bergerak, peredaran darah, dan lain-lain. Selain itu, unsur nutrisi ini juga dibutuhkan ayam ras petelur dalam berproduksi. Nutrisi yang dimaksud berupa protein, energi metabolis, lemak, serat kasar, kalsium, dan fosfor. Keseluruhan nutrisi ini harus disusun dalam jumlah yang tepat dan seimbang dalam makanan yang berupa ransum. Pemberian nutrisi kepada ayam ras petelur tidak sama antara satu dengan yang lainnya. Kadar nutrisi yang diberikan dipengaruhi oleh umur atau periode ayam. Dalam penelitian ini digunakan ayam yang sedang berada pada periode bertelur atau fase *layer* yang berumur 18 minggu sampai afkir. Detail nutrisi yang dibutuhkan akan ditunjukkan pada Tabel 2.1.

Tabel 2.1 Kebutuhan nutrisi di dalam ransum

Protein (%)	18
Lemak(%)	5
Serat Kasar(%)	4
ME(kcal/kg)	2850
Kalsium(%)	3
Fosfor(%)	0,5

Sumber : North, 1984.

2.4.1 Protein

Protein merupakan komponen organik kompleks yang terdiri dari karbon, hidrogen, oksigen, nitrogen, dan sulfur. Protein juga dibentuk dari berbagai jenis asam amino yang dirangkai oleh ikatan peptida (Rasyaf, 2003). Protein sangat diperlukan dalam beberapa proses berikut :

1. Pembentukan dan pembangunan jaringan tubuh, pembentukan dan perkembangan organ tubuh, dan pertumbuhan bulu.
2. Pembentukan enzim-enzim yang berperan dalam proses pencernaan.
3. Produksi telur. Protein yang terkandung dalam telur juga berasal dari tubuh ayam. Oleh karena itu ayam dalam fase bertelur akan membutuhkan protein lebih banyak.

Kebutuhan unsur protein dapat terpenuhi dari bahan makanan yang mengandung protein nabati maupun hewani, seperti contoh berikut ini :

1. Protein nabati : jagung, bungkil kacang kedelai, bungkil kelapa, bungkil kacang tanah, dll.
2. Protein hewani : tepung ikan, tepung kerang, tepung susu, dll.

Ayam yang membutuhkan protein dalam jumlah yang cukup tinggi adalah anak ayam periode awal (*starter*) dan ayam dalam proses produksi (*layer*). Bagi ayam dewasa yang sedang dalam tahap produksi (*layer*) membutuhkan kadar protein yang tinggi untuk menggantikan jaringan yang rusak dan untuk memproduksi. Kadar protein yang dibutuhkan oleh ayam tipe *layer* adalah 18%. Persamaan yang digunakan untuk menghitung kadar protein dalam ransum ditunjukkan pada Persamaan 2.2.

$$Protein = \frac{Bobot\ pakan\ i,j\ (\%)}{100} \times kadar\ protein\ bahan\ i(\%) \dots \dots \dots (2.2)$$

Variabel *i* dalam bobot pakan pada Persamaan 2.2 menunjukkan bobot pakan pada kromosom yang diuji dan variable *j* menunjukkan bobot pakan pada gen yang diuji. Sedangkan variabel *i* pada kadar protein bahan menunjukkan nilai

dari kadar protein dari bahan makanan yang sedang diuji. Daftar kadar protein bahan makanan yang digunakan dalam penelitian ini disajikan dalam Lampiran 1.

2.4.2 Lemak

Lemak berperan penting dalam melarutkan vitamin A, D, E, K, sebagai sumber dan cadangan energy, sebagai lapisan lemak tubuh di bawah kulit, dan sumber asam lemak esensial. Asam lemak yang terdapat pada jaringan lemak terdiri atas asam lemak jenuh dan asam lemak tidak jenuh (Anggorodi, 1985).

Penggunaan lemak pada ransum dimaksudkan untuk mendapatkan keuntungan-keuntungan sebagai berikut (Anggorodi, 1985) :

1. Pada ayam ras petelur, penambahan lemak pada ransum akan menghasilkan daya produksi yang lebih tinggi dan mendapatkan bentuk telur yang lebih besar.
2. Sebagai penyedia cadangan panas dan energy.
3. Memiliki nilai kalori lebih tinggi dibandingkan dengan bahan makanan lainnya sehingga dapat menghemat ransum karena adanya peningkatan efisiensi.

Dalam tubuh ayam petelur tipe *layer*, kandungan lemak yang dibutuhkan adalah sebesar 5%. Persamaan yang digunakan untuk menghitung kadar lemak dalam ransum ditunjukkan pada Persamaan 2.3.

$$\text{Lemak} = \frac{\text{Bobot pakan } i, j (\%)}{100} \times \text{kadar lemak bahan } i (\%) \dots \dots \dots (2.3)$$

Variabel *i* dalam bobot pakan pada Persamaan 2.3 menunjukkan bobot pakan pada kromosom yang diuji dan variable *j* menunjukkan bobot pakan pada gen yang diuji. Sedangkan variabel *i* pada kadar lemak bahan menunjukkan nilai dari kadar lemak dari bahan makanan yang sedang diuji. Daftar kadar lemak bahan makanan yang digunakan dalam penelitian ini disajikan dalam Lampiran 1.

2.4.3 Serat Kasar

Serat kasar merupakan komponen yang sebagian besar terdiri dari selulosa dan lignin yang hampir tidak dapat dicerna oleh unggas. Bahan makanan yang memiliki kandungan serat kasar tinggi memiliki nilai energy yang rendah, terkecuali untuk makanan yang juga mengandung kadar lemak tinggi.

Serat kasar (termasuk selulosa) merupakan sumber panas dan energi bila dicerna. Zat tersebut juga mencegah terjadinya penggumpalan makanan dalam lambung dan usus hewan dengan cara memberi pengaruh pencahar dan mempertahankan tensus otot yang wajar dalam saluran pencernaan.

Fungsi serat kasar pada unggas antara lain :

1. Memelihara fungsi normal dari saluran pencernaan.
2. Memperbaiki penyerapan nutrisi.
3. Mencegah kanibalisme.

Persamaan yang digunakan untuk menghitung kadar serat kasar dalam ransum ditunjukkan pada Persamaan 2.4.

$$\text{Serat kasar} = \frac{\text{Bobot pakan } i, j (\%)}{100} \times \text{kadar serat bahan } i (\%) \dots\dots(2.4)$$

Variabel i dalam bobot pakan pada Persamaan 2.4 menunjukkan bobot pakan pada kromosom yang diuji dan variable j menunjukkan bobot pakan pada gen yang diuji. Sedangkan variabel i pada kadar serat bahan menunjukkan nilai dari kadar serat dari bahan makanan yang sedang diuji. Daftar kadar serat kasar bahan makanan yang digunakan dalam penelitian ini disajikan dalam Lampiran 1.

2.4.4 Energi Metabolis

Metabolism Energy (ME) berasal dari karbohidrat dan lemak yang berada di dalam ransum. Energi yang dihasilkan akan digunakan untuk keseluruhan aktivitas di dalam tubuh, misalnya seperti proses pencernaan, pernapasan, reproduksi, dan lain-lain. Jika ayam kekurangan energy di dalam tubuhnya, maka lemak akan digunakan sebagai cadangan energy. Dan apabila cadangan energi dalam bentuk lemak ini masih belum bisa mencukupi kebutuhan, maka protein yang akan dirombak menjadi energi (Sudarmono, 2003).

Ransum yang mengandung energi metabolis tinggi akan menyebabkan ayam kurang mengkonsumsi ransum. Sehingga ayam akan kekurangan zat gizi lainnya. Untuk itu pemberian makanan yang mengandung ME harus disesuaikan. Pada ayam ras petelur tipe layer, kandungan ME yang disarankan adalah sebesar 2850 kkal/kg. Persamaan yang digunakan untuk menghitung kadar energy metabolis dalam ransum ditunjukkan pada Persamaan 2.5.

$$\text{Energi Metabolis} = \frac{\text{Bobot pakan } i, j (\%)}{100} \times \text{kadar EM bahan } i (\%) \dots (2.5)$$

Variabel i dalam bobot pakan pada Persamaan 2.5 menunjukkan bobot pakan pada kromosom yang diuji dan variabel j menunjukkan bobot pakan pada gen yang diuji. Sedangkan variabel i pada kadar EM bahan menunjukkan nilai dari kadar EM dari bahan makanan yang sedang diuji. Daftar kadar EM bahan makanan yang digunakan dalam penelitian ini disajikan dalam Lampiran 1.

2.4.5 Kalsium

Kalsium merupakan salah satu komponen yang kandungannya melimpah dalam tubuh unggas. Kalsium dibutuhkan unggas sebagai komponen esensial dalam pembentukan tulang, tulang rawan, dan esoskeleton krustasea (Abun, 2008).

Menurut Abun (2008), kalsium di dalam tubuh unggas berfungsi sebagai berikut :

1. Kalsium berguna dalam proses penggumpalan darah dengan menstimulir pengeluaran thromboplastin dari plasma darah.
2. Kalsium merupakan aktivator untuk beberapa enzim seperti enzim lipase pankreatik, asam phosphatase, cholinesterase, dan suksinik dehydrogenase.
3. Kalsium menstimulir kontraksi otot dan mengatur tranmisi impuls dari satu sel ke sel lainnya melalui pengontrolan produksi acetyl cholin.
4. Kalsium terikat kuat dengan phospholipid di dalam membran sel. Phospholipid mempunyai peranan penting dalam pengaturan membran sel dan karena itu pengambilan nutrien lebih banyak melalui sel.

5. Kalsium bisa menyerap vitamin B12 dari saluran pencernaan hasil produksi mikroba.
 6. Kalsium menjaga integritas membran sel dan jaringan skeletal.
- Persamaan yang digunakan untuk menghitung kadar kalsium dalam ransum ditunjukkan pada Persamaan 2.6.

$$\text{Kalsium} = \frac{\text{Bobot pakan } i, j (\%)}{100} \times \text{kadar kalsium bahan } i (\%) \dots \dots \dots (2.6)$$

Variabel i dalam bobot pakan pada Persamaan 2.6 menunjukkan bobot pakan pada kromosom yang diuji dan variable j menunjukkan bobot pakan pada gen yang diuji. Sedangkan variabel i pada kadar kalsium bahan menunjukkan nilai dari kadar kalsium dari bahan makanan yang sedang diuji. Daftar kadar kalsium bahan makanan yang digunakan dalam penelitian ini disajikan dalam Lampiran 1.

2.4.6 Fosfor

Dalam tubuh unggas, fosfor memainkan peranan penting dalam pembentukan tulang rawan, metabolisme energi, metabolisme karbohidrat, asam amino dan lemak, metabolisme jaringan syaraf, kimiawi darah normal, pertumbuhan kerangka, dan pengangkutan asam lemak Anggorodi (1985).

Tingkat fosfor dalam pakan tidak boleh diabaikan karena kelebihan fosfor akan menghambat pelepasan kalsium tulang dan pembentukan kalsium karbonat dalam kelenjar kerabang sehingga akan mengurangi kualitas kerabang telur.

Persamaan yang digunakan untuk menghitung kadar fosfor dalam ransum ditunjukkan pada Persamaan 2.7.

$$\text{Fosfor} = \frac{\text{Bobot pakan } i, j (\%)}{100} \times \text{kadar fosfor bahan } i (\%) \dots \dots \dots (2.7)$$

Variabel i dalam bobot pakan pada Persamaan 2.7 menunjukkan bobot pakan pada kromosom yang diuji dan variable j menunjukkan bobot pakan pada gen yang diuji. Sedangkan variabel i pada kadar fosfor bahan menunjukkan nilai

dari kadar fosfor dari bahan makanan yang sedang diuji. Daftar kadar fosfor bahan makanan yang digunakan dalam penelitian ini disajikan dalam Lampiran 1.

2.5 Algoritma Genetika

Algoritma genetika merupakan algoritma yang metode penyelesaiannya berdasarkan pada mekanisme seleksi alam dan prinsip-prinsip genetika (Goldberg, 1989), yang bertujuan untuk menentukan individu yang berkualitas tinggi pada sebuah populasi. Algoritma Genetika dapat diterapkan dalam penyelesaian masalah optimasi kompleks yang sulit dilakukan oleh metode konvensional.

Berikut adalah prosedur atau struktur umum Algoritma Genetika :

1. Pembangkitan populasi secara acak sejumlah n kromosom pada generasi ke - 0.
2. Evaluasi masing-masing kromosom dengan menghitung nilai *fitness*-nya.
3. Seleksi beberapa kromosom dari sejumlah n individu yang memiliki nilai *fitness* terbaik.
4. Rekombinasi kromosom terpilih dengan cara melakukan proses *crossover* dan proses *mutation*.
5. Proses seleksi untuk membentuk populasi baru.

Dalam melakukan proses optimasi menggunakan Algoritma Genetika dibutuhkan parameter-parameter genetika yang berperan dalam pengendalian operator-operator genetika. Beberapa parameter yang paling sering digunakan diantaranya adalah :

1. Fungsi *fitness* yang dimiliki oleh masing-masing individu, yang digunakan untuk menentukan tingkat kesesuaian individu tersebut dengan kriteria yang akan dicapai.
2. Ukuran populasi yang menunjukkan banyaknya kromosom pada sebuah populasi. Dimana besarnya populasi akan mempengaruhi kecepatan konvergensi. Jika ukuran populasi terlalu kecil akan memungkinkan terjadinya konvergensi dini / *premature*. Namun jika terlalu besar akan mengakibatkan lamanya waktu yang dibutuhkan algoritma genetika dalam menghasilkan solusi terbaik.

3. Jumlah generasi yang akan dibentuk. Sama halnya dengan ukuran populasi, jumlah generasi juga mempengaruhi kecepatan konvergensi.
4. Kemungkinan terjadinya *crossover*.
5. Kemungkinan terjadinya mutasi.

Dalam Algoritma Genetika terdapat tiga operator dasar yang sering digunakan untuk membentuk generasi baru diantaranya adalah reproduksi / seleksi, pindah silang (*crossover*), dan mutasi. Dalam proses reproduksi, setiap individu dari sebuah populasi pada suatu generasi akan diseleksi berdasarkan nilai *fitness*-nya untuk menghasilkan keturunan. Probabilitas terpilihnya suatu individu untuk bereproduksi adalah sebesar nilai *fitness* individu tersebut dibagi dengan jumlah nilai *fitness* seluruh individu dalam sebuah populasi (Davis, 1991).

Pindah silang (*crossover*) adalah proses pemilihan posisi gen pada sebuah kromosom secara acak dan menukar karakter-karakter gennya untuk menghasilkan kromosom anak. Sedangkan operator mutasi dioperasikan untuk melakukan modifikasi satu atau lebih gen dalam individu yang sama guna menciptakan individu baru. Proses modifikasi ini dilakukan sebagai cara untuk mengembalikan materi genetik yang hilang. Mutasi mencegah kehilangan total materi genetik setelah reproduksi dan pindah silang

2.6 Implementasi Algoritma Genetika

Dalam algoritma genetika, proses pencarian solusi diawali dengan tahap pembangkitan populasi awal secara acak. Populasi ini disusun atas beberapa kromosom. Setiap kromosom merupakan gambaran solusi atas pemecahan masalah. Dari sebuah populasi ini akan dihasilkan keturunan-keturunan baru yang didapat dari proses reproduksi. Proses reproduksi ini meliputi *crossover* dan mutasi. Dari keturunan baru yang dihasilkan ini diharapkan bisa memiliki sifat yang lebih baik sehingga solusi terbaik dapat dicapai dengan terus mengulang proses pencarian keturunan.

2.6.1 Representasi Kromosom

Representasi merupakan gambaran hasil akhir dari suatu permasalahan yang akan diselesaikan. Representasi kromosom berupa proses pengkodean sebuah solusi dari suatu permasalahan, dimana pengkodean tersebut meliputi penyandian gen, dengan satu gen mewakili satu variabel. Dalam penelitian ini digunakan pengkodean *real*, dimana kromosomnya mewakili bobot dari masing-masing bahan makanan.

2.6.2 Fitness

Perhitungan nilai *fitness* perlu dilakukan untuk setiap kromosom. Hasil nilai *fitness* nantinya akan digunakan sebagai acuan dalam membentuk generasi baru. Semakin besar nilai *fitness* maka semakin baik kromosom tersebut untuk dijadikan calon solusi. Penyusunan fungsi persamaan untuk perhitungan *fitness* didasarkan pada biaya dan nilai penalty yang berbanding terbalik dengan optimasi. Untuk mencegah terjadinya hasil *fitness* yang terlalu kecil, maka dibuatlah sebuah *konstanta* (K) yang berfungsi sebagai pengali senilai 1000.

Penyusunan fungsi persamaan untuk perhitungan *fitness* didasarkan pada biaya dan nilai penalti yang berbanding terbalik dengan optimasi. Untuk mencegah terjadinya hasil *fitness* yang terlalu kecil, maka dibuatlah sebuah *konstanta* (K) yang berfungsi sebagai pengali senilai 1000. Dan juga untuk membuat selisih antara penalty dan harga tidak terlalu besar maka pada sisi penalty diberi sebuah konstanta pengali sebesar 20.

Fungsi *fitness* yang digunakan ditunjukkan pada Persamaan 2.8 berikut :

$$Fitness = \frac{1}{cost + (penalti \times \alpha)} \times K \dots \dots \dots (2.8)$$

Penalti ini merupakan perbandingan antara kebutuhan nutrisi ayam dengan total nutrisi yang terdapat di bahan pakan. Persamaan dari penalty ini ditunjukkan pada Persamaan 2.9 berikut ini (Sari, 2014).

$$Penalti_{1...6} = \begin{cases} 0, & TotalNut \geq KebNut \\ KebNut - TotalNut, & TotalNut < KebNut \end{cases} \dots\dots\dots(2.9)$$

2.6.3 Inisialisasi Populasi Awal

Inisialisasi dilakukan untuk membangkitkan himpunan solusi baru secara *random* yang terdiri atas sejumlah string kromosom dan ditempatkan pada sebuah populasi (Mahmudy, 2014).

Ukuran populasi yang akan dibangkitkan dalam dalam tugas akhir ini berupa bilangan acak dengan *range* bilangan yang telah ditentukan. Ukuran populasi didefinisikan sebagai banyaknya kromosom dalam sebuah populasi. Panjang setiap string kromosom (*stringLen*) dihitung berdasarkan presisi variabel solusi yang kita cari (Mahmudy, 2014). Dalam penelitian ini, panjang setiap *string* kromosom adalah sejumlah bahan makanan yang akan diuji yaitu sebanyak 40. Sehingga setiap kromosom memiliki panjang 40.

2.6.4 Operator Algoritma Genetika

Algoritma Genetika merupakan algoritma dengan proses pencarian yang *heuristic* dan *random* sehingga pemilihan operator yang digunakan sangat menentukan keberhasilan Algoritma Genetika dalam menemukan solusi optimum dari sebuah permasalahan. Kebanyakan operator genetika digunakan setelah proses evaluasi tahap pertama selesai dilakukan. Proses yang dilakukan oleh operator genetika ini memiliki tujuan untuk membentuk sebuah populasi baru. Operator-operator yang dimaksud adalah operator *crossover*, mutasi, dan seleksi (Desiani, 2007).

2.6.4.1 Crossover

Proses *crossover* dilakukan guna menambah keanekaragaman individu dalam suatu populasi dengan menyilangkan antar string yang didapat dari reproduksi sebelumnya.

Metode *crossover* yang paling mudah untuk diterapkan adalah metode *one cut point*. Metode ini dilakukan dengan memisahkan suatu string menjadi dua bagian kemudian menukarkan salah satu bagian dengan bagian yang lain dari string yang berbeda yang telah dipisahkan dengan cara yang sama.

Misalkan secara random didapat P_1 dan P_2 sebagai *parent* terpilih dan titik potong (*cut point*) adalah 4, maka akan didapat 2 kromosom anak (*offspring*) yang diberi nama C_1 dan C_2 sebagai berikut :

P_1	1	5	7	10	5	6	8	4	7	9
P_2	10	1	2	5	2	3	2	6	4	6
C_1	1	5	7	10	5	3	2	6	4	6
C_2	10	1	2	5	2	6	8	4	7	9

Gambar 2.1 Metode *One Cut Point*

2.6.4.2 Mutasi

Operator mutasi digunakan untuk menghindari terjadinya konvergensi premature dan juga untuk tetap menjaga perbedaan (*divergesity*) kromosom pada sebuah populasi.

Proses mutasi dilakukan dengan mengubah gen-gen tertentu yang dipilih secara acak sehingga menghasilkan berbagai macam calon solusi. Terdapat beberapa cara proses mutasi yang dapat diterapkan dalam Algoritma Genetika menurut jenis pengkodean terhadap *phenotype*, diantaranya adalah mutasi dalam pengkodean biner, mutasi dalam pengkodean permutasi, dan mutasi dalam pengkodean nilai.

Dalam permasalahan ini, proses mutasi yang digunakan adalah metode *reciprocal exchange*. Tahapan prosesnya adalah dengan memilih dua posisi dari kromosom kemudian menukarkan nilainya satu sama lain.

Contoh proses mutasi :

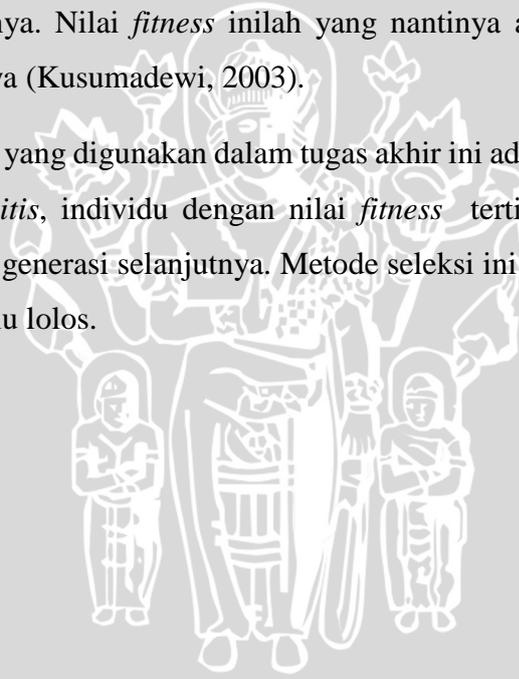
P10	10	3	7	9	4	2	3	6	6	6
C3	10	3	7	3	4	2	9	6	6	6

Gambar 2.2 Metode *Reciprocal Exchange*

2.6.4.3 Seleksi

Proses seleksi bertujuan untuk memberikan kesempatan reproduksi yang lebih besar bagi anggota populasi yang paling fit (mempunyai nilai *fitness* terbaik). Oleh sebab itu, langkah pertama yang harus dilakukan proses seleksi adalah mencari nilai *fitness*-nya. Nilai *fitness* inilah yang nantinya akan diproses pada tahap seleksi berikutnya (Kusumadewi, 2003).

Metode seleksi yang digunakan dalam tugas akhir ini adalah metode seleksi *elitis*. Pada metode *elitis*, individu dengan nilai *fitness* tertinggi adalah yang akan terpilih untuk menjadi generasi selanjutnya. Metode seleksi ini menjamin individu yang terbaik akan selalu lolos.



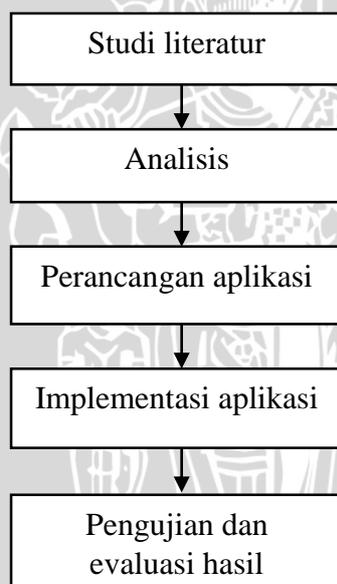
BAB III

METODOLOGI PENELITIAN

Pada bab ini akan membahas beberapa hal, yaitu tahapan penelitian, analisis kebutuhan sistem, formulasi permasalahan, dan siklus penyelesaian masalah menggunakan algoritma genetika.

3.1 Tahapan Penelitian

Pada bab tahapan penelitian akan dijelaskan langkah – langkah yang akan digunakan dalam pembuatan aplikasi optimasi pakan ternak. Tahapan-tahapannya adalah seperti pada Gambar 3.1 berikut ini.



Gambar 3.1 Diagram tahap penelitian

Berdasarkan pada Gambar 3.1, pembuatan aplikasi “Penentuan Komposisi Pakan Ternak Untuk Memenuhi Kebutuhan Nutrisi Ayam Petelur Dengan Biaya Minimum Menggunakan Algoritma Genetika” dapat dijabarkan sebagai berikut :

1. Melakukan studi literatur mengenai algoritma genetika dan kebutuhan nutrisi ayam petelur. Studi literatur yang digunakan dalam penelitian ini

berasal dari jurnal, buku, skripsi, dan artikel. Nantinya studi literatur ini akan digunakan sebagai landasan teori untuk dipelajari dan dianalisis terkait dengan permasalahan yang ada guna menunjang penelitian.

2. Menganalisa kebutuhan sistem. Pada tahap ini dilakukan proses analisis yang bertujuan untuk menganalisis permasalahan optimasi pakan ayam petelur dengan menggunakan algoritma genetika.
3. Melakukan proses perancangan sistem perangkat lunak berdasarkan pada hasil pembelajaran pada tahap sebelumnya.
4. Melakukan proses implementasi sistem berdasarkan analisis dan perancangan yang telah dibuat.
5. Melakukan uji coba dan evaluasi pada sistem yang telah berhasil diimplementasikan.

3.2 Analisa Kebutuhan Sistem

Pada bab ini akan membahas mengenai analisis kebutuhan dan perancangan pada aplikasi penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum menggunakan algoritma genetika. Untuk membuat sebuah aplikasi yang diinginkan harus diketahui terlebih dahulu hal-hal apa saja yang dibutuhkan untuk menjalankan sebuah sistem, sehingga sistem dapat bekerja secara maksimal.

Tahap analisa kebutuhan sistem memiliki tujuan untuk memodelkan informasi yang akan digunakan dalam tahap perancangan. Tahap analisis kebutuhan perangkat lunak yang dibutuhkan meliputi deskripsi umum sistem, daftar kebutuhan sistem, dan data yang digunakan dalam proses algoritma.

3.2.1 Deskripsi Umum Sistem

Pada penelitian ini akan dibangun sebuah aplikasi optimasi komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum menggunakan algoritma genetika. Pembuatan aplikasi ini bertujuan

untuk memudahkan peternak dalam menyusun ransum yang akan mereka berikan kepada ayam ternak mereka. Dan juga untuk mendapatkan kombinasi bahan makanan dengan biaya yang minimum namun kebutuhan gizi tetap terpenuhi. Penggunaan metode algoritma genetika diharapkan dapat menemukan solusi yang efisien karena algoritma genetika mampu membuat beberapa kemungkinan yang lebih baik secara otomatis dan cepat.

3.2.2 Daftar Kebutuhan Sistem

Tahapan ini bertujuan untuk menjelaskan apa saja yang dibutuhkan oleh sistem yang akan dibuat. Berikut adalah beberapa kebutuhan sistem yang harus terpenuhi :

1. Sistem ini memerlukan data berupa tabel kebutuhan nutrisi ayam petelur dan data nutrisi serta harga dalam setiap bahan pakan yang digunakan.
2. Sistem membutuhkan *inputan* berupa berat ayam (dalam *gram*) dan produksi telur (%).
3. Dari sisi algoritma genetika, *inputan* yang dibutuhkan adalah berupa jumlah populasi (*popSize*), banyak generasi, *crossover rate* (*cr*), dan *mutation rate* (*mr*).

3.2.3 Data Yang Digunakan

Data-data yang digunakan dalam membangun aplikasi ini adalah sebagai berikut:

1. Daftar nutrisi bahan makanan ternak diperoleh melalui sebuah sumber yaitu (Scott, 1976).
2. Daftar kebutuhan nutrisi ayam ternak diperoleh dari (North, 1984).
3. Daftar harga bahan makanan ternak didapat berdasarkan informasi dari Dinas Peternakan Jawa Timur (*update* Januari 2015).

3.3 Formulasi Permasalahan

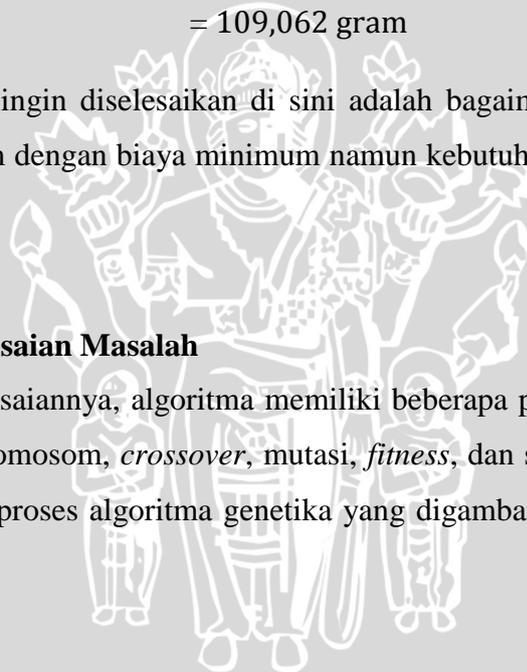
Dimisalkan *user* memasukkan bobot tubuh ayam petelur seberat 1800 gram, produksi telur sebesar 70 %, dan ayam sedang berada dalam fase bertelur (fase *layer*), serta jumlah bahan pakan yang digunakan adalah sebanyak 10 bahan. Bahan pakan yang digunakan adalah bahan pakan bernomor 1 sampai 10 yang telah dilampirkan pada Lampiran 1. Tahap awal perhitungan manual adalah dengan menghitung kebutuhan pakan per hari menggunakan Persamaan 2.1 sebagai berikut :

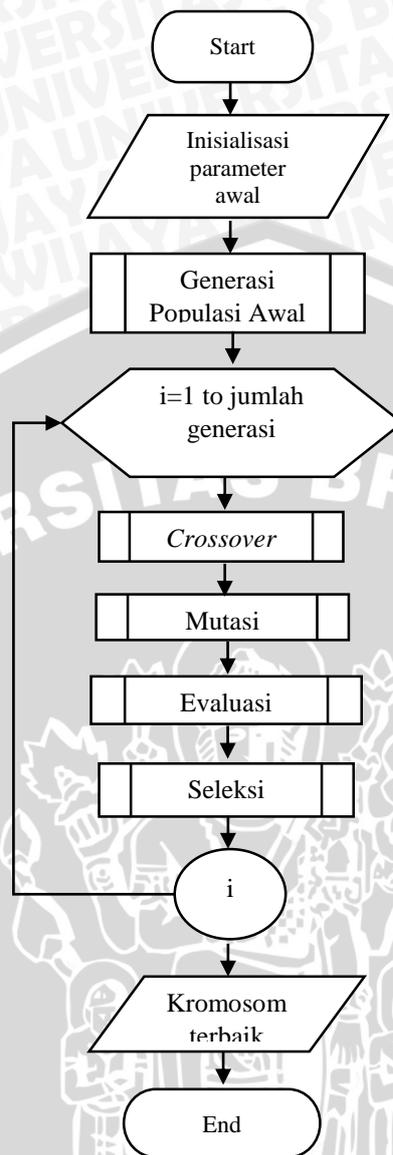
$$\begin{aligned} \text{Jumlah kebutuhan pakan/hari} &= \frac{8,3 + 2,2 \times \frac{1800}{454} + 0,1 \times 70}{100} \times 454 \\ &= 109,062 \text{ gram} \end{aligned}$$

Masalah yang ingin diselesaikan di sini adalah bagaimana mendapatkan kombinasi pakan ayam dengan biaya minimum namun kebutuhan nutrisinya tetap terpenuhi.

3.4 Siklus Penyelesaian Masalah

Dalam penyelesaiannya, algoritma memiliki beberapa proses, diantaranya adalah representasi kromosom, *crossover*, mutasi, *fitness*, dan seleksi. Berikut ini adalah *flowchart* dari proses algoritma genetika yang digambarkan pada Gambar 3.2.





Gambar 3.2 *Flowchart* Algoritma Genetika

Pada Gambar 3.2 menggambarkan proses algoritma genetika secara garis besar yang meliputi :

1. Inisialisasi parameter awal yang berupa inputan awal parameter algoritma genetika dan inputan *user*,
2. Membangkitkan generasi awal yang terdiri dari sejumlah individu yang telah ditentukan sebelumnya secara acak,
3. Melakukan proses *crossover* yang bertujuan untuk menghasilkan *offspring*,
4. Melakukan proses mutasi berdasarkan nilai *mutation rate*,
5. Melakukan proses evaluasi dengan menghitung nilai *fitness* setiap individu,

6. Melakukan proses seleksi untuk menghasilkan sebuah solusi dan menentukan populasi baru.

Algoritma genetika digunakan untuk menentukan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum. Untuk parameter algoritma genetika, misalkan *user* menginputkan data seperti berikut ini :

Ukuran populasi : 10

Jumlah generasi : 1

Crossover rate : 0,5

Mutation rate : 0,1

Contoh siklus algoritma genetika akan diuraikan pada sub-bab 3.4.1 sampai 3.4.6.

3.4.1 Representasi Kromosom Dan Perhitungan *Fitness*

Representasi merupakan gambaran hasil akhir dari suatu permasalahan yang akan diselesaikan. Dalam penelitian ini digunakan representasi *real code*. Tiap-tiap kromosom berisi bilangan pecahan pada *interval* [1...10] yang menunjukkan bobot pada masing-masing bahan pakan. Panjang kromosom yang digunakan adalah sepanjang jumlah bahan pakan yang tersedia yaitu sebanyak 10. Kromosom ini berisikan gen yang berupa bobot bahan makanan dalam satuan persen (%). Setiap kolom merupakan penunjuk bahwa gen pada kolom 1 merupakan bobot dari bahan makanan pada nomor 1, gen pada kolom 2 berupa bobot dari bahan makanan nomor 2, dan seterusnya. Untuk lebih jelasnya akan diberikan contoh sebuah kromosom pada Gambar 3.3.

B1	B2	B3	B4	B5	B6	B7	B8	B9	B10
5	9	1	3	10	7	1	3	8	10

Gambar 3.3 Contoh kromosom

Berdasarkan pada Gambar 3.3 diketahui bahwa sebuah kromosom memiliki bobot bahan 1 sebesar 5 %, bobot bahan 2 sebesar 9 %, bobot bahan 3 sebesar 3 %, bobot bahan 4 sebesar 3 %, bobot bahan 5 sebesar 10 %, bobot bahan 6 sebesar 7 %, bobot bahan 7 sebesar 1 %, bobot bahan 8 sebesar 3 %, bobot bahan 9 sebesar 8 %, bobot bahan 10 sebesar 10 %.

Selanjutnya akan dilakukan perhitungan *fitness* untuk setiap kromosom. Hasil nilai *fitness* inilah yang nantinya akan digunakan sebagai acuan dalam membentuk generasi baru. Semakin besar nilai *fitness* maka semakin baik kromosom tersebut untuk dijadikan calon solusi. Penyusunan fungsi persamaan untuk perhitungan *fitness* didasarkan pada biaya dan nilai penalty yang berbanding terbalik dengan optimasi. Untuk mencegah terjadinya hasil *fitness* yang terlalu kecil, maka dibuatlah sebuah *konstanta* (K) yang berfungsi sebagai pengali senilai 1000. Contoh perhitungan *fitness* dari contoh kromosom pada Gambar 3.2 adalah sebagai berikut.

Langkah awal yaitu melakukan *sorting* pada tiap-tiap kromosom. *Sorting* dilakukan dengan mengambil 6 dari 10 gen pada sebuah kromosom. Pengambilan 6 gen ini didasarkan pada gen yang memiliki bobot tertinggi. Jika 6 gen dengan nilai bobot tertinggi telah ditentukan, maka gen yang tidak terpilih otomatis nilainya menjadi 0. Hasil dari proses *sorting* ini ditunjukkan pada Gambar 3.4.

5	9	0	0	10	7	0	0	8	10
---	---	---	---	----	---	---	---	---	----

Gambar 3.4 Hasil *Sorting* Bobot Bahan Makanan

Karena total bobot pakan belum mencapai 100 %, maka harus dinormalisasi terlebih dahulu sehingga total hasilnya adalah 100%. Normalisasi bobot pakan menggunakan persamaan seperti berikut ini:

$$\text{Normalisasi bobot pakan } j = \frac{\text{bobot bahan } i,j \text{ (\%)}}{\text{Total bobot bahan terpilih}} \times 100\% \dots(3.1)$$

Pada Persamaan 3.1, variable j pada normalisasi bobot pakan merupakan sebuah nomor yang menunjukkan kolom keberapa sebuah gen berada. Variabel i pada bobot bahan pakan menunjukkan bobot bahan makanan pada kromosom yang diuji. Dan variabel j menunjukkan bobot bahan pakan pada gen yang diuji. Persamaan 3.1 diimplementasikan untuk menghitung normalisasi bobot seperti berikut.

$$\text{Normalisasi bobot pakan}_1 (\%) = \frac{5}{49} \times 100 \%$$

$$= 10.20408$$

$$\text{Normalisasi bobot pakan}_2 (\%) = \frac{9}{49} \times 100 \%$$

$$= 18.36735$$

$$\text{Normalisasi bobot pakan}_5 (\%) = \frac{10}{49} \times 100 \%$$

$$= 20.40816$$

$$\text{Normalisasi bobot pakan}_6 (\%) = \frac{7}{49} \times 100 \%$$

$$= 14.28571$$

$$\text{Normalisasi bobot pakan}_9 (\%) = \frac{8}{49} \times 100 \%$$

$$= 16.32653$$

$$\text{Normalisasi bobot pakan}_{10} (\%) = \frac{10}{49} \times 100 \%$$

$$= 20.40816$$

Selanjutnya melakukan perhitungan jumlah nutrisi bahan pakan dengan menggunakan Persamaan 2.2 – 2.7.

- Protein₁ = $\frac{10.20408}{100} \times 12 \%$
= 1.22449 %

- $ME_1 = \frac{10.20408}{100} \times 2860 \text{ Kcal/kg}$
 $= 291.8367 \text{ Kcal/kg}$
- $Lemak_1 = \frac{10.20408}{100} \times 12\%$
 $= 1.22449 \%$
- $Serat\ Kasar_1 = \frac{10.20408}{100} \times 3 \%$
 $= 0.306122 \%$
- $Kalsium_1 = \frac{10.20408}{100} \times 0,04 \%$
 $= 0.004082 \%$
- $Fosfor_1 = \frac{10.20408}{100} \times 1,4 \%$
 $= 0.142857 \%$

Perhitungan nutrisi dilakukan ke semua gen sampai selesai dengan menggunakan Persamaan 2.2 sampai 2.7 seperti yang telah dilakukan pada contoh perhitungan nutrisi di atas. Tabel 3.1 memaparkan semua hasil dari perhitungan nutrisi pada masing-masing gen. Dari hasil perhitungan nutrisi inilah dapat ditentukan penalti yang nantinya akan digunakan sebagai pertimbangan dalam menghitung nilai *fitness*.

Tabel 3.1 Hasil perhitungan nutrisi

No	Protein	ME	Lemak	Serat kasar	Kalsium	Fosfor
1	1.22449	291.8367	1.22449	0.306122	0.004082	0.142857
2	0.734694	244.2857	0.156122	0.826531	0.008265	0.033061
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	8.571429	461.2245	0.387755	3.469388	0.040816	0.122449
6	5.857143	260	0.271429	1.857143	0.028571	0.171429
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	7.346939	311.8367	0.816327	0.816327	0.326531	0.244898
10	6.530612	557.1429	1.020408	0.081633	0.265306	0.183673
Total	30.26531	2126.327	3.876531	7.357143	0.673571	0.898367

Setelah kebutuhan nutrisi diketahui, langkah selanjutnya adalah membandingkan total nutrisi diatas dengan kebutuhan nutrisi ayam yang terdapat pada Tabel 2.2 dengan Persamaan (2.9) seperti yang dilansir pada (Sari, 2014) berikut ini.

$$Penalti_{1...6} = \begin{cases} 0, & TotalNut \geq KebNut \\ KebNut - TotalNut, & TotalNut < KebNut \end{cases}$$

$$Penalti Protein = 30.26531 > 18 \text{ maka penalti} = 0$$

$$Penalti ME = 2126.327 < 2850 \text{ maka } 2850 - 2126.327 = 723.6735$$

$$Penalti Lemak = 3.876531 < 5 \text{ maka } 5 - 3.876531 = 1.123469$$

$$Penalti Serat = 7.357143 > 4 \text{ maka penalti} = 0$$

$$Penalti Kalsium = 0.673571 < 3 \text{ maka } 3 - 0.673571 = 2.326429$$

$$Penalti Fosfor = 0.898367 > 0,5 \text{ maka penalty} = 0$$

Nilai penalti yang didapat dari penalti ME harus dikonversikan terlebih dahulu menjadi prosentase agar dapat dijumlahkan dengan penalti nutrisi yang lainnya.

$$\text{Penalti ME} = \frac{723.6735 \text{ Kcal/kg}}{2850 \text{ Kcal/kg}} \times 100 \% = 25.39205 \%$$

Setelah didapat nilai dari penalti masing-masing nutrisi, langkah selanjutnya adalah menghitung total keseluruhan penalti.

$$\begin{aligned} \text{Penalti} &= 0 + 25.39205 + 0.033898 + 0 + 0.542372881 + 0 \\ &= 28.84195 \end{aligned}$$

Selanjutnya adalah melakukan perhitungan biaya. Untuk perhitungan biaya (*cost*) caranya adalah dengan mengalikan tiap bilangan kromosom dengan harga bahan pakan per gram. Data harga bahan makanan ini tersaji pada Lampiran 2.

Perhitungan biaya bahan pakan ternak ditunjukkan pada Persamaan 3.2 berikut ini.

$$\text{Harga} = \left(\frac{\text{Bobot bahan } i,j}{100} \times \text{keb. pakan/hari} \right) \times \text{harga bahan } i \dots (3.2)$$

Pada Persamaan 3.2 variabel *i,j* menunjukkan bobot bahan yang diuji terletak pada kromosom ke-*I* dan gen ke-*j*. Sedangkan variable *i* pada harga bahan pakan menunjukkan bahwa harga bahan pakan yang dipakai dalam perhitungan terletak pada baris ke-*i*. Implementasi Persamaan 3.2 adalah sebagai berikut.

$$\text{Harga}_1 = \left(\frac{10.20408}{100} \times 109,062 \right) \times 1,6 = 17,806$$

$$\text{Hargaa}_2 = \left(\frac{18.36735}{100} \times 109,062 \right) \times 3 = 60.095$$

$$\text{Harga}_5 = \left(\frac{20.40816}{100} \times 109,062 \right) \times 3,5 = 77.901$$

$$\text{Harga}_6 = \left(\frac{14.28571}{100} \times 109,062 \right) \times 2 = 31.161$$

$$\text{Harga}_9 = \left(\frac{16.32653}{100} \times 109,062 \right) \times 2 = 35.612$$

$$Harga_{10} = \left(\frac{20.40816}{100} \times 109,062 \right) \times 2,5 = 55.643$$

$$Total\ harga = 278,2193$$

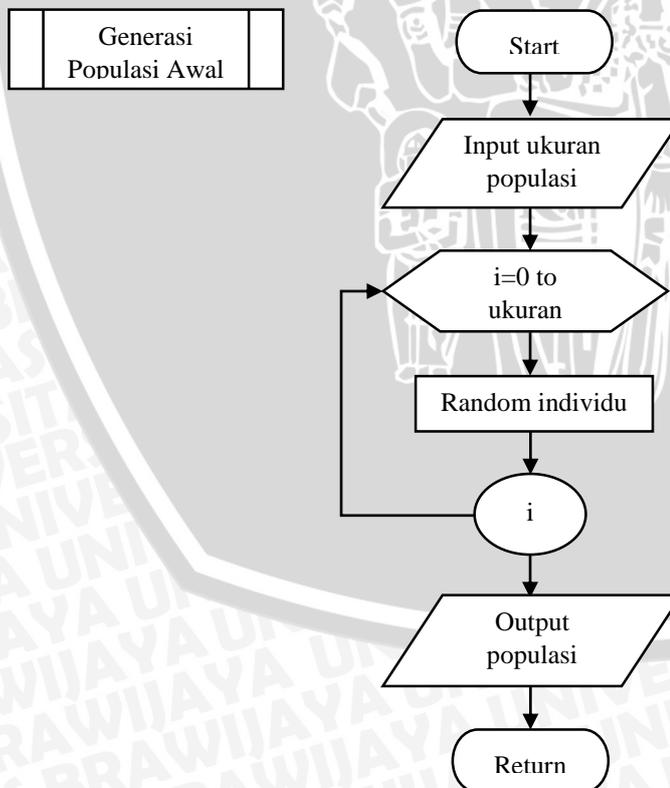
Setelah diketahui total harga dan nilai penaltinya, selanjutnya menghitung nilai *fitness*nya dengan Persamaan 2.8.

$$Fitness = \frac{1}{cost + (penalti \times \alpha)} \times K$$

$$Fitness = \frac{1}{278,2193 + (28,84195 \times 20)} \times 1000 = 1.169511$$

3.4.2 Inisialisasi Populasi Awal

Pada proses ini dilakukan inisialisasi populasi awal sebanyak jumlah populasi yang telah ditentukan sebelumnya. Diagram alir dari proses inisialisasi populasi awal ini dapat dilihat pada Gambar 3.5.



Gambar 3.5 Flowchart membangun generasi awal

Tahapan inialisasi populasi awal ini dilakukan untuk membuat populasi awal sebanyak jumlah ukuran populasi yang telah ditentukan dengan cara membangkitkan nilai-nilai *random*. Representasi kromosom yang digunakan dalam penelitian ini berupa representasi *real*. Dimana tiap-tiap kromosom diisikan dengan nilai random pada *interval* [1...10] yang berupa bobot bahan pakan ternak dalam %. Panjang setiap kromosomnya adalah sejumlah bahan makanan yang akan digunakan dalam penelitian yaitu sejumlah 10 jenis bahan makanan, sehingga panjang kromosomnya adalah 10. Bobot bahan makanan akan dianggap sebagai gen, dimana gen-gen ini memiliki label protein, ME, lemak, serat kasar, kalsium, dan fosfor, serta harga. Hasil dari inialisasi populasi awal dijabarkan pada Tabel 3.2 berikut ini.

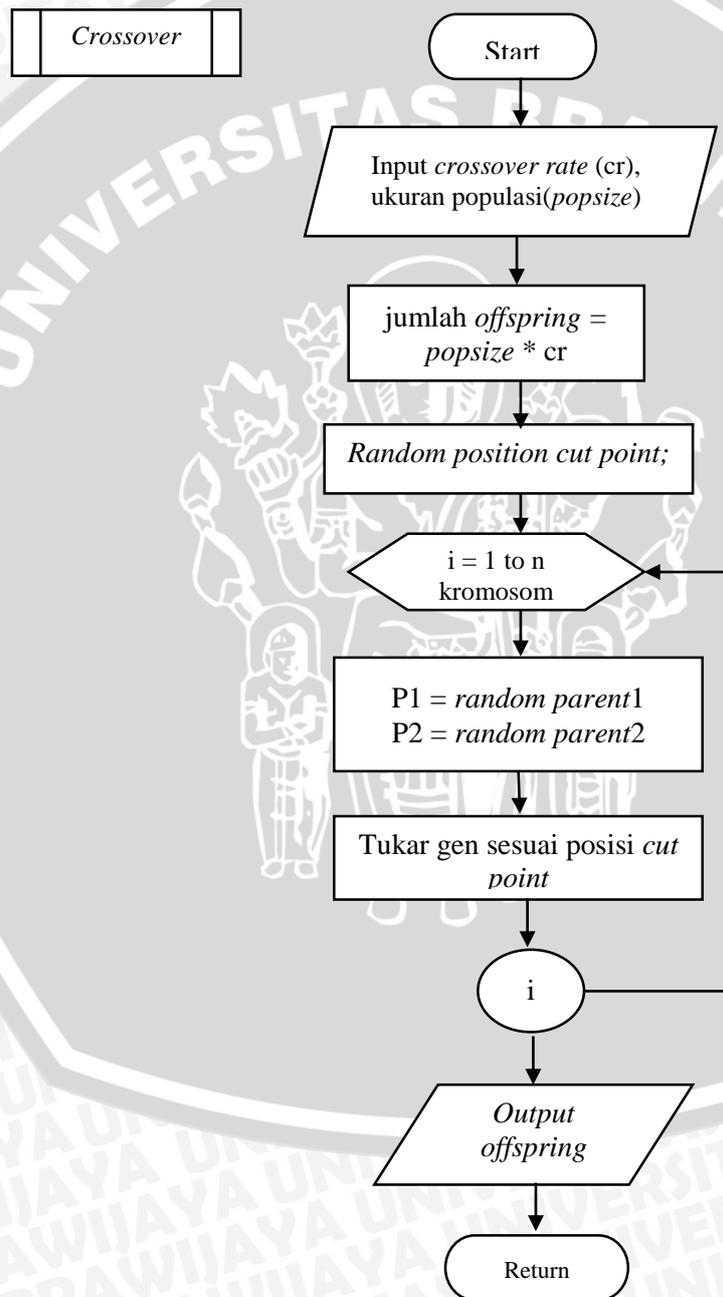
Tabel 3.2 Inialisasi Populasi Awal

<i>Parent</i>	Kromosom									
1	5	9	1	3	10	7	1	3	8	10
2	9	6	5	1	7	3	2	3	10	7
3	1	5	7	10	5	6	8	4	7	9
4	3	5	8	5	3	9	9	1	2	9
5	10	1	2	5	2	3	2	6	4	6
6	7	5	3	10	8	4	8	4	3	1
7	1	8	6	9	4	8	5	1	9	2
8	3	9	9	7	7	3	5	7	3	8
9	3	4	3	4	7	7	5	5	4	9
10	10	3	7	9	4	2	3	6	6	6

Berdasarkan pada Tabel 3.2 diketahui bahwa *Parent* 1 memiliki bobot bahan 1 sebesar 5 %, bobot bahan 2 sebesar 9 %, bobot bahan 3 sebesar 3 %, bobot bahan 4 sebesar 3 %, bobot bahan 5 sebesar 10 %, bobot bahan 6 sebesar 7 %, bobot bahan 7 sebesar 1 %, bobot bahan 8 sebesar 3 %, bobot bahan 9 sebesar 8 %, bobot bahan 10 sebesar 10 %, dst.

3.4.3 Crossover

Proses *crossover* merupakan sebuah proses reproduksi yang akan menghasilkan kromosom anak (*offspring*). Jumlah dari kromosom anak (*offspring*) yang akan dihasilkan dihitung berdasarkan *crossover rate* dan ukuran populasi yang telah ditentukan sebelumnya. Berikut ini adalah diagram alir pada proses *crossover* yang ditampilkan pada Gambar 3.6.



Gambar 3.6 Flowchart proses *crossover*

Metode yang digunakan dalam proses *crossover* pada penelitian ini adalah metode *one cut point crossover*. Metode *one cut point*, memilih satu titik potong secara acak dan menukarkan gen yang terletak pada bagian kanan. Kromosom baru yang dihasilkan dari proses *crossover* ini selanjutnya akan dijadikan kromosom anak (*offspring*). Banyaknya *offspring* yang dibutuhkan mengacu pada *crossover rate* dan ukuran populasinya. Sebelumnya telah ditentukan *crossover ratenya* sebesar 0,5 dan ukuran populasinya sebesar 10. Sehingga *offspring* yang dibutuhkan sebesar $0,5 \times 10 = 5$. Proses *crossover* ditunjukkan pada Gambar 3.7.

P1	5	9	1	3	10	7	1	3	8	10
P2	9	6	5	1	7	3	2	3	10	7
C1	5	9	1	3	10	3	2	3	10	7
C2	9	6	5	1	7	7	1	3	8	10

P3	1	5	7	10	5	6	8	4	7	9
P5	10	1	2	5	2	3	2	6	4	6
C3	1	5	7	10	5	3	2	6	4	6
C4	10	1	2	5	2	6	8	4	7	9

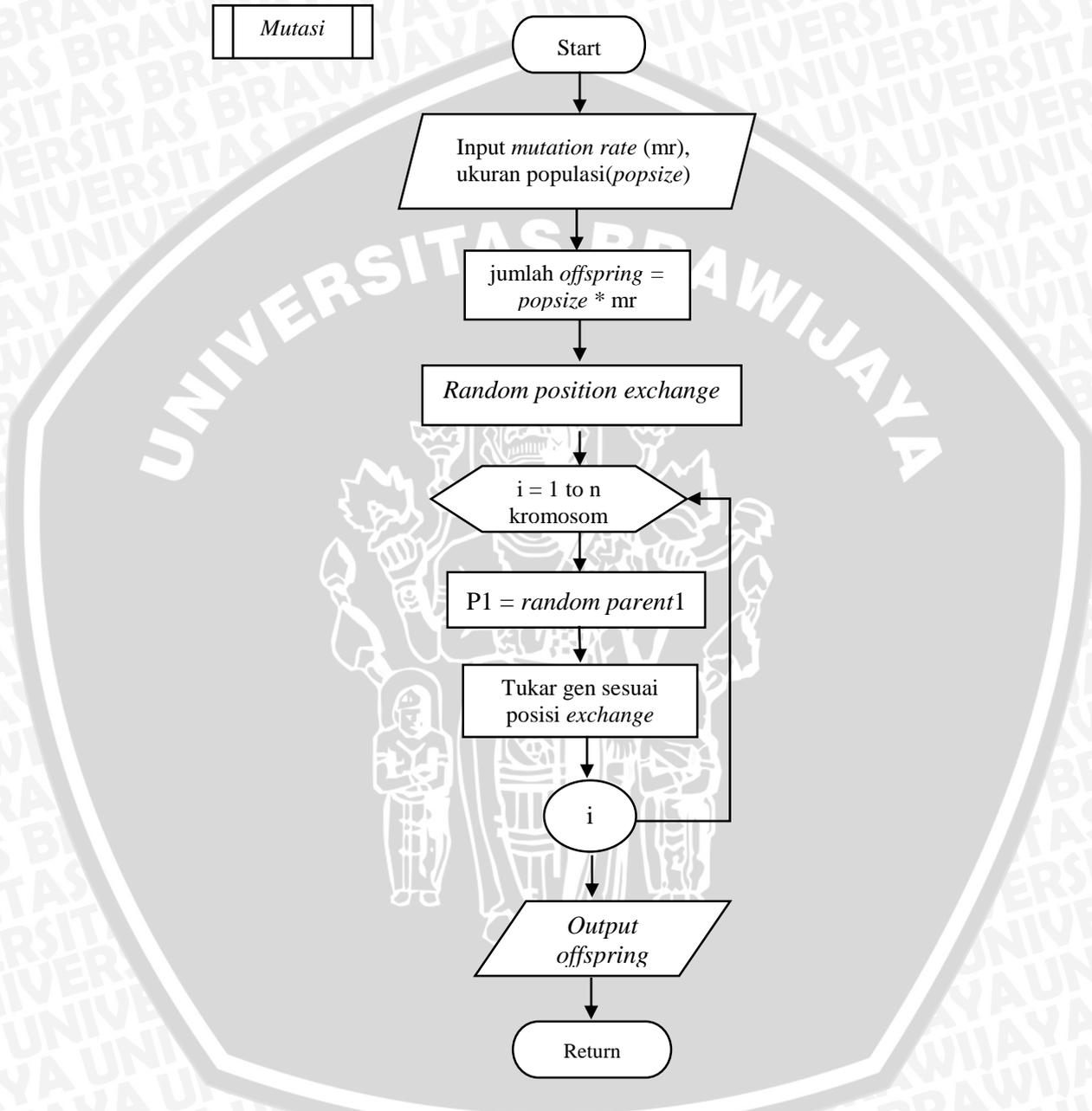
P8	3	9	9	7	7	3	5	7	3	8
P9	3	4	3	4	7	7	5	5	4	9
C5	3	9	9	7	7	3	5	5	4	9

Gambar 3.7 Proses *crossover*

3.4.4 Mutasi

Pada proses mutasi hampir sama dengan proses *crossover*. Mutasi juga merupakan proses reproduksi yang nantinya akan menghasilkan kromosom anak (*offspring*). Banyaknya kromosom anak (*offspring*) yang dibutuhkan dalam proses mutasi ini adalah berdasarkan pada besarnya *mutation rate* dan ukuran populasi

yang telah di-inisialisasikan sebelumnya. Gambar 3.8 berikut ini adalah *flowchart* dari proses mutasi.



Gambar 3.8 *Flowchart* proses mutasi

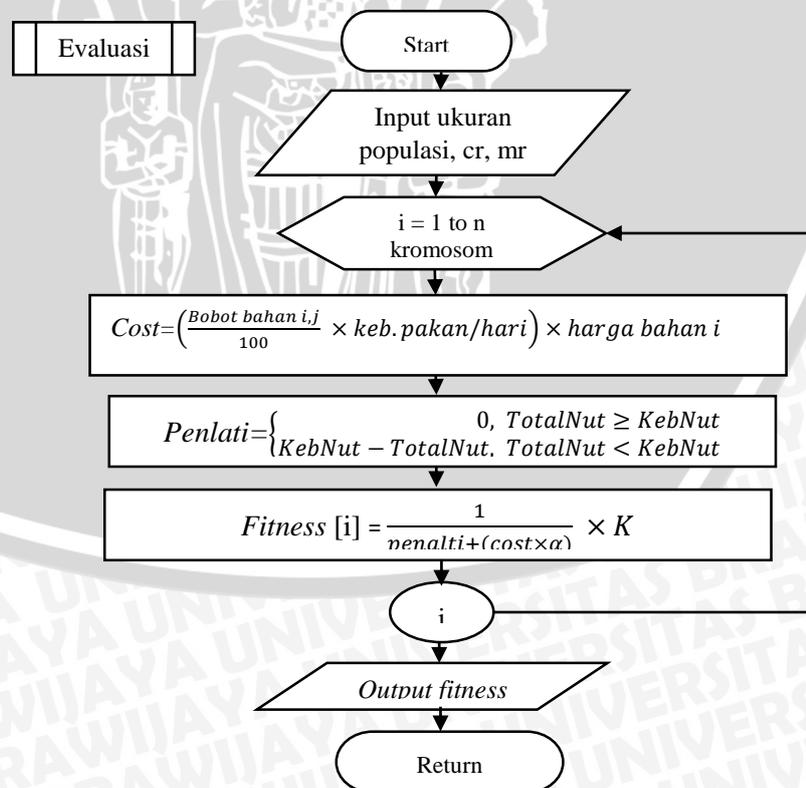
Pada proses mutasi, metode yang digunakan adalah metode *reciprocal exchange*. Metode ini bekerja dengan memilih dua posisi (*exchange point*) secara random kemudian menukarkan nilai pada posisi tersebut (Mahmudy, 2014). Banyaknya *offspring* yang dibutuhkan mengacu pada *mutation rate* dan *popSize* yang telah ditentukan sebelumnya yaitu *mutation rate* sebesar 0,1 dan ukuran populasinya sebesar 10. Sehingga *offspring* yang dibutuhkan sebesar $0,1 \times 10 = 1$. Proses mutasi ditunjukkan pada Gambar 3.9.

P10	10	3	7	9	4	2	3	6	6	6
C6	10	3	7	3	4	2	9	6	6	6

Gambar 3.9 Proses mutasi

3.4.5 Evaluasi

Setelah hasil dari reproduksi didapatkan, maka akan dilakukan proses perhitungan nilai *fitness*. Semua kromosom pada populasi awal dan juga hasil dari *offspring* pada proses *crossover* dan mutasi dihitung nilai *fitness*nya. Penjelasan singkatnya dapat dilihat dari diagram *flowchart* yang ditunjukkan pada Gambar 3.10 berikut ini :



Gambar 3.10 Flowchart proses hitung *fitness*

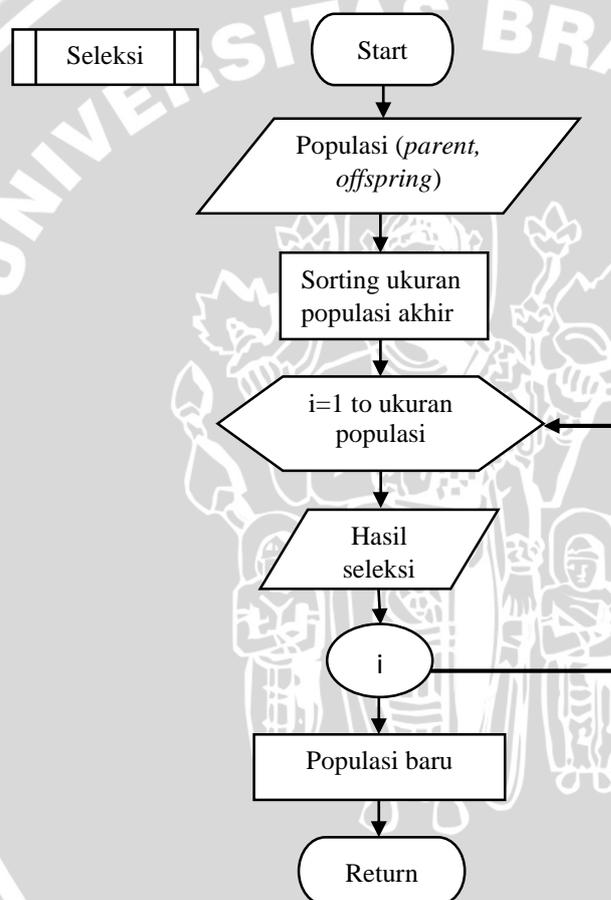
Proses evaluasi dilakukan dengan melakukan perhitungan nilai *fitness* untuk setiap kromosom. Hasil nilai *fitness* ini nantinya akan digunakan sebagai acuan dalam membentuk generasi baru. Semakin besar nilai *fitness* maka semakin baik kromosom tersebut untuk dijadikan calon solusi. Perhitungan nilai *fitness* dilakukan terhadap seluruh individu. Proses perhitungan nilai *fitness* telah dijelaskan pada subbab sebelumnya yaitu pada subbab 3.4.1. *Fitness* seluruh individu pada iterasi ke-1 ditunjukkan Tabel 3.3 .

Tabel 3.3 Hasil Perhitungan *Fitness*

Parent	Penalti	Harga	<i>Fitness</i>
P1	2.594112281	418.1326	2.127592
P2	25.21334737	329.0994	1.199952
P3	17.45055088	405.7476	1.324927
P4	14.07136491	473.7727	1.324153
P5	13.2380807	338.2802	1.65826
P6	12.47649825	506.0859	1.323424
P7	25.17241404	446.5901	1.052589
P8	12.55734035	440.3635	1.44611
P9	18.41410175	355.8379	1.380987
P10	16.40282456	419.9844	1.336825
C1	8.390635088	320.7927	2.046641
C2	29.00553684	328p.1262	1.101034
C3	25.8331193	384.9697	1.1091
C4	7.211403509	396.2586	1.850184
C5	14.19517895	404.5626	1.452504
C6	16.12206667	459.4789	1.278903

3.4.6 Seleksi

Dalam penelitian ini menggunakan seleksi *elitis* sebagai proses seleksinya. Seleksi *elitis* bekerja dengan menggunakan acuan nilai *fitness* yang telah ditentukan sebelumnya. Seleksi dilakukan dengan mengurutkan induk dan *offspring* berdasarkan nilai *fitness*nya mulai dari individu dengan nilai *fitness* tertinggi sampai individu dengan nilai *fitness* terendah. Jumlah individu hasil seleksi harus berdasarkan pada jumlah ukuran populasi awal, yaitu sebanyak 10 individu. Diagram alir dari proses seleksi ini adalah sebagai berikut :



Gambar 3.11 Flowchart proses seleksi

Seleksi dilakukan untuk memilih 10 dari 16 individu (10 *parent* + 6 *child*) yang dipertahankan hidup pada generasi berikutnya. Pemilihan individu baru dilakukan dengan cara mengurutkan keseluruhan individu mulai dari *parent* sampai *child* dari individu yang bernilai *fitness* tinggi sampai ke individu bernilai *fitness* rendah kemudian diambil 10 individu teratas yang memiliki nilai *fitness* yang tinggi. Setelah terpilih 10 individu baru, maka dilakukan perhitungan iterasi kedua

dengan cara yang sama dengan iterasi pertama namun dengan kromosom baru yang dihasilkan dari proses seleksi dengan metode *Elitis* (seperti pada proses inisialisasi) yaitu 10 individu.

Tabel 3.4 Hasil Seleksi

(p+1)	Hasil seleksi	Fitness
P1	P1	2.127592
P2	C1	2.046641
P3	C4	1.850184
P4	P5	1.65826
P5	C5	1.452504
P6	P8	1.44611
P7	P9	1.380987
P8	P10	1.336825
P9	P3	1.324927
P10	P4	1.324153

Tabel 3.4 merupakan hasil dari seleksi menggunakan metode *elitism*. Tabel tersebut menunjukkan bahwa *Parent 1* (P1) memiliki nilai *fitness* tertinggi yaitu sebesar 2.127592. Sehingga dapat diambil kesimpulan bahwa *Parent 1* (P1) lolos pada tahapan seleksi dan digunakan sebagai solusi dalam permasalahan “Penentuan Komposisi Pakan Ternak Untuk Memenuhi Nutrisi Ayam Petelur Dengan Biaya Minimum Menggunakan Algoritma Genetika”.

BAB IV PERANCANGAN

4.1 Perancangan Database

Sistem penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum menggunakan algoritma genetika ini menggunakan database PHP MyAdmin dengan nama database pakan ternak. Dalam database pakan ternak ini hanya terdapat satu tabel yaitu tabel bahan makanan. Tabel bahan makanan ini berfungsi untuk menyimpan data bahan makanan yang digunakan dalam penelitian. Bahan makanan yang tersimpan dalam database sejumlah 40 data. Tabel bahan makanan ini memiliki kolom nomor, nama bahan makanan, kadar protein, kadar ME, kadar lemak, kadar serat kasar, kadar kalsium, dan kadar fosfor, serta harga. Tabel database sistem akan ditunjukkan pada Gambar 4.1.

+ Options		no	bahan_makanan	protein	me	lemak	serat	kalsium	fosfor	harga
<input type="checkbox"/>	Edit Copy Delete	1	bekatul	12	2860	12	3	0.04	1.4	1.6
<input type="checkbox"/>	Edit Copy Delete	2	beras merah	8	2660	1.7	9	0.09	0.36	3
<input type="checkbox"/>	Edit Copy Delete	3	beras putih	7.5	3110	0.4	0.4	0.03	0.12	3
<input type="checkbox"/>	Edit Copy Delete	4	bungkil kacang kedele	45	2240	0.9	6	0.32	0.67	5.5
<input type="checkbox"/>	Edit Copy Delete	5	bungkil kacang tanah	42	2260	1.9	17	0.2	0.6	3.5
<input type="checkbox"/>	Edit Copy Delete	6	bungkil biji kapas	41	1820	1.9	13	0.2	1.2	2
<input type="checkbox"/>	Edit Copy Delete	7	bungkil kelapa (indonesia)	18.58	2212	12.55	15.38	0.21	0.6	2.5
<input type="checkbox"/>	Edit Copy Delete	8	bungkil biji bunga matahari	43	1760	2.8	14	0.4	1	2.5
<input type="checkbox"/>	Edit Copy Delete	9	bungkil wijen	45	1910	5	5	2	1.5	2
<input type="checkbox"/>	Edit Copy Delete	10	buttermilk dikeringkan	32	2730	5	0.4	1.3	0.9	2.5
<input type="checkbox"/>	Edit Copy Delete	11	dedak gandum	15	1300	4	10	0.14	1.1	2
<input type="checkbox"/>	Edit Copy Delete	12	dedak halus	12	1630	13	12	0.12	1.5	3
<input type="checkbox"/>	Edit Copy Delete	13	dedak jagung	10.6	2950	6	5	0.04	0.5	2
<input type="checkbox"/>	Edit Copy Delete	14	dedak lunteh	13.5	1630	13	12	0.12	1.5	2.5
<input type="checkbox"/>	Edit Copy Delete	15	dedak terigu	15	1300	4	10	0.14	1.1	2
<input type="checkbox"/>	Edit Copy Delete	16	gabah	7.3	2668	2.3	9	0.04	0.26	2.5
<input type="checkbox"/>	Edit Copy Delete	17	gandum	12.7	3071	2.2	3	0.05	0.36	3.5
<input type="checkbox"/>	Edit Copy Delete	18	jagung kuning	8.6	3370	3.9	2	0.02	0.3	3
<input type="checkbox"/>	Edit Copy Delete	19	kacang buncis	23.5	2330	1.4	4.5	0.13	0.6	3
<input type="checkbox"/>	Edit Copy Delete	20	kacang hijau	23.5	2330	1.2	4.5	0.13	0.6	3.5
<input type="checkbox"/>	Edit Copy Delete	21	kacang kapri	22	2200	1.1	6	0.15	0.3	3
<input type="checkbox"/>	Edit Copy Delete	22	kacang kedele	38	3510	18	5	0.25	0.6	5.5

Gambar 4.1 Tabel database bahan pakan

4.2 Perancangan User Interface

Dalam perancangan desain antarmuka program penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum menggunakan algoritma genetika ini terdiri dari satu halaman utama. Di dalam halaman utama ini terdapat tiga *tab*. *Tab* yang pertama berisikan informasi data bahan pakan ternak. *Tab* kedua berisi hasil optimasi atas data yang telah diinputkan. Dan *tab* terakhir merupakan informasi dari hasil optimasi yang dijabarkan secara lebih mendetail.

Halaman utama pada program aplikasi yang akan dibuat terdiri dari dua sisi. Sisi sebelah kiri merupakan ruang untuk menginputkan data yang dibutuhkan oleh system yaitu data parameter algoritma genetika (ukuran populasi, jumlah generasi, *crossover rate*, *mutation rate*) dan data user (bobot ayam dan prosentase produksi telur). Di bawah panel input juga terdapat informasi kebutuhan nutrisi standart yang dibutuhkan oleh ayam yang diuji.

Pada sisi sebelah kanan terdapat tiga *tab*. *Tab* yang pertama merupakan halaman data bahan makanan. Halaman data bahan makanan memuat informasi tentang kandungan nutrisi pada bahan makanan, yang terdiri dari protein, ME, lemak, serat kasar, kalsium, dan fosfor. Pada halaman ini juga disertakan informasi harga pada masing-masing bahan makanan.

No	Nutrisi	Jumlah Kebutuhan
1	Protein	18 %
2	ME	2850 Kcal/kg
3	Lemak	5 %
4	Serat Kasar	4 %
5	Kalsium	3 %
6	Fosfor	0.5 %

Gambar 4.2 Rancangan Halaman Utama

Tab yang kedua berisikan hasil dari proses algoritma genetika. Pada halaman ini terdapat hasil optimasi yang berisikan detail hasil seleksi yang terdiri dari generasi ke-, nama kromosom, kromosom, penalty, harga, dan *fitness*. Selain itu juga ditampilkan hasil seleksi dari proses perhitungan menggunakan algoritma genetika.

Penentuan Komposisi Pakan Ternak Dengan Biaya Minimum Menggunakan GA's

Input Data GA's

Ukuran populasi:

Jumlah Generasi:

Crossover rate:

Mutation rate:

Input Data User

Berat ayam (gram):

Produksi telur (%):

Proses

Kebutuhan Nutrisi

No	Nutrisi	Jumlah Kebutuhan
1	Protein	18 %
2	ME	2850 Kcal/kg
3	Lemak	5 %
4	Serat Kasar	4 %
5	Kalsium	3 %
6	Fosfor	0.5 %

Data Bahan Makanan | Hasil Proses GA's | Detail Nutrisi Hasil Optimasi

Hasil Optimasi Seleksi

Generasi ke-	Nama	Kromosom	Penalti	Harga	Fitness
--------------	------	----------	---------	-------	---------

Hasil Seleksi

Generasi ke-	Nama	Kromosom	Penalti	Harga	Fitness
--------------	------	----------	---------	-------	---------

Gambar 4.3 Rancangan Hasil Proses GA's

Tab yang ketiga berisikan halaman yang menunjukkan detail dari kromosom yang terpilih untuk generasi selanjutnya. Diantaranya adalah nama bahan makanan, kadar protein, ME, lemak, serat kasar, kalsium, dan fosfor, serta harga dari bahan makanan terpilih.

Penentuan Komposisi Pakan Ternak Dengan Biaya Minimum Menggunakan GA's

Input Data GA's

Ukuran populasi:

Jumlah Generasi:

Crossover rate:

Mutation rate:

Input Data User

Berat ayam (gram):

Produksi telur (%):

Proses

Kebutuhan Nutrisi

No	Nutrisi	Jumlah Kebutuhan
1	Protein	18 %
2	ME	2850 Kkal/kg
3	Lemak	5 %
4	Serat Kasar	4 %
5	Kalsium	3 %
6	Fosfor	0.5 %

Gambar 4.4 Rancangan Detail Hasil Optimasi

4.3 Perancangan Uji Coba dan Evaluasi

Untuk melakukan evaluasi program perlu dilakukan beberapa kali uji coba karena tidak adanya metode yang pasti untuk menentukan parameter algoritma genetika (Pratiwi, 2014). Uji coba yang perlu dilakukan adalah uji coba pada parameter algoritma genetika, diantaranya adalah :

1. Uji coba pada ukuran populasi,
2. Uji coba pada banyaknya generasi,
3. Uji coba pada kombinasi *crossover rate* dan *mutation rate*.

4.3.1 Uji Coba Ukuran Populasi

Uji coba pada ukuran populasi perlu dilakukan karena berfungsi untuk mengetahui ukuran populasi yang optimal untuk permasalahan penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum. Menurut Mahmudy (2014), semakin besar ukuran populasi maka akan semakin baik, namun jika terlalu besar akan memperlambat proses *system*. Dalam pengujian ini, ukuran populasi yang digunakan adalah kelipatan 20. Dan parameter yang digunakan dalam pengujian banyak generasi adalah :

1. Ukuran populasi : 20-200
2. Ukuran generasi : 250
3. *Crossover rate* : 0,5
4. *Mutation rate* : 0,1

Tabel 4.1 Rancangan uji coba ukuran populasi

Ukuran Populasi	Nilai <i>Fitness</i>										Rata- rata fitness	Rata rata waktu
	Percobaan populasi ke-											
	1	2	3	4	5	6	7	8	9	10		
20												
40												
60												
80												
100												

4.3.2 Uji Coba Banyak Generasi

Uji coba banyaknya generasi perlu dilakukan karena berfungsi untuk mengetahui banyaknya generasi yang optimal untuk permasalahan penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum. Dalam uji coba ini, jumlah generasi yang digunakan adalah kelipatan 250. Dan parameter yang digunakan dalam pengujian banyak *generasi* adalah :

1. Ukuran populasi : Ukuran populasi optimal dari pengujian ukuran populasi
2. Ukuran generasi : 250-2250
3. *Crossover rate* : 0,5
4. *Mutation rate* : 0,1

Tabel 4.2 Rancangan uji coba banyak generasi

Jumlah generasi	Nilai fitness										Rata-rata fitness
	Percobaan Generasi Ke-										
	1	2	3	4	5	6	7	8	9	10	
250											
500											
750											
1000											
1250											
1500											

4.3.3 Uji Coba Kombinasi Crossover Rate dan Mutation Rate

Uji coba kombinasi *crossover rate* dan *mutation rate* perlu dilakukan karena berfungsi untuk mengetahui *crossover rate* dan *mutation rate* yang optimal untuk permasalahan penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum. Uji coba kombinasi *crossover rate* (cr) dan *mutation rate* (mr) akan dilakukan dengan memberikan nilai-nilai yang berbeda pada *range* 0 hingga 1. Hal ini dilakukan karena tidak adanya metode pasti untuk menentukan parameter genetika yang digunakan seperti *crossover rate* dan *mutation rate*.

Parameter yang digunakan dalam uji coba kombinasi cr dan mr adalah sebagai berikut :

1. Ukuran populasi = Hasil populasi terbaik pada uji coba populasi
2. Banyak generasi = Hasil generasi terbaik pada uji coba generasi

Tabel 4.3 Rancangan uji coba kombinasi Cr dan Mr .

Kombinasi		Nilai fitness										Rata-rata fitness
		Percobaan kombinasi Cr dan Mr Ke-										
Cr	Mr	1	2	3	4	5	6	7	8	9	10	
1	0											
0,9	0,1											
0,8	0,2											
0,7	0,3											
0,6	0,4											
0,5	0,5											
0,4	0,6											
0,3	0,7											
0,2	0,8											
0,1	0,9											
0	1											

BAB V IMPLEMENTASI

5.1 Lingkungan Implementasi

Pada proses implementasi terdapat dua lingkungan implementasi yang digunakan yaitu lingkungan perangkat keras dan lingkungan perangkat lunak.

5.1.1 Lingkungan Perangkat Keras

Perangkat keras yang digunakan selama proses implementasi pada penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum menggunakan algoritma genetika adalah sebagai berikut :

1. Prosesor Intel® Core™ i3-3217U 1,8 GHz
2. Memori 2GB DDR3
3. *Harddisk* dengan kapasitas 500 GB
4. Monitor 14"

5.1.2 Lingkungan Perangkat Lunak

Perangkat lunak yang digunakan selama proses implementasi pada penentuan komposisi pakan ternak untuk memenuhi kebutuhan nutrisi ayam petelur dengan biaya minimum menggunakan algoritma genetika adalah sebagai berikut :

1. Sistem operasi yang digunakan adalah Windows8 *ultimate edition* 64bit
2. *Java Editor* yang digunakan adalah Netbeans IDE 8.0.1

5.2 Implemenatsi Program

Implementasi program dijalankan berdasarkan pada metodologi dan perancangan system yang yang terdapat pada bab 3 dan bab 4. Pada subbab impementasi ini akan dipaparkan implementasi dari proses-proses yang telah dirancang sebelumnya ke dalam sebuah system dengan menggunakan bahasa pemrograman Java.

5.2.1 Inisialisasi Populasi Awal

Pada proses ini digunakan untuk inisialisasi kromosom dengan mengacak array sebanyak jumlah populasi awal. *Soerce code* 5.1 merupakan proses pembangkitan populasi awal.

```
Object[][] permutasiAwal() {
    permutasi = new Object[populasi][45];
    panjangNut = 0;
    for (int i = 0; i < populasi; i++) {
        permutasi[i][0] = "P" + (i + 1);
        System.out.println(permutasi[i][0]);
        for (int j = 1; j < 41; j++) {
            permutasi[i][j] = rand.nextInt(10) + 1;
            System.out.print(" " + permutasi[i][j]);
        }
        System.out.println("");
    }
    return permutasi;
}
```

Source code 5.1 Membangkitkan populasi awal

5.2.2 Perhitungan *Fitness*

Proses perhitungan *fitness* dilakukan dengan tujuan untuk mendapatkan kualitas sebuah kromosom dalam populasi. Parameter yang digunakan dalam menghitung *fitness* adalah data bahan makanan yang terdiri dari kandungan nutrisi dan bahan. Proses perhitungan *fitness* dapat dilihat pada *source code* 5.2.

```
void setDaftarBahanPakan(double[][] daftarBahanPakan)
{
    this.daftarBahanPakan = daftarBahanPakan;
}
void setCrossoverrate(double cr) {
    Crossoverrate = cr;
    offspringCross = (int) (populasi
    *Crossoverrate);
}
```

```
void setMutationrate(double mr) {
    Mutationrate = mr;
    offspringMutasi = (int) (populasi *
Mutationrate);
}
void setPopulasi(int pop) {
    populasi = pop;
}

int getOffspring() {
    return offspringCross + offspringMutasi;
}
void setJumlahPakanHarian(double beratRata2, int
produksiTelur) {
    jumlahPakanHarian = ((8.3 + (2.2 * (beratRata2
/ 454)) + (0.1 * produksiTelur)) / 100) * 454;
}
Object[][] hitungJumlahNutrisi(Object
permutasiNutri[][[]], int pop) {
    jumlahNutrisi = new Object[40 * pop][10];

    int counter1, counter2 = 0;
    double harga, konversiGram;
    for (int i = 0; i < pop; i++) {
        counter1 = counter2;
        jumlahNutrisi[counter1][8] =
permutasiNutri[i][0];
        total = 0;
        harga = 0;
        for (int j = 0; j < 40; j++) {
            konversiGram =
(Double.valueOf(permutasiNutri[i][j + 1].toString()) /
100) * jumlahPakanHarian;
            harga = harga + (konversiGram *
daftarBahanPakan[j + 1][7]);
            jumlahNutrisi[counter1][6] =
(konversiGram * daftarBahanPakan[j + 1][7]); //harga
            counter1++;
        }
    }
}
```

```

    }
    permutasiNutri[i][43] = harga;
    double totalNutrisi, penalti = 0;
    for (int k = 0; k < 6; k++) {
        counter1 = counter2;
        totalNutrisi = 0;
        for (int j = 0; j < 40; j++) {
            jumlahNutrisi[counter1][k] =
Double.valueOf(permutasiNutri[i][j + 1].toString()) *
daftarBahanPakan[j + 1][k + 1] / 100;
            totalNutrisi = totalNutrisi +
Double.valueOf(jumlahNutrisi[counter1][k].toString());
            counter1++;
        }
    if (k == 1) {
        penalti = (( penalti +
(Math.abs(kebutuhanNutrisi[k] - totalNutrisi) ) ) /
2850) * 100 * 20;
    } else {
        penalti = penalti +
Math.abs(kebutuhanNutrisi[k] - totalNutrisi) * 20;
    }
    permutasiNutri[i][42] = penalti;
    permutasiNutri[i][44] = 1000 / (harga +
penalti);
    counter2 = counter1;
    }
    return permutasiNutri;
}

```

Source code 5.2 Perhitungan fitness

5.2.3 Proses Crossover dan Mutasi

Metode *crossover* yang digunakan dalam penelitian ini adalah metode *one cut point crossover*. Banyak kromosom yang mengalami proses *crossover* tergantung pada ukuran populasi dan *crossover rate* yang telah diinputkan.

Sedangkan metode yang digunakan dalam proses mutasi adalah metode *reciprocal exchange*. Jumlah kromosom yang mengalami proses mutasi diketahui dari ukuran populasi dan juga dari *mutation ratenya*. Berikut adalah *source code* dari proses *crossover* dan mutasi yang ditunjukkan dalam *Source code 5.4*.

```
Object[][] crossoverReproduksi(Object[][] seleksi)
{
    //double permutasi[][]

    crossoverArray = new Object[offspringCross +
    offspringMutasi][45]; //C 207
    hasilCross = new Object[offspringCross][45];
    hasilMutasi = new Object[offspringMutasi][45];
    int randomParent[] = new int[offspringMutasi +
    offspringCross];

    System.out.println("offspring " +
    (offspringMutasi + " " + offspringCross));

    System.arraycopy(shuffleArray(offspringMutasi +
    offspringCross), 0, randomParent, 0, offspringCross +
    offspringMutasi);

    int cross;
    System.out.println("CrossOver offspring " +
    offspringCross);

    //proses Crossover
    for (int i = 0; i < offspringCross; i = i + 2) {
        hasilCross[i][0] =
        (seleksi[randomParent[i]][0] != null) ? ("C" + (i + 1) +
        " " + seleksi[randomParent[i]][0]) : null;
        if ((i + 1) < offspringCross) {

            hasilCross[i + 1][0] =
            (seleksi[randomParent[i + 1]][0] != null) ? ("C" + (i +
            2) + " " + seleksi[randomParent[i + 1]][0]) : null;

            hasilCross[i + 1][1] =
            (seleksi[randomParent[i]][1] != null) ?
            seleksi[randomParent[i]][1] : null;
        }
    }
}
```

```
        hasilCross[i][1] = (seleksi[randomParent[i + 1]][1] != null) ? seleksi[randomParent[i + 1]][1] : null;

        cross = rand.nextInt(39) + 1;

        System.out.println("parent " + (randomParent[i] + 1) + " " + (randomParent[i + 1] + 1));

        System.out.println("crossPos " + cross);
        for (int j = 1; j < cross; j++) {
            hasilCross[i][j] = seleksi[randomParent[i]][j];
            if ((i + 1) < offspringCross) {
                hasilCross[i + 1][j] = seleksi[randomParent[i + 1]][j];
            }
        }
        for (int j = cross; j < 45; j++) {
            hasilCross[i][j] = (seleksi[randomParent[i + 1]][j] != null) ? seleksi[randomParent[i + 1]][j] : null;
            if ((i + 1) < offspringCross) {
                hasilCross[i + 1][j] = (seleksi[randomParent[i]][j] != null) ? seleksi[randomParent[i]][j] : null;
            }
        }
        System.out.println("");
    }
    System.out.println("parent \tkomposisi ... ");

    for (int i = 0; i < offspringCross; i++) {
        for (int j = 0; j < 45; j++) {
            System.out.print(hasilCross[i][j] + " ");
        }
        System.out.println("");
    }
```

```
}
//proses Mutasi
Object temp1, temp2;
System.out.println("Mutasi offspring " +
offspringMutasi);
for (int i = 0; i < offspringMutasi; i++) {
    System.out.println("");
    for (int j = 0; j < 45; j++) {
        hasilMutasi[i][j] =
(seleksi[randomParent[i + offspringCross]][j] != null) ?
seleksi[randomParent[i + offspringCross]][j] : null;
        System.out.print(hasilMutasi[i][j] + "
");
    }
    hasilMutasi[i][0] = "M" + (i + 1) + " " +
hasilMutasi[i][0];

    System.out.println("");
    cross = rand.nextInt(39) + 1;
    temp1 = cross;
    System.out.println("cross " + (cross - 2));
    cross = rand.nextInt(39) + 1;
    System.out.println("cross " + (cross - 2));
    temp2 = hasilMutasi[i][cross];
    hasilMutasi[i][cross] = hasilMutasi[i][(int)
temp1];
    hasilMutasi[i][(int) temp1] = temp2;
    System.out.println(temp2 + " " +
hasilMutasi[i][cross]);
    for (int j = 0; j < 45; j++) {
        System.out.print(((hasilMutasi[i][j] !=
null) ? hasilMutasi[i][j] : null) + " ");
    }
    System.out.println("");
}
System.arraycopy(hasilCross, 0, crossoverArray,
0, offspringCross);
```

```

        System.arraycopy(hasilMutasi, 0, crossoverArray,
        offspringCross, offspringMutasi);
        return crossoverArray;
    }

```

Source code 5.3 Proses crossover dan mutasi

5.2.4 Proses Evaluasi

Proses evaluasi pada implementasi program ini hanya memanggil proses perhitungan harga dan fitness yang terdapat pada metode `hitungJumlahNutrisi()`.

```

Object[][] evaluasi(Object[][] parent, Object[][]
child) {

    Object[][] evaluasi = new Object[populasi +
offspringCross + offspringMutasi][45];
    for (int i = 0; i < populasi; i++) {
        System.arraycopy(parent[i], 0, evaluasi[i],
0, 45);
    }
    for (int i = populasi; i < offspringCross +
offspringMutasi + populasi; i++) {
        System.arraycopy(child[i - populasi], 0,
evaluasi[i], 0, 45);
    }
    return evaluasi;
}

```

Source code 5.4 Proses evaluasi

5.2.5 Proses Seleksi *Elitis*

Pada proses seleksi, metode yang digunakan adalah *elitism*. Proses seleksi *elitism* dalam implementasi program menggunakan *bubble sort*. Source code dari proses seleksi *elitism* ditunjukkan pada Source code 5.5.

```

Object[][] seleksi(Object[][] evaluasi) {
    Object[][] seleksiT = BubbleSort(evaluasi);
}

```

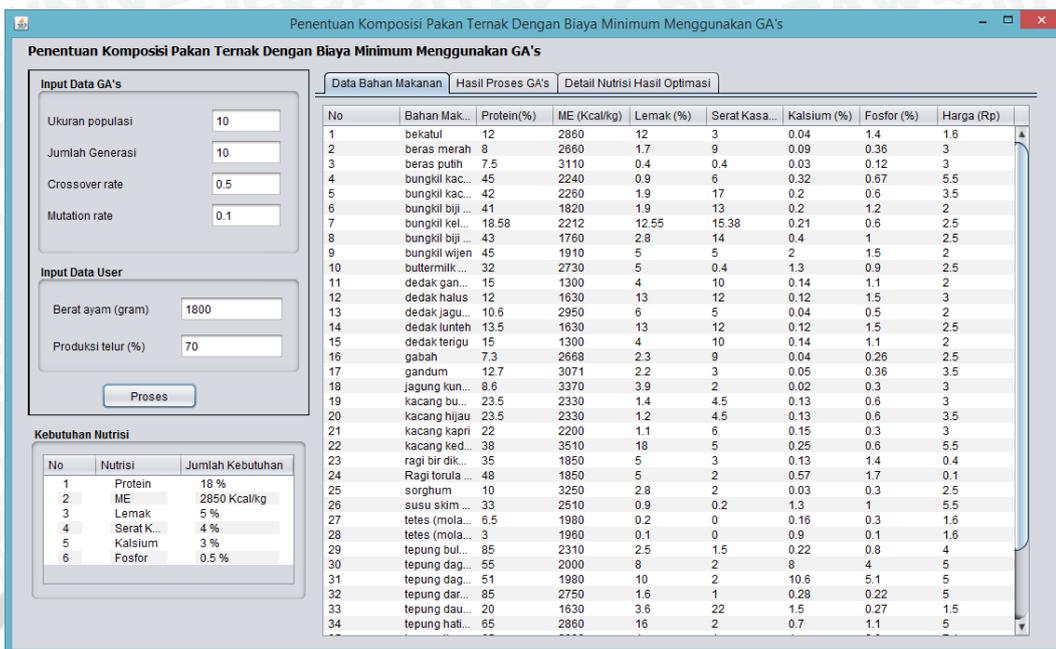
```
Object[][] seleksi = new
Object[populasi][45];
System.out.println("pop " + populasi);
System.arraycopy(seleksiT, 0, seleksi, 0,
populasi);
return seleksi;
}
```

Source code 5.5 Proses seleksi

5.3 Implementasi User Interface

Halaman utama pada program aplikasi optimasi pakan ternak terdiri dari dua sisi. Sisi sebelah kiri merupakan ruang untuk menginputkan data yang dibutuhkan oleh system yaitu data parameter algoritma genetika (ukuran populasi, jumlah generasi, *crossover rate*, *mutation rate*) dan data user (bobot ayam dan prosentase produksi telur). Di bawah panel input juga terdapat informasi kebutuhan nutrisi standart yang dibutuhkan oleh ayam yang diuji.

Pada sisi sebelah kanan terdapat tiga *tab*. *Tab* yang pertama merupakan halaman data bahan makanan. Halaman data bahan makanan memuat informasi tentang kandungan nutrisi pada bahan makanan, yang terdiri dari protein, ME, lemak, serat kasar, kalsium, dan fosfor. Pada halaman ini juga disertakan informasi harga pada masing-masing bahan makanan, seperti yang ditunjukkan pada Gambar 5.1.



Gambar 5.1 Implementasi halaman utama

Tab yang kedua berisikan hasil dari proses algoritma genetika. Pada halaman ini terdapat hasil optimasi yang berisikan detail hasil seleksi yang terdiri dari generasi ke-, nama kromosom, kromosom, penalty, harga, dan *fitness*. Selain itu juga ditampilkan hasil seleksi dari proses perhitungan menggunakan algoritma genetika. Gambar 5.2 merupakan hasil optimasi dari proses algoritma genetika.

Generasi ke-	Nama	Kromosom	Penalti	Harga	Fitness
0	C2 P9	6 9 4 2 9 1 1 2 3 6 4 8 ...	490.22	323.10	1.23
1	C4 C2 P9	6 9 4 2 9 1 1 2 3 6 4 8 ...	255.94	347.28	1.66
2	C4 C2 P9	6 9 4 2 9 1 1 2 3 6 4 8 ...	255.94	347.28	1.66
3	C4 C2 P9	6 9 4 2 9 1 1 2 3 6 4 8 ...	255.94	347.28	1.66
4	M1 C4 C4 C2 P9	6 9 4 2 9 1 1 2 3 6 4 8 ...	232.71	299.44	1.88
5	M1 C4 C4 C2 P9	6 9 4 2 9 1 1 2 3 6 4 8 ...	232.71	299.44	1.88
6	M1 C2 M1 C4 C4 C2 P9	6 9 4 2 1 0 1 1 2 3 6 4 ...	222.23	300.86	1.91
7	M1 M1 C4 C4 C2 P9	10 9 4 2 9 1 1 2 3 6 4 ...	188.94	282.22	2.12
8	M1 M1 C4 C4 C2 P9	10 9 4 2 9 1 1 2 3 6 4 ...	188.94	282.22	2.12
9	M1 M1 C4 C4 C2 P9	10 9 4 2 9 1 1 2 3 6 4 ...	188.94	282.22	2.12

Gambar 5.2 Implementasi hasil proses GA's

Tab yang ketiga berisikan halaman yang menunjukkan detail dari kromosom yang terpilih untuk generasi selanjutnya. Diantaranya adalah nama bahan makanan, kadar protein, ME, lemak, serat kasar, kalsium, dan fosfor, serta harga dari bahan makanan terpilih.

Bahan Makanan	Protein (%)	ME (Kcal/...	Lemak (%)	Serat Ka...	Kalsium ...	Fosfor (%)	Berat (gr...	Harga
dekatal	2.11	501.75	2.11	0.53	0.01	0.25	19.13	30.61
beras merah	1.26	420.00	0.27	1.42	0.01	0.06	17.22	51.66
bungkil kacang tanah	6.63	356.84	0.30	2.98	0.03	0.09	17.22	60.27
dedak jagung	1.67	465.79	0.95	0.79	0.01	0.08	17.22	34.44
jagung kuning	1.51	591.23	0.68	0.35	0.00	0.05	19.13	57.40
sorghum	1.75	570.18	0.49	0.35	0.01	0.05	19.13	47.83
Total	14.93684...	2905.78...	4.796491...	6.12280...	0.067894...	0.58140...	109.062	282.221...

Gambar 5.3 Implementasi detail hasil optimasi

BAB VI PENGUJIAN DAN ANALISA

6.1 Pengujian dan Analisa Ukuran Populasi

Pengujian ukuran populasi dilakukan untuk mengetahui ukuran populasi yang optimal pada permasalahan optimasi biaya kebutuhan nutrisi pakan ayam petelur. Data yang digunakan pada pengujian adalah 40 data bahan makanan, dan data ayam petelur yang digunakan adalah :

- a. Berat ayam = 1800 gram
- b. Produksi telur = 70 %

Sedangkan parameter algoritma genetika yang digunakan pada uji coba populasi adalah sebagai berikut :

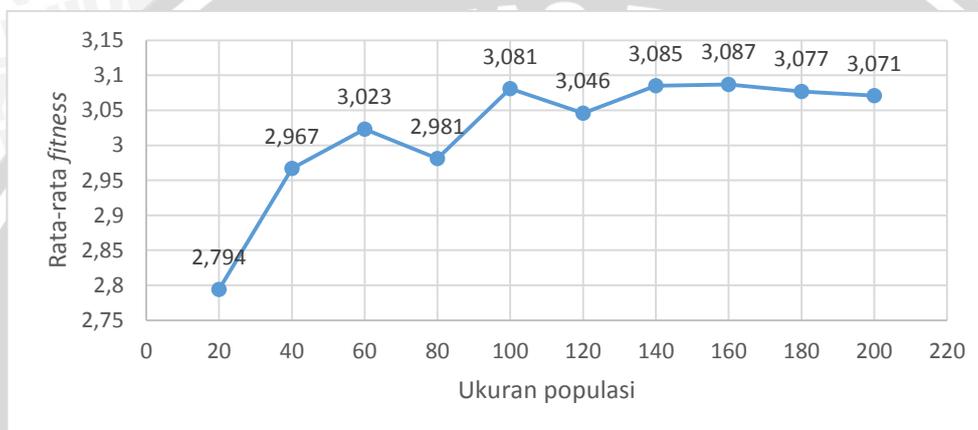
- a. Ukuran populasi = kelipatan 20 sampai 200
- b. Ukuran generasi = 250 generasi
- c. *Crossover rate* (cr) = 0,5
- d. *Mutation rate* (mr) = 0,1

Pengujian dilakukan sebanyak 10 kali. Hasil dari kesepuluh percobaan yang telah dilakukan nantinya akan dihitung rata-rata nilai *fitness*nya untuk mengetahui populasi yang optimal dari masing-masing populasi. Tabel 6.1 berikut merupakan hasil dari uji coba ukuran populasi.

Tabel 6.1 Hasil uji coba ukuran populasi

PopSize	Percobaan Ke-										Rata-rata <i>fitness</i>	Rata-rata waktu (s)
	1	2	3	4	5	6	7	8	9	10		
20	2.52	2.79	3.07	2.62	2.61	2.52	3.07	2.66	2.92	3.16	2.794	2
40	2.86	2.82	2.94	3.09	2.9	3.03	3.07	2.95	2.83	3.18	2.967	4
60	3.01	3.18	3.29	2.42	3.25	3.15	3.01	3.05	3.03	2.84	3.023	6
80	3.04	2.78	3.1	3.09	2.98	2.91	2.93	3.15	2.73	3.1	2.981	8

100	3.18	3.02	2.77	3.09	3.2	3.29	3.01	2.93	3.13	3.19	3.081	12
120	3.06	3.04	3.06	3.13	2.51	3.28	2.78	3.21	3.19	3.2	3.046	18
140	3.03	3.04	3.23	2.91	3.15	3.23	2.94	3.2	2.97	3.15	3.085	22
160	3.18	2.94	3.03	3.09	3.15	3.12	3.16	2.97	3.2	3.03	3.087	23
180	3.1	3.21	3.04	3.23	3.03	2.89	3.14	2.94	3.1	3.09	3.077	26
200	3.2	2.95	3.1	2.88	2.93	3.34	2.97	3.23	3.18	2.93	3.071	34



Gambar 6.1 Grafik uji coba ukuran populasi

Dari grafik yang ditunjukkan pada Gambar 6.1 dapat disimpulkan bahwa nilai rata-rata *fitness* terbaik 10 kali percobaan dari ukuran populasi 20 hingga ukuran populasi 200 adalah pada ukuran populasi ke-160 dengan nilai *fitness* sebesar 3,087. Dengan hasil ini, dapat dikatakan bahwa ukuran populasi 160 merupakan ukuran populasi yang optimal untuk menyelesaikan permasalahan optimasi biaya pada pemenuhan nutrisi ayam petelur. Setelah ukuran populasi 140 tidak ada kenaikan rata-rata *fitness* yang signifikan lagi dan bisa dikatakan konvergen.

6.2 Pengujian dan Analisa Banyak Generasi

Pengujian banyak generasi dilakukan untuk mengetahui ukuran generasi yang optimal pada permasalahan optimasi biaya kebutuhan nutrisi pakan ayam

petelur. Data yang digunakan pada pengujian adalah 40 data bahan makanan, dan data ayam petelur yang digunakan adalah :

- a. Berat ayam = 1800 gram
- b. Produksi telur = 70 %

Parameter yang digunakan pada uji coba populasi adalah sebagai berikut :

- a. Banyak generasi = kelipatan 250 sampai 1500
- b. Ukuran populasi = 160
- c. *Crossover rate* (cr) = 0,5
- d. *Mutation rate* (mr) = 0,1

Pengujian dilakukan sebanyak 10 kali. Hasil dari kesepuluh percobaan yang telah dilakukan nantinya akan dihitung rata-rata nilai *fitness*nya untuk mengetahui generasi yang optimal dari masing-masing generasi. Tabel 6.2 berikut ini merupakan hasil dari uji coba banyak generasi.

Tabel 6.2 Hasil uji coba banyak generasi

Banyak generasi	Percobaan Ke-										Rata-rata <i>fitness</i>	Rata-rata waktu (s)
	1	2	3	4	5	6	7	8	9	10		
250	3.13	3.09	3.21	3.14	3.03	3.26	3.14	3.22	2.95	3.03	3.12	33
500	3.06	3.09	3.06	3.04	3.1	3.28	3.22	3.23	3.27	3.07	3.142	52
750	3.22	3.29	3.23	3.04	2.99	3.39	2.92	3.29	2.89	2.94	3.12	64
1000	3.24	2.97	3.03	3.24	2.91	3.38	2.9	3.25	3.09	3.04	3.105	85
1250	3.26	3.09	2.89	3.24	3.31	3.15	3.15	2.96	2.95	3.15	3.115	104
1500	2.95	2.98	3.1	3.21	3.25	2.94	3.17	3.23	3.09	3.22	3.114	119
1750	3.05	3.03	3.16	3.17	3.1	3.24	3.28	3.04	2.93	3.04	3.104	133
2000	3.23	3.09	3.1	3.13	3.23	3.13	2.71	3.2	3.23	2.71	3.076	149
2250	3.28	2.94	3.09	3.04	3.23	3.25	2.78	3.23	2.92	3.09	3.085	165
2500	3.23	2.95	3.23	3.29	3.06	2.92	3.06	3.03	3.09	3.23	3.109	178
2750	3.07	2.94	3.28	3.15	3.08	2.93	3.14	2.95	3.21	3.04	3.079	201



Gambar 6.2 Grafik uji coba banyak generasi

Dari grafik yang ditunjukkan pada Gambar 6.2 dapat disimpulkan bahwa nilai rata-rata *fitness* terbaik untuk 10 kali percobaan dari generasi ke 250 hingga generasi ke 2250 adalah pada generasi ke 500 dengan nilai rata-rata *fitness* sebesar 3,142. Dengan hasil ini, dapat dikatakan bahwa pada generasi ke 500 merupakan generasi yang optimal untuk menyelesaikan permasalahan optimasi biaya pada pemenuhan nutrisi ayam petelur. Setelah generasi ke 750 tidak ada kenaikan rata-rata *fitness* yang signifikan lagi bahkan nilai rata-rata *fitness* cenderung turun sehingga bisa dikatakan sudah konvergen.

6.3 Pengujian dan Analisa Kombinasi *Crossover Rate* dan *Mutation Rate*

Pengujian kombinasi *crossover rate* dan *mutation rate* dilakukan untuk mengetahui kombinasi *crossover rate* dan *mutation rate* yang optimal pada permasalahan optimasi biaya kebutuhan nutrisi pakan ayam petelur. Uji coba kombinasi *crossover rate* (cr) dan *mutation rate* (mr) akan dilakukan dengan memberikan nilai-nilai yang berbeda pada range 0 hingga 1. Data yang digunakan pada pengujian adalah 40 data bahan makanan, dan data ayam petelur yang digunakan adalah :

- a. Berat ayam = 1800 gram
- b. Produksi telur = 70 %

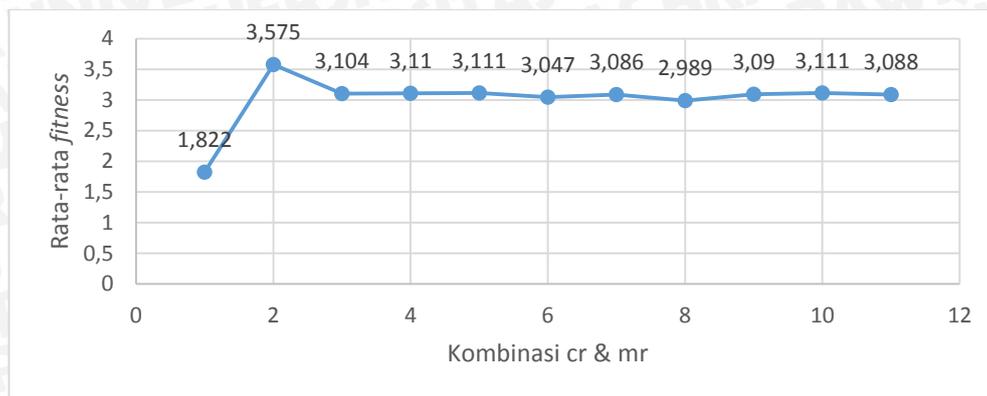
Sedangkan ukuran populasi dan banyak generasi yang digunakan dalam pengujian ini adalah ukuran populasi terbaik dan banyak generasi terbaik yang didapatkan dari hasil pengujian yang telah dilaksanakan pada subbab sebelumnya, yaitu :

- a. Ukuran populasi = 160
- b. Banyak generasi = 500

Pengujian dilakukan sebanyak 10 kali. Hasil dari kesepuluh percobaan yang telah dilakukan nantinya akan dihitung rata-rata nilai *fitness*nya untuk mengetahui kombinasi *crossover rate* dan *mutation rate* yang optimal dari masing-masing kombinasi. Tabel 6.3 berikut merupakan hasil dari uji coba kombinasi *crossover rate* dan *mutation rate*.

Tabel 6.3 Hasil uji coba kombinasi *crossover rate* dan *mutation rate*

Kombinasi		Nilai fitness										Rata-rata fitness
		Percobaan kombinasi <i>Cr</i> dan <i>Mr</i> Ke-										
<i>Cr</i>	<i>Mr</i>	1	2	3	4	5	6	7	8	9	10	
0	1	2.13	3.07	2.94	3.14	3.2	2.39	2.96	3.1	3.15	3.23	2.931
1	0	2.32	2.12	0.92	2.61	1.42	1.57	1.36	2.2	1.89	1.81	1.822
0.9	0.1	3.1	3.29	7.12	3.06	3.23	3.04	3.17	3.21	3.36	3.17	3.575
0.8	0.2	3.15	3.23	3.29	3.09	3.13	2.88	2.92	3.03	3.23	3.09	3.104
0.7	0.3	3.13	2.96	3.23	3.09	3.28	3.2	3.03	3.01	3.04	3.13	3.11
0.6	0.4	3.36	3.2	3.06	2.51	3.18	3.28	3.03	3.01	3.25	3.23	3.111
0.5	0.5	2.93	2.88	3.22	3.1	3.16	3.15	3.13	3.09	2.85	2.96	3.047
0.4	0.6	2.82	3.21	3.28	3.1	3.2	3.03	3.1	3.04	2.92	3.16	3.086
0.3	0.7	3.23	2.78	3.13	3.39	2.58	2.9	2.93	3.04	3.09	2.82	2.989
0.2	0.8	3.26	3.15	3.22	3.29	2.96	3.2	2.9	2.91	2.92	3.09	3.09
0.1	0.9	3.38	3.14	3.15	3.21	3.09	3.23	2.85	3.14	2.99	2.93	3.111
0	1	2.89	3.18	3.15	2.9	3.06	2.82	3.23	3.22	3.21	3.22	3.088



Gambar 6.3 Grafik uji coba kombinasi *crossover* dan mutasi

Dari grafik yang ditunjukkan pada Gambar 6.3 dapat disimpulkan bahwa kenaikan rata-rata *fitness* terbaik untuk 10 kali percobaan dari kombinasi *crossover rate* dan *mutation rate* adalah pada *crossover rate* sebesar 0,9 dan *mutation rate* sebesar 0,1 dengan nilai rata-rata *fitness* sebesar 3,575. Dengan hasil ini, dapat dikatakan bahwa pada *crossover rate* sebesar 0,9 dan *mutation rate* sebesar 0,1 merupakan kombinasi yang optimal untuk menyelesaikan permasalahan optimasi biaya pada pemenuhan nutrisi ayam petelur. Pada kombinasi *crossover rate* sebesar 0,8 dan *mutation rate* sebesar 0,2 nilai rata-rata *fitness*nya tidak mengalami kenaikan yang signifikan sehingga percobaan bisa dikatakan konvergen.

6.4 Analisa Perbandingan Kebutuhan Nutrisi Harian Ayam Petelur dengan Nutrisi yang Dihasilkan Kromosom Terbaik dari Hasil Optimasi

Perbandingan kebutuhan nutrisi dilakukan untuk mengetahui kecukupan nutrisi dari kromosom yang dihasilkan oleh proses algoritma genetika. Untuk mengetahuinya dilakukan pengujian pada aplikasi optimasi pakan ayam petelur dengan menggunakan parameter optimal yang didapatkan dari pengujian sebelumnya yaitu ukuran populasi sebesar 160, banyak generasi sebesar 500 generasi, dan kombinasi *cr* dan *mr* sebesar 0,9 dan 0,1. Dari hasil pengujian didapatkan kromosom terbaik dengan kombinasi bahan pakan yang paling optimal. Solusi yang dihasilkan ini memiliki nilai penalti dan harga yang terkecil bila dibandingkan dengan kromosom lainnya. Namun demikian, masih terdapat penalti pada setiap kromosom yang terpilih sebagai solusi. Dengan terdapatnya penalti

pada kromosom hasil seleksi ini menunjukkan bahwa pada kombinasi bahan pakan yang terpilih masih belum dapat memenuhi kebutuhan nutrisi harian ayam petelur. Hal ini bisa disebabkan oleh bobot bahan pakan yang diberikan dan juga kandungan nutrisi pada masing-masing bahan pakan yang digunakan memiliki nilai yang relatif kecil, sedangkan kebutuhan nutrisi harian yang dibutuhkan oleh ayam sangat tinggi mengingat bahwa ayam pada fase *layer* memiliki kebutuhan nutrisi tertinggi daripada fase-fase lainnya. Selain itu rumus *fitness* yang digunakan juga mempengaruhi hasil dari nutrisi yang ada pada kromosom hasil proses algoritma genetika.



BAB VII

KESIMPULAN DAN SARAN

7.1 Kesimpulan

Dari hasil uji coba yang telah dilakukan dalam optimasi pakan ayam petelur, terdapat beberapa kesimpulan yaitu :

1. Bentuk representasi kromosom yang efisien memiliki panjang kromosom sepanjang jumlah bahan pakan yang digunakan yaitu sebanyak 40. Dan setiap gen diisi dengan nilai bilangan pecahan (*real*) dengan interval $[1 \dots 10]$ yang mewakili bobot bahan pakan.
2. Berdasarkan uji coba parameter algoritma genetika dapat disimpulkan bahwa seluruh parameter sangat berpengaruh terhadap solusi yang akan dihasilkan. Pada hasil uji coba, parameter terbaik dengan rata – rata nilai *fitness* tertinggi adalah sebagai berikut :
 - a. Ukuran populasi : 160
 - b. Banyak generasi : 500
 - c. *Crossover Rate* : 0.9
 - d. *Mutation rate* : 0.1
3. Untuk mengukur tingkat kualitas solusi dari permasalahan optimasi pakan ternak ini menggunakan perhitungan nilai *fitness* yang diperoleh dari nilai penalti nutrisi dan harga bahan pakan yang digunakan. Namun pada solusi yang didapatkan masih terdapat penalti nutrisi, yang artinya bahwa solusi yang dihasilkan belum memenuhi kebutuhan nutrisi harian ayam petelur.

7.2 Saran

Pengembangan penelitian ini dapat dilakukan untuk menyelesaikan masalah optimasi pakan ternak dengan menggunakan metode *crossover*, mutasi dan seleksi yang berbeda sehingga mempengaruhi nilai *fitness* suatu individu. Selain itu, perbaikan solusi juga bisa dilakukan dengan melakukan hibridasi algoritma genetika dengan algoritma lainnya untuk mendapatkan hasil yang lebih optimal. Serta pada penelitian selanjutnya dapat dilakukan perubahan rumus pada bobot

bahan pakan harian yang harus diberikan kepada ayam dengan menghitung 100 ekor ayam sekaligus untuk mendapatkan bobot bahan pakan yang lebih besar agar terhindar dari penalti. Dengan standar bobot pakan yang tinggi maka kemungkinan sebuah kromosom untuk terkena penalti akan semakin kecil.



DAFTAR PUSTAKA

- Abun. 2008. *Nutrisi Mineral pada Unggas*. Fakultas Peternakan. Universitas Padjadjaran : Jatinangor.
- Anggorodi. 1985. *Kemajuan Mutakhir Dalam Ilmu Makanan Ternak Unggas*. UI-Press : Jakarta.
- Amrullah, I. K. 2003. *Nutrisi Ayam Petelur. Seri Beternak Mandiri*. Lembaga Satu Gunungbudi : Bogor
- Aribowo, A, Lukas, S, Gunawan, M. 2008. *Penerapan Algoritma Genetika Pada Penentuan Komposisi Pakan Ayam Petelur*. Seminar Nasional Aplikasi Teknologi Informasi 2008 (SNATI 2008). Universitas Pelita Harapan : Tangerang.
- Banong, S. 2012. *Manajemen Industri Ayam Ras Petelur*. Masagena Press : Makassar.
- Cahyono, B. 1995. *Ayam Petelur (Gallus sp)*. Pustaka Nusatama : Yogyakarta.
- Davis, L. 1991. *Handbook of Genetic Algorithms*. Van Nostrand Reinhold : New York.
- Desiani, Anita dan Arhami, Muhammad. 2007. *Konsep Kecerdasan Buatan*. Andi : Yogyakarta.
- Goldberg, D. E. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Massachusetts: AddisonWesley Publishing Company, Inc.
- Homayouni, SM, Tang, SH, dan Motlagh, Omid. 2013. *A Genetic Algorithm for Optimization of Integrated Scheduling of Cranes, Vehicles, And Storage Platforms At Automated Container Terminals*. *Journal of Computational and Applied Mathematics*, Vol 270, PP. 545-556.
- Kusumadewi, S. 2003. *Artificial Intelligence (Teknik dan Aplikasinya)*. Graha Ilmu : Yogyakarta.
- Mahmudy, Wayan Firdaus. 2014. *Algoritma Evolusi*. Program Teknologi Informasi dan Ilmu Komputer. Universitas Brawijaya : Malang.

Michalewics, Z. 1996. *Genetic Algorithm + Data Structure = Evolution Programs*. Springer-Verlag 3rd Edition : New York.

North, M. O. 1984. *Commercial Chicken Production Manual*. Connecticut : AVI Publication Co.

Nugroho, Adi. 2014. *Budidaya Ayam Petelur*. PT Charoen Pokhpand Indonesia : Jombang.

Pratiwi, MI, Mahmudy, WF & Dewi, C 2014, *Implementasi Algoritma Genetika Pada Optimasi Biaya Pemenuhan Kebutuhan Gizi*, *DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya*, vol. 4, no. 6.

Rasyaf, Muhammad. 2003. *Beternak Ayam Petelur*. Penebar Swadaya : Jakarta.

Sari, AP, Mahmudy, WF & Dewi, C 2014, *Optimasi asupan gizi pada ibu hamil dengan menggunakan algoritma genetika*, *DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya*, vol. 4, no. 5.

Sudarmono, AS. 2003. *Pedoman Pemeliharaan Ayam Ras Petelur*. Kanisius : Yogyakarta.

Zakariah, Askari. 2011. *Pengaruh Penggunaan Serat Terhadap Kadar Kolesterol Unggas*. Pascasarjana Fakultas Peternakan. Universitas Gadjah Mada : Yogyakarta.

LAMPIRAN

Lampiran 1. Daftar nutrisi bahan makanan

No	Bahan	Protein (%)	ME (kkal/kg)	Lemak (%)	Serat kasar (%)	Kalsium (%)	Fosfor (%)
1	Bekatul	12	2860	12	3	0.04	1.4
2	Beras merah	8	2660	1.7	9	0.09	0.36
3	Beras putih	7.5	3110	0.4	0.4	0.03	0.12
4	Bungkil kacang kedele	45	2240	0.9	6	0.32	0.67
5	Bungkil kacang tanah	42	2260	1.9	17	0.2	0.6
6	Bungkil biji kapas	41	1820	1.9	13	0.2	1.2
7	Bungkil kelapa	21	1540	1.8	15	0.2	0.6
8	Bungkil biji bunga matahari	43	1760	2.8	14	0.4	1
9	Bungkil wijen	45	1910	5	5	2	1.5
10	<i>Buttermilk</i> dikeringkan	32	2730	5	0.4	1.3	0.9
11	Dedak gandum	15	1300	4	10	0.14	1.1
12	Dedak halus	12	1630	13	12	0.12	1.5

13	Dedak jagung	10.6	2950	6	5	0.04	0.5
14	Dedak lunteh	13.5	1630	13	12	0.12	1.5
15	Dedak terigu	15	1300	4	10	0.14	1.1
16	Gabah	7.3	2668	2.3	9	0.04	0.26
17	Gandum	12.7	3071	2.2	3	0.05	0.36
18	Jagung kuning	8.6	3370	3.9	2	0.02	0.3
19	Kacang buncis	23.5	2330	1.4	4.5	0.13	0.6
20	Kacang hijau	23.5	2330	1.2	4.5	0.13	0.6
21	Kacang kapri	22	2200	1.1	6	0.15	0.3
22	Kacang kedele	38	3510	18	5	0.25	0.6
23	Ragi bir dikeringkan	35	1850	5	3	0.13	1.4
24	Ragi torula dikeringkan	48	1850	5	2	0.57	1.7
25	Sorghum	10	3250	2.8	2	0.03	0.3
26	Susu skim dikeringkan	33	2510	0.9	0.2	1.3	1
27	Tetes (Molase dari beet)	6.5	1980	0.2	0	0.16	0.3
28	Tetes (Molase dari tebu)	3	1960	0.1	0	0.9	0.1

29	Tepung bulu ayam	85	2310	2.5	1.5	0.22	0.8
30	Tepung daging	55	2000	8	2	8	4
31	Tepung daging dan tulang	51	1980	10	2	10.6	5.1
32	Tepung darah	85	2750	1.6	1	0.28	0.22
33	Tepung daun alfalfa	20	1630	3.6	22	1.5	0.27
34	Tepung hati dan kelenjar	65	2860	16	2	0.7	1.1
35	Tepung ikan ancovetta	65	2830	4	1	4	2.6
36	Tepung ikan herring	72	3190	10	1	2	1.5
37	Tepung ikan menhaden	61	3080	9	1	5.5	2.8
38	Tepung kepiting	31	1870	2	11	15	1.6
39	Tepung tapioka	1.8	3720	1.3	1.8	0.3	0.35
40	Whey dikeringkan	13	1910	0.8	0	0.9	0.8

Sumber : Scott, 1976.

Lampiran 2. Daftar Harga Bahan Pakan

No	Bahan	Harga/gram (rupiah)
1	Bekatul	1.6
2	Beras merah	3
3	Beras putih	3
4	Bungkil kacang kedele	5.5
5	Bungkil kacang tanah	3.5
6	Bungkil biji kapas	2
7	Bungkil kelapa (indonesia)	2.5
8	Bungkil biji bunga matahari	2.5
9	Bungkil wijen	2
10	<i>Buttermilk</i> dikeringkan	2.5
11	Dedak gandum	2
12	Dedak halus	3
13	Dedak jagung	2
14	Dedak lunteh	2.5
15	Dedak terigu	2
16	Gabah	2.5
17	Gandum	3.5
18	Jagung kuning	3
19	Kacang buncis	3
20	Kacang hijau	3.5
21	Kacang kapri	3
22	Kacang kedele	5.5
23	Ragi bir dikeringkan	0.4
24	Ragi torula dikeringkan	0.1
25	Sorghum	2.5
26	Susu skim dikeringkan	5
27	Tetes (Molase dari beet)	1.6
28	Tetes (Molase dari tebu)	1.6

29	Tepung bulu ayam	4
30	Tepung daging	5
31	Tepung daging dan tulang	5
32	Tepung darah	5
33	Tepung daun alfalfa	1.5
34	Tepung hati dan kelenjar	5
35	Tepung ikan ancovetta	7.1
36	Tepung ikan herring	7.7
37	Tepung ikan menhaden	6.8
38	Tepung kepiting	5
39	Tepung tapioka	2.3
40	Whey dikeringkan	5

