

**PENERAPAN ALGORITMA GENETIKA PADA PENENTUAN
KOMPOSISI PAKAN IKAN**

SKRIPSI

Oleh :

**LAMHOT PRANATA S
(0610960038-96)**



**PROGRAM STUDI ILMU KOMPUTER
JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2011**

UNIVERSITAS BRAWIJAYA



LEMBAR PENGESAHAN

PENERAPAN ALGORITMA GENETIKA PADA PENENTUAN KOMPOSISI PAKAN IKAN

Oleh :

LAMHOT PRANATA S
(0610960038-96)

Setelah dipertahankan di depan Majelis Penguji

Pada tanggal 31 Januari 2011

Dan dinyatakan memenuhi syarat untuk memperoleh gelar
Sarjana Komputer dalam bidang Ilmu Komputer

Pembimbing I

Pembimbing II

Drs. Muh.Arif Rahman, M.Kom
NIP : 19660423 199111 1 001

Candra Dewi, S.Kom., M.Sc
NIP : 19771114 200312 2 001

Mengetahui,
Ketua Jurusan Matematika
Fakultas MIPA Universitas Brawijaya

Dr. Abdul Rouf Alghofari, MSc
NIP: 19670907 199203 1 001

UNIVERSITAS BRAWIJAYA



LEMBAR PERNYATAAN

Saya yang bertandatangan di bawah ini :

Nama

: Lamhot Pranata

NIM

: 0610960038

Jurusan

: Matematika

Penulis Tugas Akhir Berjudul : Penerapan Algoritma Genetika
Pada Penentuan Pakan Ikan

Dengan ini menyatakan bahwa :

1. Isi dari skripsi yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain. Selain nama-nama yang termaktub di isi dan tertulis di daftar pustaka di skripsi ini.
2. Apabila di kemudian hari ternyata skripsi yang saya tulis terbukti hasil jiplakan, maka saya akan bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan segala kesadaran

Penulis,

Lamhot Pranata
NIM : 0610960038

UNIVERSITAS BRAWIJAYA



PENERAPAN ALGORITMA GENETIKA PADA PENENTUAN KOMPOSISI PAKAN IKAN

ABSTRAK

Ada banyak kemungkinan kombinasi pakan yang dapat digunakan untuk meningkatkan produksi dan kualitas ikan. Namun demikian, tidak semua kombinasi pakan akan memberikan solusi terbaik. Pada skripsi ini akan dilakukan penelitian tentang penentuan pakan ikan menggunakan algoritma genetika.

Algoritma genetika yang memiliki kehandalan dalam menghasilkan output yang optimal dapat dimanfaatkan untuk memecahkan masalah tersebut. Proses-proses yang terlibat dalam algoritma genetika adalah *crossover*, mutasi dan seleksi. *Crossover* dan mutasi merupakan proses untuk membentuk kromosom baru, sedangkan seleksi merupakan proses pemilihan kromosom yang akan digunakan pada populasi baru. Dalam kasus ini terdapat beberapa input yang dibutuhkan, yaitu jumlah bahan yang dikomposisikan, berat pakan yang ingin dihasilkan, jumlah populasi awal, jumlah generasi, probabilitas crossover dan probabilitas mutasi. Hasil pemrosesan merupakan kombinasi pakan yang merepresentasikan penyelesaian masalah ini. Hanya kromosom terbaik yang akan diberikan sebagai hasil.

Penelitian yang telah dilakukan, digunakan 34 jenis bahan pakan yang digunakan pada pengujian. Dari jenis bahan pakan tersebut, dipilih 10 jenis bahan pakan. Melalui pengujian yang telah dilakukan dengan parameter algoritma genetika yang bervariasi, hasil yang optimum diperoleh pada pengujian dengan jumlah bahan 10, jumlah kromosom 15, probabilitas crossover 90%, probabilitas mutasi 90%, dan jumlah generasi 1000 dengan nilai *fitness* 0.091491807.

Kata kunci: Algoritma Genetika, Optimisasi, Komposisi, Bahan Pakan.

UNIVERSITAS BRAWIJAYA



PENERAPAN ALGORITMA GENETIKA PADA PENENTUAN KOMPOSISI PAKAN IKAN

ABSTRACT

There are many possible combinations of feed that can be used to improve production and quality of fish. However, not all combinations of feed will provide the best solution. In this thesis research will be done on the determination of fish feed using genetic algorithms.

Genetic algorithms have the reliability in producing the optimal output can be utilized to solve the problem. The processes involved in the genetic algorithm are crossover, mutation and selection. Crossover and mutation is a process to form a new chromosome, while selection is the process of selecting chromosomes that will be on the new population. In this case there are several inputs that are needed, namely the amount of ingredient composed, the weight of feed to generate, the number of initial population, number of generations, crossover probability and mutation probability. The processing result is a combination of feed that represents the solution of this problem. Only the best chromosome will be given as a result.

The research that has been done, used 34 types of feed ingredients to be used in testing. From the type of feed ingredients, the selected 10 types of feed ingredients. Through testing has been done with varying parameter genetic algorithm, optimum results obtained in testing with the amount of ingredients 10, the number of chromosomes 15, crossover probability 90%, mutation probability 90%, and the total generation 1000 with fitness value 0.091491807.

Keywords: Genetic Algorithm, Optimization, Composition, Feed Ingredients.

UNIVERSITAS BRAWIJAYA



KATA PENGANTAR

Puji Syukur kehadirat Tuhan Yesus Kristus, karena berkat kasih dan karunia-Nyalah penulis dapat menyelesaikan skripsi yang berjudul **“Penerapan Algoritma Genetika Pada Penentuan Komposisi Pakan Ikan”**. Terselesaikannya laporan skripsi tentu tidak lepas dari bantuan dari berbagai pihak. Untuk itu penulis ingin menyampaikan ucapan terima kasih kepada :

1. Drs. Muh. Arif Rahman, M.Kom, selaku pembimbing I dari Program Studi Ilmu Komputer, Jurusan Matematika, Fakultas MIPA Universitas Brawijaya.
2. Candra Dewi, S. Kom, M.Sc selaku pembimbing II dari Program Studi Ilmu Komputer, Jurusan Matematika, Fakultas MIPA Universitas Brawijaya.
3. Dr. Abdul Rouf Alghofari, MSc selaku Ketua Jurusan Matematika Fakultas MIPA Universitas Brawijaya.
4. Drs. Marji, MT, selaku Ketua Program Studi Ilmu Komputer, Jurusan Matematika, Fakultas MIPA Universitas Brawijaya.
5. Bondan Sapto Prakoso, S.T selaku dosen pembimbing akademik.
6. Seluruh Bapak dan Ibu Dosen Program Studi Ilmu Komputer, Jurusan Matematika, Fakultas MIPA Universitas Brawijaya, terima kasih atas bimbingan dan pengajaran yang diberikan selama ini.
7. Papa, Mama, kakak dan kedua adik tersayang, terima kasih atas dukungan, perhatian dan kasih sayang yang tak terhingga.
8. Yunita Rosalyn yang telah senantiasa memberikan semangat, doa, serta dukungannya.
9. Teman-teman ILKOM seperjuangan Lupi, Suryana, Yoga, Lia, David dan semua teman-teman angkatan 2006 yang tidak bisa disebutkan satu persatu. Terima kasih atas dukungan, kerjasama dan persahabatan kalian semua.
10. Sahabat-sahabat PMK, Sandi, Ari, Vegi, Ricky, Eliezer, Yere dan teman-teman PMK Phils yang tidak bisa disebutkan satu persatu, terima kasih atas dukungan, semangat dan persahabatan kalian selama ini.
11. Dan semua pihak yang telah membantu dalam penyelesaian skripsi ini.

Semoga Tuhan Yesus Kristus senantiasa memberikan Kasih dan Karunia-Nya kepada semua pihak yang telah memberikan segala bantuan tersebut. Skripsi ini tentu saja masih jauh dari sempurna, sehingga penulis dengan senang hati menerima kritik demi perbaikan. Penulis berharap semoga skripsi ini dapat memberikan manfaat bagi yang membacanya.

Malang, 23 Januari 2011

Penulis



DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	iii
LEMBAR PERNYATAAN	v
ABSTRAK	vii
ABSTRACT	ixix
KATA PENGANTAR	xi
DAFTAR ISI	xxiii
DAFTAR GAMBAR	Error! Bookmark not defined.ii
DAFTAR TABEL	Error! Bookmark not defined.ix
DAFTAR GRAFIK	Error! Bookmark not defined.
DAFTAR SOURCECODE	Error! Bookmark not defined.
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	3
1.5 Manfaat penelitian.....	3
1.6 Metodologi Pemecahan Masalah.....	3
1.7 Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA	5
2.1. Komposisi Pakan Ikan	5
2.2. Algoritma Genetika.....	5
2.2.1 Solusi dalam Algoritma Genetika	6
2.2.2 Struktur dalam Algoritma Genetika	7
2.2.3 Pengkodean	7
2.2.4 Mendefinisikan Individu	8
2.2.5 Mendefinisikan Nilai <i>Fitness</i>	10
2.2.6 Pembangkitan Populasi Awal	11
2.2.7 Perkawinan Silang (<i>crossover</i>)	11
2.2.8 Mutasi	12
2.2.9 Seleksi	13
BAB III METODOLOGI DAN PERANCANGAN	15
3.1. Deskripsi Sistem	15

3.2.	Langkah-Langkah Proses Penyelesaian Penentuan Komposisi Pakan Ikan Dengan Menggunakan Algoritma Genetika	15
3.2.1	Representasi Kromosom	18
3.2.2	Membangkitkan Populasi Awal	18
3.2.3	Nilai <i>Fitness</i>	19
3.2.4	<i>Crossover</i>	22
3.5.5	Mutasi.....	26
3.5.6	Seleksi	28
3.3	Perhitungan Manual	30
3.3.1	Pembangkitan Populasi Awal.....	30
3.3.2	Contoh Perhitungan Nilai <i>Fitness</i>	30
3.3.3	<i>Crossover</i>	32
3.3.4	Mutasi.....	34
3.3.5	Seleksi	35
3.4	Rancangan Uji Coba	37
BAB IV	IMPLEMENTASI DAN PEMBAHASAN	39
4.1.	Deskripsi Sistem.....	39
4.2.	Sumber Data.....	39
4.3.	Lingkungan Implementasi.....	39
4.3.1	Lingkungan Perangkat Keras	39
4.3.2	Lingkungan Perangkat Lunak	40
4.4	Implementasi Program Penentuan Komposisi Pakan Ikan Menggunakan Algoritma Genetika	40
4.4.1	Struktur Data	40
4.4.2	Inisialisasi Kromosom.....	42
4.4.3	Perhitungan <i>Fitness</i>	42
4.4.4	Proses <i>Crossover</i>	43
4.4.4.1	Prosedur Pilih Parent <i>Crossover</i>	44
4.4.4.2	Prosedur Discrete <i>Crossover</i>	44
4.4.5	Proses Mutasi	46
4.4.5.1	Prosedur Pilih Induk Mutan	46
4.4.5.2	Prosedur Mutasi.....	47
4.4.6	Proses Seleksi	48
4.5.	Implementasi <i>Database</i>	49
4.6.	Implementasi Antarmuka Penentuan Komposisi Pakan Ikan Menggunakan Algoritma Genetika	50
4.7.	Penerapan Aplikasi Penentuan Komposisi Pakan Ikan Menggunakan Algoritma Genetika	52

4.8.	Analisa Hasil	54
BAB V	KESIMPULAN DAN SARAN	59
5.1.	Kesimpulan	59
5.2.	Saran	60
	DAFTAR PUSTAKA.....	61
	LAMPIRAN 1	63
	LAMPIRAN 2	65
	LAMPIRAN 3	67
	LAMPIRAN 4	69
	LAMPIRAN 5	71



UNIVERSITAS BRAWIJAYA



DAFTAR GAMBAR

Gambar 2.1 Contoh kromosom pada pengkodean biner	7
Gambar 2.2 Contoh kromosom pada pengkodean permutasi.....	7
Gambar 2.3 Contoh kromosom pada pengkodean nilai.....	8
Gambar 2.4 Contoh kromosom pada pengkodean pohon.....	8
Gambar 2.5 Struktur umum individu.....	10
Gambar 3.1 <i>Flowchart</i> algoritma genetika untuk menentukan komposisi pakan ikan.....	17
Gambar 3.2 Representasi Kromosom.....	18
Gambar 3.3 <i>Flowchart</i> proses hitung <i>fitness</i>	22
Gambar 3.4 <i>Flowchart Crossover</i>	23
Gambar 3.5 Proses <i>Discrete Crossover</i>	24
Gambar 3.6 <i>Flowchart Discrete Crossover</i>	25
Gambar 3.7 Proses Mutasi.....	26
Gambar 3.8 <i>Flowchart Mutasi</i>	27
Gambar 3.9 <i>Flowchart</i> proses seleksi.....	29
Gambar 4.1 Tampilan utama program	50
Gambar 4.2 Tampilan form input data bahan.....	51
Gambar 4.3 Tampilan form edit data bahan	51
Gambar 4.4 Implementasi program	53

UNIVERSITAS BRAWIJAYA



DAFTAR TABEL

Tabel 3.1 Populasi setelah <i>crossover</i> dan mutasi	35
Tabel 3.2 Individu dan fitnessnya	36
Tabel 3.3 Hasil pengurutan individu berdasarkan besar fitnessnya	36
Tabel 3.4 Percobaan nilai <i>fitness</i> dengan peluang <i>crossover</i> dan peluang mutasi yang berbeda	38
Tabel 3.5 Percobaan konvergensi dengan peluang crossover dan peluang mutasi yang berbeda.....	38
Tabel 3.6 Percobaan <i>fitness</i> dengan komposisi bahan yang berbeda	38
Tabel 4.1 Struktur data dalam algoritma	41
Tabel 4.2 Tabel Bahan.....	49
Tabel 4.3 Data parameter genetika.....	52
Tabel 4.4 Data komponen pakan	52
Tabel 4.5 Nilai <i>Fitness</i> pada probabilitas <i>crossover</i> dan probabilitas mutasi.....	54
Tabel 4.6 Generasi konvergensi pada probabilitas crossover dan probabilitas mutasi	56
Tabel 4.7 Nilai <i>fitness</i> pada komposisi bahan yang berbeda dengan probabilitas mutasi dan <i>crossover</i> 90%	58

UNIVERSITAS BRAWIJAYA



DAFTAR GRAFIK

Grafik 4.1 Perbandingan nilai rata-rata <i>fitness</i> dari masing-masing probabilitas <i>crossover</i> dan mutasi	55
Grafik 4.2 Perbandingan Generasi Konvergensi dengan Probabilitas Mutasi dan Probabilitas <i>Crossover</i>	57



UNIVERSITAS BRAWIJAYA



DAFTAR SOURCECODE

Sourcecode 4.1 Struktur data.....	40
Sourcecode 4.2 Inisialisasi kromosom	42
Sourcecode 4.3 Perhitungan <i>fitness</i>	42
Sourcecode 4.4 Pemilihan induk yang akan di-crossover	44
Sourcecode 4.5 Prosedur <i>Discrete Crossover</i>	44
Sourcecode 4.6 Proses pembangkitan bilangan <i>random</i>	45
Sourcecode 4.7 Proses <i>Crossover</i>	46
Sourcecode 4.8 Proses pemilihan induk mutan	46
Sourcecode 4.9 Prosedur Mutasi	47
Sourcecode 4.10 Proses <i>Ranking Selection</i>	58



UNIVERSITAS BRAWIJAYA



UNIVERSITAS BRAWIJAYA



UNIVERSITAS BRAWIJAYA



BAB I

PENDAHULUAN

1.1 Latar Belakang

Kebutuhan masyarakat terhadap protein hewani ikan terus meningkat seiring dengan peningkatan populasi penduduk, oleh karena itu dibutuhkan suatu usaha pembudidayaan ikan.

Keberhasilan usaha pembudidayaan ikan sangat dipengaruhi oleh 3 faktor, yaitu *breeding* (pemuliaan biakan, bibit), *feeding* (pakan) dan *management* (tata laksana). Namun jika dilihat dari total biaya produksi dalam usaha budidaya ikan, maka kontribusi pakan adalah yang paling tinggi sekitar 75% (Khairuman, 2002). Hal ini menyebabkan pakan ikan menjadi faktor penentu utama biaya produksi pembudidayaan ikan, oleh karena itu pemilihan pakan sangat menentukan keberhasilan pembudidayaan ikan.

Pakan yang dimakan ikan berasal alam (disebut pakan alami) dan dari buatan manusia (disebut pakan buatan). Dalam praktiknya, pakan alami sudah terdapat secara alami dalam perairan kolam tempat pemeliharaan ikan. Sedangkan pakan buatan diramu dari beberapa bahan baku yang memiliki kandungan nutrisi spesifik (Khairuman, 2002).

Dalam menentukan komposisi pakan ikan harus didasarkan pada pertimbangan kebutuhan nutrien ikan, kualitas bahan baku, dan nilai ekonomis. Masing-masing bahan baku memiliki nutrisi yang berbeda-beda, oleh karena itu ada banyak kemungkinan kombinasi bahan baku yang dapat digunakan untuk meningkatkan kualitas ikan. Namun demikian, tidak semua kombinasi bahan baku akan memberikan solusi terbaik. Solusi terbaik adalah kombinasi bahan pakan yang dapat memenuhi kebutuhan standar gizi ikan. Daftar kebutuhan standar gizi ikan dapat dilihat pada lampiran 1.

Permasalahan pemilihan komposisi bahan pakan ini termasuk dalam ruang lingkup kombinatorik. Sebagai contoh kasus, dalam penelitian ini terdapat 34 jenis bahan pakan dengan kandungan nutrisi yang berbeda-beda (daftar jenis bahan pakan ikan dan kandungan nutrisinya dapat dilihat pada lampiran 2), jika dari 34 jenis bahan pakan akan dipilih 10 jenis bahan pakan, maka akan didapatkan 10^{34} kombinasi bahan pakan. Kombinasi 10^{34} ini

merupakan jumlah solusi yang terjadi sehingga untuk menemukan satu solusi dari 10^{34} solusi yang ada akan membutuhkan waktu yang sangat lama.

Salah satu metode yang bisa digunakan untuk menyelesaikan masalah kombinatorik adalah algoritma genetika (Sanjoyo, 2006). Setiap solusi pada algoritma genetika diwakili oleh satu individu atau satu kromosom. Algoritma genetika didasarkan atas mekanisme evolusi biologis (Kusumadewi, 2003). Dengan menggunakan konsep evolusi biologis maka akan dihasilkan suatu keluaran berupa komposisi dari bahan pakan yang sebaiknya dikonsumsi untuk memenuhi kebutuhan nutrisi ikan.

Pada penelitian sebelumnya telah dilakukan penelitian terhadap masalah formulasi pakan. Salah satunya oleh Toriq (2009) yaitu masalah optimasi formulasi pakan ternak menggunakan algoritma genetika. Pada penelitian Toriq ini fokus permasalahannya adalah menentukan komposisi pakan dengan biaya termurah dengan mempertimbangkan setiap diskon harga pada bahan-bahan pakan.

Pada skripsi ini akan dilakukan penelitian tentang penentuan pakan ikan menggunakan algoritma genetika. Algoritma genetika diharapkan mampu menghasilkan suatu komposisi pakan yang terbaik bagi ikan.

1.1 Rumusan Masalah

Rumusan masalah dalam skripsi ini adalah:

1. Bagaimana menerapkan seleksi *Ranking* pada permasalahan penentuan komposisi pakan ikan untuk menemukan hasil yang optimal?
2. Berapa persen peluang *crossover* dan peluang mutasi pada permasalahan penentuan komposisi pakan ikan untuk menemukan hasil yang optimal?
3. Bagaimana menentukan komposisi bahan pada permasalahan penentuan komposisi pakan ikan untuk menemukan hasil yang optimal?

1.2 Batasan Masalah

Batasan masalah dalam skripsi ini adalah data komposisi bahan pakan yang digunakan diambil dari buku *Budi Daya Ikan di*

Perairan Umum karangan Ir. Cahyono Bambang, 2001 dan buku *Membuat Pakan Ikan Konsumsi* karangan Khairuman, 2002.

1.3 Tujuan

Tujuan dari penulisan skripsi ini adalah :

1. Menerapkan seleksi *Ranking* pada permasalahan penentuan komposisi pakan ikan untuk menemukan hasil yang optimal.
2. Menentukan peluang *crossover* dan peluang mutasi pada permasalahan penentuan komposisi pakan ikan untuk menemukan hasil yang optimal.
3. Menentukan komposisi bahan pada permasalahan penentuan komposisi pakan ikan untuk menemukan hasil yang optimal.

1.4 Manfaat

Adapun manfaat dari pembuatan tugas akhir ini adalah tersedianya perangkat lunak untuk menentukan komposisi pakan ikan dengan menggunakan algoritma genetika sehingga didapatkan komposisi pakan ikan yang terbaik yaitu sesuai dengan standar kebutuhan nutrisi ikan, serta skripsi ini juga bermanfaat untuk memperoleh pengaruh dari perubahan probabilitas *crossover* dan mutasi terhadap nilai *fitness*.

1.5 Metodologi Pemecahan Masalah

Untuk mencapai tujuan yang dirumuskan sebelumnya, maka metodologi yang digunakan dalam penulisan tugas akhir ini adalah:

1. Studi literatur

Mempelajari teori-teori yang berhubungan dengan konsep algoritma genetik dari berbagai referensi.

2. Pendefinisian dan analisis masalah

Mendefinisikan dan menganalisis masalah untuk mencari solusi yang tepat.

3. Perancangan dan implementasi sistem

Membuat perancangan piranti lunak dengan analisis terstruktur dan mengimplementasikan hasil rancangan tersebut yaitu penerapan algoritma genetik pada penentuan komposisi pakan ikan.

4. Uji coba dan analisa hasil implementasi

Menguji piranti lunak, dan menganalisa hasil dari

implementasi tersebut apakah sudah sesuai dengan tujuan yang dirumuskan sebelumnya, untuk kemudian menarik kesimpulannya.

1.6 Sistematika Penulis

Untuk memberikan gambaran tentang skripsi ini, berikut disajikan secara garis besar pembahasan dari keseluruhan isi laporan skripsi untuk setiap bab, sebagai berikut :

BAB I :PENDAHULUAN

Berisi latar belakang penulisan, permasalahan yang dihadapi, tujuan, batasan masalah, dan manfaat serta sistematika penulisan skripsi.

BAB II :TINJAUAN PUSTAKA

Menguraikan teori tentang pakan ikan, dan algoritma genetika, serta teori-teori yang berhubungan dengan penggunaan metode algoritma genetika.

BAB III :METODOLOGI DAN PERANCANGAN

Berisi metode-metode yang digunakan dalam membuat sistem yang mampu memecahkan permasalahan pada *optimasi penentuan komposisi pakan ikan*.

BAB IV : HASIL DAN PEMBAHASAN

Berisi tentang penjelasan implementasi sistem dan hasil pengujian yang dilakukan dari penerapan algoritma genetika dalam menentukan komposisi pakan ikan.

BAB V :PENUTUP

Bab ini berisi kesimpulan yang diperoleh dari hasil pengujian dan saran-saran untuk pengembangan lebih lanjut.

BAB II

TINJAUAN PUSTAKA

2.1 Komposisi Pakan ikan

Pakan bagi ikan merupakan salah satu faktor yang menentukan keberhasilan suatu budidaya perikanan. Pakan ikan diberikan pada ikan dengan harapan antara lain (Murtidjo, 2001) :

- a. Pakan merupakan salah satu faktor input bersama-sama dengan faktor pengelolaan yang baik, akan membuat benih ikan memiliki pertumbuhan yang baik.
- b. Pakan merupakan kebutuhan dasar bagi ikan untuk kelangsungan hidup dan proses biologis dalam tubuh.
- c. Pakan merupakan biaya yang terbesar dari total biaya produksi budidaya ikan.

Pakan yang diberikan harus memenuhi standar nutrisi (gizi) bagi ikan agar kelangsungan hidupnya tinggi dan pertumbuhannya cepat. Pakan yang baik memiliki komposisi zat gizi yang lengkap seperti protein, lemak, karbohidrat, serat, vitamin, dan mineral. Oleh karena itu diperlukan suatu pemilihan komposisi bahan pakan yang tepat baik dari kuantitas maupun kualitasnya. Daftar analisis nilai gizi komposisi bahan pakan dapat dilihat pada lampiran 2.

Dalam pembuatan piranti lunak formulasi pakan menurut Perry (2004) terdapat beberapa kebutuhan pokok, antara lain fasilitas komputer yang memadai, sumber daya manusia yang terlatih, informasi kebutuhan nutrien ternak, informasi jenis bahan pakan yang tepat, informasi komposisi bahan pakan dan informasi harga bahan pakan yang tersedia.

2.2 Algoritma Genetika

Algoritma genetika adalah algoritma pencarian heuristik yang didasarkan atas mekanisme evolusi biologis (Kusumadewi, 2003). Konsep dasar yang mengilhami timbulnya algoritma genetika adalah teori evolusi alam yang dikemukakan oleh Charles Darwin. Teori tersebut menjelaskan bahwa pada proses evolusi alami, setiap individu harus melakukan adaptasi terhadap lingkungan disekitarnya agar dapat bertahan hidup (Kuswara, 2003).

Algoritma genetika pertama kali dikembangkan pada tahun 1975 oleh John Holland bersama rekan kerja dan murid-muridnya dari Universitas Michigan. John Holland mengatakan bahwa setiap masalah yang berbentuk adaptasi (alami maupun buatan) dapat diformulasikan dalam terminologi genetika (Kusumadewi, 2003).

2.2.1 Solusi dalam Algoritma Genetika

Algoritma genetika berangkat dari himpunan solusi yang dihasilkan secara acak ataupun terstruktur yang disebut populasi. Sedangkan setiap individu dalam populasi disebut kromosom yang merupakan representasi dari solusi.

Solusi dari suatu populasi diambil dan digunakan untuk membentuk populasi baru berikutnya. Dengan harapan, populasi baru berikutnya akan lebih baik dari populasi sebelumnya. Individu yang dipilih adalah individu yang memiliki karakteristik yang bagus yang ditandai dari nilai *fitness*-nya, yakni makin baik nilainya, maka makin besar kesempatannya untuk diproduksi (Kusumadewi, 2003). Sedangkan individu dalam populasi yang tidak terseleksi dalam reproduksi akan mati dengan sendirinya. Dengan jalan ini, beberapa generasi dengan karakteristik yang bagus akan bermunculan dalam populasi tersebut, untuk kemudian dicampur dan ditukar dengan karakter yang lain. Dengan mengawinkan semakin banyak individu, maka akan semakin banyak pula kemungkinan terbaik yang dapat diperoleh.

2.2.2 Struktur Algoritma Genetika

Struktur algoritma genetika menurut Gen,M,1997 :

1. [Start] Generate populasi yang beranggotakan n kromosom secara acak (solusi yang memenuhi untuk permasalahan yang dihadapi).
2. [Kecocokan] Evaluasi kecocokan $f(x)$ dari setiap kromosom x dalam populasi tersebut.
3. [Populasi baru] Bentuk sebuah populasi baru dengan cara mengulang langkah-langkah 1 dan 2 sampai sebuah populasi baru terbentuk.
 - a. [Crossover] Dengan melihat probabilitas *crossover* lakukan *crossover* antara dua parent untuk membentuk *offspring* baru.

- b. [Mutasi] Dengan melihat probabilitas mutasi, lakukan mutasi pada individu dalam populasi awal untuk membentuk *offspring* baru.
- c. [Seleksi] Individu-individu yang terkumpul dari populasi awal dan offspring akan dipilih sejumlah individu ada populasi awal sebagai populasi baru.
4. [Menggantikan] Gunakan populasi yang baru dibentuk untuk penggunaan algoritma selanjutnya.
5. [Pengujian] Jika hasil akhir memuaskan, berhenti dan kembali ke solusi terbaik dalam populasi saat ini.
6. [Pengulangan] Ulangi langkah 2.

2.2.3 Pengkodean

Pengkodean adalah cara untuk merepresentasikan masalah ke dalam bentuk kromosom. Berikut adalah beberapa jenis pengkodean yang umum digunakan (Kuswara, 2003).

1. Pengkodean Biner

Pengkodean biner adalah pengkodean yang paling umum dan paling sederhana dalam merepresentasikan masalah pada algoritma genetika. Pada pengkodean biner setiap kromosom terdiri atas barisan string bit 0 atau 1. Contoh dapat dilihat pada Gambar 2.1.

Kromosom A	0101101100010011
Kromosom B	1011010110110101

Gambar 2.1 Contoh kromosom pada pengkodean biner.

2. Pengkodean Permutasi

Pengkodean permutasi dapat digunakan pada masalah pengurutan data (*ordering problems*), seperti wiraniaga (*Travelling Salesman Problem*) atau masalah pengurutan tugas (*Task Ordering Problem*). Pada pengkodean permutasi, setiap kromosom terdiri dari barisan angka, yang merepresentasikan angka pada urutan. Contoh pengkodean permutasi dapat dilihat pada gambar 2.2.

Kromosom A	8549102367
Kromosom B	9102438576

Gambar 2.2 Contoh kromosom pada pengkodean permutasi.

3. Pengkodean Nilai

Pengkodean nilai dapat digunakan pada masalah yang sangat kompleks, dimana nilai yang dikodekan langsung merupakan representasi dari masalah. Penggunaan pengkodean biner pada tipe masalah yang kompleks akan menjadi lebih susah.

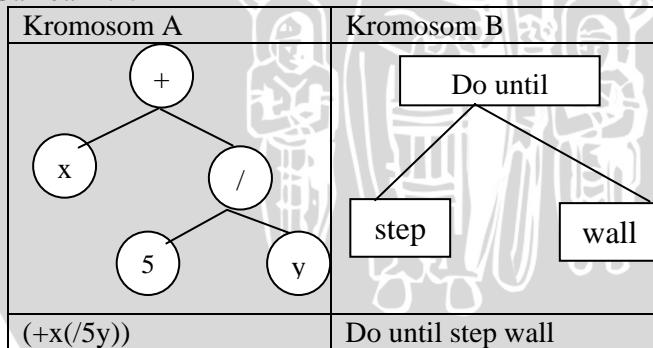
Pada pengkodean nilai, setiap kromosom adalah barisan dari beberapa nilai. Nilai dapat berupa apa saja, seperti bilangan biasa, bilangan riil, karakter sampai dengan obyek-obyek yang rumit. Contoh pengkodean nilai dapat dilihat pada Gambar 2.3.

Kromosom A	[red], [black], [blue], [yellow], [red]
Kromosom B	1.875, 3.9821, 9.1283, 6.8344, 4.116
Kromosom C	A, B, C, D, E, F, G, H, I, J, K, L, M, N

Gambar 2.3 Contoh kromosom pada pengkodean nilai.

4. Pengkodean Pohon

Pengkodean pohon lebih banyak digunakan untuk menyusun program atau ekspresi, bagi pemrograman genetika (*genetic programming*). Pada pengkodean pohon, setiap kromosom merupakan pohon dari sejumlah obyek, seperti fungsi atau perintah pada bahasa pemrograman. Contoh pengkodean pohon dapat dilihat pada Gambar 2.4.



Gambar 2.4 Contoh kromosom dengan pengkodean pohon.

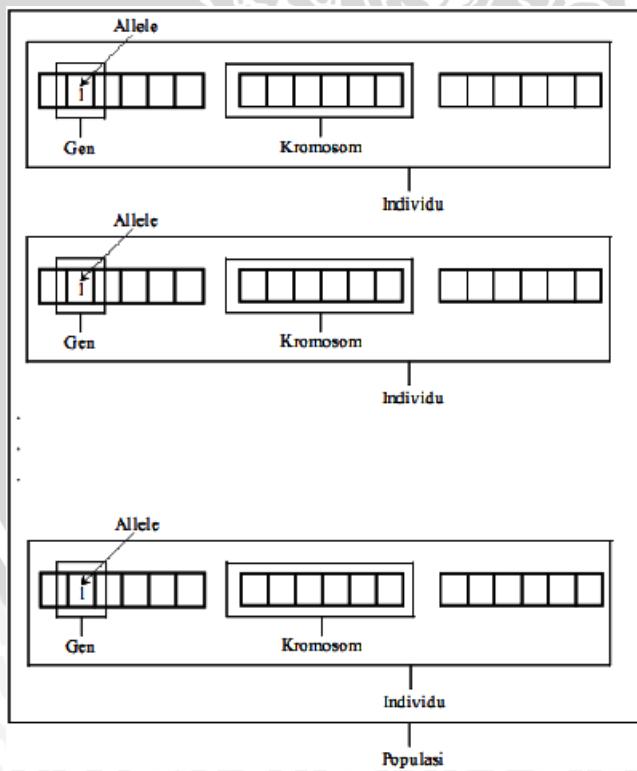
2.2.4 Mendefinisikan individu

Individu menyatakan salah satu solusi yang mungkin. Individu bisa dikatakan sama dengan kromosom, yang merupakan

kumpulan gen. Beberapa definisi yang perlu diperhatikan dalam individu sebagai berikut :

- *Genotype* (Gen), sebuah nilai yang menyatakan satuan dasar yang membentuk suatu arti tertentu dalam satu kesatuan gen yang dinamakan kromosom. Dalam algoritma genetika, gen ini bisa berupa nilai biner, integer maupun karakter, atau kombinatorial.
- *Allele*, nilai dari gen
- Kromosom, gabungan gen-gen yang membentuk nilai tertentu.
- Individu, menyatakan satu nilai atau keadaan yang menyatakan salah satu solusi yang mungkin dari permasalahan.
- Populasi, merupakan sekumpulan individu yang akan diproses bersama dalam satu siklus proses evolusi.
- Generasi, menyatakan satu siklus proses atau iterasi didalam algoritma genetika.

Untuk lebih jelas tentang definisi individu dapat dilihat pada Gambar 2.5.



Gambar 2.5 Struktur umum individu (Kusumadewi, 2003).

2.2.5 Mendefinisikan nilai *fitness*

Nilai *fitness* adalah nilai yang menyatakan baik tidaknya suatu solusi (individu). Perhitungan *fitness* berdasarkan pada nilai bobot bahan dan kandungan nutrisi bahan. Untuk menentukan nilai bobot bahan menggunakan langkah-langkah sebagai berikut:

- Menentukan nilai bobot sementara tiap bahan dengan cara merandom nilai sebesar berat pakan.
- Menentukan bobot total dengan cara menjumlahkan seluruh nilai bobot sementara tiap bahan.

Cara menentukan kandungan nutrisi bahan yaitu dengan mengalikan bobot bahan dengan masing-masing kandungan nutrisi (protein, lemak, karbohidrat, serat dan abu) kemudian dibagi dengan seratus.

Besar kecilnya nilai kandungan nutrisi berpengaruh terhadap besar kecilnya *fitness*. Nilai *fitness* ini yang dijadikan acuan dalam mencapai nilai optimal dalam algoritma genetika. Algoritma genetika bertujuan mencari individu dengan nilai *fitness* yang paling tinggi.

Rumus untuk mencari nilai *fitness* :

$$fitness = \frac{1}{((Abs(a - p)) + (Abs(b - q)) + (Abs(c - r)) + (Abs(d - s)) + (Abs(e - t)) + 1)}$$

(2.1)

Keterangan :

- a = Protein standar
- b = lemak standar
- c = Karbohidrat standar
- d = Serat standar
- e = Abu standar
- p = Jumlah Protein
- q = Jumlah Lemak
- r = Jumlah karbohidrat
- s = Jumlah serat

t = Jumlah Abu

1 = bilangan konstan untuk menghindari pembagian dengan nol

Untuk nilai nutrisi bahan pakan (protein, lemak, karbohidrat, serat dan abu) dapat dilihat dalam lampiran 2 (Cahyono, 2001) dan untuk kebutuhan standar masing-masing nutrisi pada ikan dapat dilihat dalam lampiran 1 (Craig, 2009).

2.2.6 Pembangkitan populasi awal

Membangkitkan populasi awal adalah proses membangkitkan sejumlah individu secara acak atau melalui prosedur tertentu. Populasi itu sendiri terdiri dari sejumlah kromosom yang merepresentasikan solusi yang diinginkan.

2.2.7 Perkawinan silang

Proses perkawinan silang (*crossover*) berfungsi untuk menghasilkan keturunan dari dua buah kromosom induk yang terpilih. Kromosom anak yang dihasilkan merupakan kombinasi gen-gen yang dimiliki oleh kromosom induk (Kuswara, 2003).

Secara umum, mekanisme kawin silang adalah sebagai berikut:

- a. memilih dua buah kromosom sebagai induk.
- b. memilih secara acak posisi dalam kromosom, biasa disebut titik perkawinan silang, sehingga masing-masing kromosom induk terpecah menjadi dua segmen.
- c. lakukan pertukaran antar segmen kromosom induk untuk menghasilkan kromosom anak.

Jumlah individu yang mengalami proses *crossover* tergantung pada jumlah probabilitas *crossover* (P_c) yang ditentukan. Jika ditentukan nilai P_c adalah 20%, maka diharapkan akan ada 20% dari jumlah individu pada populasi yang akan mengalami *crossover* (Michalewicz. 1996). Pada tingkat probabilitas *crossover* yang cukup tinggi, proses pencarian solusi optimum dapat menjelajah ke ruang explorasi yang lebih luas sehingga kemungkinan terperangkap pada nilai optimum lokal yang salah dapat dihindari. Salah satu cara menentukan individu yang akan mengalami *crossover* adalah dengan membangkitkan nilai r [0..1] sejumlah individu dalam populasi secara random. Jika

nilai $r < P_c$, maka individu tersebut akan mengalami *crossover* (Michalewicz. 1996).

2.2.7.1 Discrete Crossover

Pada skripsi ini, teknik *crossover* yang digunakan adalah *discrete crossover*. Pada teknik ini, digunakan dua buah kromosom untuk membuat kromosom baru. Untuk menghasilkan kromosom baru, kromosom induk dipecah menjadi tiga bagian. Setelah itu bilangan acak dibangkitkan sebanyak tiga buah. Bilangan acak tersebut bernilai satu atau dua. Bilangan satu menyatakan induk yang pertama dan bilangan dua menyatakan induk yang kedua. Kemudian kromosom yang baru dibentuk sesuai dengan bilangan acak yang telah didapatkan.

2.2.8 Mutasi

Setelah melalui proses perkawinan silang, pada *offspring* dapat dilakukan proses mutasi. Variabel *offspring* dimutasi dengan menambahkan nilai *random* yang sangat kecil, dengan probabilitas yang rendah. Peluang mutasi mengendalikan banyaknya gen baru yang akan dimunculkan untuk dievaluasi. Jika peluang mutasi terlalu kecil, banyak gen yang mungkin berguna tidak pernah dievaluasi. Tetapi bila peluang mutasi terlalu besar, maka akan terlalu banyak gangguan acak, sehingga anak akan kehilangan kemiripan dari induknya. Mutasi berperan untuk menggantikan gen yang hilang dari populasi akibat proses seleksi yang memungkinkan munculnya kembali gen yang tidak muncul pada inisialisasi populasi, sehingga meningkatkan variasi populasi (Kusumadewi, 2003).

Tujuan dari mutasi adalah agar individu-individu yang ada dalam populasi semakin bervariasi. Mutasi akan sangat berperan jika pada populasi awal hanya ada sedikit solusi yang mungkin terpilih. Sehingga, operasi itu sangat berguna dalam mempertahankan keanekaragaman individu dalam populasi meskipun dengan mutasi tidak dapat diketahui apa yang terjadi pada individu baru (Kuswara, 2003).

Jumlah individu yang mengalami proses mutasi tergantung pada jumlah probabilitas mutasi (P_m) yang ditentukan. Salah satu cara menentukan individu yang akan mengalami mutasi adalah

dengan membangkitkan nilai r [0..1] sejumlah individu dalam populasi secara *random*. Jika nilai $r < P_m$, maka individu tersebut akan mengalami mutasi (Michalewicz, 1996).

2.2.8.1 Mutasi Random

Pada algoritma genetika, mutasi memainkan peran penting, yaitu pertama, menggantikan gen yang hilang dari populasi selama proses seleksi. Kedua, menyediakan gen yang tidak pernah ditampilkan pada populasi awal (Sanjoyo, 2006). Jumlah individu yang mengalami proses mutasi tergantung pada jumlah probabilitas mutasi (P_m) yang ditentukan. Teknik mutasi yang digunakan adalah mutasi random. Langkah-langkah proses mutasi random adalah sebagai berikut (Al-Hajri and Abido, 2009):

- Menentukan P_m (10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% dan 100%).
- Membangkitkan bilangan random [0..1] sebanyak jumlah individu.
- Jika bilangan random $< P_m$, individu terpilih untuk dikenai mutasi.
- Memilih gen yang akan dimutasi (k) secara random.
- Membangkitkan bilangan random antara 1 sampai jumlah bahan pakan sebanyak 1 bilangan (m) dengan ketentuan bilangan (m) merupakan bilangan yang belum pernah terpakai pada populasi awal.
- Mengganti nilai k dengan nilai m .

2.2.9 Seleksi

Seleksi digunakan untuk mendapatkan calon induk yang baik. Langkah pertama yang dilakukan dalam seleksi ini adalah pencarian nilai *fitness*. Seleksi mempunyai tujuan untuk memberikan kesempatan reproduksi yang lebih besar bagi anggota populasi yang mempunyai nilai *fitness* terbaik. Semakin tinggi nilai *fitness* suatu individu semakin besar kemungkinannya untuk terpilih. Ada beberapa metode seleksi dalam algoritma genetika, tetapi metode seleksi yang akan dipakai dan dibahas dalam skripsi ini adalah metode seleksi *Ranking*.

Seleksi *Ranking* dimulai dari mengurutkan nilai *fitness* dari yang tertinggi sampai yang terendah dalam satu populasi, setelah itu

individu dengan nilai fitness tertinggi akan terpilih sebagai induk, sedangkan individu dengan nilai *fitness* terendah akan dibuang.

UNIVERSITAS BRAWIJAYA



BAB III

METODOLOGI DAN PERANCANGAN

Pada bab metode dan perancangan ini akan dibahas langkah-langkah yang digunakan dalam pembuatan piranti lunak dan metode percobaan. Tahapan pembuatannya adalah sebagai berikut :

3.1 Deskripsi Sistem

Jika ada sejumlah bahan yang ingin dikombinasikan menjadi pakan ikan dengan komposisi nutrisi tertentu dan bahan-bahan tersebut memiliki komposisi nutrisi yang berbeda, seorang formulator dapat menentukan bahan apa yang akan dipakai, persentase bahan pakan dan jumlah masing-masing nutrisi bahan tersebut dengan algoritma genetika. Algoritma genetika diharapkan dapat menemukan solusi yang memenuhi hal-hal tersebut dengan pemilihan komposisi yang dapat memenuhi kebutuhan standar gizi untuk ikan.

Sistem yang akan dibangun ini menggunakan algoritma genetika. Tahap-tahap yang akan dilakukan sebagai berikut :

1. Representasi kromosom.
2. Evaluasi dengan menghitung *fitness* dari masing-masing individu.
3. Proses *crossover* untuk mendapatkan individu baru.
4. Proses mutasi yang berfungsi untuk meningkatkan variasi dalam populasi atau mencegah konvergensi dini.
5. Proses seleksi untuk membentuk populasi baru

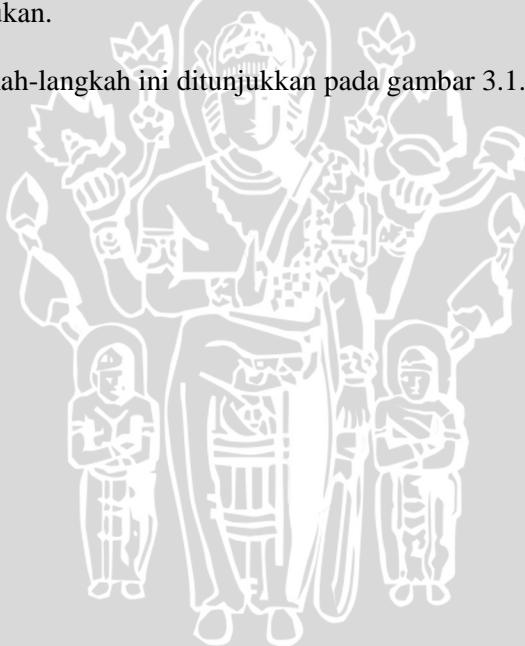
3.2 Langkah-Langkah Proses Penyelesaian Penentuan Komposisi Pakan Ikan Dengan Menggunakan Algoritma Genetika

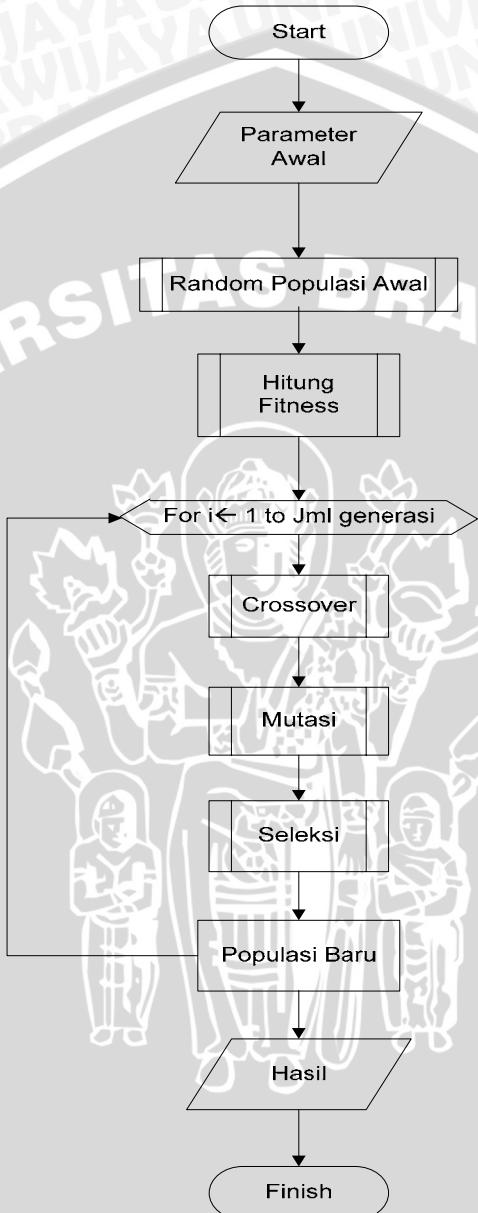
Langkah-langkah implementasi algoritma genetika untuk Menentukan komposisi pakan ikan adalah sebagai berikut :

1. Tentukan semua parameter awal yang diperlukan, antara lain :
 - Jumlah gen dalam satu individu
 - Jumlah individu pada setiap populasi
 - Jumlah generasi
 - Nilai probabilitas *crossover* dan mutasi
2. Bangkitkan populasi awal secara acak (random).
3. Evaluasi setiap individu pada populasi dengan menghitung

- nilai fitness.
4. Bentuk populasi baru dengan melakukan langkah-langkah berikut ini :
 - a. Dilakukan proses *crossover* dengan metode *Discrete Crossover* pada individu yang telah terpilih, kemudian dilakukan perhitungan nilai *fitness*.
 - b. Dilakukan mutasi pada individu yang terpilih, kemudian hitung nilai fitness.
 - c. Dilakukan proses seleksi dari populasi dengan metode *Ranking Selection*.
 5. Digunakan populasi yang baru untuk dilakukan kembali langkah no. 4.
 6. Dilakukan proses-proses diatas sesuai dengan jumlah iterasi yang telah ditentukan.

Flowchart dari langkah-langkah ini ditunjukkan pada gambar 3.1.





Gambar 3.1 Flowchart Algoritma Genetika Untuk Menentukan Komposisi Pakan Ikan

3.2.1 Representasi Kromosom

Kromosom terdiri dari gen-gen dimana gen tersebut merupakan *encoding* yang digunakan dalam merepresentasikan masalah penentuan komposisi pakan ikan. Jumlah gen dalam tiap kromosom sama dengan jumlah bahan yang akan dikomposisikan. Teknik *encoding* yang digunakan pada persoalan ini adalah permutasi *encoding*. Pada permutasi *encoding*, setiap bahan pakan diwakili oleh sebuah angka dan angka tersebut merupakan sebuah gen yang ada pada sebuah kromosom.

Setiap kromosom yang ada pada satu generasi merepresentasikan komposisi bahan pakan yang digunakan untuk mendapatkan hasil yang optimal dan kromosom yang mewakili solusi tersebut boleh mempunyai lebih dari satu jenis bahan pakan yang sama. Dalam hal ini, apabila terjadi beberapa nama bahan pakan yang muncul lebih dari satu kali pada setiap kromosom maka akan dianggap bahan pakan yang dipakai akan sebanyak kemunculan angka yang sama. Berikut ini adalah gambar kromosom pada persoalan ini :

12	01	07	14	09	08	07	14	10	06	20	18
----	----	----	----	----	----	----	----	----	----	----	----



14 = id bahan tepung terigu

Gambar 3.2 Representasi Kromosom

3.2.2 Membangkitkan Populasi Awal

Populasi merupakan kumpulan beberapa kromosom. Langkah pertama yang dilakukan adalah membangkitkan populasi awal dengan cara membangkitkan bilangan acak antara 1 sampai dengan jumlah bahan pakan yang ingin dikombinasikan. Jumlah bahan pakan yang ada pada skripsi ini adalah 34. Maka bilangan acak yang akan dibangkitkan antara 1 sampai 34. Bilangan acak ini merupakan bahan pakan yang akan dikombinasikan untuk menyelesaikan masalah komposisi pakan tersebut.

Misalkan jumlah populasi awal yang dibangkitkan adalah 4, maka populasinya sebagai berikut :

Individu 1 : 12 1 7 14 9 20
Individu 2 : 19 2 17 4 9 24
Individu 3 : 13 10 30 1 12 6
Individu 4 : 34 31 5 23 22 2

Dari populasi awal tersebut diketahui individu 1 : 12 1 7 14 9 20, maka dari individu tersebut jika dikembalikan dalam bentuk bahan pakan menjadi dedak jagung, tepung ikan, tepung bekicot, tepung terigu, tepung daging dan bungkil kacang kedele.

3.2.3 Nilai Fitness

Langkah selanjutnya adalah menghitung *fitness* dari masing-masing kromosom. Perhitungan *fitness* berdasarkan pada langkah-langkah sebagai berikut:

1. Menentukan bobot (berat) bahan.

Untuk menentukan nilai bobot bahan menggunakan langkah-langkah sebagai berikut:

- Menentukan nilai bobot sementara tiap bahan dengan cara merandom nilai sebesar berat pakan.
- Menentukan bobot total dengan cara menjumlahkan seluruh nilai bobot sementara tiap bahan
- Membagikan nilai bobot sementara tiap bahan dengan bobot total, setelah itu dikalikan dengan berat pakan, hasil tersebut merupakan nilai bobot pakan sebenarnya.

Contoh :

Misalkan ada 3 bahan yang akan dikomposisikan dengan berat pakan yang akan dihasilkan adalah 100kg:

- Dedak halus
- Daun turi
- Bungkil kelapa
- Menentukan nilai bobot sementara tiap bahan
 - Dedak halus 50kg
 - Daun turi 20kg
 - Bungkil kelapa 40kg
- Menentukan bobot total
 - Bobot total = $50+20+40=110\text{kg}$
- Membagikan nilai bobot sementara tiap bahan dengan bobot total, setelah itu dikalikan dengan berat pakan

- Dedak halus = $(50/110)*100 = 45,45\text{kg}$
- Daun turi = $(20/110)*100 = 18,18\text{kg}$
- Bungkil kelapa = $(40/110)*100 = 36,36\text{kg}$

2. Menentukan kandungan nutrisi bahan

Cara menentukan kandungan nutrisi bahan yaitu dengan mengalikan bobot bahan dengan masing-masing kandungan nutrisi (protein, lemak, karbohidrat, serat dan abu) kemudian dibagi dengan seratus.

Contoh :

Misalkan dedak halus memiliki kandungan protein 30, lemak 10, karbohidrat 5, serat 10 dan abu 0,5. Maka

$$\text{Jumlah protein} = (50*30)/100 = 15$$

$$\text{Jumlah lemak} = (50*10)/100 = 5$$

$$\text{Jumlah karbohidrat} = (50*5)/100 = 2,5$$

$$\text{Jumlah serat} = (50*10)/100 = 5$$

$$\text{Jumlah abu} = (50*0,5)/100 = 0,25$$

Misalkan Daun turi memiliki kandungan protein 15, lemak 5, karbohidrat 5, serat 0,5 dan abu 0,5. Maka

$$\text{Jumlah protein} = (20*15)/100 = 3$$

$$\text{Jumlah lemak} = (20*5)/100 = 1$$

$$\text{Jumlah karbohidrat} = (20*5)/100 = 1$$

$$\text{Jumlah serat} = (20*0,5)/100 = 0,01$$

$$\text{Jumlah abu} = (20*0,5)/100 = 0,01$$

Misalkan Bungkil kelapa memiliki kandungan protein 20, lemak 5, karbohidrat 50, serat 30 dan abu 0. Maka

$$\text{Jumlah protein} = (40*20)/100 = 8$$

$$\text{Jumlah lemak} = (40*5)/100 = 2$$

$$\text{Jumlah karbohidrat} = (40*50)/100 = 20$$

$$\text{Jumlah serat} = (40*30)/100 = 12$$

$$\text{Jumlah abu} = (40*0)/100 = 0$$

Maka Jumlah Protein bahan = $15+3+8 = 26$

$$\text{Jumlah Lemak bahan} = 5+1+2 = 8$$

$$\text{Jumlah karbohidrat bahan} = 2,5+1+20 = 23,5$$

$$\text{Jumlah serat bahan} = 5+0,01+12 = 17,01$$

$$\text{Jumlah abu bahan} = 0,25 + 0,01 + 0 = 0,26$$

3. Mencari nilai *fitness*

Untuk mencari nilai *fitness* terlebih dahulu harus mengetahui nilai standar nutrisi bagi ikan. Nilai protein standar ikan 30%, lemak 6%, karbohidrat 17.5%, serat 6% dan abu 8,5% (terdapat pada lampiran 1).

$$fitness = \frac{1}{((\text{Abs}(a - p)) + (\text{Abs}(b - q)) + (\text{Abs}(c - r)) + (\text{Abs}(d - s)) + (\text{Abs}(e - t)) + 1)}$$

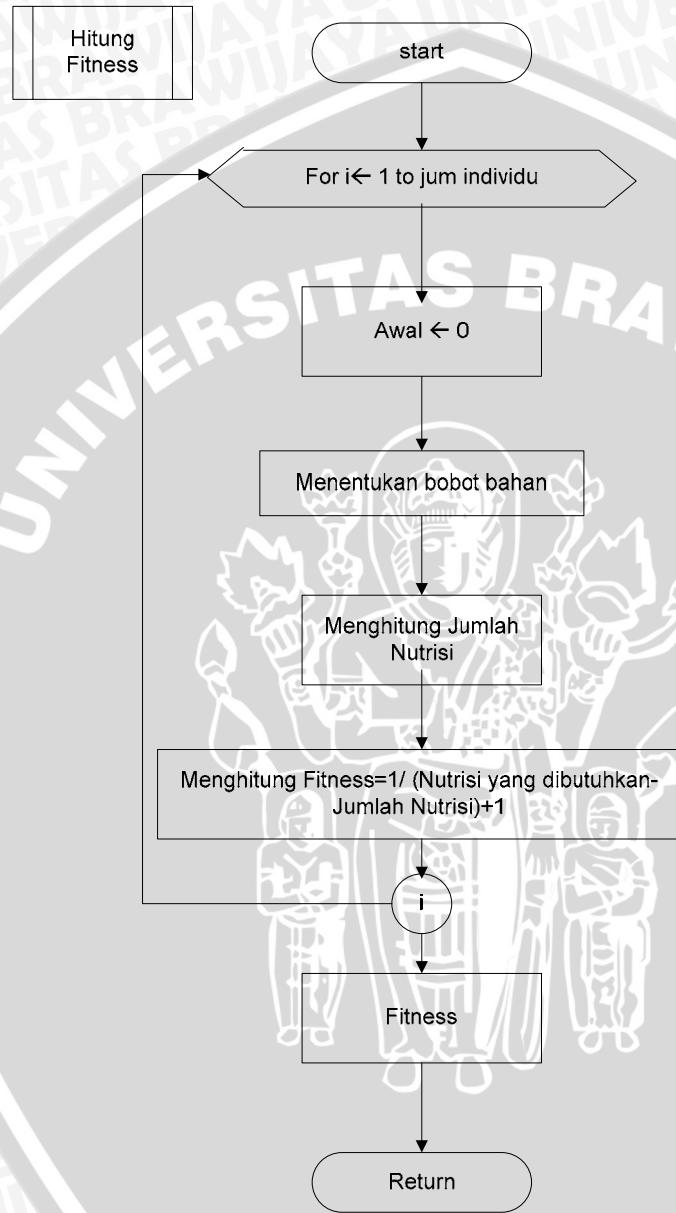
$$fitness = \frac{1}{((\text{Abs}(30 - 26)) + (\text{Abs}(16 - 8)) + (\text{Abs}(17,5 - 23,3)) + (\text{Abs}(6 - 17,01)) + (\text{Abs}(8,5 - 0,26)) + 1)}$$

$$fitness = \frac{1}{4 + 8 + 5,8 + 11,01 + 8,24 + 1}$$

$$fitness = \frac{1}{38,05}$$

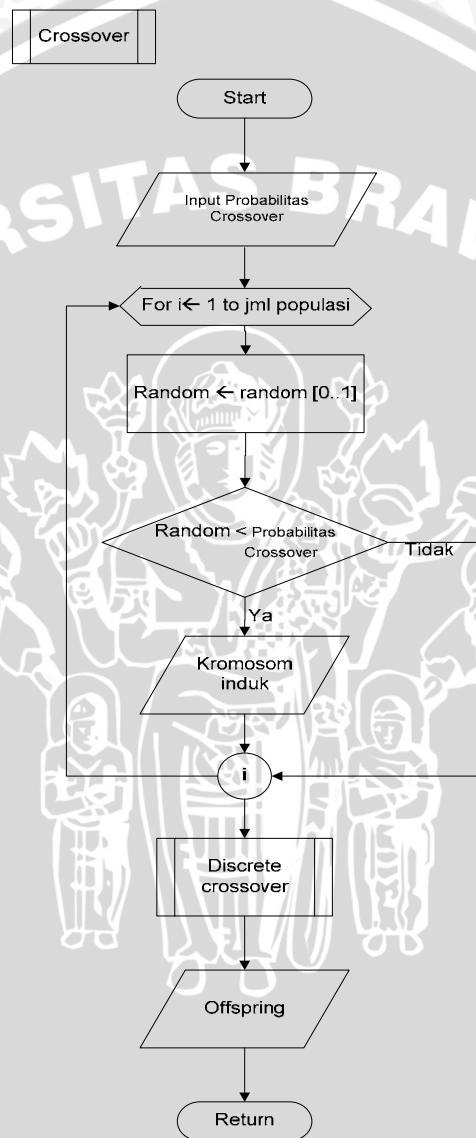
$$fitness = 0,02628$$

Proses menghitung nilai *fitness* dari tiap individu ditunjukkan pada gambar 3.3.



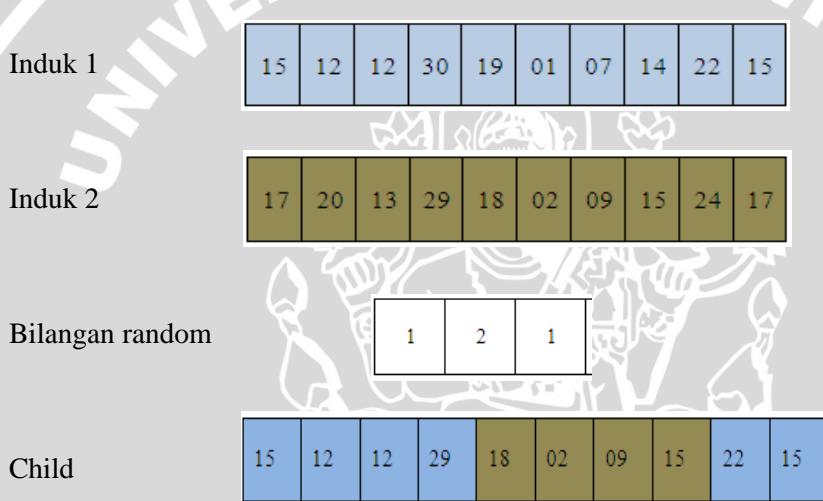
3.2.4 Crossover

Untuk diagram alir proses *crossover* ditunjukkan pada gambar 3.4.



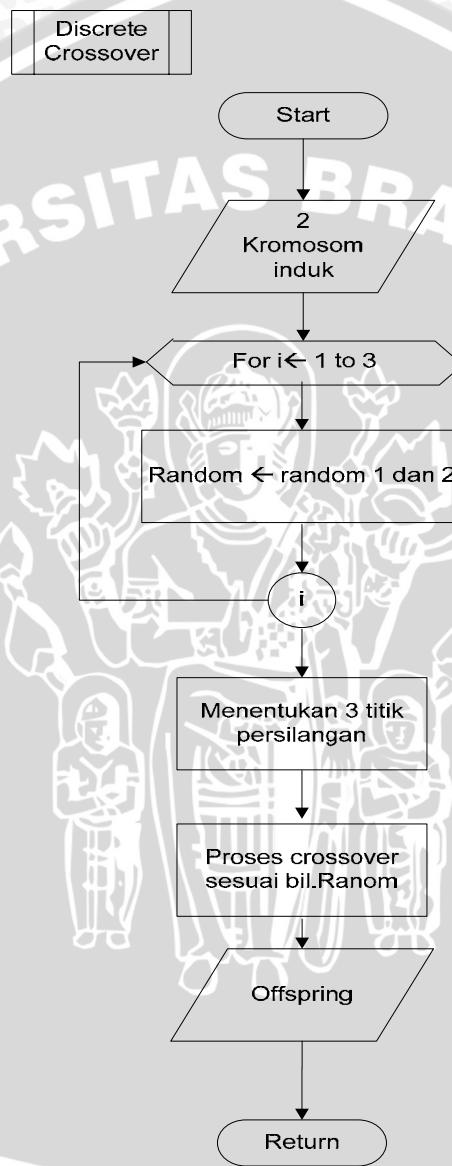
Gambar 3.4 Flowchart *Crossover*

Pada teknik ini, digunakan dua buah kromosom untuk membuat kromosom baru. Untuk menghasilkan kromosom baru, kromosom induk dipecah menjadi tiga bagian. Setelah itu bilangan acak dibangkitkan sebanyak tiga buah. Bilangan acak tersebut bernilai satu atau dua. Bilangan satu menyatakan induk yang pertama dan bilangan dua menyatakan induk yang kedua. Kemudian kromosom yang baru dibentuk sesuai dengan bilangan acak yang telah didapatkan. Contoh proses *discrete crossover* dapat dilihat pada gambar 3.5.



Gambar 3.5 Proses *Discrete Crossover*

Untuk diagram alir proses *Discrete Crossover* ditunjukkan pada gambar 3.6.



Gambar 3.6 Flowchart *Discrete Crossover*

3.2.5 Mutasi

Pada algoritma genetika, mutasi memainkan peran penting, yaitu pertama, menggantikan gen yang hilang dari populasi selama proses seleksi. Kedua, menyediakan gen yang tidak pernah ditampilkan pada populasi awal (sanjoyo, 2006). Jumlah individu yang mengalami proses mutasi tergantung pada jumlah probabilitas mutasi (P_m) yang ditentukan. Teknik mutasi yang digunakan adalah mutasi random.

Contoh proses mutasi dapat dilihat pada gambar 3.7.

Kromosom asal

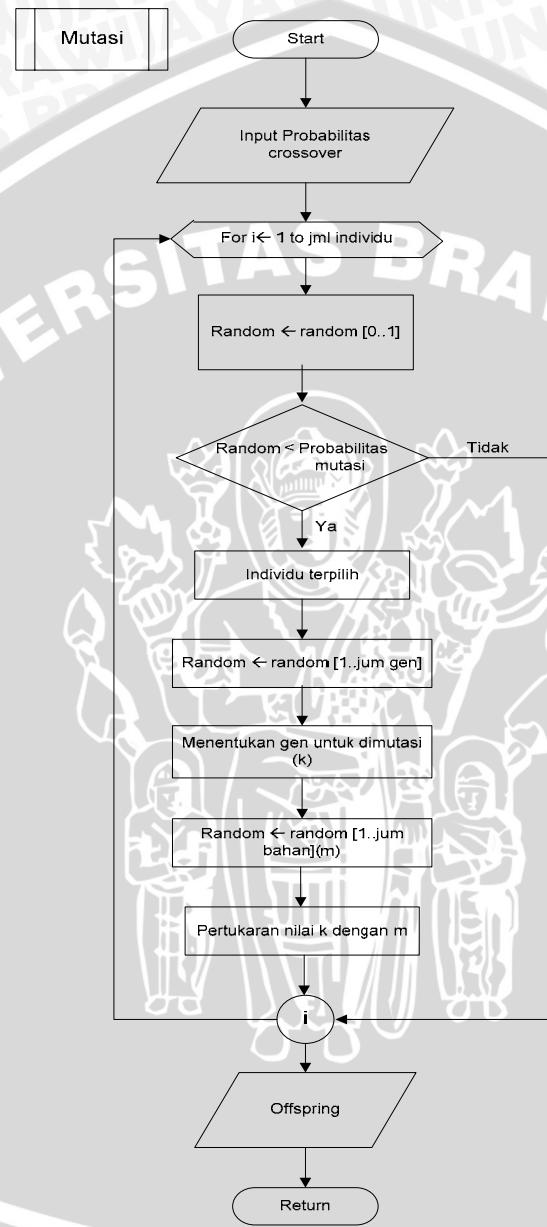
15	12	17	24	19	01	07	14	22	15
----	----	----	----	----	----	----	----	----	----

Kromosom hasil mutasi

15	12	17	24	19	01	13	14	22	15
----	----	----	----	----	----	----	----	----	----

Gambar 3.7 Proses Mutasi

Pada gen yang bernilai 07 ditukar dengan gen yang bernilai 13, nilai 13 merupakan nilai yang tidak pernah terpakai pada populasi awal. Untuk diagram alir proses mutasi ditunjukkan pada gambar 3.8.



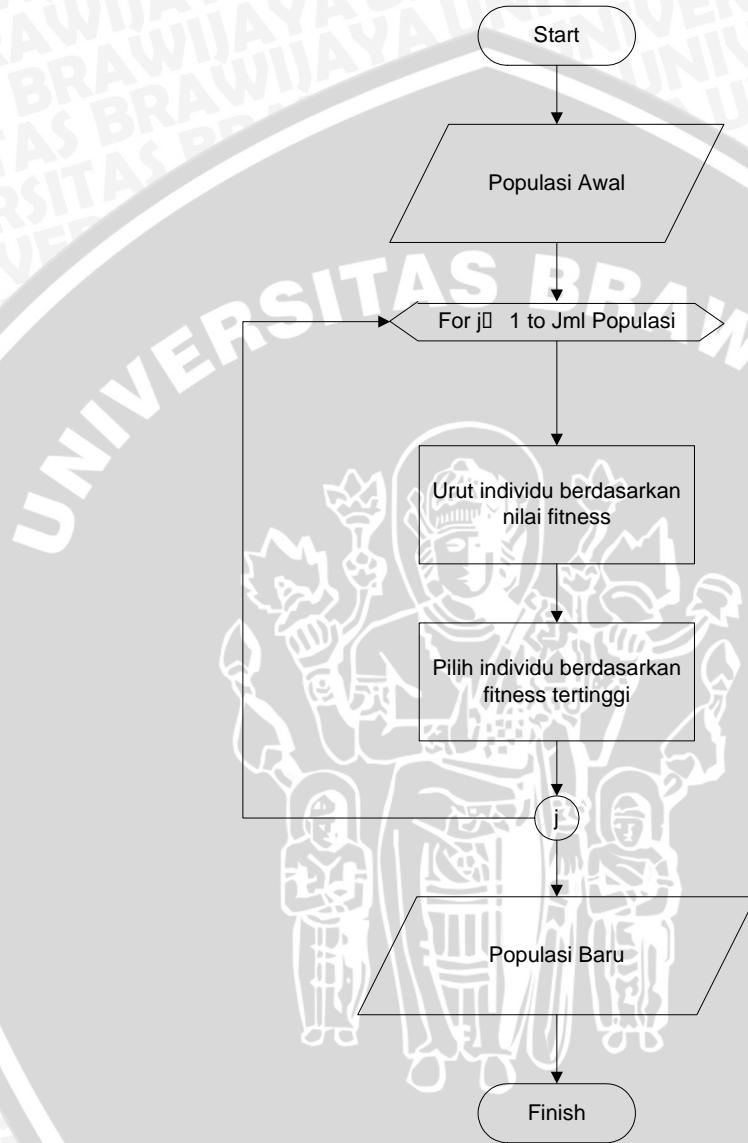
Gambar 3.8 Flowchart Mutasi

3.2.6 Seleksi

. Seleksi mempunyai tujuan untuk memberikan kesempatan reproduksi yang lebih besar bagi anggota populasi yang mempunyai nilai *fitness* terbaik. Metode seleksi yang digunakan dalam penyelesaian permasalahan penentuan komposisi pakan ikan adalah metode *Ranking Selection*.

Cara kerja metode *Ranking Selection* ini adalah semakin besar nilai fitness sebuah kromosom, maka sudah pasti untuk terpilih. Seleksi *Ranking* dimulai dari mengurutkan nilai fitness dari yang tertinggi sampai yang terendah dalam satu populasi, setelah itu individu dengan nilai fitness tertinggi akan terpilih sebagai induk, sedangkan individu dengan nilai fitness terendah akan dibuang. Untuk diagram alir proses seleksi ditunjukkan pada gambar 3.9.





Gambar 3.9 Flowchart Seleksi

3.3 Perhitungan Manual

3.3.1 Pembangkitan Populasi Awal

Membangkitkan populasi awal adalah proses membangkitkan sejumlah individu secara acak. Misalkan telah dibangkitkan populasi awal dengan jumlah bahan sebanyak 5, jumlah populasi sebanyak 4 dan berat pakan yang ingin dihasilkan 100kg sebagai berikut :

Individu 1 : 2 21 6 23 21

Individu 2 : 8 14 4 11 33

Individu 3 : 8 33 34 30 3

Individu 4 : 18 1 16 13 28

3.3.2 Contoh perhitungan nilai *fitness*

Perhitungan Fitness Individu 1 :

Individu 1: 2 21 6 23 21

Bobot sementara masing-masing bahan : 76, 4, 16, 72, 58.

Maka bobot sebenarnya: $(76/226)*100 = 33.628$

$$(4/226)*100 = 1.769$$

$$(16/226)*100 = 7.079$$

$$(72/226)*100 = 31.858$$

$$(58/226)*100 = 25.663$$

Jumlah Protein : 29.3019

Jumlah Lemak : 4.1159

Jumlah Karbohidrat : 1.0761

Jumlah Serat : 6.3717

Jumlah Abu : 4.3752

$$fitness = \frac{1}{((\text{Abs}(a - p)) + (\text{Abs}(b - q)) + (\text{Abs}(c - r)) + (\text{Abs}(d - s)) + (\text{Abs}(e - t)) + 1)}$$

$$fitness = \frac{1}{((\text{Abs}(30 - 29.3019)) + (\text{Abs}(18 - 4.1159)) + (\text{Abs}(17.5 - 1.0761)) + (\text{Abs}(6 - 6.3717)) + (\text{Abs}(8.5 - 4.3752)) + 1)}$$

$$fitness = 0.02739540 \ 04819651$$

Perhitungan Fitness Individu 2 :

Individu 2: 8 14 4 11 33

Bobot sementara masing-masing bahan : 8, 36, 49, 19, 15

Maka bobot sebenarnya: $(8/127)*100 = 6.299$

$$(36/127)*100 = 28.346$$

$$(49/127)*100 = 38.583$$

$$(19/127)*100 = 14.960$$

$$(15/127)*100 = 11.811$$

Jumlah Protein : 29.5135

Jumlah Lemak : 5.2657

Jumlah Karbohidrat : 28.3659

Jumlah Serat : 9.3070

Jumlah Abu : 7.0682

$$fitness = \frac{1}{((Abs(a - p)) + (Abs(b - q)) + (Abs(c - r)) + (Abs(d - s)) + (Abs(e - t)) + 1)}$$

$$fitness = \frac{1}{((Abs(30 - 29.5135)) + (Abs(18 - 5.2657)) + (Abs(17,5 - 28.3659)) + (Abs(6 - 9.3070)) + (Abs(8,5 - 7.0682)) + 1)}$$

$$fitness = 0.03352825 \ 48147366$$

Perhitungan Fitness Individu 3 :

Individu 3 : 8 33 34 30 3

Bobot sementara masing-masing bahan : 89, 64, 8, 94, 59

Maka bobot sebenarnya: $(89/314)*100 = 28.343$

$$(64/314)*100 = 20.382$$

$$(8/314)*100 = 2.547$$

$$(94/314)*100 = 29.936$$

$$(59/314)*100 = 18.789$$

Jumlah Protein : 33.0232

Jumlah Lemak : 9.7041

jumlah Karbohidrat : 3.8825

jumlah Serat : 6.5832

jumlah Abu : 12.5994

$$fitness = \frac{1}{((\text{Abs}(a - p)) + (\text{Abs}(b - q)) + (\text{Abs}(c - r)) + (\text{Abs}(d - s)) + (\text{Abs}(e - t)) + 1)}$$

$$fitness = \frac{1}{((\text{Abs}(30 - 33.0232)) + (\text{Abs}(18 - 9.7041)) + (\text{Abs}(17,5 - 3.8825)) + (\text{Abs}(6 - 6.5832)) + (\text{Abs}(8,5 - 12.5994)) + 1)}$$

$$fitness = 0.03265920\ 84406372$$

Perhitungan Fitness Individu 4 :

Individu 4: 18 1 16 13 28

Bobot sementara masing-masing bahan : 4, 84, 18, 65, 60

Maka bobot sebenarnya: $(4/231)*100 = 1.731$

$$(84/231)*100 = 36.363$$

$$(18/231)*100 = 7.792$$

$$(65/231)*100 = 28.138$$

$$(60/231)*100 = 25.974$$

Jumlah Protein : 45.7164

Jumlah Lemak : 5.5598

Jumlah Karbohidrat : 2.4197

Jumlah Serat : 3.23

Jumlah Abu : 4.3516

$$fitness = \frac{1}{((\text{Abs}(a - p)) + (\text{Abs}(b - q)) + (\text{Abs}(c - r)) + (\text{Abs}(d - s)) + (\text{Abs}(e - t)) + 1)}$$

$$fitness = \frac{1}{((\text{Abs}(30 - 45.7164)) + (\text{Abs}(18 - 5.5598)) + (\text{Abs}(17,5 - 2.4197)) + (\text{Abs}(6 - 3.23)) + (\text{Abs}(8,5 - 4.3516)) + 1)}$$

$$fitness = 0.01954834\ 02528252$$

3.3.3 Crossover

Proses *crossover* tergantung pada suatu parameter yaitu probabilitas *crossover*. Misalkan probabilitas *crossover* adalah 70%, itu diharapkan 70% individu dari populasi yang akan mengalami *crossover*. Langkah pertama yang harus dilakukan adalah

membangkitkan nilai random r [0..1]. jika r < 0,70 maka individu tersebut yang akan dikenai proses *crossover*. Misalkan nilai r yang terbentuk antara lain 0,05; 0,85; 0,98; 0,42. Dari nilai r yang terbentuk, diperoleh individu 1 dan individu 4 yang akan dikenakan proses *crossover*.

Langkah 1 :

Individu 1 (induk 1) :

2	21	6	23	21
---	----	---	----	----

Individu 4 (induk 2) :

18	1	16	13	28
----	---	----	----	----

Langkah 2 :

Bilangan Random :

Indeks	1 sampai 2	3	4 sampai 5
Bil. Random	2	2	1

Hasil *crossover*:

Child :

18	1	16	23	21
----	---	----	----	----

Perhitungan Fitness Individu Child :

Individu Child: 18 1 16 23 21

Bobot sementara masing-masing bahan : 85, 82, 35, 61, 30

Maka bobot sebenarnya: $(85/293)*100 = 29.01$

$$(82/293)*100 = 27.98$$

$$(35/293)*100 = 11.945$$

$$(61/293)*100 = 20.819$$

$$(30/293)*100 = 10.238$$

Jumlah Protein : 28.2845

Jumlah Lemak : 5.5693

Jumlah Karbohidrat : 8.6116

Jumlah Serat : 12.1786

Jumlah Abu : 6.3825

$$fitness = \frac{1}{((\text{Abs}(a - p)) + (\text{Abs}(b - q)) + (\text{Abs}(c - r)) + (\text{Abs}(d - s)) + (\text{Abs}(e - t))) + 1}$$

$$fitness = \frac{1}{((\text{Abs}(30 - 28.2645)) + (\text{Abs}(18 - 5.5693)) + (\text{Abs}(17,5 - 8.6116)) + (\text{Abs}(6 - 12.1786)) + (\text{Abs}(8,5 - 6.3825))) + 1}$$

$$fitness = 0.03093030\ 54710749$$

3.3.4 Mutasi

Proses mutasi tergantung pada suatu parameter yaitu probabilitas mutasi. Misalkan probabilitas mutasi adalah 25%, itu diharapkan 25% individu dari populasi yang akan mengalami mutasi. Langkah pertama adalah membangkitkan bilangan random $r [0..1]$. Jika nilai $r < 0,25$ maka individu tersebut akan dikenakan proses mutasi. Misalkan nilai r yang terbentuk adalah 0,73; 0,03; 0,4 dan 0,88, maka individu yang akan mengalami proses mutasi adalah individu 2. Berikut ini adalah langkah-langkah proses mutasi.

1. Pilih posisi gen (Random gen ke 1 sampai gen ke 10).
2. Diperoleh posisi gen yang akan dikenai mutasi (k).
3. Random nilai antara 1 sampai jumlah bahan(m) dimana nilai (m) belum pernah tampil pada populasi awal.
4. Menukar nilai k dengan nilai m .

Individu 2 :

8	14	4	11	33
---	----	---	-----------	----

Hasil mutasi :

Individu' 2:

8	14	4	27	33
---	----	---	-----------	----

Perhitungan Fitness Individu' 2:

Individu' 2 : 8 14 4 27 33

Bobot sementara masing-masing bahan : 30, 57, 71, 68, 96

Maka bobot sebenarnya: $(30/322)*100 = 9.316$

$$(57/322)*100 = 17.701$$

$$(71/322)*100 = 22.049$$

$$(68/322)*100 = 21.118$$

$$(96/322)*100 = 29.813$$

Jumlah Protein : 1

Jumlah Lemak : 5.4595

Jumlah Karbohidrat : 11.1538

Jumlah Serat : 12.1786

Jumlah Abu:11.7799

$$fitness = \frac{1}{((\text{Abs}(a - p)) + (\text{Abs}(b - q)) + (\text{Abs}(c - r)) + (\text{Abs}(d - s)) + (\text{Abs}(e - t))) + 1)}$$

$$fitness = \frac{1}{((\text{Abs}(30 - 1)) + (\text{Abs}(18 - 5.4595)) + (\text{Abs}(17,5 - 11.1538)) + (\text{Abs}(6 - 12.1786)) + (\text{Abs}(8,5 - 11.7799))) + 1)}$$

$$fitness = 0.01713796$$

Setelah proses *crossover* dan mutasi maka terbentuk individu-individu baru seperti pada tabel 3.1. Individu-individu yang terkumpul adalah individu awal, individu hasil crossover dan individu hasil mutasi. Padahal yang akan diproses adalah individu sejumlah populasi awal, sehingga perlu dilakukan seleksi untuk menentukan individu mana yang akan digunakan untuk proses genetika berikutnya.

Tabel 3.1 Populasi Setelah *Crossover* Dan Mutasi

Indeks	Susunan Komposisi Bahan Pakan
Individu 1	2 21 6 23 21
Individu 2	8 14 4 11 33
Individu 3	8 33 34 30 3
Individu 4	18 1 16 13 28
Child	18 1 16 23 21
Individu' 2	8 14 4 27 33

3.3.5 Seleksi

Individu-individu yang terkumpul akan diseleksi dengan menggunakan metode *Ranking Selection*. Cara kerja metode ini adalah individu yang memiliki *fitness* lebih besar akan mempunyai peluang yang lebih besar pula untuk terpilih. Berikut ini merupakan langkah-langkah seleksi dengan menggunakan metode *Ranking Selection*.

Tabel 3.2 Individu dan fitnessnya.

Indeks	Fitness
Individu 1	0.0273954004819651
Individu 2	0.0335282548147366
Individu 3	0.0326592084406372
Individu 4	0.0195483402528252
Child	0.0309303054710749
Individu' 2	0.01713796

Langkah berikutnya adalah mengurutkan individu berdasarkan besar *fitness*. Hasil pengurutan dapat dilihat pada tabel 3.3.

Tabel 3.3 Hasil pengurutan individu berdasarkan besar fitnessnya.

Indeks	Fitness
Individu 2	0.0335282548147366
Individu 3	0.0326592084406372
Child	0.0309303054710749
Individu 1	0.0273954004819651
Individu 4	0.0195483402528252
Individu' 2	0.01713796

Maka dari hasil pengurutan diatas individu yang akan dipilih adalah individu yang memiliki nilai *fitness* tertinggi. Individu yang dipilih sebanyak populasi awal yaitu :

Individu 2 : 8 14 4 11 33

Individu 3 : 8 33 34 30 3

Child : 18 1 16 23 21

Individu 1 : 2 21 6 23 21

Individu yang tidak terpilih akan dibuang sedangkan individu-individu yang terpilih akan melanjutkan proses genetika selanjutnya. Individu-individu yang terbentuk ini disebut populasi generasi ke-1.

3.4 Rancangan Uji Coba

Uji coba sistem untuk pembentukan komposisi pakan ikan ini akan melakukan evaluasi terhadap hasil yang diperoleh. Pada pengujian menggunakan algoritma genetika akan diberikan nilai yang berbeda-beda pada parameter genetika, yaitu nilai probabilitas *crossover* dan probabilitas mutasinya.

3.4.1 Tujuan Uji Coba

Tujuan dari uji coba ini adalah :

- ▶ Untuk mengetahui hubungan peluang *crossover* dan peluang mutasi dengan hasil yang didapat.
- ▶ Untuk mengetahui tingkat konvergensi.
- ▶ Untuk mengetahui komposisi bahan pakan yang dibutuhkan untuk memperoleh hasil yang optimal.

3.4.2 Skenario Uji Coba

Percobaan-percobaan yang dilakukan menggunakan algoritma genetika. Percobaan-percobaan tersebut dilakukan dengan parameter-parameter sebagai berikut :

1. Jumlah Bahan
2. Peluang *Crossover*
3. Peluang Mutasi
4. Jumlah Iterasi

Probabilitas *crossover* dan mutasi yang akan diujikan adalah 10 %, 30%, 50%, 70% dan 90%. Masing-masing pengujian akan dilakukan pada populasi yang memiliki 15 individu dan sebanyak 1.000 generasi dengan jumlah bahan sebanyak 10 dan berat pakan yang akan dihasilkan sebanyak 100 kg. Setiap pengujian dilakukan untuk mengetahui perubahan nilai *fitness* dan generasi konvergensi sehingga diketahui pada nilai probabilitas *crossover* dan mutasi ke berapa didapatkan hasil yang lebih optimal.

Tabel 3.4- Tabel Percobaan nilai *fitness* dengan Peluang *Crossover* (Pc) dan Peluang Mutasi (Pm) yang berbeda.

Prob. CO	Prob. Mutasi				
	10 %	30 %	50 %	70 %	90 %
10 %					
30 %					
50 %					
70%					
90 %					

Tabel 3.5 - Tabel Percobaan konvergensi dengan Peluang Crossover (Pc) dan Peluang Mutasi (Pm) yang berbeda.

Prob. CO	Prob. Mutasi				
	10 %	30 %	50 %	70 %	90 %
10 %					
30 %					
50 %					
70%					
90 %					

Tabel 3.6 - Tabel Percobaan nilai *fitness* dengan komposisi bahan yang berbeda.

ID Bahan	Individu Terbaik	Fitness

BAB IV

IMPLEMENTASI DAN PEMBAHASAN

4.1 Deskripsi Sistem

Program aplikasi ini merupakan program yang digunakan untuk menyelesaikan permasalahan penentuan komposisi pakan ikan. Jika ada sejumlah bahan yang ingin dikombinasikan menjadi pakan ikan dengan komposisi nutrisi tertentu dan bahan-bahan tersebut memiliki komposisi nutrisi yang berbeda, seorang formulator dapat menentukan batasan persentase bahan dan jumlah total masing-masing nutrisi dengan algoritma genetika. Algoritma genetika diharapkan dapat menemukan solusi yang memenuhi batasan-batasan tersebut dengan pemilihan komposisi yang dapat memenuhi kebutuhan standar gizi untuk ikan.

4.2 Sumber Data

Sumber data komposisi bahan pakan yang digunakan pada aplikasi ini diambil dari buku *Budi Daya Ikan di Perairan Umum* karangan Ir. Cahyono Bambang, 2001 dan buku *Membuat Pakan Ikan Konsumsi* karangan Khairuman, 2002.

4.3 Lingkungan Implementasi

Proses implementasi merupakan tahapan penerapan rancangan pada bahasa pemrograman yang dapat dimengerti oleh komputer. Lingkungan implementasi yang akan dijelaskan pada bab ini meliputi lingkungan implementasi perangkat keras dan perangkat lunak.

4.3.1 Lingkungan Perangkat Keras

Perangkat keras yang digunakan dalam pembuatan sistem penentuan komposisi pakan ikan menggunakan algoritma genetika ini adalah sebagai berikut :

1. Processor Intel Pentium Core 2 Duo 2,10GHz.
2. RAM 2,00 GB
3. Harddisk dengan kapasitas 250 GB
4. Monitor 14'
5. Keyboard
6. Mouse

Perangkat-perangkat keras ini digunakan sebagai pendukung dalam penelitian ini.

4.3.2 Lingkungan Perangkat Lunak

Perangkat lunak yang digunakan dalam pembuatan sistem penentuan komposisi pakan ikan menggunakan algoritma genetika ini adalah sebagai berikut :

1. Sistem Operasi Microsoft Windows XP Professional Edition Service Pack 3.
2. Borland Delphi 7 sebagai software development dalam implementasi rancangan sistem.
3. Microsoft Acces sebagai media penyimpanan data.

4.4 Implementasi Program Penentuan Komposisi Pakan Ikan Menggunakan Algoritma Genetika

Berdasarkan perancangan proses yang terdapat pada bab tiga, maka pada subbab ini akan dijelaskan implementasi proses-proses tersebut.

4.4.1 Struktur Data

Struktur data yang digunakan dalam implementasi penentuan komposisi pakan ikan menggunakan algoritma genetika ditampilkan pada *sourcecode 4.1* dibawah ini :

```
Type  
  
TKromosom = array [1..jumlah_bahan] of integer;  
TIndividu = record  
    Kromosom: TKromosom;  
    Fitness: real;  
    bobot:array[1..34]of Real;  
end;  
TPopulasi = array [1..150] of TIndividu;  
  
Var  
  
    bahan1:array[1..100,1..5]of String;  
    indi:Tindividu;  
    BanyakPopulasi,BanyakIterasi:integer;  
    pop: TPopulasi;  
    PanjangKromosom :integer;
```

```

jumProtein, jumLemak, jumKarbohidrat,
jumSerat, jumAbu:Real;
child:TIndividu;
cp:array [1..2] of integer;
diskrit:array[1..3]of Integer;

```

Sourcecode 4.1 Struktur data.

Keterangan struktur data dapat dilihat pada tabel 4.1

Tabel 4.1 Struktur data dalam algoritma

Struktur Data	Keterangan
TKromosom	Tipe data bertipe <i>array</i> untuk menampung sejumlah bahan pakan.
TIndividu	Tipe data <i>record</i> untuk menyimpan data individu, yang terdiri dari kromosom, <i>fitness</i> dan bobot
individu	Variabel bertipe <i>array</i> untuk menyimpan kromosom dengan masing-masing record bertipe Trute
BanyakPopulasi	Variabel bertipe <i>integer</i> untuk menyimpan total populasi
Bahan1	Variabel bertipe <i>array</i> 2 dimensi untuk menyimpan nilai nutrisi
Indi	Variabel bertipe TIndividu untuk menyimpan data individu
jumProtein	Variabel bertipe <i>real</i> untuk menyimpan total protein bahan
jumLemak	Variabel bertipe <i>real</i> untuk menyimpan total lemak bahan
jumKarbohidrat	Variabel bertipe <i>real</i> untuk menyimpan total karbohidrat bahan
jumSerat	Variabel bertipe <i>real</i> untuk menyimpan total serat bahan
jumAbu	Variabel bertipe <i>real</i> untuk menyimpan total abu bahan
TPopulasi	Tipe data berupa <i>array</i> yang bertipe TIndividu untuk menyimpan data populasi
bobot	Tipe data berupa <i>array</i> yang bertipe <i>real</i> untuk menyimpan bobot (berat) setiap bahan pakan

4.4.2 Inisialisasi Kromosom

Pada prosedur inisialisasi kromosom panjang *array* dibuat dinamis karena jumlah bahan yang akan dikomposisikan berbeda. Proses inisialisasi kromosom dapat dilihat pada *sourcecode* 4.2.

```
Randomize;  
  
PanjangKromosom:= strToInt(Form1.Edit6.Text);  
  
for j:=1 to PanjangKromosom do  
begin  
    Indi.Kromosom[j] :=random(jumlah_bahan)+1;  
end;  
  
Ind:=Indi;  
end;
```

Sourcecode 4.2 Inisialisasi kromosom

4.4.3 Perhitungan Fitness

Perhitungan *fitness* dilakukan pada masing-masing individu. Proses perhitungan nilai *fitness* kromosom dapat dilihat pada *sourcecode* 4.3.

```
jumProtein:=0;  
jumLemak:=0;  
jumKarbohidrat:=0;  
jumSerat:=0;  
jumAbu:=0;  
  
bobot_total:=0;  
  
for i:=1 to PanjangKromosom do  
begin  
    bobot[i]:=random(berat_pakan)+1;  
    bobot_total:=bobot_total+bobot[i];  
end;  
  
for i:=1 to PanjangKromosom do  
begin  
    bobot[i]:=(bobot[i]/bobot_total)*berat_pakan;  
    individu.bobot[i]:=bobot[i];  
end;  
  
j:=1;  
while j<=PanjangKromosom do
```

```

begin
    a:=StrToFloat(bahan1[Indi.Kromosom[j],1]);
    b:=StrToFloat(bahan1[Indi.Kromosom[j],2]);
    c:=StrToFloat(bahan1[Indi.Kromosom[j],3]);
    d:=StrToFloat(bahan1[Indi.Kromosom[j],4]);
    e:=StrToFloat(bahan1[Indi.Kromosom[j],5]);

    hasil_protein[j]:= (bobot[j]*a)/100;
    hasil_lemak[j]:= (bobot[j]*b)/100;
    hasil_karbohidrat[j]:= (bobot[j]*c)/100;
    hasil_serat[j]:= (bobot[j]*d)/100;
    hasil_abu[j]:= (bobot[j]*e)/100;

    nutrisi_protein:=hasil_protein[j];
    nutrisi_lemak:=hasil_lemak[j];
    nutrisi_karbohidrat:=hasil_karbohidrat[j];
    nutrisi_serat:=hasil_serat[j];
    nutrisi_abu:=hasil_abu[j];

    inc(j);
    jumProtein:=jumProtein+nutrisi_protein;
    jumLemak:=jumLemak+nutrisi_lemak;
    jumKarbohidrat:=jumKarbohidrat+nutrisi_karbo
    hidrat;
    jumSerat:=jumSerat+nutrisi_serat;
    jumAbu:=jumAbu+nutrisi_abu;
end;

fitness:=1/( (Abs(ProStd-jumProtein))+(Abs(LemStd-
    jumLemak))+(Abs(KarStd-
    jumKarbohidrat))+(Abs(SerStd-
    jumSerat))+(Abs(AbuStd-jumAbu))+1);

Individu.Fitness:=fitness;

end;

```

Sourcecode 4.3 Perhitungan nilai fitness

4.4.4 Proses Crossover

Pada proses *crossover* terdiri dari 2 prosedur. Prosedur yang pertama adalah pemilihan individu-individu dari populasi untuk dijadikan sebagai *parent*, dan prosedur kedua adalah proses *discrete crossover*.

4.4.4.1 Prosedur Pilih Parent Crossover

Prosedur Pilih Parent Crossover adalah pemilihan individu-individu dari populasi untuk dijadikan sebagai *parent* sesuai dengan nilai probabilitas *crossover* yang telah ditentukan. Pemilihan individu yang akan di-*crossover* (*parent*) dapat dilihat pada *source code* 4.4.

```
BanyakPopulasi:=strToInt(Form1.Edit3.Text);
Randomize;
PanjangKromosom:=StrToInt(Edit6.Text);
prob:=strToInt(Edit5.Text);
probCO:=prob/100;

P:=1;
for i:=1 to BanyakPopulasi do
begin
    NilaiRand:=random;
    if(nilaiRand<probCO)then
    begin
        parent[p]:=pop[i];
        p:=p+1;
    end;
end;
BanyakPopulasi_CO:=p-1;
```

Sourcecode 4.4 Pemilihan induk yang akan di-*crossover*

4.4.4.2 Prosedur Discrete Crossover

Metode *crossover* yang digunakan adalah metode *discrete crossover*. Pada prosedur ini terdapat 3 tahapan. Tahapan yang pertama adalah kromosom dari individu yang terpilih akan dibagi menjadi 3 bagian. *Sourcecode* untuk membagi kromosom menjadi 3 bagian dapat dilihat pada *sourcecode* 4.5.

```

if BanyakPopulasi_CO mod 2<>0 then
    BanyakPopulasi_CO:=BanyakPopulasi_CO-1;

p:=1;
while p<=BanyakPopulasi_CO do
begin
    BanyakPopulasi:=BanyakPopulasi+1;
    cp[1]:=Random(PanjangKromosom-1)+1;
repeat
    cp[2]:=random(PanjangKromosom-1)+1;
until
    cp[2]<>cp[1];

```

Sourcecode 4.5 Membagi kromosom

Tahap berikutnya adalah membangkitkan bilangan random 1 dan 2 sebanyak 3 bilangan. Proses pembangkitan bilangan random ini dapat dilihat pada *sourcecode 4.6*.

```

for i:=1 to 3 do
begin
    diskrit[i]:=random(2)+1;
end;

for i:=1 to 2 do
begin
    for j:=i+1 to 2 do
begin
    if cp[i]>cp[j] then
begin
        temp:=cp[i];
        cp[i]:=cp[j];
        cp[j]:=temp;
    end;
end;
end;

```

Sourcecode 4.6 Proses pembangkitan bilangan random

Langkah terakhir adalah menukar kedua kromosom *parent* sesuai dengan bilangan *random* dan pembagian kromosom 3 bagian. Proses *crossover* dapat dilihat pada *sourcecode 4.7*.

```

j:=1;
for i:=1 to PanjangKromosom do
begin
    if ((i>=1)and(i<=cp[1]))then
begin

```

```

if diskrit[1]=1 then
    child.Kromosom[i]:=parent[p].Kromos
    om[i]
else

    child.Kromosom[i]:=parent[p+1].Krom
    osom[i];
end;

if (i>cp[1])and(i<=cp[2])then
begin
    if diskrit[2]=1 then

        child.Kromosom[i]:=parent[p].Kromos
        om[i]
    else

        child.Kromosom[i]:=parent[p+1].Kromoso
        m[i];
end;

if (i>cp[2])and(i<=PanjangKromosom)then
begin
    if diskrit[3]=1 then

        child.Kromosom[i]:=parent[p].Krom
        osom[i]
    else

        child.Kromosom[i]:=parent[p+1].Kr
        omosom[i];
end;
end;

```

Sourcecode 4.7 Proses crossover

4.4.5 Proses Mutasi

Pada proses mutasi juga terdiri dari 2 prosedur. Prosedur pertama hampir sama dengan crossover yaitu pemilihan induk yang akan dimutasi sedangkan prosedur kedua adalah proses mutasi.

4.4.5.1 Prosedur Pilih Induk Mutan

Langkah awal proses mutasi sama dengan langkah awal proses crossover, yaitu memilih individu yang akan dilakukan proses mutasi

sesuai dengan nilai probabilitas mutasi yang ditentukan. Proses pemilihan individu pada populasi ditunjukkan pada *sourcecode* 4.8.

```
BanyakPopulasi:=strtoint(Form1.Edit3.Text);
Randomize;
prob:=strtoint(Edit7.Text);
probMut:=prob/100;
PanjangKromosom:=StrToInt(Edit6.Text);

a:=1;
for i:=1 to BanyakPopulasi do
begin
    rand:=random;
    if(rand<probMut)then
    begin
        mutant[a]:=pop[i];
        a:=a+1;
    end;
end;
```

Sourcecode 4.8 Proses pemilihan induk yang akan dimutasi

4.4.5.2 Prosedur Mutasi

Langkah berikutnya yaitu menentukan gen yang akan mengalami mutasi secara acak, setelah itu menentukan nilai yang belum pernah tampil pada populasi awal, jika semua nilai sudah pernah ditampilkan maka akan diacak nilai antara 1 sampai 24 (banyak bahan yang tersedia). Kemudian dilakukan pertukaran antara kedua gen yang akan mengalami mutasi dengan bilangan acak tersebut. Proses mutasi dapat dilihat pada *sourcecode* 4.9.

```
BanyakPopulasi_Mut:=a-1;

if BanyakPopulasi_Mut<>0 then
begin
    for a:=1 to BanyakPopulasi_Mut do
begin
    BanyakPopulasi:=BanyakPopulasi+1;
    indexMut:=random(PanjangKromosom)+1;
    genMutant:=random(24)+1;
    i:=1;
    while i<=PanjangKromosom do
begin
    if genMutant<>pop[a].Kromosom[i] then
    begin
        i:=i+1;
    end
```

```
        else
        begin
            genMutan:=random(24)+1;
            i:=1;
        end;
    end;

mutant[a].Kromosom[indexMut]:=genMutan ;
```

Sourcecode 4.9 Proses mutasi

Setelah dilakukan proses *crossover* dan mutasi maka terbentuklah individu-individu baru. Individu-individu yang terkumpul adalah individu-individu dari populasi awal, individu hasil *crossover* dan individu hasil mutasi. Padahal yang akan dievaluasi adalah individu sejumlah populasi awal, sehingga perlu dilakukan seleksi untuk menentukan individu mana yang akan digunakan untuk proses genetika berikutnya.

4.4.6 Proses Seleksi

Proses seleksi atau pemilihan individu dilakukan dengan metode *Ranking Selection*. Cara kerja metode *Ranking Selection* ini adalah semakin besar nilai fitness sebuah kromosom, maka sudah pasti untuk terpilih. Seleksi *Ranking* dimulai dari mengurutkan nilai fitness dari yang tertinggi sampai yang terendah dalam satu populasi, setelah itu individu dengan nilai fitness tertinggi akan terpilih sebagai induk, sedangkan individu dengan nilai fitness terendah akan dibuang. Proses *Ranking Selection* dapat dilihat pada sourcecode 4.10.

```

for i:=1 to BanyakPopulasi do
begin
    for j:=i+1 to BanyakPopulasi do
    begin
        if pop[i].Fitness<pop[j].Fitness then
        begin
            tempIndi:=pop[i];
            pop[i]:=pop[j];
            pop[j]:=tempIndi;
        end;
    end;
end;

for i:=1 to PopulasiAwal do
begin
    pop[i]:=pop[i];
end;

```

Sourcecode 4.10 Proses Ranking Selection

4.5 Implementasi Database

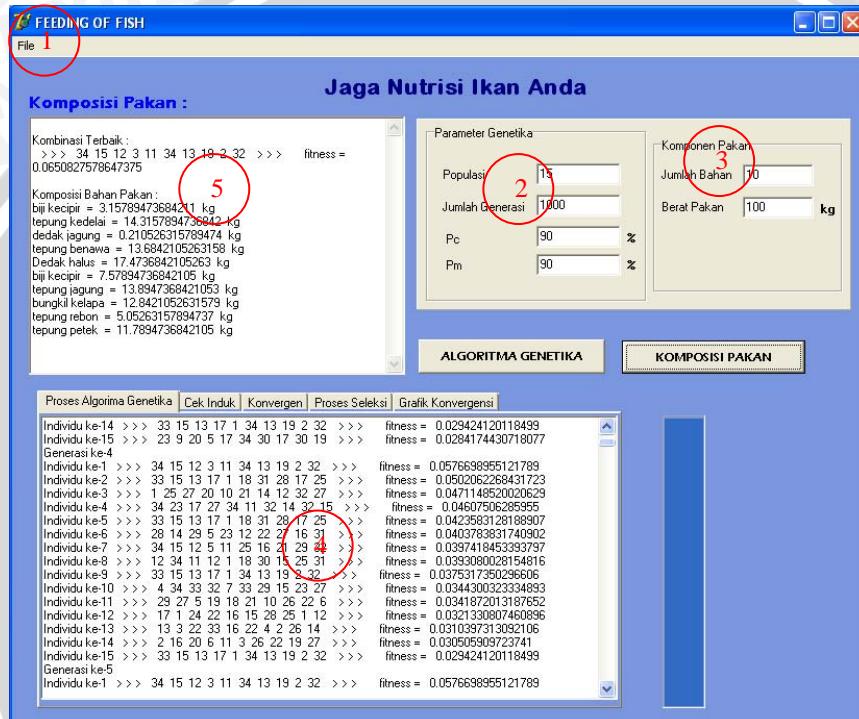
Database yang digunakan adalah Microsoft Access dimana hanya terdapat 1 tabel yaitu tabel yang digunakan untuk menyimpan data bahan pakan yang dapat dilihat pada tabel 4.1.

Tabel 4.2 Tabel Bahan

Field	Type	Null	Ket.
ID	Number	No	Id tiap bahan
Nama_Bahan	Text	No	Nama bahan pakan ikan
Protein	Number	No	Kandungan protein
Lemak	Number	No	Kandungan lemak
Karbohidrat	Number	No	Kandungan karbohidrat
Serat	Number	No	Kandungan serat
Abu	Number	No	Kandungan abu

4.6 Implementasi Antarmuka Penentuan Komposisi Pakan Ikan Menggunakan Algoritma Genetika

Tampilan utama dari program *Penentuan Komposisi Pakan Ikan* menggunakan algoritma genetika dapat dilihat pada gambar 4.1.



Gambar 4.1 Tampilan utama program

Program aplikasi diatas terdiri dari beberapa bagian, yang akan dijelaskan sebagai berikut :

1. Main Menu ‘File’ yang terdiri dari 2 fungsi pengolahan data bahan pakan, antara lain :

- a. Input data bahan pakan.

Menu file ini berfungsi untuk menginput data bahan pakan atau untuk menambah data bahan pakan. Semua data yang dimasukkan akan disimpan ke dalam

database Microsoft Access. Tampilan form input data terdapat pada gambar 4.2.

The screenshot shows a Microsoft Access form titled "Input Data Pakan". The form has a blue header bar with the title and standard window controls. The main area is titled "Data Bahan Pakan" and contains seven text input fields with percentage signs (%). The fields are labeled: "ID Bahan" (containing "37"), "Nama Bahan", "Protein", "Lemak", "Karbohidrat", "Serat", and "Abu". To the right of each percentage field is a small "%". At the bottom right are two buttons: a green "Simpan" button with a checkmark icon and a red "Batal" button with a cross icon.

Gambar 4.2 Tampilan form input data bahan

- b. Lihat Data Bahan
- c. Edit Data Bahan

Menu file ini berfungsi untuk mengedit data bahan pakan. Tampilan form input data terdapat pada gambar 4.3.

The screenshot shows a Microsoft Access form titled "Edit Data Pakan". The form has a blue header bar with the title and standard window controls. The main area is titled "Edit Data Bahan Pakan" and contains seven text input fields with percentage signs (%). The fields are labeled: "ID Bahan" (containing "1"), "Nama Bahan", "Protein", "Lemak", "Karbohidrat", "Serat", and "Abu". To the right of each percentage field is a small "%". Above the "Abu" field is a "Tampil" button. At the bottom right are two buttons: a green "Ubah" button with a checkmark icon and a red "Batal" button with a cross icon.

Gambar 4.3 Tampilan form edit data bahan

2. Parameter Algoritma Genetika
Parameter algoritma genetika terdiri dari :
 - a. Jumlah populasi
 - b. Jumlah generasi
 - c. Nilai probabilitas *crossover*
 - d. Nilai probabilitas mutasi
3. Komponen Pakan
 - a. Jumlah bahan yang akan dikomposisikan
 - b. Berat pakan yang ingin dihasilkan
4. Tabsheet untuk menampilkan hasil penyelesaian Penentuan Komposisi Pakan Ikan Menggunakan Algoritma Genetika.
5. Hasil komposisi pakan.

4.7 Penerapan Aplikasi Penentuan Komposisi Pakan Ikan Menggunakan Algoritma Genetika

Aplikasi diterapkan dengan memasukkan data-data yang diperlukan, yaitu Parameter genetika dan komponen pakan yang dapat dilihat pada tabel 4.2 dan 4.3 :

Tabel 4.3 Data parameter genetika

Data	
Jumlah Populasi	15
Jumlah Generasi	1000
Probabilitas crossover (Pc)	40
Probabilitas mutasi (Pm)	20

Tabel 4.4 Data Komponen Pakan

Data	
Jumlah Bahan	10
Berat pakan	100 kg

Pada tabel 4.2 dapat dilihat bahwa terdapat 20 individu dalam satu populasi dan 1.000 generasi. Probabilitas crossover adalah 40% dan probabilitas mutasi sebanyak 20%. Sehingga

diharapkan akan terdapat 40% individu yang akan mengalami crossover dan 20% individu yang Mengalami proses mutasi.

Pada tabel 4.3 dapat dilihat bahwa jumlah bahan yang akan dikomposisikan sebanyak 10 dan berat pakan yang ingin dihasilkan adalah 100 kg. Tampilan implementasi dapat dilihat pada gambar 4.4.



Gambar 4.4 Implementasi program

Pada gambar 4.4 dapat dilihat proses genetika pada contoh kasus, dimana terdapat 15 individu dalam setiap generasi. Pada masing-masing individu memiliki komposisi yang berbeda dengan nilai *fitness* yang berbeda sesuai dengan perhitungan.

Setelah dilakukan proses genetika, diperoleh hasil individu yang terbaik adalah individu dengan *fitness* 0.0649328633235146 dan kromosom 27 17 12 1 18 2 1 12 24 7. Dari kromosom ini maka diperoleh bahan-bahan dan bobotnya sebagai berikut:

➤ tepung ketan	= 9.65189873417722 kg
➤ bungkil kacang tanah	= 4.27215189873418 kg
➤ dedak jagung	= 14.2405063291139 kg
➤ tepung ikan	= 9.96835443037975 kg
➤ bungkil biji kapuk	= 15.3481012658228 kg
➤ tepung rebon	= 6.17088607594937 kg
➤ tepung ikan	= 6.9620253164557 kg
➤ dedak jagung	= 7.43670886075949 kg
➤ daun turi	= 11.0759493670886 kg
➤ tepung bekicot	= 14.873417721519 kg

Jika ada hasil bahan-bahan tersebut yang muncul 2 kali maka hal itu mengindikasikan bahwa bahan pakan tersebut dipakai dalam 2 porsi.

4.8 Analisa Hasil

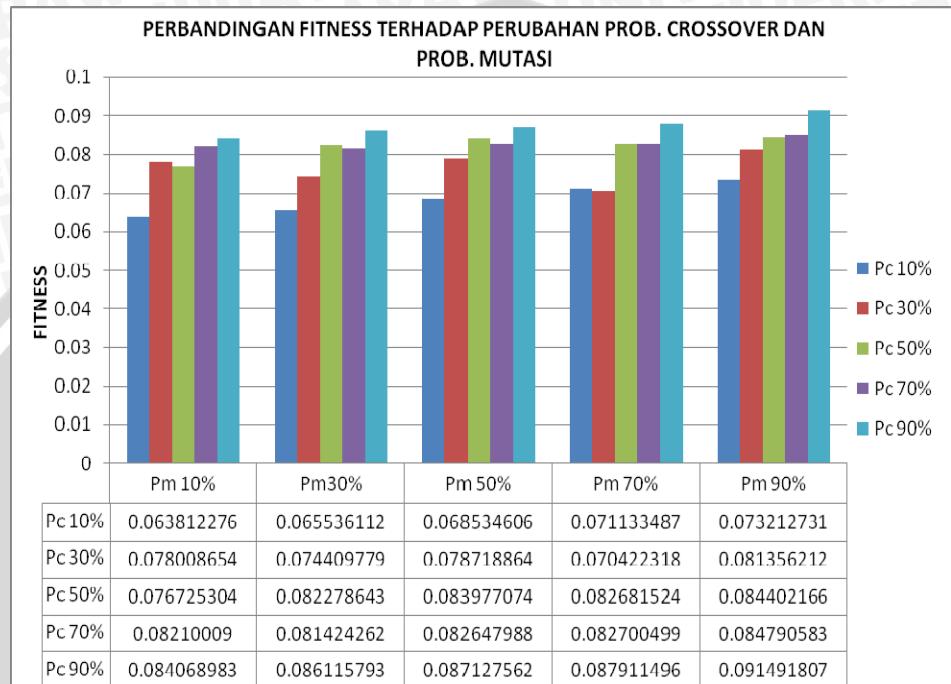
Perubahan nilai fitness setiap individu dari inisialisasi awal hingga generasi terakhir dipengaruhi oleh jumlah generasi, probabilitas *crossover*, probabilitas mutasi dan jumlah bahan. Pada penelitian ini, akan dianalisa pengaruh jumlah bahan, probabilitas *crossover* dan mutasi terhadap nilai fitness dan konvergensi.

Pengujian dilakukan pada probabilitas *crossover* 10%, 30%, 50%, 70% dan 90%, dan probabilitas mutasi 10%, 30%, 50%, 70% dan 90% juga. Uji coba dilakukan sebanyak 5 kali percobaan untuk masing-masing probabilitas. Berikut ini hasil uji coba untuk masing-masing probabilitas *crossover* dan mutasi.

Tabel 4.5 Nilai *Fitness* pada probabilitas *crossover* dan probabilitas mutasi.

Prob. Crossover	Prob. Mutasi				
	10 %	30%	50 %	70 %	90 %
10 %	0.063812276	0.065536112	0.068534606	0.071133487	0.073212731
30 %	0.078008654	0.074409779	0.078718864	0.070422318	0.081356212
50 %	0.076725304	0.082278643	0.083977074	0.082681524	0.084402166
70%	0.08210009	0.081424262	0.082647988	0.082700499	0.084790583
90 %	0.084068983	0.086115793	0.087127562	0.087911496	0.091491807

Perbandingan nilai rata-rata *fitness* dari masing-masing probabilitas *crossover* dan mutasi dapat dilihat pada grafik 4.1.



Grafik 4.1 Perbandingan *Fitness* dengan Probabilitas Mutasi dan Probabilitas *Crossover*

Berdasarkan hasil uji coba diketahui bahwa perubahan nilai probabilitas *crossover* dan mutasi memberikan perubahan terhadap nilai *fitness*-nya. Pada grafik 4.1 diketahui bahwa peningkatan nilai probabilitas *crossover* dan mutasi membuat nilai *fitness* semakin tinggi.

Misalkan pada keadaan probabilitas *crossover* 10% dan probabilitas mutasi 10% diperoleh nilai *fitness* 0.063812276, sedangkan pada keadaan probabilitas *crossover* 50% dan probabilitas mutasi 50% diperoleh nilai *fitness* 0.083977074. Hal ini menunjukkan bahwa peningkatan nilai probabilitas *crossover* dan probabilitas mutasi menyebabkan kecenderungan peningkatan nilai *fitness*, karena semakin tinggi nilai probabilitas *crossover* dan probabilitas mutasi yang ditetapkan maka semakin tinggi peluang terbentuk

variasi individu yang lebih baik dan nilai *fitness* yang dihasilkan semakin tinggi.

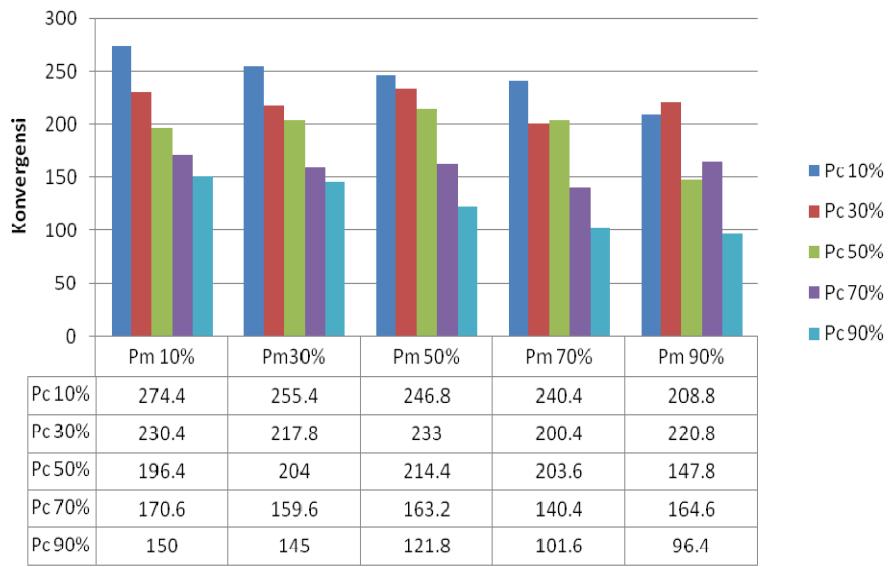
Probabilitas *crossover* dan mutasi juga berpengaruh pada generasi konvergensi. Perubahan generasi konvergensi dari masing-masing probabilitas crossover dan probabilitas mutasi dapat dilihat pada tabel 4.5.

Tabel 4.6 Generasi konvergensi pada probabilitas *crossover* dan probabilitas mutasi.

Prob. CO	Prob. Mutasi				
	10 %	30%	50 %	70 %	90 %
10 %	274.4	255.4	246.8	240.4	208.8
30 %	230.4	217.8	233	200.4	220.8
50 %	196.4	204	214.4	203.6	147.8
70%	170.6	159.6	163.2	140.4	164.6
90 %	150	145	121.8	101.6	96.4

Perbandingan generasi konvergensi dari masing-masing probabilitas *crossover* dan mutasi dapat dilihat pada grafik 4.2.

PERBANDINGAN KONVERGENSITERHADAP PERUBAHAN PROB. CROSSOVER DAN PROB. MUTASI



Grafik 4.2 Perbandingan Generasi Konvergensi dengan Probabilitas Mutasi dan Probabilitas *Crossover*

Dari uji coba diatas diperoleh hasil bahwa perubahan nilai probabilitas *crossover* atau mutasi memberikan perubahan terhadap generasi konvergensi. Misalkan pada keadaan probabilitas *crossover* 10% dan probabilitas mutasi 10% diperoleh konvergensi pada generasi 274.4, sedangkan pada keadaan probabilitas *crossover* 50% dan probabilitas mutasi 50% diperoleh konvergensi pada generasi 214.4.

Hal ini menunjukkan bahwa peningkatan nilai probabilitas *crossover* dan mutasi mengakibatkan nilai generasi konvergensi semakin rendah atau diperoleh konvergensi pada generasi yang lebih awal. Hal ini disebabkan karena nilai probabilitas *crossover* atau mutasi yang lebih tinggi membuat area pencarian terhadap individu yang lebih baik semakin luas, sehingga kemungkinan ditemukan individu terbaik juga semakin cepat.

Probabilitas *crossover* dan mutasi yang disarankan untuk digunakan menyelesaikan permasalahan penentuan komposisi pakan ikan menggunakan algoritma genetika ini adalah 70% hingga 90%. Karena pada rentang nilai tersebut diperoleh hasil *fitness* yang optimal.

Selain probabilitas *crossover* dan probabilitas mutasi, komposisi bahan yang berbeda juga menentukan perubahan nilai *fitness*. Pengujian dilakukan pada probabilitas *crossover* 90%, dan probabilitas mutasi 90%. Uji coba dilakukan sebanyak 10 kali percobaan untuk masing-masing komposisi bahan.

Perubahan nilai *fitness* dari masing-masing perbedaan komposisi bahan dengan probabilitas mutasi dan *crossover* 90% dapat dilihat pada tabel 4.6.

Tabel 4.7 Nilai *Fitness* pada komposisi bahan yang berbeda dengan probabilitas mutasi dan *crossover* 90%.

ID Bahan	Individu Terbaik	Fitness
1 sampai 10	2 3 8 4 7 8 4 8	0.041104478318127
11 sampai 20	17 12 17 12 16 20 16 17	0.0647918526367313
21 sampai 34	31 25 32 25 30 34 33 34	0.0874689255133045

Dari uji coba diatas diperoleh hasil bahwa perubahan komposisi bahan memberikan perubahan terhadap nilai *fitness*. Misalkan pada komposisi bahan dengan id bahan 1 sampai 10 diperoleh nilai *fitness* 0.041104478318127, sedangkan komposisi bahan dengan id bahan 21 sampai 34 memiliki nilai *fitness* 0.0874689255133045.

Hal ini menunjukkan bahwa nilai *fitness* yang paling besar dimiliki oleh komposisi bahan dengan id bahan 21 sampai 34. Oleh karena itu komposisi bahan dengan id bahan 21 sampai 34 memiliki nilai *fitness* yang paling optimal.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

- Kesimpulan yang dapat diambil dari tugas akhir ini adalah :
1. Pada penelitian ini telah dibuat suatu program aplikasi untuk menyelesaikan permasalahan Penentuan Komposisi Pakan Ikan Menggunakan Algoritma.
 2. Seleksi *Ranking* diterapkan dengan cara mengurutkan nilai *fitness* dari yang tertinggi sampai yang terendah dalam satu populasi, setelah itu individu dengan nilai *fitness* tertinggi akan terpilih sebagai induk, sedangkan individu dengan nilai *fitness* terendah akan dibuang.
 3. Probabilitas *crossover* dan mutasi memberi pengaruh pada nilai *fitness* dan generasi konvergensi. Peningkatan nilai probabilitas *crossover* atau mutasi membuat nilai *fitness* semakin tinggi dan konvergensi pada generasi lebih awal, karena semakin tinggi probabilitas *crossover* atau mutasi yang diberikan membuat peluang terbentuknya individu yang lebih baik semakin besar sehingga nilai *fitness* semakin tinggi dan nilai *fitness* yang terbaik juga lebih cepat ditemukan.
 4. Dari hasil uji coba diperoleh hasil pada nilai probabilitas *crossover* 90% dan probabilitas mutasi 90% diperoleh nilai *fitness* yang optimal.
 5. Perubahan komposisi bahan juga memberikan pengaruh terhadap perubahan nilai *fitness*. Dari hasil uji coba diperoleh komposisi bahan mulai dari id bahan 21 sampai 34 memiliki nilai *fitness* yang paling optimal.

5.2. Saran

Diharapkan aplikasi ini dapat dikembangkan dengan menambahkan komponen kandungan nutrisi bahan, tidak hanya protein, lemak, karbohidrat, dan serat, tetapi juga melibatkan vitamin, zat perekat dan mineral.

UNIVERSITAS BRAWIJAYA



DAFTAR PUSTAKA

- Al-hajri, M.T. Abido, M.A. 2009. *Assessment of Genetic Algorithm Selection, Crossover and Mutation Techniques in Reactive Power Optimization*. Congress on Evolutionary Computation. Saudi Arabia.
- Aribowo, A. Lukas, S. Gunawan, M. 2008. *Penerapan Algoritma Genetika Pada Penentuan Komposisi Pakan Ayam Petelur*. Seminar Nasional Aplikasi Teknologi Informasi : Yokyakarta.
- Cahyono. 2001. *Budi Daya Ikan di Perairan Umum*. Kanisius:Yokyakarta.
- Cinantya. 2010. *IMPLEMENTASI ALGORITMA GENETIKA UNTUK PENYELESAIAN CAPACITATED VEHICLE ROUTING WITH TIME WINDOWS*. Skripsi. Universitas Brawijaya:Malang.
- Gen, M. Cheng, R. 1997. *Genetic Algorithms and Engineering Design*. John Wiley & Son, Inc. USA.
- Gen, M. Cheng, R. 1997. *Genetic Algorithms and Engineering Optimization*. John Wiley & Son, Inc. USA.
- Ghufran. 2007. Meramu Pakan Ikan untuk Ikan Karnivor. Aneka Ilmu:Semarang
- Khairuman. 2002. *Membuat Pakan Ikan Konsumsi*. agroMedia Pustaka:Jakarta-Depok.
- Kusumadewi. 2003. *Artificial Inteligence*. Graha Ilmu. Yogyakarta.
- Kuswara, Setiawan. 2003. *Paradigma Sistem Cerdas*. Bayumedia Publishing. Malang.
- Michalewicz, Z. 1996. *Genetic Algorithm + Data Structures =*

Evolution Programs, 3rd, revised and extended edition.
Springer-Verlag.

Mitchell. 1999. *An introduction to genetic algorithms*. A Bradford Book The MIT Press. London, England.

Murtidjo. 2001. *Pedoman Meramu Pakan Ikan*.
Kanisius:Yogyakarta.

Perry,T.W., Cullison, A.E., and Lowrey, R.S.,. 2004. *Feeds and Feeding 6th Edition*. Pearson Education, Inc. Upper Saddle River, New Jersey.

Sanjoyo. 2006. *Aplikasi Algoritma Genetika*.

Steven, Craig. 2009. *Understanding Fish Nutrition, Feeds, and Feeding*. <http://pubs.ext.vt.edu/420/420-256/420-256.html>.
Tanggal akses 15 Juli 2010.

Toriq. 2009. *Optimasi Formulasi Pakan Ternak Dengan Menggunakan Algoritma Genetika*. Skripsi. Universitas Brawijaya:Malang.

LAMPIRAN 1
Tabel Kebutuhan Standar Gizi Ikan

Tabel 1 Kebutuhan standar gizi untuk ikan (Steven, Craig. 2009)

Nutrisi	Standar (%)
Protein	30
Lemak	18
Karbohidrat	17,5
Serat	6
Abu	8,5



UNIVERSITAS BRAWIJAYA



LAMPIRAN 2
Tabel Analisis Nilai Gizi Komposisi Bahan Pakan

Tabel 1 Analisis nilai gizi komposisi bahan pakan (Cahyono, Bambang, 2001)

Bahan						
ID	Nama	Protein	Lemak	Karbohidrat	Serat	Abu
1	tepung ikan	50	7.8	0	0.6	7.97
2	tepung rebon	59.4	3.6	3.2	0	0
3	tepung benawa	23.38	25.33	0.06	11.82	31.41
4	tepung kepala udang	57.74	6.65	0	14.61	7.72
5	tepung darah	84.1	1.4	13.12	0.7	0.76
6	tepung tulang	25.54	3.8	0	1.8	61.8
7	tepung bekicot	54.29	4.18	30.45	0	32.67
8	telur ayam dan itik	12.8	11.5	0.7	0	0
9	tepung daging	57.8	10.2	0	1.7	12.87
10	tepung teri	51	4.21	0	3.6	27.13
11	Dedak halus	13.6	8.2	28.62	8	1.28
12	dedak jagung	9.7	7.2	0	6.8	0.3
13	tepung jagung	9.8	4.2	0	4.2	0
14	tepung terigu	8.9	1.3	77.3	0	0.06
15	tepung kedelai	46.36	5.31	29.5	3.54	7.2
16	ampas tahu	23.55	5.54	26.92	16.53	17.03
17	bungkil kacang tanah	47.9	10.9	25	3.6	4.8
18	bungkil biji kapuk	27.4	5.6	18.6	25.3	7.3
19	bungkil kelapa	17.09	9.44	23.77	30.4	5.92
20	bungkil kacang kedelai	44.4	4	0	6.2	0.98
21	bungkil kelapa sawit	18	8.1	0	18	0
22	tepung singkong	0.85	0.3	0	0	3.64
23	daun ubi kayu	8.1	1.3	0	4.1	0

Bahan

ID	Nama	Protein	Lemak	Karbohidrat	Serat	Abu
24	daun turi	6.5	0.8	0	3.6	0
25	Tepung Sagu	7.25	0.55	66.21	11.24	1.53
26	Daun Lamturo	23.2	2.4	0	20.1	0
27	tepung ketan	8.21	2.13	83.12	2.26	2.96
28	tepung bulu unggas	86.5	3.9	0	0.4	0
29	tepung kembung	40.36	5.25	1.26	0	31.96
30	tepung kepala cumi	70.19	2.16	0	0	0
31	tepung beras	14.1	15.1	0	12.8	12.8
32	tepung petek	66	11.2	7.36	0	19.5
33	daun kacang tanah	15.85	3.22	18.02	20.94	32.86
34	biji kecipir	29.8	15	0	3.7	0

LAMPIRAN 3

Tabel Pengujian Probabilitas Crossover dan Mutasi Terhadap Nilai *Fitness*

Tabel 1 Nilai *fitness* pada probabilitas *crossover* 10% dan probabilitas mutasi 10% - 90%

PERCOBAAN	Pm 10%	Pm 30%	pm 50%	Pm 70%	Pm 90%
1	0.063520308	0.057262682	0.056201439	0.067922083	0.066002828
2	0.058917268	0.063075151	0.074698504	0.069881838	0.072423436
3	0.062882525	0.060614093	0.063423012	0.078286938	0.066569925
4	0.065417487	0.071024172	0.073537689	0.071963397	0.088300337
5	0.068323795	0.07570446	0.074812387	0.067613182	0.072767129
RATA-RATA	0.063812276	0.065536112	0.068534606	0.071133487	0.073212731

Tabel 2 Nilai *fitness* pada probabilitas *crossover* 30% dan probabilitas mutasi 10% - 90%

PERCOBAAN	Pm 10%	Pm 30%	pm 50%	Pm 70%	Pm 90%
1	0.073036926	0.079083572	0.074928977	0.064582898	0.083020288
2	0.091529006	0.074851328	0.080892968	0.062750694	0.081540303
3	0.089932263	0.087108275	0.087131054	0.0697266	0.079271928
4	0.072447994	0.067593274	0.078202815	0.067344263	0.082075929
5	0.063097081	0.063412444	0.072438505	0.087707137	0.080872612
RATA-RATA	0.078008654	0.074409779	0.078718864	0.070422318	0.081356212

Tabel 3 Nilai *fitness* pada probabilitas *crossover* 50% dan probabilitas mutasi 10% - 90%

PERCOBAAN	Pm 10%	Pm 30%	pm 50%	Pm 70%	Pm 90%
1	0.076589468	0.099326652	0.094934807	0.076382102	0.097402345
2	0.087231372	0.082037073	0.063923573	0.098856598	0.070358661
3	0.07605529	0.082356898	0.091549607	0.089940133	0.065490809
4	0.067563201	0.058261229	0.082497954	0.07355977	0.106208956
5	0.076187189	0.089411362	0.086979431	0.074669015	0.082550057
RATA-RATA	0.076725304	0.082278643	0.083977074	0.082681524	0.084402166

Tabel 4 Nilai *fitness* pada probabilitas *crossover* 70% dan probabilitas mutasi 10% - 90%

PERCOBAAN	Pm 10%	Pm 30%	pm 50%	Pm 70%	Pm 90%
1	0.085823927	0.088741276	0.092042294	0.104160574	0.087183958
2	0.077218405	0.086101532	0.086599102	0.086246945	0.080237406
3	0.072440889	0.078413801	0.071837106	0.082495383	0.098602365
4	0.086162258	0.078574404	0.076475122	0.074207055	0.074238049
5	0.088854971	0.075290298	0.086286317	0.066392538	0.083691138
RATA-RATA	0.08210009	0.081424262	0.082647988	0.082700499	0.084790583

Tabel 5 Nilai *fitness* pada probabilitas *crossover* 90% dan probabilitas mutasi 10% - 90%

PERCOBAAN	Pm 10%	Pm 30%	pm 50%	Pm 70%	Pm 90%
1	0.077762764	0.109478654	0.104982237	0.080937795	0.089774584
2	0.083908883	0.078253373	0.07343198	0.092318446	0.096509881
3	0.09065225	0.073333844	0.098922433	0.084303157	0.088039265
4	0.085604915	0.082579798	0.069967909	0.080647248	0.089486559
5	0.082416101	0.086933294	0.088333249	0.101350835	0.093648747
RATA-RATA	0.084068983	0.086115793	0.087127562	0.0879111496	0.091491807

LAMPIRAN 4
Tabel Pengujian Probabilitas Crossover dan Mutasi
Terhadap Generasi Konvergensi

Tabel 1 Generasi konvergensi pada probabilitas *crossover* 10% dan probabilitas mutasi 10% - 90%

PERCOBAAN	Pm 10%	Pm 30%	pm 50%	Pm 70%	Pm 90%
1	261	415	146	112	157
2	240	133	279	255	323
3	301	89	246	524	84
4	293	478	221	203	272
5	277	162	342	108	208
RATA-RATA	274.4	255.4	246.8	240.4	208.8

Tabel 2 Generasi konvergensi pada probabilitas *crossover* 30% dan probabilitas mutasi 10% - 90%

PERCOBAAN	Pm 10%	Pm 30%	pm 50%	Pm 70%	Pm 90%
1	115	308	306	174	265
2	14	280	159	62	76
3	470	253	48	166	186
4	209	145	305	350	350
5	344	103	347	250	227
RATA-RATA	230.4	217.8	233	200.4	220.8

Tabel 3 Generasi konvergensi pada probabilitas *crossover* 50% dan probabilitas mutasi 10% - 90%

PERCOBAAN	Pm 10%	Pm 30%	pm 50%	Pm 70%	Pm 90%
1	147	64	61	98	148
2	220	39	146	218	49
3	103	401	153	324	151
4	302	164	437	61	193
5	210	352	275	317	198
RATA-RATA	196.4	204	214.4	203.6	147.8

Tabel 4 Generasi konvergensi pada probabilitas *crossover* 70% dan probabilitas mutasi 10% - 90%

PERCOBAAN	Pm 10%	Pm 30%	pm 50%	Pm 70%	Pm 90%
1	318	125	360	74	98
2	167	101	113	93	211
3	176	324	207	168	180
4	113	108	79	321	187
5	79	140	57	46	147
RATA-RATA	170.6	159.6	163.2	140.4	164.6

Tabel 5 Generasi konvergensi pada probabilitas *crossover* 90% dan probabilitas mutasi 10% - 90%

PERCOBAAN	Pm 10%	Pm 30%	pm 50%	Pm 70%	Pm 90%
1	172	185	85	106	82
2	380	203	87	74	59
3	40	137	122	102	108
4	60	172	70	67	108
5	98	28	245	159	125
RATA-RATA	150	145	121.8	101.6	96.4

LAMPIRAN 5

Pengujian Komposisi Bahan Terhadap Nilai Fitness

Komposisi 1

Komposisi bahan pakan dengan id bahan mulai dari 1 sampai 10

Percobaan 1

Individu 2 8 3 1 4 10 3 8

fitness = 0.0394213828767464

Komposisi Bahan Pakan :

tepung rebon = 8.04289544235925 kg

telur ayam dan itik = 25.4691689008043 kg

tepung benawa = 3.48525469168901 kg

tepung ikan = 8.84718498659517 kg

tepung kepala udang = 17.9624664879357 kg

tepung teri = 0.268096514745308 kg

tepung benawa = 21.4477211796247 kg

telur ayam dan itik = 14.4772117962466 kg

Percobaan 2

Individu : 1 4 1 7 3 5 7 9

fitness = 0.032703080863542

Komposisi Bahan Pakan :

tepung ikan = 0.920598388952819 kg

tepung kepala udang = 11.1622554660529 kg

tepung ikan = 3.68239355581128 kg

tepung bekicot = 5.63866513233602 kg

tepung benawa = 5.40851553509781 kg

tepung darah = 10.7019562715765 kg

tepung bekicot = 11.5074798619102 kg

tepung daging = 2.41657077100115 kg

Percobaan 3

Individu : 3 7 4 8 4 1 5 7

fitness = 0.0328435564806133

Komposisi Bahan Pakan :

tepung benawa = 2.54901960784314 kg
tepung bekicot = 2.94117647058823 kg
tepung kepala udang = 14.1176470588235 kg
telur ayam dan itik = 2.15686274509804 kg
tepung kepala udang = 11.7647058823529 kg
tepung ikan = 1.17647058823529 kg
tepung darah = 7.25490196078431 kg
tepung bekicot = 13.921568627451 kg

Percobaan 4

Individu : 8 6 9 8 3 3 7 3
fitness = 0.0354648204400722

Komposisi Bahan Pakan :

telur ayam dan itik = 6.666666666666667 kg
tepung tulang = 0.289855072463768 kg
tepung daging = 21.1594202898551 kg
telur ayam dan itik = 26.9565217391304 kg
tepung benawa = 14.7826086956522 kg
tepung benawa = 10.7246376811594 kg
tepung bekicot = 9.85507246376812 kg
tepung benawa = 9.56521739130435 kg

Percobaan 5

Individu : 8 3 10 7 3 8 8 4
fitness = 0.0397990500621671

Komposisi Bahan Pakan :

telur ayam dan itik = 14.1772151898734 kg
tepung benawa = 4.30379746835443 kg
tepung teri = 1.77215189873418 kg
tepung bekicot = 10.6329113924051 kg
tepung benawa = 16.4556962025316 kg
telur ayam dan itik = 16.9620253164557 kg
telur ayam dan itik = 12.1518987341772 kg
tepung kepala udang = 23.5443037974684 kg

Percobaan 6

Individu : 8 8 8 2 3 7 3 5
fitness = 0.0409981431753827

Komposisi Bahan Pakan :

telur ayam dan itik	= 16.804979253112 kg
telur ayam dan itik	= 17.4273858921162 kg
telur ayam dan itik	= 13.9004149377593 kg
tepung rebon	= 12.448132780083 kg
tepung benawa	= 10.1659751037344 kg
tepung bekicot	= 1.03734439834025 kg
tepung benawa	= 16.1825726141079 kg
tepung darah	= 12.0331950207469 kg

Percobaan 7

Individu : 4 7 7 2 8 8 3 1

fitness = 0.0394812518698949

Komposisi Bahan Pakan :

tepung kepala udang	= 12.80276816609 kg
tepung bekicot	= 2.07612456747405 kg
tepung bekicot	= 2.7681660899654 kg
tepung rebon	= 10.0346020761246 kg
telur ayam dan itik	= 12.4567474048443 kg
telur ayam dan itik	= 31.1418685121107 kg
tepung benawa	= 20.7612456747405 kg
tepung ikan	= 7.95847750865052 kg

Percobaan 8

Individu : 2 3 8 4 7 8 4 8

fitness = 0.041104478318127

Komposisi Bahan Pakan :

tepung rebon	= 7.19424460431655 kg
tepung benawa	= 13.1894448441247 kg
telur ayam dan itik	= 16.0671462829736 kg
tepung kepala udang	= 4.79616306954436 kg
tepung bekicot	= 7.67386091127098 kg
telur ayam dan itik	= 18.4652278177458 kg
tepung kepala udang	= 16.7865707434053 kg
telur ayam dan itik	= 15.8273381294964 kg

Percobaan 9

Individu: 4 8 10 8 9 9 3 8
fitness = 0.0369175920236905

Komposisi Bahan Pakan :

tepung kepala udang	= 11.9047619047619 kg
telur ayam dan itik	= 9.76190476190476 kg
tepung teri	= 5.95238095238095 kg
telur ayam dan itik	= 18.0952380952381 kg
tepung daging	= 2.38095238095238 kg
tepung daging	= 14.047619047619 kg
tepung benawa	= 18.5714285714286 kg
telur ayam dan itik	= 19.2857142857143 kg

Percobaan 10

Individu: 1 8 1 4 8 3 2 3
fitness = 0.0391331153528564

Komposisi Bahan Pakan :

tepung ikan	= 5.0314465408805 kg
telur ayam dan itik	= 23.2704402515723 kg
tepung ikan	= 4.08805031446541 kg
tepung kepala udang	= 18.8679245283019 kg
telur ayam dan itik	= 14.4654088050314 kg
tepung benawa	= 0.314465408805031 kg
tepung rebon	= 6.28930817610063 kg
tepung benawa	= 27.6729559748428 kg

Komposisi 2

Komposisi bahan pakan dengan id bahan mulai dari 11 sampai 20

Percobaan 1

Individu : 20 17 16 19 17 12 16 16

fitness = 0.0603818383711231

Komposisi Bahan Pakan :

bungkil kacang kedelai	= 4.23728813559322 kg
bungkil kacang tanah	= 16.1016949152542 kg
ampas tahu	= 1.69491525423729 kg
bungkil kelapa	= 0.282485875706215 kg
bungkil kacang tanah	= 17.7966101694915 kg
dedak jagung	= 27.9661016949153 kg
ampas tahu	= 12.9943502824859 kg
ampas tahu	= 18.9265536723164 kg

Percobaan 2

Individu : 17 13 13 16 15 15 20 11

fitness = 0.0621366301648485

Komposisi Bahan Pakan :

bungkil kacang tanah	= 27.3333333333333 kg
tepung jagung	= 15 kg
tepung jagung	= 16 kg
ampas tahu	= 20.3333333333333 kg
tepung kedelai	= 14 kg
tepung kedelai	= 3.3333333333333 kg
bungkil kacang kedelai	= 3.6666666666667 kg
Dedak halus	= 0.3333333333333 kg

Percobaan 3

Individu : 16 15 17 16 12 16 11 20

fitness = 0.0563626393600626

Komposisi Bahan Pakan :

ampas tahu	= 13.550135501355 kg
tepung kedelai	= 5.1490514905149 kg
bungkil kacang tanah	= 15.7181571815718 kg
ampas tahu	= 6.77506775067751 kg

dedak jagung	= 13.550135501355 kg
ampas tahu	= 11.6531165311653 kg
Dedak halus	= 11.1111111111111 kg
bungkil kacang kedelai	= 22.4932249322493 kg

Percobaan 4

Individu : 11 17 15 12 17 16 15 12

fitness = 0.0618436207454142

Komposisi Bahan Pakan :

Dedak halus	= 3.97553516819572 kg
bungkil kacang tanah	= 27.5229357798165 kg
tepung kedelai	= 1.52905198776758 kg
dedak jagung	= 5.81039755351682 kg
bungkil kacang tanah	= 7.3394495412844 kg
ampas tahu	= 21.1009174311927 kg
tepung kedelai	= 9.48012232415902 kg
dedak jagung	= 23.2415902140673 kg

Percobaan 5

Individu : 13 15 11 15 19 20 11 20

fitness = 0.0506888158390003

Komposisi Bahan Pakan :

tepung jagung	= 18.4818481848185 kg
tepung kedelai	= 1.98019801980198 kg
Dedak halus	= 4.62046204620462 kg
tepung kedelai	= 29.3729372937294 kg
bungkil kelapa	= 2.31023102310231 kg
bungkil kacang kedelai	= 5.94059405940594 kg
Dedak halus	= 20.4620462046205 kg
bungkil kacang kedelai	= 16.8316831683168 kg

Percobaan 6

Individu : 17 11 18 12 15 20 15 16

fitness = 0.0541923206861603

Komposisi Bahan Pakan :

bungkil kacang tanah	= 25.4452926208651 kg
Dedak halus	= 15.7760814249364 kg
bungkil biji kapuk	= 5.34351145038168 kg

dedak jagung	= 19.0839694656489 kg
tepung kedelai	= 7.88804071246819 kg
bungkil kacang kedelai	= 10.1781170483461 kg
tepung kedelai	= 0.508905852417303 kg
ampas tahu	= 15.7760814249364 kg

Percobaan 7

Individu : 16 20 17 15 12 15 15 13

fitness = 0.060340161321196

Komposisi Bahan Pakan :

ampas tahu	= 20.1680672268908 kg
bungkil kacang kedelai	= 6.51260504201681 kg
bungkil kacang tanah	= 8.61344537815126 kg
tepung kedelai	= 14.7058823529412 kg
dedak jagung	= 20.5882352941176 kg
tepung kedelai	= 3.36134453781513 kg
tepung kedelai	= 14.4957983193277 kg
tepung jagung	= 11.5546218487395 kg

Percobaan 8

Individu : 12 13 16 17 15 20 17 16

fitness = 0.0631947775668968

Komposisi Bahan Pakan :

dedak jagung	= 5.28052805280528 kg
tepung jagung	= 24.4224422442244 kg
ampas tahu	= 23.1023102310231 kg
bungkil kacang tanah	= 18.8118811881188 kg
tepung kedelai	= 12.2112211221122 kg
bungkil kacang kedelai	= 4.95049504950495 kg
bungkil kacang tanah	= 8.25082508250825 kg
ampas tahu	= 2.97029702970297 kg

Percobaan 9

Individu : 18 20 16 17 11 12 17 15

fitness = 0.0577972558245065

Komposisi Bahan Pakan :

bungkil biji kapuk	= 0.206611570247934 kg
bungkil kacang kedelai	= 16.5289256198347 kg

ampas tahu	= 19.6280991735537	kg
bungkil kacang tanah	= 5.78512396694215	kg
Dedak halus	= 14.0495867768595	kg
dedak jagung	= 18.18181818182	kg
bungkil kacang tanah	= 17.975206115702	kg
tepung kedelai	= 7.64462809917355	kg

Percobaan 10

Individu : 17 12 17 12 16 20 16 17

fitness = 0.0647918526367313

Komposisi Bahan Pakan :

bungkil kacang tanah	= 15.7205240174673	kg
dedak jagung	= 20.0873362445415	kg
bungkil kacang tanah	= 17.0305676855895	kg
dedak jagung	= 10.4803493449782	kg
ampas tahu	= 7.4235807860262	kg
bungkil kacang kedelai	= 1.74672489082969	kg
ampas tahu	= 20.3056768558952	kg
bungkil kacang tanah	= 7.20524017467249	kg

Komposisi 2

Komposisi bahan pakan dengan id bahan mulai dari 21 sampai 34

Percobaan 1

Individu : 31 25 32 25 30 34 33 34

fitness = 0.0874689255133045

Komposisi Bahan Pakan :

tepung beras	= 21.0526315789474 kg
Tepung Sagu	= 8.33333333333333 kg
tepung petek	= 19.9561403508772 kg
Tepung Sagu	= 14.0350877192982 kg
tepung kepala cumi	= 3.28947368421053 kg
biji kecipir	= 14.0350877192982 kg
daun kacang tanah	= 2.41228070175439 kg
biji kecipir	= 16.8859649122807 kg

Percobaan 2

Individu : 32 26 24 31 30 27 26 29

fitness = 0.0622096105273005

Komposisi Bahan Pakan :

tepung petek	= 0.25062656641604 kg
Daun Lamturo	= 15.2882205513784 kg
daun turi	= 8.7719298245614 kg
tepung beras	= 13.7844611528822 kg
tepung kepala cumi	= 18.546365914787 kg
tepung ketan	= 21.3032581453634 kg
Daun Lamturo	= 1.00250626566416 kg
tepung kembung	= 21.0526315789474 kg

Percobaan 3

Individu : 29 22 28 28 31 25 34 30

fitness = 0.0564372726687983

Komposisi Bahan Pakan :

tepung kembung	= 4.41176470588235 kg
tepung singkong	= 1.83823529411765 kg
tepung bulu unggas	= 1.47058823529412 kg
tepung bulu unggas	= 1.83823529411765 kg

tepung beras	= 33.0882352941176 kg
Tepung Sagu	= 27.2058823529412 kg
biji kecipir	= 3.67647058823529 kg
tepung kepala cumi	= 26.4705882352941 kg

Percobaan 4

Individu : 32 25 34 24 24 32 24 29

fitness = 0.0619355491092025

Komposisi Bahan Pakan :

tepung petek	= 18.8775510204082 kg
Tepung Sagu	= 24.234693877551 kg
biji kecipir	= 22.4489795918367 kg
daun turi	= 0.255102040816327 kg
daun turi	= 0.76530612244898 kg
tepung petek	= 5.61224489795918 kg
daun turi	= 20.1530612244898 kg
tepung kembung	= 7.6530612244898 kg

Percobaan 5

Individu : 28 31 25 29 26 30 27 23

fitness = 0.0589028249912085

Komposisi Bahan Pakan :

tepung bulu unggas	= 1.49253731343284 kg
tepung beras	= 15.9203980099502 kg
Tepung Sagu	= 8.45771144278607 kg
tepung kembung	= 21.8905472636816 kg
Daun Lamturo	= 12.4378109452736 kg
tepung kepala cumi	= 17.4129353233831 kg
tepung ketan	= 15.6716417910448 kg
daun ubi kayu	= 6.71641791044776 kg

Percobaan 6

Individu : 32 25 31 29 25 24 32 34

fitness = 0.0785275241796981

Komposisi Bahan Pakan :

tepung petek	= 8.99280575539568 kg
Tepung Sagu	= 15.8273381294964 kg
tepung beras	= 17.6258992805755 kg

tepung kembung	= 2.87769784172662 kg
Tepung Sagu	= 8.99280575539568 kg
daun turi	= 8.27338129496403 kg
tepung petek	= 14.7482014388489 kg
biji kecipir	= 22.6618705035971 kg

Percobaan 7

Individu : 34 23 29 32 21 21 32 25

fitness = 0.0718222887086012

Komposisi Bahan Pakan :

biji kecipir	= 19.5266272189349 kg
daun ubi kayu	= 3.55029585798817 kg
tepung kembung	= 10.0591715976331 kg
tepung petek	= 1.77514792899408 kg
bungkil kelapa sawit	= 16.8639053254438 kg
bungkil kelapa sawit	= 3.25443786982249 kg
tepung petek	= 21.8934911242604 kg
Tepung Sagu	= 23.0769230769231 kg

Percobaan 8

Individu : 31 31 34 30 32 23 27 23

fitness = 0.0664382740851961

Komposisi Bahan Pakan :

tepung beras	= 9.28381962864722 kg
tepung beras	= 19.0981432360743 kg
biji kecipir	= 22.2811671087533 kg
tepung kepala cumi	= 0.530503978779841 kg
tepung petek	= 20.9549071618037 kg
daun ubi kayu	= 2.38726790450928 kg
tepung ketan	= 22.5464190981432 kg
daun ubi kayu	= 2.91777188328912 kg

Percobaan 9

Individu : 27 32 22 21 27 31 22 21

fitness = 0.0671047981124686

Komposisi Bahan Pakan :

tepung ketan	= 12.0996441281139 kg
tepung petek	= 28.8256227758007 kg

tepung singkong	= 4.98220640569395 kg
bungkil kelapa sawit	= 19.5729537366548 kg
tepung ketan	= 6.04982206405694 kg
tepung beras	= 25.2669039145907 kg
tepung singkong	= 2.84697508896797 kg
bungkil kelapa sawit	= 0.355871886120996 kg

Percobaan 10

Individu : 22 31 21 32 33 30 25 30

fitness = 0.0545300993997132

Komposisi Bahan Pakan :

tepung singkong	= 4.73537604456824 kg
tepung beras	= 19.7771587743733 kg
bungkil kelapa sawit	= 9.74930362116992 kg
tepung petek	= 27.8551532033426 kg
daun kacang tanah	= 2.22841225626741 kg
tepung kepala cumi	= 7.52089136490251 kg
Tepung Sagu	= 26.4623955431755 kg
tepung kepala cumi	= 1.67130919220056 kg