

**OPTIMASI KOMPOSISI PAKAN SISTEM POLIKULTUR IKAN
DAN UDANG MENGGUNAKAN ALGORITMA
EVOLUTION STRATEGIES**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Nenovli Widerti
NIM: 125150201111007



TEKNIK INFORMATIKA
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

PENGESAHAN

OPTIMASI KOMPOSISI PAKAN SISTEM POLIKULTUR IKAN DAN UDANG
MENGUNAKAN ALGORITMA *EVOLUTION STRATEGIES*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Nenovli Widerti

NIM: 125150201111007

Skripsi ini telah diuji dan dinyatakan lulus pada
24 Maret 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Imam Cholissodin, S.Si, M.Kom

NIK: 201201 850719 1 001

Dian Eka Ratnawati, S.Si, M.Kom

NIP: 19730619 200212 2 001

Mengetahui

Ketua Program Studi Informatika dan Ilmu Komputer

Drs. Marji, M.T

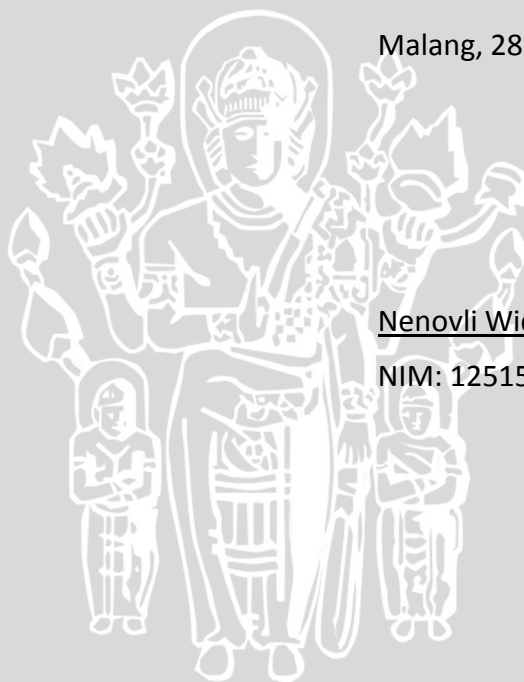
NIP. 19670801 199203 1 001

PERNYATAAN ORISINALITAS

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

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 28 Februari 2016



Nenovli Widerti

NIM: 125150201111007

KATA PENGANTAR

Puji Syukur kehadirat Allah SWT karena atas rahmat dan karunianya penulis dapat menyelesaikan penyusunan skripsi sebagai syarat untuk memperoleh gelar sarjana dengan judul “ Optimasi Komposisi Pakan Sistem Polikultur Ikan dan Udang Menggunakan Algoritma *evolution Strategies*” .

Penelitian ini membahas tentang permasalahan optimasi Komposisi Pakan sistem polikultur ikan dan udang. Diharapkan dengan adanya aplikasi yang dibuat dapat membantu para pembudidaya ikan dalam menentukan komposisi pakan yang optimal dengan memperhatikan kebutuhan nutrisi ikan dan dengan biaya yang minimal.

Penulis menyadari bahwa dalam penyusunan dan penyelesaian skripsi ini tidak akan dapat berjalan dengan baik tanpa bantuan dan keterlibatan berbagai pihak, oleh karena itu penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Imam Cholissodin, S.Si., M.Kom selaku dosen pembimbing pertama yang telah memberikan bimbingan dan masukan dalam penulisan skripsi ini.
2. Dian Eka Ratnawati, S.Si., M.Kom selaku dosen pembimbing kedua yang telah memberikan bimbingan dan masukan dalam penulisan skripsi ini.
3. Ir.Sutrisno, M.T, Ir. Heru Nurwasito, M. Kom, Himawat Aryadita, ST., Msc, dan Edy Santoso, S.Si., M.Kom selaku Ketua, Wakil Ketua I, Wakil Ketua II, dan Wakil Ketua III Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
4. Drs. Mardji, MT dan Issa Arwani, S.Kom., M.Sc selaku Ketua dan Sekertaris Program Studi Informatika dan Ilmu Komputer Universitas Brawijaya.
5. Muhammad Fakhri, S.Pi., M.Sc Selaku dosen Fakultas Perikanan dan Ilmu Kelautan Universitas Brawijaya yang bersedia membantu dan membagi ilmu dalam proses penyelesaian skripsi ini.
6. Segenap Bapak dan Ibu dosen atas kesediaan membagi ilmu kepada penulis selama menempuh pendidikan di Program Studi Informatika dan Ilmu Komputer Universitas Brawijaya.
7. Segenap staf dan karyawan yang membantu penulis selama proses penyelesaian skripsi ini.
8. Kedua orang tua atas segala do'a, nasihat, perhatian, kesabaran serta dukungan baik moril ataupun materil yang tanpa henti selalu diberikan kepada penulis.
9. Saudara perempuan saya dan segenap keluarga atas dukungan, nasehat serta kepedulian dan pengertiannya.

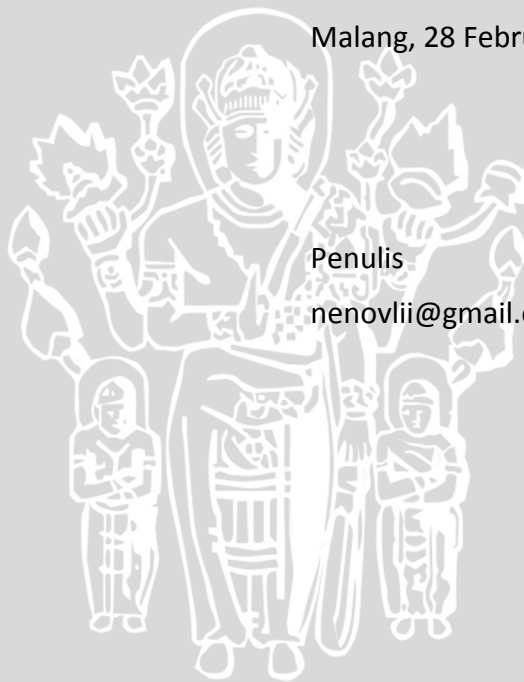
10. Yanda, Mayang dan A'yun teman seperjuangan atas kesabarannya dan ketersediaannya membantu selama menempuh pendidikan bersama di Teknik Informatika Universitas Brawijaya.

11. Segenap teman-teman TIF 2012, Terutama TIF E 2012 yang telah memberikan bantuan dan dorongan selama menempuh pendidikan di Universitas Brawijaya.

Semoga segala bantuan mendapatkan balasan yang terbaik. Dengan segala kerendahan hati, penulis menyadari bahwa skripsi ini masih memiliki banyak kekurangan. Untuk itu saran dan kritik yang sifatnya membangun sangat penulis harapkan sehingga dikesempatan berikutnya penulis dapat menyusun laporan dengan lebih baik lagi. Penulis berharap semoga skripsi ini dapat bermanfaat bagi *civitas academica* Teknik Informatika pada khususnya, dan Universitas Brawijaya pada umumnya.

Malang, 28 Februari 2016

Penulis
nenovlii@gmail.com



ABSTRAK

Banyak pembudidaya membudidayakan ikan dan udang dengan sistem polikultur karena budidaya polikultur dapat memberikan keuntungan lebih dari pada budidaya monokultur. Untuk meningkatkan produktivitas hasil budidaya ikan dan udang dapat menggunakan pakan buatan. Biaya pakan buatan bisa mencapai 60% dari biaya produksi. Untuk menekan biaya pakan maka, pakan buatan dapat dibuat sendiri, namun tidak semua pembudidaya mengetahui formulasi komposisi bahan pakan yang tepat sesuai dengan kebutuhan ikan. Oleh karena itu penelitian ini dilakukan untuk mengatasi masalah komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *evolution strategies*.

Untuk mengimplementasi algoritma *evolution strategies* dalam penentuan komposisi pakan sistem polikultur ikan dan udang yang harus dilakukan adalah menentukan kebutuhan nutrisi dan jumlah pakan dari ikan yang akan dibudidayakan dengan sistem polikultur. Dalam proses algoritma *evolution strategies* langkah pertama yang dilakukan adalah membangkitkan individu sebanyak jumlah populasi. Individu dibangkitkan dengan cara *random* bilangan dengan rentang tertentu. Kemudian akan dilakukan proses mutasi untuk menghasilkan keturunan, sehingga individu yang dihasilkan lebih beragam. Setelah proses mutasi selesai dilakukan selanjutnya yaitu melakukan proses seleksi dengan menggunakan seleksi elitism dengan memilih individu yang memiliki nilai *fitness* tertinggi sebagai individu yang lolos ke generasi selanjutnya. Individu yang dipilih sebanyak jumlah populasi.

Dari Hasil pengujian solusi optimal komposisi pakan sistem polikultur ikan dan udang diperoleh dari rentang kromosom 1-100, ukuran populasi 30, ukuran *offspring* 8 μ , jumlah generasi 500 pada siklus ES ($\mu/r+\lambda$) memperoleh rata-rata nilai *fitness* tertinggi yaitu 77.08264553. Hasil akhir dari penelitian ini adalah rekomendasi komposisi pakan ikan sistem polikultur ikan dan udang.

Kata kunci: sistem polikultur ikan dan udang, optimasi komposisi pakan, *evolution strategies*

ABSTRACT

Many cultivators cultivate fish and shrimp in polyculture systems because polyculture cultivation can provide more benefits than monoculture cultivation. To improve the productivity of the cultivation fish and shrimp can use artificial feed. Artificial feed costs could reach 60% of the cost production. To reduce the cost of feed, the artificial feed can be made, but not all cultivators know the composition of feed ingredient formulations appropriate to the needs of the fish. Therefore, this research was conducted to address the problem the composition of feed fish and shrimp polyculture system using evolution strategies algorithm.

To implement evolution strategies algorithm in determining the composition of feed fish and shrimp polyculture system that must be done is to determine the nutritional needs and the amount of feed of the fish to be cultivated with a polyculture system. In the process of evolution strategies algorithm first step is to generate of individuals as much as the population sizes. Individuals generated by way of random numbers with a certain range. Then, it will be done the mutation process to produce offspring, so that the individual who generated more diverse. Once the process is complete mutation do next is to do the selection process using elitism selection by selecting individuals who have the highest fitness value as individuals who qualify for the next generation. Individuals who been selected as many as the number of population.

The test results are the optimal solution composition of feed fish and shrimp polyculture system obtained range of chromosomes 1-100 , population size 30, offspring size 8μ , generation number 500, and cycles $(\mu / r + \lambda)$ gained an average of the highest fitness value is 77.08264553. The end result of this research is recommendation the composition of fish feed fish and shrimp polyculture system.

Keywords: polyculture systems of fish and shrimp, optimization of the feed composition, evolution strategies algorithm.

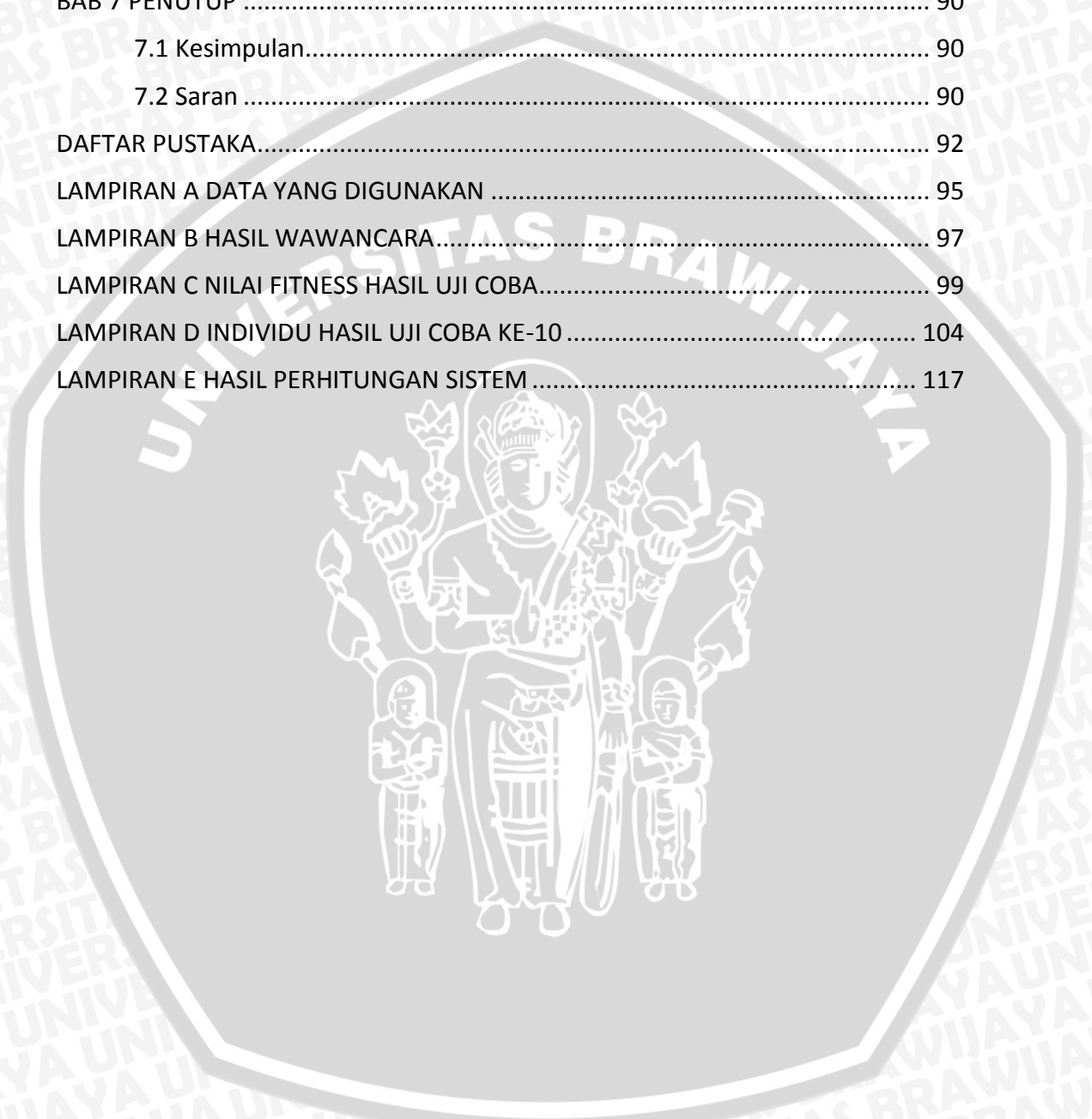
DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiv
DAFTAR LAMPIRAN	xv
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan masalah	3
1.6 Sistematika penulisan	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Pakan Ikan	7
2.3 Pakan Ikan Buatan	8
2.4 Aspek Nutrisi Pakan Ikan	10
2.4.1 Protein.....	10
2.4.2 Karbohidrat	10
2.4.3 Lemak	10
2.4.4 Serat	11
2.5 Kebutuhan Nurisi Pakan Ikan dan Udang	11
2.6 Bahan Baku Pakan Ikan.....	12
2.7 Konsumsi Makanan Harian Ikan	13
2.8 Budidaya Polikultur.....	14
2.8.1 Jenis Ikan Sistem Polikultur.....	15

2.8.2 Padat Tebar Ikan dan Udang yang Dipelihara dengan Sitem Polikultur	16
2.8.3 Strategi Pakan ikan sistem polikultur Ikan dan Udang	17
2.9 Algoritma Evolusi	17
2.10 <i>Evolution Strategies (ES)</i>	19
2.10.1 Proses <i>Evolution Strategies</i>	19
2.10.2 Representasi Kromosom	20
2.10.3 Inisialisasi	20
2.10.4 Reproduksi	21
2.10.5 Perhitungan <i>fitness</i>	23
2.10.6 Seleksi	23
BAB 3 METODOLOGI	25
3.1 Tahapan Penelitian	25
3.2 Studi Literatur	26
3.3 Pengumpulan Data	26
3.4 Analisis Kebutuhan	26
3.4.1 Deskripsi Umum Sistem	26
3.4.2 Data yang Digunakan	27
3.5 Perancangan Sistem	27
3.6 Implementasi Sistem	28
3.7 Pengujian	28
3.8 Kesimpulan dan Saran	28
BAB 4 PERANCANGAN SISTEM	29
4.1 Formulasi Permasalahan	29
4.2 Siklus Pemecahan Masalah	30
4.2.1 <i>Generate</i> Populasi Awal	32
4.2.2 Reproduksi	33
4.2.3 Perhitungan nilai <i>fitness</i>	34
4.2.4 Proses Seleksi	35
4.3 Contoh Perhitungan Manual	37
4.3.1 Inisialisasi	38
4.3.2 Membuat populasi awal	38

4.3.3	Proses Generasi ke 1	38
4.3.4	Proses Generasi Ke 2	50
4.3.5	Hasil Rekomendasi Komposisi Pakan	62
4.4	Perancangan Antarmuka	62
4.4.1	Rancangan Tampilan Halaman Input Data Ikan dan Bahan	62
4.4.2	Rancangan Tampilan Halaman Proses Algoritma <i>Evolution Strategies</i>	64
4.5	Perancangan Uji Coba dan Evaluasi	65
4.5.1	Uji Coba Rentang Kromosom	65
4.5.2	Uji Coba Ukuran Populasi	65
4.5.3	Uji Coba Ukuran <i>Offspring</i>	66
4.5.4	Uji Coba Banyaknya Generasi	67
4.5.5	Uji Coba Siklus Algoritma <i>Evolution Strategies</i>	67
4.5.6	Uji Coba Persamaan <i>Fitness</i>	67
BAB 5	IMPLEMENTASI SISTEM	70
5.1	Spesifikasi Sistem	70
5.1.1	Spesifikasi Perangkat Keras	70
5.1.2	Spesifikasi Perangkat Lunak	70
5.2	Implementasi Program	70
5.2.1	Implementasi Generate Populasi Awal	71
5.2.2	Implementasi Reproduksi	72
5.2.3	Implementasi Proses Perhitungan <i>fitness</i>	73
5.2.4	Implementasi Proses Seleksi	75
5.3	Implementasi Antarmuka	77
5.3.1	Tampilan Halaman Input Data Ikan dan Bahan	77
5.3.2	Tampilan Halaman Proses Algoritma	78
BAB 6	PENGUJIAN DAN ANALISIS	80
6.1	Sistematika Pengujian	80
6.2	Analisis dan Hasil Pembahasan	80
6.2.1	Hasil Uji Coba Rentang Kromosom	80
6.2.2	Hasil Uji Coba Ukuran Populasi	81
6.2.3	Hasil Uji Coba Ukuran <i>Offspring</i>	83

6.2.4 Hasil Uji Coba Jumlah Generasi.....	84
6.2.5 Hasil Uji Coba Siklus Algoritma Evolution Strategies	86
6.2.6 Hasil Uji Coba Persamaan Fitness	87
BAB 7 PENUTUP	90
7.1 Kesimpulan.....	90
7.2 Saran	90
DAFTAR PUSTAKA.....	92
LAMPIRAN A DATA YANG DIGUNAKAN	95
LAMPIRAN B HASIL WAWANCARA.....	97
LAMPIRAN C NILAI FITNESS HASIL UJI COBA.....	99
LAMPIRAN D INDIVIDU HASIL UJI COBA KE-10	104
LAMPIRAN E HASIL PERHITUNGAN SISTEM.....	117



DAFTAR TABEL

Tabel 2.1 Perbandingan Penelitian Penulis dengan Penelitian Sebelumnya.....	6
Tabel 2.2 Komposisi Nutrisi Pakan Ikan Bandeng	11
Tabel 2.3 Komposisi Nutrisi Pakan Ikan Nila	11
Tabel 2.4 Komposisi Nutrisi Pakan Udang Windu.....	12
Tabel 2.5 Bahan Baku Pakan Ikan Dan Nutrisinya	12
Tabel 2.6 Dosis Pakan Ikan Bandeng.....	13
Tabel 2.7 Dosis Pakan Ikan Nila.....	13
Tabel 2.8 Dosis Pakan Udang Windu	14
Tabel 2.9 Contoh Inisialisasi	20
Tabel 2.10 Contoh Hasil Rekombinasi.....	21
Tabel 2.11 Contoh Mutasi.....	22
Tabel 2.12 Contoh Seleksi	24
Tabel 4.1 Hasil <i>Random</i> Pembangkitan Nilai r	39
Tabel 4.2 Hasil Perhitungan Individu Baru	40
Tabel 4.3 Contoh perhitungan total harga p_1	41
Tabel 4.4 Hasil perhitungan total cost <i>parent</i>	41
Tabel 4.5 Hasil perhitungan total cost <i>offspring</i>	42
Tabel 4.6 Contoh Perhitungan Total Nutrisi	42
Tabel 4.7 Total Nutrisi <i>Parent</i>	44
Tabel 4.8 Total Nutrisi <i>Offspring</i>	45
Tabel 4.9 Perhitungan <i>Penalty Parent</i>	46
Tabel 4.10 Perhitungan <i>Penalty Offspring</i>	47
Tabel 4.11 Hasil perhitungan nilai <i>fitness</i> parrent	48
Tabel 4.12 Hasil perhitungan nilai <i>fitness offspring</i>	48
Tabel 4.13 hasil seleksi.....	49
Tabel 4.14 Hasil <i>Random</i> Pembangkitan Nilai r	51
Tabel 4.15 Hasil Perhitungan Individu Baru	52
Tabel 4.16 Contoh perhitungan total harga p_1	52
Tabel 4.17 Hasil perhitungan total cost <i>parent</i>	53
Tabel 4.18 Hasil perhitungan total cost <i>offspring</i>	53

Tabel 4.19 Contoh Perhitungan Total Nutrisi	54
Tabel 4.20 Total Nutrisi <i>Parent</i>	56
Tabel 4.21 Total Nutrisi <i>Offspring</i>	56
Tabel 4.22 Perhitungan <i>Penalty Parent</i>	58
Tabel 4.23 Perhitungan <i>Penalty Offspring</i>	59
Tabel 4.24 Hasil perhitungan nilai <i>fitness</i> parrent	60
Tabel 4.25 Hasil perhitungan nilai <i>fitness offspring</i>	60
Tabel 4.26 hasil seleksi.....	61
Tabel 4.27 Rancangan Uji Coba Rentang Kromosom.....	65
Tabel 4.28 Rancangan Uji Coba Ukuran Populasi	66
Tabel 4.29 Rancangan Uji Coba Ukuran <i>Offspring</i>	66
Tabel 4.30 Rancangan Uji Coba Banyaknya Generasi.....	67
Tabel 4.31 Rancangan Uji Coba Tipe Algoritma Evolution Strategies.....	67
Tabel 4.32 Rancangan Uji Coba Pengaruh Persmaan Fitness.....	69
Tabel 4.33 Rancangan Uji Coba Perbandingan nutrisi.....	69
Tabel 5.1 Spesifikasi Perangkat Keras	70
Tabel 5.2 Spesifikasi Perangkat Lunak	70
Tabel 6.1 Hasil Uji Coba Rentang Kromosom.....	80
Tabel 6.2 Hasil Uji Coba Ukuran Populasi	82
Tabel 6.3 Hasil Uji Coba Ukuran <i>offspring</i>	83
Tabel 6.4 Hasil Uji Coba Jumlah Generasi.....	85
Tabel 6.5 Hasil Uji Coba Tipe Algoritma Evolution Strategies.....	86
Tabel 6.6 Uji Coba Persamaan Fitness	87
Tabel 6.7 Rekomendasi Komposisi Bahan Pakan.....	88
Tabel 6.8 Perbandingan Nutrisi yang Tersedia dan Dibutuhkan	88
Tabel 6.9 Hasil Rekomendasi dari Bahan yang direkomendasikan.....	89

DAFTAR GAMBAR

Gambar 2.1 Contoh Gambar jenis jenis pakan ikan.....	7
Gambar 2.2 Contoh Pakan Buatan.....	9
Gambar 2.3 Skema Umum Proses Algoritma Evolusi	18
Gambar 3.1 Tahapan Penelitian.....	25
Gambar 3.2 Alur Sistem	27
Gambar 4.1 Flowchart Proses algoritma <i>evolution strategies</i>	31
Gambar 4.2 Flowchart Generate Populasi Awal	32
Gambar 4.3 <i>Flowchart</i> proses reproduksi.....	33
Gambar 4.4 <i>Flowchart</i> menghitung nilai <i>fitness</i>	34
Gambar 4.5 <i>Flowchart</i> menghitung nilai <i>fitness</i> (lanjutan)	35
Gambar 4.6 <i>Flowchart</i> Proses Seleksi	36
Gambar 4.7 Rancangan Tampilan Halaman Inout Data Ikan Bahan.....	63
Gambar 4.8 Rancangan Antarmuka Halaman Algoritma Genetika	64
Gambar 5.1 Tampilan Halaman Input Data Ikan.....	77
Gambar 6.1 Hasil Uji Coba Rentang Kromosom	81
Gambar 6.2 Grafik Hasil Uji Coba Ukuran Populasi	82
Gambar 6.3 Grafik Hasil Uji Coba Ukuran <i>Offspring</i>	84
Gambar 6.4 Hasil Uji Coba Jumlah Generasi.....	85
Gambar 6.5 Hasil Uji Coba Tipe Algoritma Evolution Strategies	87

DAFTAR LAMPIRAN

LAMPIRAN A DATA YANG DIGUNAKAN	95
A.1 Kebutuhan Nutri Ikan Bandeng	95
A.2 Kebutuhan Nutrisi Ikan Nila.....	95
A.3 Dosis Pakan Ikan Bandeng.....	95
A.4 Dosisi Pakan Ikan Nila	95
A.5 Daftar Harga Bahan Pakan.....	96
LAMPIRAN B HASIL WAWANCARA.....	97
LAMPIRAN C NILAI FITNESS HASIL UJI COBA.....	99
C.1 Hasil Uji Coba Rentang Kromosom	99
C.2 Hasil Uji Coba Ukuran Populasi.....	100
C.3 Hasil Uji Coba Ukuran <i>Offspring</i>	101
C.4 Hasil Uji Coba Jumlah Generasi	102
C.5 Hasil Uji Coba Siklus Algoritma Evolution Strategies.....	103
C.6 Hasil Uji Coba Persamaan Fitness.....	103
LAMPIRAN D INDIVIDU HASIL UJI COBA KE-10	104
D.1 Hasil Uji Coba ke-10 Rentang Kromosom 1-100	104
D.2 Hasil Uji Coba ke-10 Ukuran Populasi 30	106
D.3 Hasil Uji Coba ke-10 Ukuran <i>Offspring</i> 8μ	108
D.4 Hasil Uji Coba ke-10 Jumlah Generasi 500	110
D.5 Hasil Uji Coba ke -10 Siklus Algoritma Evolution Strategies ($\mu/r+\lambda$) .	112
D.6 Hasil Uji Coba ke-10 Persamaan Fitness	114
LAMPIRAN E HASIL PERHITUNGAN SISTEM	117
E.1 Individu awal	117
E.2 Individu hasil rekombinasi	117
E.3 Individu Hasil mutasi.....	120
E.4 Individu Hasil Seleksi Generasi ke 1.....	123
E.5 Individu Hasil Seleksi Generasi ke 500.....	124

BAB 1 PENDAHULUAN

Bab ini menjelaskan mengenai latar belakang permasalahan, rumusan masalah, tujuan, manfaat, batasan masalah serta sistematika penulisan.

1.1 Latar Belakang

Dewasa ini masyarakat yang mengkonsumsi ikan terus meningkat. Menurut data statistik Kementerian Kelautan dan Perikanan konsumsi ikan penduduk Indonesia tahun 2010-2014 terus mengalami peningkatan yaitu pada tahun 2010 konsumsi ikan mencapai 30,48 Kg/Kapita/Tahun, tahun 2011 mencapai 32,25Kg/Kapita/Tahun, tahun 2012 mencapai 33,89Kg/Kapita/Tahun, tahun 2013 mencapai 35,21 Kg/Kapita/Tahun dan tahun 2014 mencapai 38,00 Kg/Kapita/Tahun. Konsumsi ikan terus meningkat karena harga ikan relatif murah dan dapat memenuhi gizi masyarakat. Dalam membudidayakan ikan ada yang dikenal dengan istilah budidaya sistem polikultur

Menurut Afrianto dan Liviawaty (1998) sistem polikultur yaitu pada satu kolam dipelihara berbagai jenis ikan yang membutuhkan jenis makanan yang berbeda sehingga setiap jenis ikan tidak akan bersaing dalam mencari makanan. Menurut Morachman, *et al.*, (2010) polikultur merupakan salah satu alternatif untuk mengatasi permasalahan air yang mengakibatkan penurunan produksi ikan di kolam.

Sistem polikultur dapat mengefisienkan lahan dan dapat menguntungkan karena dalam satu lahan dapat membudidayakan lebih dari satu jenis. Selain itu hasil produksi akan lebih tinggi karena dalam satu periode pembudidayaan hasil pemanenan akan mendapatkan lebih satu jenis ikan. Contoh penerapan budidaya polikultur adalah pembudidayaan ikan dan udang seperti pada penelitian yang dilakukan oleh Kholifah, Trisyani dan Yuniar pada tahun 2008 yang meneliti tentang sistem polikultur ikan bandeng dan udang windu. Selain ikan bandeng dan udang windu terdapat juga penelitian tentang polikultur ikan nila dan udang vaname. Polikultur ikan nila dan udang vaname terdapat dalam penelitian yang dilakukan oleh junior *et al* pada tahun 2012 dan Simão *et al* (pada tahun 2013 yang meneliti tentang strategi pakan pada sistem polikultur nila dan udang. Untuk mendapatkan hasil pemanenan yang baik dapat dilakukan dengan meningkatkan produktifitas ikan. Untuk meningkatkan produktifitas ikan dapat menggunakan pakan buatan.

Menurut Wardhani, Safrizal dan Chairi (2011) salah satu upaya yang dapat dilakukan untuk meningkatkan produktivitas ikan dalam budidaya per unit lahan adalah melalui penggunaan pakan buatan, terutama ketika produksi pakan alami sudah tidak mampu lagi memenuhi kebutuhan nutrisi ikan yang ditebar dengan kepadatan tinggi. Tujuan dari pemberian pakan buatan pada ikan adalah untuk menyediakan kebutuhan gizi dan nutrisi ikan agar kesehatan dan pertumbuhan ikan baik. Pakan yang berkualitas, baik secara kegunaan maupun fisik merupakan kunci untuk mencapai tujuan produksi dan ekonomis budidaya ikan.

Pengetahuan tentang gizi ikan dan pakan ikan berperan penting dalam mendukung pengembangan budidaya ikan (Dharmawan, 2012).

Mahalnya harga pasaran pakan ikan produksi pabrik membuat banyak pembudidaya membuat pakan sendiri. Menurut Dharmawan (2012) biaya pakan ikan dapat mencapai 60% dari biaya produksi. Jika pakan dibuat sendiri maka akan mampu menekan biaya pembelian pakan. Pakan buatan merupakan campuran dari bahan-bahan pakan yang memiliki kandungan nutrisi dan harga yang berbeda-beda (Wardhani, LK, Safrizal, M dan Chairi, 2011). Namun, tidak semua pembudidaya mengetahui formulasi komposisi bahan pakan yang tepat sesuai dengan kebutuhan ikan. Kesalahan penentuan bahan-bahan pakan yang akan digunakan dapat berdampak pada rendahnya kandungan nutrisi dan tingginya biaya pakan buatan yang dihasilkan (Wardhani, LK, Safrizal, M dan Chairi, 2011).

Pakan yang tepat adalah pakan yang memenuhi kebutuhan ikan serta kandungan gizi dan nutrisi seperti kandungan protein, lemak, karbohidrat, serat kasar dan air. Bahan-bahan yang dapat digunakan untuk komposisi pakan ikan banyak tersedia dan nutrisi yang terkandung juga berbeda-beda. Dalam mencari optimasi komposisi pakan yang sesuai dengan kebutuhan nutrisi ikan yang baik diambil dari kombinasi bahan pakan yang terbaik. Terdapat penelitian sebelumnya yang membahas optimasi komposisi bahan pakan ikan yang dilakukan oleh Wardhani, Safrizal dan Chairi (2011). Dalam penelitian tersebut menunjukkan bahwa algoritma genetika dapat menyelesaikan permasalahan optimasi komposisi pakan ikan.

Algoritma *Evolution Strategies* (ES) merupakan cabang dari *evolution algorithm* yang baik digunakan untuk optimasi. ES merupakan algoritma yang sering digunakan untuk eksperimen empiris khususnya pada masalah-masalah optimasi numerik. Kecepatan proses ES lebih baik dibandingkan dengan GA untuk masalah optimasi bernilai real (Suyanto, 2008).

Algoritma *evolution strategies* pernah digunakan untuk optimasi pakan ternak sapi. Penelitian tersebut dilakukan oleh Milah dan Mahmudy (2015) yang menunjukkan ukuran populasi (μ) sebanyak 40, ukuran *offspring* $\lambda=4\mu$, dan banyak generasi 1000 memperoleh *fitness* tertinggi yaitu 0.203. Hasil akhir dari penelitian tersebut berupa rekomendasi komposisi pakan ternak sapi.

Berdasarkan permasalahan di atas, solusi yang ditawarkan adalah dengan mengoptimasi komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *Evolution Strategies*. Penelitian ini bertujuan untuk membuat komposisi pakan ikan yang bergizi sesuai dengan nutrisi yang dibutuhkan oleh ikan dengan biaya yang minimal.

1.2 Rumusan masalah

Berdasarkan uraian latar belakang di atas, maka dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana mengimplementasikan algoritma *Evolution Strategies* untuk optimasi komposisi pakan ikan sistem *polikultur* ?
2. Bagaimana nilai *fitness* yang dihasilkan oleh *Evolution Strategies* dalam mengoptimasi komposisi pakan ikan pada sistem budidaya *polikultur*?

1.3 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini adalah :

1. Untuk mengimplementasikan algoritma *Evolution Strategies* untuk optimasi komposisi pakan ikan pada sistem budidaya *polikultur*
2. Untuk mengetahui nilai *fitness* yang dihasilkan oleh *Evolution Strategies* dalam mengoptimasi pakan ikan pada sistem budidaya *polikultur*.

1.4 Manfaat

Penelitian ini diharapkan dapat bermanfaat untuk berbagai pihak. Manfaat dari penelitian ini adalah sebagai berikut:

1. Dapat membantu menentukan komposisi pakan ikan berdasarkan kebutuhan nutrisi yang sesuai dengan harga yang minimal.

1.5 Batasan masalah

Adapun batasan masalah pada penelitian ini adalah :

1. Penelitian ini digunakan untuk menentukan rekomendasi optimasi komposisi pakan ikan.
2. Pada penelitian ini hanya digunakan data ikan yang memungkinkan untuk dilakukan sistem budidaya *polikultur* yaitu data ikan bandeng dengan udang dan ikan nila dengan udang
3. Nutrisi yang digunakan untuk membuat komposisi pakan ikan adalah protein, lemak dan serat.
4. Pada penelitian ini menggunakan algoritma *Evolution Strategies* tanpa membandingkan dengan metode lain.
5. Pada penelitian ini representasi kromosom menggunakan representasi real code. Pada proses seleksi menggunakan metode seleksi *elitism*.

1.6 Sistematika penulisan

Sistematika penulisan pada penelitian ini tersusun sebagai berikut:

BAB I : Pendahuluan

Bab ini menjelaskan mengenai latar belakang permasalahan, rumusan masalah, tujuan, manfaat, batasan masalah, sistematika penulisan serta jadwal penelitian.

BAB II : Landasan Kepustakaan

Bab ini menjelaskan kajian pustaka dan dasar teori yang menjadi acuan dalam penulisan proposal yang diperoleh dari beberapa literatur, jurnal dan halaman website.

BAB III : Metodologi

Bab ini berisi metode atau langkah langkah yang digunakan dalam penelitian yang terdiri dari tahap penelitian, studi literature, pengumpulan data, analisis kebutuhan, perancangan sistem, implementasi sistem, pengujian, kesimpulan dan saran.

BAB IV : Perancangan Sistem

Bab ini berisi perancangan sistem yang terdiri dari formulasi permasalahan, siklus pemecahan masalah, contoh perhitungan manual, perancangan antar muka dan perancangan uji coba.

BAB V : Implementasi Sistem

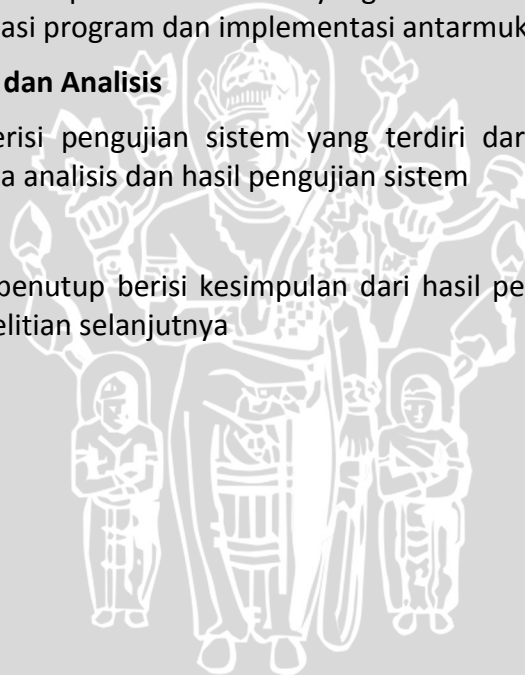
Bab ini berisi implementasi sistem yang terdiri dari spesifikasi sistem, implementasi program dan implementasi antarmuka.

BAB VI : Pengujian dan Analisis

Bab ini berisi pengujian sistem yang terdiri dari siklus pengujian sistem serta analisis dan hasil pengujian sistem

BAB VII : Penutup

Pada bab penutup berisi kesimpulan dari hasil penelitian dan saran untuk penelitian selanjutnya



BAB 2 LANDASAN KEPUSTAKAAN

Secara umum bab landasan kepastakaan berisi kajian pustaka dan dasar teori difokuskan pada pakan ternak ikan sebagai objek penelitian dan *Evolution Strategies* sebagai algoritma yang akan diimplementasikan dalam penelitian ini.

2.1 Kajian Pustaka

Telah terdapat beberapa penelitian yang membahas optimasi komposisi pakan ikan, penelitian tersebut dilakukan dengan berbagai macam objek dan metode. Penelitian pertama yang dilakukan oleh Wardhani, Safrizal, Chairi pada tahun 2011 tentang optimasi komposisi bahan pakan ikan menggunakan metode *multi-objective genetic algorithm*. Penelitian tersebut memiliki tujuan untuk meminimalkan perbedaan kebutuhan nutrisi dan efisiensi biaya yang dihasilkan. Berdasarkan pengujian yang telah dilakukan dengan parameter algoritma genetika yang bervariasi, penerapan metode pada aplikasi ikan dapat bekerja dengan rata-rata tingkat keberhasilan pemenuhan nilai gizi mencapai 100% dan tingkat efisiensi biaya mencapai 46.5%. Kombinasi optimal untuk menghasilkan pakan sejumlah 6kg tersebut dapat dicapai dengan menggunakan parameter panjang gen 5, jumlah kromosom 200, probabilitas *crossover* 0.01, probabilitas mutasi 0.2, probabilitas *elitism* 0.03 dan jumlah generasi 5. Perbedaan penelitian yang akan dilakukan dengan penelitian ini adalah pada penelitian yang akan dilakukan tujuan peneliti adalah untuk menemukan solusi optimal masalah pakan ikan yang dibudidayakan dengan sistem polikultur sedangkan pada penelitian ini menemukan solusi optimal untuk pakan ikan yang terdiri dari satu jenis ikan tawar dan algoritma yang digunakan adalah *multi-objective genetic algorithm* sedangkan dalam penelitian ini algoritma yang digunakan adalah *evolution strategies*.

Penelitian yang kedua yang dilakukan oleh Milah, Mahmudy pada tahun 2015 yang berjudul implementasi algoritma *Evolution Strategies* untuk optimasi komposisi pakan ternak sapi potong. Dalam penelitian tersebut berusaha untuk mengatasi masalah komposisi pakan ternak sapi potong menggunakan algoritma *Evolution Strategies (ES)*. Tipe proses ES yang digunakan adalah $(\mu/r + \lambda)$ sehingga proses reproduksi melibatkan rekombinasi dan hasil mutasi nantinya akan diikuti bersama parent dalam proses seleksi dan seleksi menggunakan *elitism selection*. Solusi optimal diperoleh dari ukuran populasi sebanyak 50, ukuran *offspring* 4, rentang kromosom 0-20 (*real code*) dan jumlah generasi 500 memperoleh rata-rata nilai *fitness* tertinggi yaitu 0.998. Hasil akhir berupa rekomendasi komposisi pakan ternak sapi potong. perbedaan penelitian yang akan dilakukan dengan penelitian ini adalah pada penelitian yang akan dilakukan objek dari penelitian adalah pakan ikan pada sistem budidaya polikultur sedangkan penelitian yang telah dilakukan objeknya adalah tentang pakan sapi.

Penelitian yang ketiga yang dilakukan oleh Abad dan Elipe pada tahun 2015 yang berjudul *Evolution strategies for computing periodic orbits*. Dalam penelitian tersebut Algoritma strategi evolusi yang dikembangkan untuk

mendeteksi orbit periodik dalam masalah dinamik. Algoritma ini diterapkan pada masalah gerak partikel di bawah medan gravitasi dari kawat melingkar yang solid. Teknik algoritma yang digunakan adalah $\mu/\rho + \lambda$. Hasil dari penelitian tersebut adalah Algoritma *Evolution Strategies* yang disesuaikan dapat mengkonversi masalah dalam mencari oriiit priodik ke dalam pencariin fungsi minimal. dapat mengurangi jumlah variabel yang digunakan, yang secara dramatis mengurangi waktu komputasi. Perbedaan antara penelitian ini dengan penelitian yang dilakukan adalah objek yang digunakan pada penelitian ini adalah priodik orbit sedangkan pada penelitian yang akan dilakukan adalah ikan sistem polikultur.

Untuk melihat lebih jelas persamaan dan perbedaan penelitian ini dengan penelitian sebelumnya berdasarkan studi literatur yang telah dilakukan ditunjukkan pada Tabel.2.1

Tabel 2.1 Perbandingan Penelitian Penulis dengan Penelitian Sebelumnya

No	Judul	Objek	Metode	Perbedaan	
				Studi Literatur	Skripsi Penulis
1	Optimasi Komposisi Bahan Pakan Ikan Air tawar Menggunakan Metode <i>Multi-Objective Genetic Algorithm</i>	Ikan Air Tawar	<i>Multi-Objective Genetic Algorithm</i>	Ikan Air Tawar, <i>Multi-Objective Genetic Algorithm</i>	Ikan dengan sitem polikultur, Algoritma <i>Evolution Strategies</i>
2	Implementasi Algoritma <i>Evolution Strategies</i> Untuk Optimasi Komposisi Pakan Ternak Sapi Potong	Sapi Potong	<i>Evolution Strategies</i> , eknik ES ($\mu/r+\lambda$)	Sapi Potong, <i>Evolution Strategies</i> , Teknik ES ($\mu/r+\lambda$) reproduksi melibatkan mutasi dan rekombinasi	Ikan dengan sitem polikultur, Algoritma <i>Evolution Strategies</i> Teknik ES ($\mu+\lambda$) reproduksi hanya melibatkan mutasi tanpa rekombinasi

3	Evolution strategies for computing periodic orbits	Priodik orbit	Evolution Strategies, teknik ES ($\mu/\rho + \lambda$).	Priodik orbit, Evolution Strategies, teknik ES ($\mu/\rho + \lambda$).	Ikan dengan sistem polikultur, Algoritma <i>Evolution Strategies</i> Teknik ES ($\mu + \lambda$)
---	--	---------------	---	--	--

Sumber : Diolah oleh penulis.

Pada penelitian ini, penulis akan mencoba menyelesaikan masalah optimasi pakan sistem polikultur ikan dan udang dengan algoritma *evolution strategies*.

2.2 Pakan Ikan

Pakan adalah makanan atau asupan yang diberikan kepada hewan ternak atau peliharaan lain seperti ikan konsumsi. sehingga pakan merupakan sumber energi bagi pertumbuhan dan kehidupan makhluk hidup termasuk ikan konsumsi (Tim Penulis CMK & B. Prasetya W, 2015). Menurut Dharmawan (2012) ada enam jenis pakan ikan yaitu pakan hidup, pakan segar, pakan beku, pakan botolan, pakan beku kering dan pakan kering. Setiap jenis pakan memiliki keuntungan dan kerugian masing masing.



Gambar 2.1 Contoh Gambar jenis jenis pakan ikan

Sumber : Purwakusuma, 2012

Gambar 2.1 merupakan Gambar berbagai macam jenis pakan ikan kering dari mulai flake, Tabelt, hingga pellet.

Bahan baku pakan berdasarkan sifatnya, maka bahan baku dibagi menjadi 2 kelompok, yaitu bahan baku nabati dan bahan baku hewani. Sekitar 70-75 % bahan baku nabati merupakan biji-bijian dan hasil olahannya, 15 – 25% limbah industri makanan dan selebihnya berupa hijauan. Bahan pakan nabati sebagian merupakan sumber energi yang baik, dan sumber vitamin yang dibutuhkan untuk pertumbuhan ikan (Juharni, 2014).

Contoh bahan baku hewani dan bahan baku nabati (Dharmawan, 2012) :

1. Bahan Hewani
 - a. Tepung ikan
 - f. Ampas minyak hati ikan

- | | |
|--------------------------------|--------------------------------------|
| b. Tepung rebon dan benawa | g. Tepung dara |
| c. Tepung kepala udang | h. Silase ikan |
| d. Tepung anak ayam | i. Arang bulu ayam dan Tepung Tulang |
| e. Tepung kepompong ulat sutra | j. Tepung Bekicot |
2. Bahan Nabati
- | | |
|----------------------|--------------------------------|
| a. Dedak Padi | h. Tepung bungkil kacang tanah |
| b. Dedak Gandum | i. Bungkil Kelapa |
| c. Jagung | j. Biji Kapas |
| d. Centel/Sorghum | k. Tepung Daun Turi |
| e. Tepung Terigu | l. Tepung Daun Lamtoro |
| f. Tepung Kedelai | m. Tepung Daun Ketela Pohon |
| g. Tepung Ampas Tahu | |

Setiap bahan pakan terkandung zat makanan yang berbeda-beda sehingga bahan pakan memiliki kelebihan dan kekurangan pada suatu zat makanan tertentu. Menurut Dharmawan (2012) terdapat suatu pengelompokan bahan pakan berdasarkan zat makanan tertentu yaitu :

1. Bahan pakan sumber energi adalah suatu bahan pakan yang mempunyai kandungan karbohidrat, lemak, protein yang berenergi tinggi. Contoh bahan pakan yaitu jagung, sorghum, minyak dan bekatul
2. Bahan pakan sumber protein adalah bahan pakan yang kaya akan kandungan protein. Contohnya tepung ikan, tepung daging, tepung darah, tepung udang, bungkil kacang tanah dan lain lain

Bahan pakan sumber Vitamin adalah bahan pakan yang digunakan untuk melengkapi kebutuhan vitamin ikan.

2.3 Pakan Ikan Buatan

Pakan buatan adalah pakan yang dibuat khusus dengan formulasi tertentu berdasarkan pertimbangan kebutuhannya. Pembuatan pakan sebaiknya didasarkan pada pertimbangan kebutuhan nutrisi ikan, kualitas bahan baku, dan nilai ekonomis. Dengan pertimbangan yang baik, dapat dihasilkan pakan buatan yang disukai ikan, tidak mudah hancur dalam air, aman bagi ikan (Dharmawan, 2012).

Pakan buatan, merupakan pakan berbentuk pelet, fleke dan crumble, pakan ini dalam kondisi kering sehingga daya tahannya antara > 4 bulan, kandungan gizinya lengkap karena dibuat sesuai dengan kebutuhan. Jenis pakan inilah yang akan dikupas lebih mendalam (Juharni, 2014).



Gambar 2.2 Contoh Pakan Buatan

Sumber : Suyanto dan Takarina, 2009

Gambar 2.2 merupakan contoh pakan buatan yang disering disebut dengan nama pakan pellet. Pakan pellet dapat digunakan untuk ikan dan udang.

Berdasarkan tingkat kebutuhannya pakan ikan dapat dibagi menjadi tiga kelompok yaitu (Dharmawan, 2012):

1. Pakan tambahan yaitu pakan yang dibuat untuk memenuhi kebutuhan pakan.
2. Pakan suplemen, yaitu pakan yang dibuat untuk memenuhi komposisi nutrisi tertentu
3. Pakan utama yaitu pakan yang sengaja dibuat untuk menggantikan sebagian besar atau keseluruhan pakan alami

Pakan buatan yang dibutuhkan harus mempunyai formula yang lengkap, mengandung bahan-bahan yang dapat meningkatkan pertumbuhan dan mempertahankan sintasan kultivan yang pada akhirnya dapat meningkatkan produktifitas dan keuntungan. Hal ini dapat diperoleh dari pakan buatan yang dibuat dengan cermat dan perhitungan kandungan nutrisi yang teliti dari bahan-bahan penyusunnya (Juhrani, 2014).

Dalam membuat pakan buatan untuk ikan/udang, hal pertama yang harus dipertimbangkan adalah persyaratan bahan baku pakan yaitu (Juhrani, 2014) :

1. Bahan baku tidak mengandung racun. Bahan baku yang mengandung racun dapat menghambat pertumbuhan, ikan mabuk dan stress bahkan dapat menyebabkan kematian ikan/udang yang dipelihara secara masal.
2. Bahan baku pakan tidak boleh bersaing dengan bahan makanan manusia.
3. Bahan baku harus tersedia dalam waktu lama, atau tersedia secara kontinyu.
4. Harga bahan baku, walaupun dapat digunakan tetapi harganya mahal. Sebenarnya murah atau mahalnya bahan baku harus dinilai dari manfaat bahan baku tersebut. Sebagai contoh tepung ikan harganya memang mahal tetapi bila dibandingkan dengan nilai kegunaannya terutama kandungan proteinnya yang tinggi dan kelengkapan asam aminonya maka penggunaan tepung ikan menjadi murah.

5. Kualitas gizi bahan baku, menjadi persyaratan penting, walaupun harganya murah, dan tersedia cukup melimpah tetapi kandungan gizinya buruk, maka bahan baku seperti ini tidak dapat digunakan.

2.4 Aspek Nutrisi Pakan Ikan

Ikan akan tumbuh dengan baik bila semua kebutuhan nutrisinya terpenuhi secara maksimal (Tim Penulis CMK & B. Prasetya W., 2015). Aspek nutrisi pakan dibutuhkan oleh ikan yaitu, protein, karbohidrat, lemak dan serat.

2.4.1 Protein

Protein sangat diperlukan oleh tubuh ikan/udang, baik untuk pertumbuhan maupun untuk menghasilkan tenaga. Protein nabati (asal dari tumbuhan), lebih sulit dicerna dari pada protein hewani (asal dari hewan), hal ini disebabkan karena protein nabati terbungkus dalam dinding selulosa yang memang sukar dicerna (Juharni, 2014). Protein digunakan ikan secara efisien sebagai sumber energy (Tim penulis CMK B. Prasetya W., 2015)

Pada umumnya kebutuhan ikan terhadap protein dapat digolongkan secara garis besar sebagai berikut: 15-30% dari total pakan bagi ikan herbivore, dan 45% bagi ikan karnivora. Sedangkan untuk ikan muda diperlukan diet dengan kandungan protein 50% (Purwakusuma, 2012).

2.4.2 Karbohidrat

Pada ikan, karbohidrat diperlukan untuk pertumbuhan dan energi. Meskipun demikian, ikan tidak memerlukan karbohidrat dalam jumlah yang besar pada makanannya. Kebanyakan Karbohidrat diketahui malah dapat menghambat pertumbuhan ikan. Hal ini tampaknya berkaitan dengan kenyataan bahwa kandungan kadar Karbohidrat yang tinggi pada makanan ikan sering berkaitan rendahnya kadar nutrisi esensial lainnya (Purwakusuma, 2012).

Karbohidrat atau hidrat arang atau zat pati, berasal dari bahan baku nabati. Kadar karbohidrat dalam pakan ikan, dapat berkisar antara 10 – 50%. Kemampuan ikan untuk memanfaatkan karbohidrat ini tergantung pada kemampuannya untuk menghasilkan enzim pemecah karbohidrat (amilase) ikan karnivora biasanya membutuhkan karbohidrat sekitar 12 % sedangkan untuk omnivore kadar karbohidratnya dapat mencapai 50% (Juharni, 2014).

2.4.3 Lemak

Lemak merupakan sumber utama pada ikan. Lemak tersimpan dalam jaringan dan berfungsi untuk menjaga stamina yang prima pada ikan yang bersangkutan. selain itu juga sebagai media penyimpanan vitamin-vitamin yang larut dalam lemak, seperti vitamin A,D,E, dan K (Purwakusuma, 2012).

Nilai gizi lemak dipengaruhi oleh kandungan asam lemak esensiilnya yaitu asam-asam lemak tak jenuh atau PUFA (*Poly Unsaturated Fatty Acid*) antara lain asam oleat, asam linoleat dan asam linolenat. Asam lemak esensiil ini banyak

terdapat di tepung kepala udang, cumi-cumi dll. Kandungan lemak angat dipengaruhi oleh faktor usuran ikan, kondisi lingkungan dan adanya sumber tenaga lain. Kebutuhan ikan akan lemak bervariasi antara 4 – 18% (Juhrani, 2014).

2.4.4 Serat

Serat dibutuhkan dalam jumlah sedikit untuk membantu proses pencernaan. Ikan karnivora tidak bisa mencerna serat sama sekali, dan di rekomendasikan serat untuk karnivora tidak lebih dari 4%. Sedangkan untuk ikan herbivora di anjurkan untuk memberikan serat dengan kadar 5-10% (Purwakusuma, 2012).

2.5 Kebutuhan Nurisi Pakan Ikan dan Udang

Setiap Ikan memiliki kebutuhan zat-zat makanan yang berbeda-beda tergantung pada tujuan produksi (Dharmawan, 2012). Dalam penelitian ini ikan yang menggunakan sistem budidaya polikultur ikan bandeng dengan udang dan ikan nila dengan udang. Kebutuhan zat makanan pada pakan ikan bandeng ditunjukkan pada Tabel 2.2, kebutuhan zat makanan pada pakan ikan nila ditunjukkan pada Tabel 2.3 dan kebutuhan zat makanan pada pakan udang windu ditunjukkan pada Tabel 2.4.

Tabel 2.2 Komposisi Nutrisi Pakan Ikan Bandeng

Karakteristik Nutrisi	Diet I (Usia 1-4 Minggu)	Diet II (Usia 5-24 Minggu)
Protein, %	30	20
Lemak, %	8-10	7-8
<i>Cholesterol</i> , %	0.8-1.0	0.8-1.0
Abu, maks, %	13	13
Serat Kasar, maks, %	4	4
Energi, K.cal/kg	3.500	3.500

Sumber : Murtidjo, 2002

Tabel 2.2 berisi komposisi nutrisi yang harus ada pada bahan pakan ikan bandeng berdasarkan umur ikan bandeng. Nutrisi yang diperlukan adalah protein, lemak, *cholesterol*, Abu, Serat Kasar. Komposisi pakan ikan bandeng dikelompokkan menjadi dua yaitu Diet I dan Diet II. Diet I diberikan pada ikan bandeng berusia 1-4 minggu sedangkan Diet II diberikan pada usia 5-24 minggu.

Tabel 2.3 Komposisi Nutrisi Pakan Ikan Nila

Berat	umur	protein	lemak	serat	Berat
0-1 g	0-2 minggu (larva)	40%	10-15%	6-8%	0-1 g
1-10 g	2-4 minggu	35-40%	10-15%	6-8%	1-10 g

10-25 g	4-7 minggu	30-35%	10-15%	6-8%	10-25 g
25-200g	8-12 minggu	30-32%	10-15%	6-8%	25-200
>200g	> 12 minggu	28-30	10-15%	6-%	>200

Sumber : FAO

Tabel 2.3 berisi komposisi nutrisi yang ada pada bahan pakan ikan nila yang berdasarkan umur ikan nila yaitu umur 0-2 minggu dengan berat 0-1g dan lebih dari 2 minggu dengan berat lebih dari 1 g. Nutrisi yang dibutuhkan ikan nila adalah protein, lemak dan serat.

Tabel 2.4 Komposisi Nutrisi Pakan Udang Windu

Umur	Berat	Protein	Lemak	Serat	Abu
<=30 hari	4-9 g/ekor	38%	2.8%	3%	16%
31-50 hari	9-15 g/ekor	37%	2.8%	3%	16%
51-70 hari	15-20 gr /ekor	37%	2.8%	3%	16%
>=90 hari	>20 g/ekor	36%	2.8%	3%	16%

Sumber : Suyanto dan Takarina, 2009

Tabel 2.4 berisi komposisi nutrisi yang harus ada pada bahan pakan udang windu berdasarkan umur ikan bandeng dan berat udang bandeng. Nutrisi yang diperlukan dalam bahan pakan adalah protein, lemak, Serat, Abu.

2.6 Bahan Baku Pakan Ikan

Ada beberapa bahan baku yang dapat digunakan untuk membuat komposisi pakan ikan untuk memenuhi kebutuhan nutrisi ikan. Daftar bahan baku yang dapat digunakan disertai oleh nutrisi yang terkandung di dalam bahan pakan ditunjukkan pada Tabel 2.5

Tabel 2.5 Bahan Baku Pakan Ikan Dan Nutrisinya

No	Bahan Baku Pakan	Nutrisi		
		Protein	Lemak	Serat
1	Tepung Ikan	60.50%	9.40%	0.70%
2	Tepung Kepala Udang	53.74%	6.65%	14.61%
3	Tepung Darah	71.45%	0.42%	7.95%
4	Arang Bulu Ayam	25.54%	3.80%	1.80%
5	Tepung Tulang	25.54%	3.80%	1.80%
6	Dedak Padi	11.35%	12.15%	24.46%
7	Dedak Gandum	11.99%	1.48%	3.75%
8	Jagung	8.50%	3.80%	2.20
9	Cental/Sorgum	13.0%	2.05%	13.5%
10	Tepung Kedelai	39.6%	14.3%	2.8%

11	Bungkil kedelai	44.0%	0.8%	7.3%
12	Tepung Ampas Tahu	23.55%	5.54%	16.53%
13	Tepung Bungkil Kacang Tanah	50.70%	10.9%	3.6%
14	Bungkil Kelapa	17.09%	9.44%	3.6%
15	Biji Kapuk/Randu	27.4%	5.6%	25.3%
16	Tepung Daun Turi	27.54%	4.73%	14.01%
17	Tepung Daun Lamtoro	36.82%	5.4%	18.14%
18	Bekatul	12.90%	13.00	11.40%

Sumber : Dharmawan, 2012

2.7 Konsumsi Makanan Harian Ikan

Ikan-ikan herbivore dan pemakan planghton nabati (phytoplankton), jumlah konsumsi makanan hariannya berbobot lebih banyak dari pada ikan karnivora. Hal ini disebabkan karena makanan nabati itu nilai kalorinya lebih rendah dari pada bahan makanan hewani. Selain itu, kandungan air bahan baku nabati jauh lebih tinggi dari pada bahan hewani (Dharmawan, 2012).

Menurut Dharmawan (2012) Jumlah makanan yang dikonsumsi oleh seekor ikan secara umum berkisar antara 5-6% berat tubuhnya perhari. Namun jumlah tersebut berubah-ubah tergantung pada suhu lingkungannya. Selain berpengaruh terhadap jumlah makanan yang dikonsumsi, suhu juga berpengaruh terhadap kegiatan metabolisme (Dharmawan, 2012).

Dalam Penelitian ini ikan yang dibudidayakan dengan sistem polikultur adalah ikan bandeng dengan udang windu dan ikan nila dengan udang. Untuk melihat dosis jumlah pakan yang dikonsumsi oleh ikan bandeng ditunjukkan pada Tabel 2.6, dosis pakan ikan nila ditunjukkan pada Tabel 2.7 dan dosis pakan untuk udang windu ditunjukkan pada Tabel 2.8

Tabel 2.6 Dosis Pakan Ikan Bandeng

Umur Ikan	Dosis pakan (%bb)
≤2 minggu	5%
>2 minggu	3%

Sumber : Tristian, 2011

Tabel 2.5 menampilkan data dosis pakan ikan bandeng dari biomasanya. sebanyak 5% dari biomasa yang diberikan pada 2 minggu pertama sedangkan untuk 2 minggu berikutnya pakan yang diberikan sebanyak 3% dari biomassa ikan.

Tabel 2.7 Dosis Pakan Ikan Nila

umur	berat	Dosis pakan
0-3 minggu	1-5 g	10-7 %
4-7 minggu	5-20g	6-4 %

8-10 minggu	20-100g	4-2.5 %
11-12minggu	100-200g	2.5-2%
13-17minggu	200-400g	2-1.5%

Sumber: Ghufran dan Kordi, 2013

Tabel 2.7 menunjukkan dosis pakan yang diberikan untuk ikan nila berdasarkan umur ikan nila. Terdapat 5 tingkatan kelompok umur ikan nila dan dosis pakan setiap tingkat umur berbeda beda.

Tabel 2.8 Dosis Pakan Udang Windu

Umur Udang (hari)	Berat Rata-rata Udang (gram)	Dosis Pakan (%bb)	Frekuensi Pemberian per Hari (kali)
1-15	0.005-1.0	75-25	2-3
16-30	1.1-2.5	25-15	2-3
31-45	2.6-5.0	15-10	3-4
45-60	5.1-8.0	10-7	3-4
61-75	8.1-14.0	7-5	4-5
76-90	14.1-20.0	5-3	4-5
91-105	20.1-26.0	5-3	4-6
106-120	26.1-30.1	4-2	4-6

Sumber : Suyanto dan Takarina, 2009

Tabel 2.6 menampilkan data data pemberian pakan udang yang disesuaikan dengan umur udang dan ukuran berat udang. Dari Tabel tersebut dapat dilihat bahwa semakin tua umur udang maka dosis yang diberikan semakin berkurang.

2.8 Budidaya Polikultur

Polikultur merupakan metode budidaya yang digunakan untuk memelihara banyak komoditas dalam satu lahan, seperti udang, bandeng, dan rumput laut. Melalui sistem ini, diperoleh manfaat, yaitu tingkat produktifitas lahan yang tinggi (Yasin, 2013).

Jika dilaksanakan sesuai dengan prinsip-prinsip budidaya ikan, pemeliharaan iakan dengan sistem polikultur dapat memberikan keuntungan bagi petani ikan, antara lain (Afrianto dan Liviawaty, 1988):

1. makanan alamiah yang tersedia di kolam dapat dimanfaatkan oleh ikan secara efektif, sehingga tidak ada lagi makanan alamiah yang terbuang sia sia.
2. penggunaan lahan menjadi efisien, karena dalam luas yang sama dapat dipelihara ikan dengan kepadatan yang lebih tinggi
3. secara keseluruhan, produksi kolam akan meningkat karena jumlah ikan yang dipelihara dalam satu kolam lebih banyak.
4. produksi tiap spesies ikan akan lebih tinggi bila dibandingkan dengan hasil pemeliharaan dengan sistem monokultur.

5. Tingkat kepadatan setiap spesies ikan pada sistem polikultur umumnya sama atau sedikit lebih rendah bila di bandingkan dengan tingkat kepadatan spesies tersebut pada sistem monokultur

Dalam praktek, sistem polikultur telah demikian berkembang, sehingga timbul beberapa variasi dari sistem ini, yaitu (Afrianto & Liviawaty, 1998):

1. Kombinasi spesies ikan yang berbeda dalam kebiasaan makan

Dalam kombinasi ini, ikan-ikan yang ditebarkan di kolam terdiri dari beberapa spesies yang berbeda dalam kebiasaan makan. Kombinasi ini dimaksudkan untuk memanfaatkan makanan alamiah yang terdapat di kolam secara efektif dan efisien.

Sebaiknya dipilih kombinasi spesies ikan yang mempunyai daerah operasi makanan di permukaan, di pertengahan, dan di dasar kolam. Dengan demikian seluruh makanan alamiah yang terdapat di kolam dapat dimanfaatkan dan tidak terjadi persaingan untuk mendapatkan makanan atau ruang gerak.

2. Kombinasi ikan yang berbeda ukuran

Dalam kombinasi ini, ikan yang ditebarkan terdiri dari satu spesies tetapi mempunyai ukuran yang berbeda, ini dilakukan karena setiap ukuran ikan mempunyai jenis makanan tertentu, meskipun ikan tersebut berasal dari satu spesies yang sama. Contohnya ikan mas (*Cyprinus carpio*). Ikan mas yang berukuran kecil biasa hidup di permukaan air dan makan plankton. Sedangkan ikan mas yang besar lebih menyukai hidup di dasar perairan dan mencari makanan dengan mengaduk ngaduk dasar kolam.

Contoh penerapan sistem budidaya polikultur adalah pembudidayaan ikan bandeng dan udang windu. Karena menurut Atjo dan Yusuf (2011) pemeliharaan udang windu secara monokultur sering terjadi kegagalan produksi sehingga untuk mengatasi masalah tersebut maka penambang menerapkan budidaya udang windu secara polikultur, salah satunya budidaya pembesaran bersamaan dengan ikan bandeng. Karena gerakan atau kepak an ikan bandeng bisa berfungsi sebagai kincir alternatif untuk pemasok oksigen terlarut maupun untuk menghindari terjadinya stratifikasi (pelapisan) suhu dalam air.

2.8.1 Jenis Ikan Sistem Polikultur

Untuk membudidayakan ikan secara polikultur harus memperhatikan kombinasi dari jenis ikan yang akan dibudidaya. Kombinasi ikan yang baik adalah kombinasi ikan yang tidak menimbulkan persaingan di antara ikan dalam mencari makan ataupun ruang gerak ikan (Afrianto dan Liviawaty, 1998).

Untuk menentukan kombinasi spesies ikan yang akan di polikultur harus mengetahui kombinasi spesies ikan di kolam. Jenis jenis ikan berdasarkan yang dimakannya (Afrianto dan Liviawaty, 1998)

1. ikan pemakan tumbuhan air adalah ikan tawes (*Puntius javanicus*), Gurami (*Oshphroneus gournamy*).

2. ikan pemakan plankton adalah tamakan (*helos-toma temmincki*), sepat siam (*Trichogaster pectoralis*)
3. ikan pemakan segala (*omnivor*) ikan mas (*Cyprinus carpio*), Nila (*Oreochromis niloticus*), Mujair (*Oreochormis mossambica*).
4. ikan pemakan daging (*carnivora*) adalah ikan lele (*Clarias batrachus*), tagih (*Mystus nemurus*)

Dalam penelitian ini ikan yang di budidayakan dengan sistem polikultur adalah ikan bandeng dengan udang dan ikan nila dengan udang. Menurut Murachman, *et al.* (2010) waktu penebaran benih ikan bandeng dan udang berbeda. Penebaran yang pertama kali di lakukan adalah penebaran ikan bandeng. Penebaran udang windu dilakukan tujuh hari setelah penebaran ikan bandeng.

2.8.2 Padat Tebar Ikan dan Udang yang Dipelihara dengan Sitem Polikultur

1. Padat tebar ikan bandeng dan udang windu

Pada penelitian yang dilakukan oleh Kholifah, Trisyani dan Yuniar pada tahun 2008 dengan judul pengaruh padat tebar yang berbeda terhadap kelangsungan hidup dan pertumbuhan pada polikultur udang windu (*penaeus monodon fab*) dan ikan bandeng (*chanos chanos*) di tambak brebes - jawa tengah. Penelitian ini bertujuan untuk mempelajari efek padat penebaran untuk tingkat kelangsungan hidup dan pertumbuhan udang windu (*penaeus monodon fab*) dan ikan bandeng (*chanos chanos*).

Hasil penelitian menunjukkan bahwa padat penebaran yang berbeda memiliki efek nyata pada tingkat kelangsungan hidup udang tetapi tidak memiliki efek nyata pada tingkat ketahanan hidup dari ikan dan tidak memberikan efek pada pertumbuhan ikan bandeng dan udang windu. Dari hasil penelitian dapat diketahui bahwa semakin tinggi pada tebar maka semakin rendah kelangsungan hidup ikan bandeng dan udang windu. Pemberian dengan kepadatan penebaran ikan 25 ekor/m³ dan udang 20 ekor/m² adalah yang terbaik.

Berdasarkan penelitian tersebut maka dalam penelitian ini perbandingan yang optimal untuk padat tebar ikan bandeng dan udang menggunakan perbandingan 2,5:20.

2. Padat tebar ikan nila dan udang vaname

Pada penelitian yang di lakukan junior et al (2012) Penelitian yang dilakukan bertujuan untuk mengevaluasi produktivitas, kinerja pertumbuhan dan kelayakan ekonomi dari polikultur nila dan udang vaname. Pada penelitian ini dianalisis padat tebar ikan dan udang dan pengaruhnya terhadap produktivitas, pertumbuhan dan kelayakan ekonomi yang dihasilkan berdasarkan padat tebar tersebut. Hasil dari penelitian tersebut diperoleh perbandingan padat tebar ikan

dan udang adalah (2T:9S) untuk budidaya polikultur. Kombinasi padat tebar (2T:9S) memberikan hasil keuntungan yang signifikan dibandingkan dengan budidaya monokultur namun dengan produktivitas dan pertumbuhan yang tidak jauh berbeda dengan sistem monokultur.

Berdasarkan penelitian tersebut maka dalam penelitian ini perbandingan yang optimal untuk padat tebar ikan nila dan udang menggunakan perbandingan 2:9.

2.8.3 Strategi Pakan ikan sistem polikultur Ikan dan Udang

Dalam proses penentuan kebutuhan pakan dalam budidaya polikultur ikan dan udang biasanya ditentukan berdasarkan budidaya ikan yang diutamakan. Pada penelitian yang dilakukan oleh Simão *et al* (2013) yang berjudul padat tebar dan strategi pakan dalam polikultur nila dan udang dalam tanki (*Stocking densities and feeding strategies in shrimp and tilapia polyculture in tanks*). Pada penelitian dilakukan dua percobaan.

Hasil dari penelitian tersebut menunjukkan bahwa pada percobaan pertama yang menjadikan udang sebagai budidaya utama dan kebutuhan pakan diberikan berdasarkan kebutuhan udang memberikan dampak yang buruk pada pertumbuhan udang karena nila mengganggu pertumbuhan udang. Sedangkan pada percobaan kedua yang menjadikan nila sebagai budidaya utama dan kebutuhan pakan berdasarkan kebutuhan nila memberikan dampak yang baik karena udang mampu bertahan hidup tanpa mengganggu kinerja nila. Hal ini tentu menjadikan polikultur menjadi lebih efisien baik dengan kombinasi (2:2.5/m²) atau (2:5/m²) dengan pakan berdasarkan kebutuhan dari ikan.

Pada penelitian yang kedua dilakukan oleh Junior *et al* (2012) tentang polikultur nila dan udang dengan padat tebar berbeda (*Polyculture of Nile tilapia and shrimp at different stocking densities*). Dalam penelitian ini nila dianggap sebagai budidaya yang diutamakan atau *main culture* sehingga pemberian pakan berdasarkan kebutuhan nutrisi dan biomas ikan. Penelitian yang dilakukan bertujuan untuk mengevaluasi produktivitas, kinerja pertumbuhan dan kelayakan ekonomi dari polikultur nila dan udang. Hasil analisis ekonomi pada penelitian ini sistem polikultur dengan padat tebar nila 2m², udang 9 dan 12 per m² memberikan keuntungan lebih besar dibandingkan dengan budidaya monokultur ikan nila dengan padat tebar yang sama yaitu 2m².

Berdasarkan dua penelitian tersebut maka dalam penelitian ini strategi pemberian kebutuhan pakan dari budidaya polikultur ikan dan udang menggunakan kebutuhan nutrisi dan biomass dari ikan.

2.9 Algoritma Evolusi

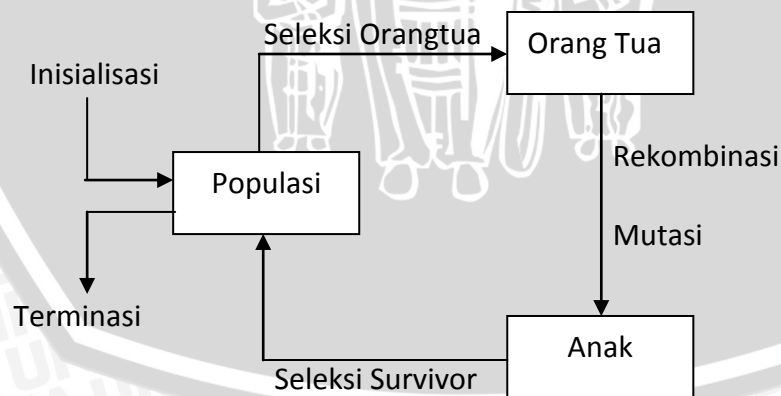
Dalam kecerdasan buatan, *Evolutionary Algorithms (EAs)* merupakan sebuah algoritma optimasi berbasis populasi metaheuristik generik. EAs menggunakan beberapa mekanisme yang terinspirasi oleh evolusi biologi: reproduksi, mutasi, rekombinasi, dan seleksi (Rifqi, Maharani dan Shaufiah, 2011).

Algoritma Evolusi (*evolutionary algorithms, EAs*) merupakan teknik optimasi yang meniru proses evolusi biologi. Menurut teori evolusi terdapat sejumlah individu dalam populasi. Dari generasi ke generasi, individu-individu ini berperan sebagai induk (*parent*) yang melakukan reproduksi menghasilkan keturunan (*offspring*). Individu-individu ini (beserta *offspring*) berevolusi dan individu-individu yang lebih baik (mampu beradaptasi dengan lingkungannya) mempunyai peluang lebih besar untuk melewati seleksi alam (*natural selection*) dan bertahan hidup. Individu yang lebih baik juga cenderung (tidak selalu tapi mempunyai kemungkinan lebih besar) menghasilkan keturunan yang lebih baik sehingga dari generasi ke generasi akan terbentuk populasi yang lebih baik (Mahmudy, 2013).

Pada setiap generasi, proses evolusi yang terjadi sebagai berikut (Suyanto, 2008) :

1. Dua individu dipilih sebagai orang tua (*parent*) menggunakan mekanisme tertentu. Kedua orang tua ini kemudian direkombinasi (*crossover*) untuk menghasilkan dua individu anak (*offspring*).
2. Dengan probabilitas tertentu, dua individu anak ini mungkin mengalami perubahan gen melalui operator mutasi (*mutation*).
3. Suatu *replacment scame* (skema penggantian individu) atau *survivor selection* (seleksi individu-individu yang bertahan hidup atau *servive* pada generasi berikutnya) tertentu diterapkan sehingga menghasilkan populasi baru
4. Proses ini akan terus berulang sampai kondisi berhenti (*stopping condition*) tertentu. Kondisi berhenti bisa berupa jumlah individu tertentu atau kriteria lainnya.

Proses algoritma evolusi dapat digambarkan oleh Gambar diagram 2.4



Gambar 2.3 Skema Umum Proses Algoritma Evolusi

Sumber : Suyanto, 2008

Pada Gambar 2.3 menggambarkan skema umum dari proses algoritma evolusi yang terjadi pada populasi yang terdiri dari individu kemudian terjadi siklus

seleksi orang tua, rekombinasi dan mutasi, serta *survivor*. Kondisi tersebut akan terus berulang hingga kondisi terminasi terpenuhi.

Menurut Suyanto (2008) algoritma evolusi dibutuhkan jika permasalahan yang memiliki satu atau lebih cirri-ciri berikut:

1. Ruang masalah yang sangat besar, kompleks, dan sulit dipahami.
2. Tidak bisa diselesaikan menggunakan metode-metode konvensional.
3. Terapat batasan waktu.
4. Solusi yang diharapkan tidak harus paling optimal atau bagus dan bisa diterima.
5. Kurang atau tidak ada pengetahuan yang memadai untuk mempersempit masalah kedalam ruang pencarian yang lebih sempit.
6. Tidak tersedia analisis matematika yang memadai.

Algoritma Evolusi memiliki banyak parameter yang harus ditentukan secara hati-hati diantaranya adalah : ukuran populasi, ukuran kromosom, probabilitas, seleksi orang tua jika menggunakan *tournament selection*, probabilitas rekombinasi atau *crossover* dan probabilitas mutasi (Suyanto, 2008)

2.10 Evolution Strategies (ES)

Menurut Suyanto (2008) algoritma *evolution strategies* (ES) diperkenalkan pertama kali oleh Ingo Rechenberg di Jerman pada era 1970-an. Ide-ide yang terdapat pada ES memang sangat mirip dengan GA. Tetapi Ingo Rechenberg mengembangkan ES secara terpisah dari GA. Berbeda dengan GA yang bisa menghasilkan perubahan signifikan, ES justru berbasis pada prinsip sebab-akibat (Suyanto, 2008).

Ciri utama *Evolution Strategies* (ES) adalah penggunaan vektor bilangan pecahan (*real-vector*) sebagai representasi solusi. Berbeda dengan GAs yang menggunakan *crossover* sebagai operator reproduksi utama dan mutasi sebagai operator penunjang, ES lebih bertumpu pada operator mutasi. Mekanisme *self-adaptation* digunakan untuk mengontrol perubahan nilai parameter pencarian. GAs dan ES bisa digunakan untuk menyelesaikan permasalahan yang sama. Tetapi mana yang terbaik di antara kedua metode tersebut sangat tergantung pada permasalahan yang dihadapi (Mahmudy, 2013).

2.10.1 Proses Evolution Strategies

Beberapa notasi digunakan oleh ES. μ (*miu*) menyatakan ukuran populasi (sama seperti *popSize* pada GAs). λ (*lamda*) menyatakan banyaknya *offspring* yang dihasilkan pada proses reproduksi (sama seperti *crossover rate* dan *mutation rate* pada GAs). Beberapa penelitian menyarankan besarnya nilai λ sebesar 7μ (Mahmudy, 2013).

Secara umum terdapat empat siklus dari algoritma ES, yaitu (Suyanto, 2008) :

1. (μ, λ) : Sejumlah μ orang tua menghasilkan sejumlah λ anak hanya menggunakan mutasi (tanpa rekombinasi). Seleksi *survivor* hanya dilakukan terhadap λ anak. Pada poses evolusi jenis ini, jumlah individu dalam populasi selalu tetap, yaitu sejumlah μ .
2. $(\mu/r, \lambda)$: sama dengan proses (μ, λ) , tetapi jumlah λ anak dihasilkan melalui proses rekombinasi dan mutasi.
3. $(\mu+\lambda)$: Sejumlah μ orang tua menghasilkan sejumlah λ anak hanya menggunakan mutasi (tanpa rekombinasi). Seleksi *survivor* hanya dilakukan gabungan anak dan orang tua $(\mu+\lambda)$. Pada poses evolusi jenis ini, jumlah individu dalam populasi selalu tetap, yaitu sejumlah μ .
4. $(\mu/r+\lambda)$: sama dengan proses $(\mu+\lambda)$, tetapi jumlah λ anak dihasilkan melalui proses rekombinasi dan mutasi.

2.10.2 Representasi Kromosom

Solusi dari suatu masalah harus dipetakan (encoding) menjadi string *chromosome*. String *chromosome* ini tersusun atas sejumlah gen yang menggambarkan variabel-variabel keputusan yang digunakan dalam solusi. Variabel keputusan (X_1 dan X_2) langsung menjadi gen string kromosom. Selain gen yang menyatakan variabel keputusan, parameter tambahan yang melekat pada setiap *chromosome* adalah σ (sigma). Nilai ini menyatakan level mutasi untuk *chromosome* tersebut. Nilai ini akan ikut berubah secara adaptif sepanjang generasi. Jika P adalah satu kromosom maka $P=(X_1, X_2, \sigma_1, \sigma_2)$ dengan panjang string sebesar 4 (Mahmudy, 2013).

2.10.3 Inisialisasi

Tahap populasi inisial dibangkitkan secara *random*. Tahap inisialisasi ini bertujuan untuk membangkitkan himpunan solusi baru secara acak yang terdiri dari sejumlah string kromosom dan ditempatkan pada penampungan yang disebut dengan populasi. Pada tahap ini μ yang menyatakan ukuran populasi harus ditentukan. Panjang string kromosom dihitung berdasarkan presisi variabel dan solusi yang dicari (Mahmudy, 2013). Contoh inisialisasi dengan menggunakan $\mu=4$ ditunjukkan pada Tabel 2.9:

Tabel 2.9 Contoh Inisialisasi

$P(t)$	x_1	x_2	σ_1	σ_2	$f(x_1, x_2)$
P1	1,48980	2,09440	0,14197	0,91090	19,8212830
P2	8,49170	2,57540	0,53801	0,86904	34,7060873
P3	-1,84610	1,70970	0,99835	0,49351	11,5863900
P4	5,81140	5,07790	0,40521	0,98911	14,5620828

Sumber : Mahmudy, 2013.

Tabel 2.9 di atas merupakan contoh inisialisasi yang nilai x_1 dan x_2 dibangkitkan secara acak sebagai bilangan pecahan. Pada Tabel tersebut terdapat suatu populasi yang terdiri dari 4 individu dari P1-P4 dimana setiap

individu terdiri dari dua gen x_1, x_2 , dua parameter σ_1, σ_2 yang menyatakan level dari kromosom dan memiliki nilai *fitness* untuk mengukur kualitas individu.

2.10.4 Reproduksi

Reproduksi bertujuan untuk menghasilkan keturunan dari individu-individu yang ada di populasi (Milah, Mahmudy, 2015). Disini proses ES menggunakan siklus $(\mu/r+\lambda)$ sehingga menggunakan rekombinasi dan mutasi dalam reproduksi.

2.10.4.1 Rekombinasi

Pada ES, rekombinasi dilakukan dengan dua cara, yaitu anak mewarisi gen gen orang tuanya (*intermediary*) dan anak yang mewarisi gen gen orang tuanya secara acak (*discrete*). Semua gen yang diwariskan bisa diperoleh dari dua orang tua secara tetap yang sering disebut dengan pewarisan lokal. Gen-gen yang diwariskan bisa juga diperoleh dari dua orang tua yang berubah-ubah yang sering disebut dengan pewarisan global (Suyanto, 2008).

Rekombinasi dilakukan untuk menghasilkan *offspring* sebanyak λ dari sejumlah μ individu dalam populasi. Setiap satu individu *offspring* dihasilkan dari beberapa induk. Induk dipilih secara acak dari populasi. Metode rekombinasi paling sederhana adalah dengan menghitung rata-rata nilai elemen induk (Mahmudy, 2013).

Contoh proses rekombinasi diberikan sebagai berikut (Mahmudy, 2013):

- Misalkan *offspring* didapatkan dari 2 induk. Jika P1 dan P3 terpilih maka akan didapatkan *offspring* $C=(-0,17815, 1,90205, 0,57016, 0,70221)$.
- Misalkan *offspring* didapatkan dari 3 induk. Jika P1, P2 dan P3 terpilih maka akan didapatkan *offspring* $C=(2,71180, 2,12650, 0,55944, 0,75782)$.

Pada studi kasus ini, misalkan $\lambda=6$ dan *offspring* didapatkan dari 2 induk. Contoh hasil rekombinasi ditunjukkan pada Tabel 2.10 :

Tabel 2.10 Contoh Hasil Rekombinasi

C(t)	Induk	x_1	x_2	σ_1	σ_2	$f(x_1, x_2)$
C1	P1 dan P3	-0,17815	1,90205	0,57016	0,70221	16,6418295
C2	P2 dan P3	3,32280	2,14255	0,76818	0,68128	19,5813015
C3	P1 dan P4	3,65060	3,58615	0,27359	0,95001	9,5700496
C4	P2 dan P4	7,15155	3,82665	0,47161	0,92907	12,5240357
C5	P1 dan P3	-0,17815	1,90205	0,57016	0,70221	16,6418295
C6	P3 dan P4	1,98265	3,39380	0,70178	0,74131	12,6500683

Sumber : Mahmudy, 2013

Tabel 2.10 di atas terdapat 6 *offspring* yang didapatkan dari 2 induk. Setiap *offspring* dari C1-C6 terdiri dari dua gen x_1, x_2 , dua parameter σ_1, σ_2 yang menyatakan level dari kromosom dan memiliki nilai *fitness* untuk mengukur

kualitas individu. Semua parameter dari *offspring* di dapatkan dari nilai rata-rata *parentnya* (P).

2.10.4.2 Mutasi

Mutasi merupakan proses yang sangat penting untuk menemukan solusi. Mutasi dilakukan untuk mengubah nilai gendengan menambah bilangan *random* yang dibangkitkan berdasarkan distribusi normal (Rifqi, Maharani dan Shaufiah, 2011).

Proses mutasi bisa menghasilkan individu baru yang berkualitas lebih buruk, sama atau menghasilkan individu baru yang lebih baik dari pada individu lama. Mutasi berfungsi untuk memanfaatkan kromosom-kromosom yang ada agar menjadi lebih baik. sehingga mutasi sering disebut operator eksploitasi karena dengan mutasi EAs seolah olah melakukan eksploitasi hanya di suatu lokasi pencarian tertentu (Suyanto, 2008).

Untuk menghitung nilai kromosom mutasi dapat menggunakan persamaan (2.1) (Mahmudy, 2013) :

$$\begin{aligned} X_1' &= X_1 + \sigma N(0,1) \\ X_2' &= X_2 + \sigma N(0,1) \end{aligned} \tag{2.1}$$

$N(0,1)$ merupakan bilangan acak yang mengikuti sebaran normal dengan rata-rata sebesar 0 dan standard deviasi sebesar 1. Pada program komputer, nilai $N(0,1)$ bisa didapatkan dengan membangkitkan dua bilangan *random* r_1 dan r_2 pada interval $[0,1]$. Rumus yang digunakan ditunjukkan pada persamaan (2.2) (Schwefel, 1995):

$$N(0,1) = \sqrt{-2 \cdot \ln r_1} \sin \pi r_2 \tag{2.2}$$

Misalkan $r_1 = 0,4749$ dan $r_2 = 0,3296$ maka didapatkan $N(0,1) = 1,0709$.

Nilai σ dinaikkan jika ada paling sedikit 20% hasil mutasi yang menghasilkan individu yang lebih baik dari induknya. Jika tidak maka nilai σ diturunkan. Misalkan $\sigma = 3 \times \mu = 12$, maka setiap individu dalam populasi akan menghasilkan 3 *offspring*. Pada kasus ini, nilai σ akan dinaikkan jika ada setidaknya 1 *offspring* yang lebih baik dari pada induknya (Mahmudy, 2013). Nilai σ' dari *offspring* yang dihasilkan jika lebih baik maka dinaikkan dengan cara $\sigma' = \sigma \times 1,1$. Jika hasilnya buruk maka diturunkan dengan cara $\sigma' = \sigma \times 0,9$ (Mahmudy, 2013). Contoh perhitungan mutasi jika diketahui $\lambda = 2\mu = 8$ ditunjukkan pada Tabel 2.11.

Tabel 2.11 Contoh Mutasi

C'(t)	N ₁ (0,1)	N ₂ (0,1)	x' ₁	x' ₂	σ'1	σ'2	f(x1,x2)
C1	0,1885	0,2747	-0,07068	2,094946	0,62717	0,77243	21,3386958
C2	-1,7947	-0,1359	1,944154	2,049965	0,84499	0,74940	19,9034447
C3	-0,5603	-1,8657	3,497309	1,813724	0,30095	1,04501	10,9809788
C4	0,6189	-0,4613	7,443427	3,398068	0,42445	0,83617	5,4075091
C5	-0,1371	-0,4201	-0,25632	1,607053	0,51314	0,63199	11,2620580

C6	1,1125	-0,2153	2,763377	3,234196	0,77195	0,81544	16,3293685
----	--------	---------	----------	----------	---------	---------	------------

Sumber : Mahmudy, 2013.

Tabel 2.11 merupakan contoh kasus mutasi. Tabel 2.11 di atas terdapat 6 *offspring* dari hasil mutasi yang terdiri dari parameter $N_1(0,1)$, $N_2(0,1)$, x'_1 , x'_2 , σ'_1 , σ'_2 dan nilai *fitness*.

2.10.5 Perhitungan *fitness*

Dalam proses ini setiap kromosom dalam populasi yang telah dibangkitkan dihitung nilai *fitnessnya*. Pada optimasi fungsi berkendala, penentuan rumus perhitungan *fitness* harus dilakukan secara tepat agar solusi optimum bisa ditemukan secara efisien (Mahmudy, 2013). Untuk menghasilkan *fitness* yang terbaik, perlu memahami batasan-batasan yang diberikan sehingga dapat menemukan prioritas dari batasan-batasan tersebut (Suyanto, 2008).

Pada proses evaluasi individu dilakukan berdasarkan fungsi *fitness*-nya. Untuk pemasalahan optimasi pencarian nilai maksimum maka fungsi *fitness* yang digunakan adalah fungsi itu sendiri. Persamaanya ditunjukkan pada persamaan (2.3) (Suyanto, 2008):

$$fitness=h \quad (2.3)$$

Sedangkan untuk permasalahan meminimalkan fungsi (minimasi) maka *fitness* tidak dapat digunakan secara langsung,. Maka untuk permasalahan meminimalkan fungsi adalah $f=1/h$, akan tetapi rumus *fitness* ini akan bermasalah jika h dapat bernilai 0 yang mengakibatkan *fitness* bernilai tak higa. Untuk itu dapat menggunakan persamaan yang ditunjukkan pada persamaan (2.4) (Suyanto, 2008):

$$Fitness = \frac{1}{h+a} \quad (2.4)$$

Dimana h adalah fungsi yang akan di minimalkan sedangkan a adalah bilangan yang dianggap sangat kecil yang berfungsi untuk menghindari pemagian dengan 0. Tujuan pada penelitian ini adalah untuk membuat optimasi komposisi bahan pakan dengan meminimalkan harga serta mempertimbangkan nutrisi pakan ikan sehingga permasalahan *fitness* yang digunakan adalah pesmaan untuk meminimalkan fungsi (minimasi).

2.10.6 Seleksi

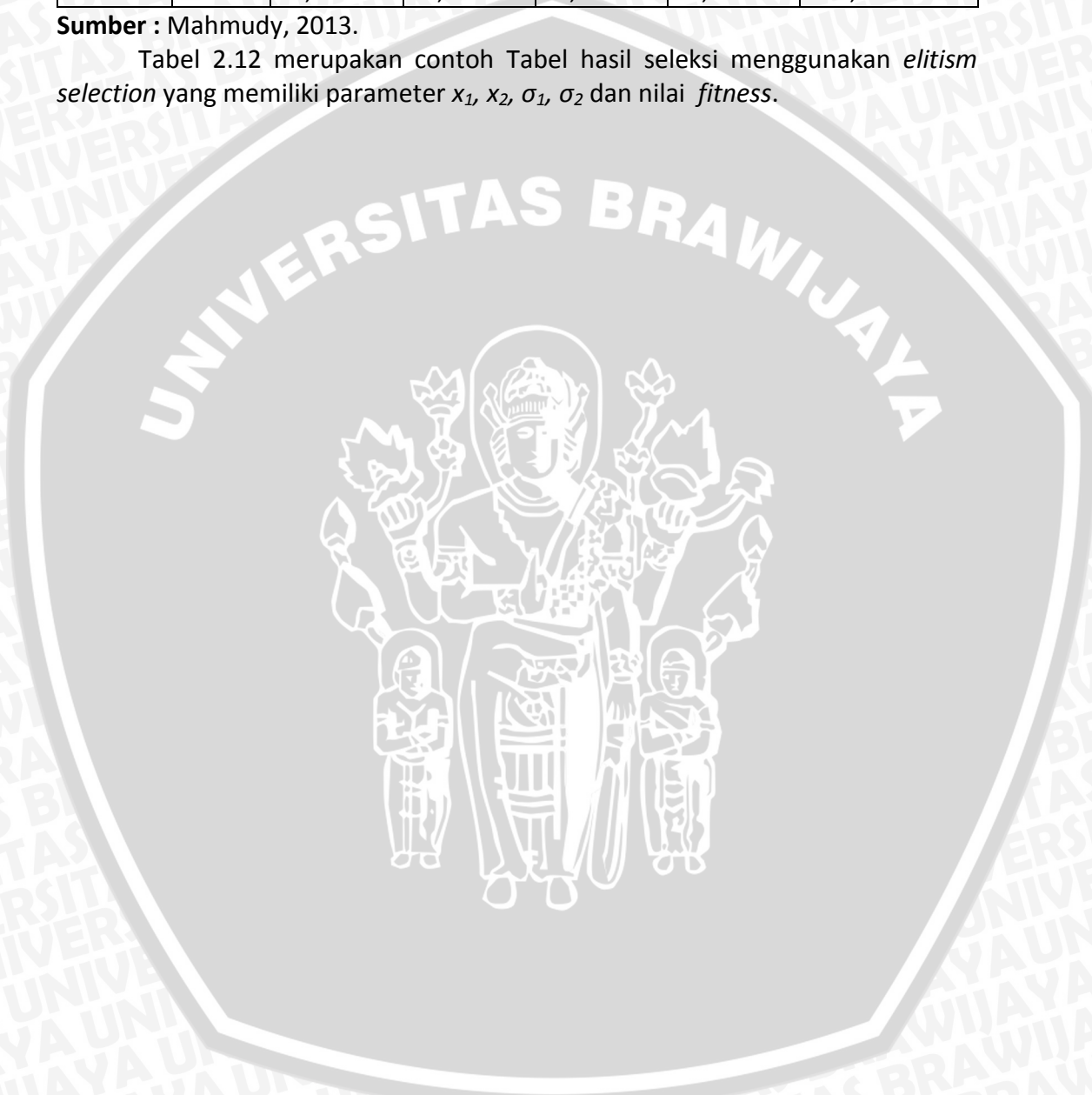
Seleksi menggunakan *elitism selection* melibatkan individu dalam *offspring* dan individu induk dalam populasi. Metode seleksi elitsim bekerja dengan cara mengumpulkan semua individu dalam populasi (*parent*) dan *offspring* dalam satu penampungan. Metode ini melakukan seleksi pada individu-individu dalam penampungan berdasarkan nilai *fitness* tertinggi. individu terbaik dalam penampungan akan lolos untuk masuk dalam generasi berikutnya. Metode seleksi elitism menjamin individu yang terbaik akan selalu lolos (Mahmudy, 2013). Contoh hasil seleksi menggunakan metode *elitism selection* ditunjukkan pada Tabel 2.12.

Tabel 2.12 Contoh Seleksi

P(t+1)	Asal	x1	x2	σ_1	σ_2	f(x1,x2)
P1	P2	8,49170	2,57540	0,53801	0,86904	34,7060873
P2	C1	-0,07068	2,09495	0,62717	0,77243	21,3386958
P3	C2	1,94415	2,04996	0,84499	0,74940	19,9034447
P4	P1	1,48980	2,09440	0,14197	0,91090	19,8212830

Sumber : Mahmudy, 2013.

Tabel 2.12 merupakan contoh Tabel hasil seleksi menggunakan *elitism selection* yang memiliki parameter x_1 , x_2 , σ_1 , σ_2 dan nilai *fitness*.

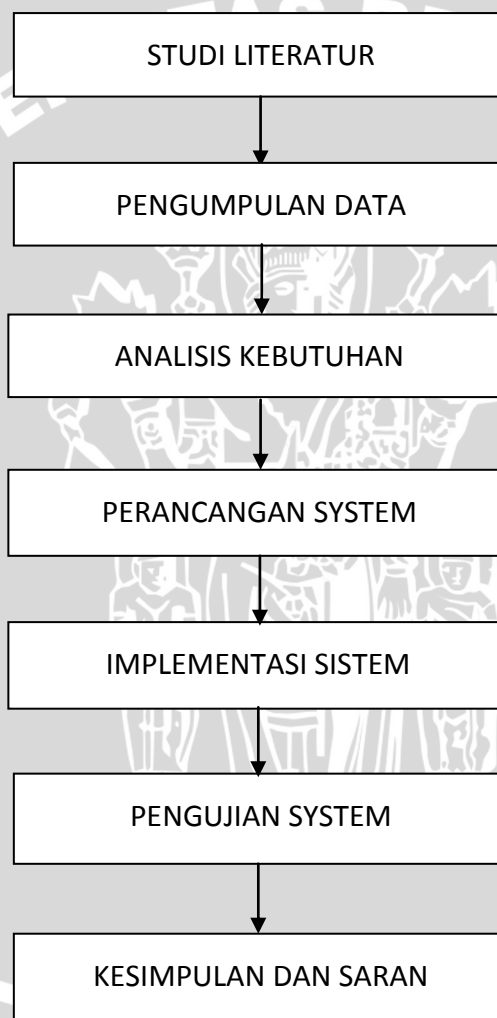


BAB 3 METODOLOGI

Pada bab ini akan dibahas tentang metode atau langkah langkah yang digunakan dalam penelitian yang terdiri dari tahap penelitian, kebutuhan sistem, deskripsi umum sistem, data yang digunakan, perancangan sistem, implementasi dan pengujian.

3.1 Tahapan Penelitian

Tahapan penelitian ini meliputi, studi literatur, pengumpulan dan analisis data, analisis dan perancangan sistem, pengujian sistem, dan kesimpulan dan saran. Tahapan penelitian dapat dilihat seperti Gambar 3.1



Gambar 3.1 Tahapan Penelitian

Berdasarkan Gambar 3.1 penjabaran langkah-langkah yang dilakukan dalam penelitian ini adalah melakukan studi literatur, pengumpulan data, analisis kebutuhan, implementasi sistem, perancangan sistem, pengujian sistem dan kesimpulan dan saran.

3.2 Studi Literatur

Studi Literatur pada penelitian ini mempelajari literatur dari beberapa bidang ilmu yang berhubungan dengan pembuatan sistem untuk optimasi pakan ikan sistem polikultur dengan algoritma *evolution strategies*. Studi literatur yang dipelajari adalah pakan ikan, ikan dengan sistem polikultur dan algoritma *evolution strategies*. Literatur yang digunakan pada penelitian ini didapat dari buku, jurnal dan internet.

3.3 Pengumpulan Data

Pengumpulan data yang akan digunakan untuk optimasi komposisi pakan sistem polikultur ikan dan udang diambil dari buku, jurnal dan survei di pasar splendid kota malang pada bulan oktober 2015. Data yang diperoleh dari buku adalah data nutrisi ikan, data dosis pemberian pakan dan data bahan pakan disertai nutrisinya. Data yang diperoleh dari jurnal adalah data perbandingan ikan bandeng dan udang windu dalam sistem budidaya polikultur dan data perbandingan ikan nila dan udang vannamei dalam sistem budidaya polikultur. Data yang diperoleh dari buku dan jurnal telah di konsultasikan ke dosen Fakultas Perikanan dan Ilmu Kelautan, Universitas Brawijaya. Data yang diperoleh dari survei adalah data harga bahan baku pakan.

3.4 Analisis Kebutuhan

Analisis Kebutuhan merupakan tahap untuk menganalisis kebutuhan sistem yang dibutuhkan dalam mengimplementasikan sistem optimasi komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *evolution strategies*.

3.4.1 Deskripsi Umum Sistem

Sistem yang akan dibangun yaitu perangkat lunak yang akan menyelesaikan masalah optimasi komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *Evolution Strategies*. Data yang dibutuhkan dalam sistem ini berupa Tabel kebutuhan nutrisi ikan, daftar kandungan nutrisi setiap bahan pakan yang ada, daftar dosis pemberian pakan dan data daftar harga bahan pakan.

Sistem ini memerlukan inputan oleh user agar sistem dapat berjalan. Masukan dari user adalah, umur ikan, jumlah ikan dalam kolam, berat ikan. Sedangkan dari sisi algoritma *Evolution Strategies* masukan yang dibutuhkan adalah rentang kromosom, jumlah populasi(μ), hasil *offspring*(λ) dan jumlah generasi. setelah menerima inputan maka sistem akan memproses inputan user yang akan digunakan untuk menghitung optimasi komposisi pakan sistem polikultur ikan dan udang. Output dari sistem ini akan menampilkan rekomendasi komposisi bahan pakan sistem polikultur ikan dan udang yang sesuai dengan nutrisi yang di butuhkan dengan harga yang minimum.

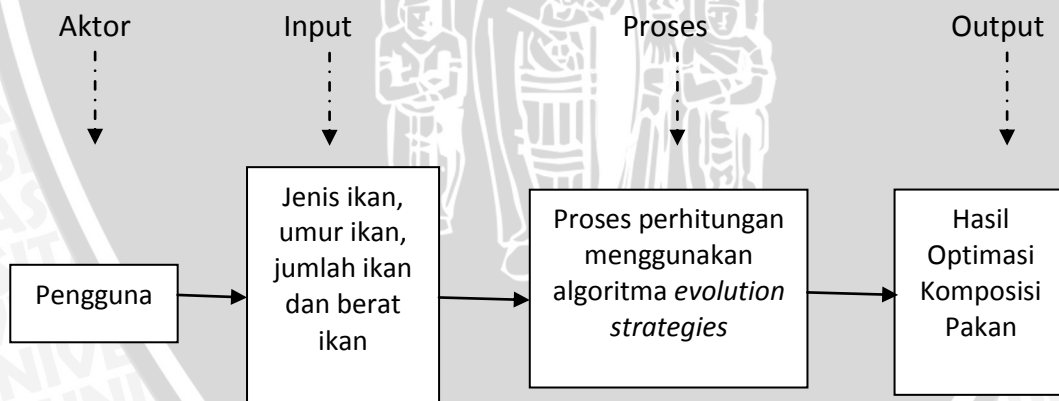
3.4.2 Data yang Digunakan

Dalam penelitian ini, data yang digunakan adalah sebagai berikut:

1. Data kebutuhan nutrisi pakan ikan yang bersumber dari buku yang akan digunakan dalam proses perhitungan untuk mencari komposisi pakan sistem polikultur ikan dan udang yang optimal.
2. Data frekuensi dosis pemberian pakan ikan yang bersumber dari buku yang akan digunakan untuk menghitung jumlah pakan yang dibutuhkan.
3. Data bahan pakan ikan dan udang disertai nutrisinya yang bersumber dari buku yang akan digunakan dalam proses perhitungan untuk mencari komposisi pakan ikan yang optimal.
4. Data harga bahan pakan ikan yang bersumber dari survei harga di pasar splendid yang akan digunakan dalam proses perhitungan untuk mencari komposisi pakan ikan yang optimal dengan harga yang minimum.
5. Data perbandingan ikan bandeng dan udang windu serta ikan nila dan udang vannamei yang bersumber dari jurnal penelitian yang akan digunakan untuk merekomendasi padat tebar udang yang optimal dalam sistem polikultur.

3.5 Perancangan Sistem

Berdasarkan analisis kebutuhan yang telah dijelaskan pada sub bab sebelumnya. Arsitektur perancangan sistem untuk optimasi komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *evolution strategies* ditunjukkan pada Gambar 3.2



Gambar 3.2 Alur Sistem

Gambar 3.2 merupakan Gambar alur sistem secara umum. Pertama pengguna sebagai aktor akan menginputkan data-data yang dibutuhkan seperti jenis ikan, umur ikan, jumlah ikan, dan berat ikan. Setelah menerima inputan kemudian sistem akan melakukan proses perhitungan optimasi komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *evolution strategies*. Output yang dihasilkan adalah rekomendasi optimasi komposisi pakan ikan yang

memiliki nutrisi yang sesuai oleh ikan dan udang yang dipolikultur dengan harga bahan pakan yang minimum.

3.6 Implementasi Sistem

Pada tahap ini akan dilakukan proses implementasi berdasarkan perancangan yang sudah dibuat. Implementasi dilakukan dengan menggunakan bahasa pemrograman java dan manajemen database menggunakan Mysql. Tahapan implementasi sistem dimulai dari implementasi antarmuka dari sistem, kemudian implementasi perhitungan optimasi pakan ikan menggunakan algoritma *evolution strategies*.

3.7 Pengujian

Pada tahap ini akan dilakukan pengujian Sistem setelah semua proses selesai dilakukan. Pengujian *Evolution Strategies* dilakukan untuk menguji pengaruh rentang kromosom, pengaruh ukuran populasi (μ), pengaruh ukuran *offspring* (λ), pengaruh banyaknya jumlah generasi, pengaruh siklus algoritma *evolution strategies* yang digunakan, pengaruh persamaan fitness yang digunakan dalam memecahkan masalah optimasi komposisi pakan sistem polikultur ikan dan udang.

3.8 Kesimpulan dan Saran

Setelah tahapan perancangan sistem, implementasi dan pengujian selesai dilakukan maka yang tahap selanjutnya adalah menarik kesimpulan. Kesimpulan diperoleh berdasarkan hasil dari pengujian system dan analisis menggunakan algoritma *evolution strategies* terhadap optimasi komposisi pakan sistem polikultur ikan dan udang. Tahapan terakhir dari penelitian ini adalah saran. Saran digunakan untuk memberikan pertimbangan untuk penelitian selanjutnya.

BAB 4 PERANCANGAN SISTEM

Pada bab ini akan dibahas perancangan sistem yang terdiri dari formulasi permasalahan, siklus pemecahan masalah, contoh perhitungan manual, perancangan *database*, perancangan antarmuka, dan yang terakhir adalah perancangan uji coba dan evaluasi.

4.1 Formulasi Permasalahan

Optimasi pakan ikan sistem polikultur menggunakan *algorithm evolution strategies* terdiri dari beberapa proses berikut:

1. Melakukan proses perhitungan nutrisi yang dibutuhkan dan jumlah pakan yang dibutuhkan dengan memasukkan data, Jenis ikan, Umur ikan, Berat ikan, Banyak ikan dalam satu tambak persamaan yang digunakan untuk menghitung jumlah pakan ditunjukkan pada persamaan (4.1)

$$\text{Jumlah pakan} = (\text{dosis pakan} \times (\text{berat ikan} / \text{jumlah ikan}) / 100 \quad (4.1)$$

2. Menghitung rekomendasi padat tebar udang. Persamaan yang digunakan untuk menghitung rekomendasi ditunjukkan pada persamaan (4.2)

$$\text{Padat tebar udang} = (\text{jumlah tebar ikan} / \text{perbandingan tebar ikan} \times \text{perbandingan tebar udang} \quad (4.2)$$

3. Menginisialisasi parameter *evoution strategies*, yaitu :

a. Jumlah populasi (μ)

b. Ukuran *offspring* (λ)

4. Membuat populasi awal sesuai dengan jumlah populasi yang di masukan.
5. Membuat populasi baru menggunakan mutasi.
6. Menghitung nilai fitness masing masing individu. Untuk menghitung nilai *fitness* yang pertama dilakukakn adalah menghitung bobot bahan pakan ikan. Untuk menghitung bobot bahan menggunakan rumus (4.3) :

$$\text{Bobot bahan} = \frac{\text{gen}}{\text{total gen}} \times \text{jumlah_pakan} \quad (4.3)$$

Menghitung biaya bahan menggunakan rumus (4.4) :

$$\text{Biaya bahan} = \text{bobot bahan} \times \text{harga bahan} \quad (4.4)$$

Menghitung kandungan nutrisi tiap bahan menggunakan umusu (4.5) :

$$\text{Kandungan bahan} = \text{bobot bahan} \times (\text{nutrisi bahan} / 100) \quad (4.5)$$

Menghitung nutrisi yang dibutuhkan menggunakan rumus (4.6)

$$\text{Kebutuhan nutrisi} = (\text{nutrisi dibutuhkan} / 100) \times \text{jumlah_pakan} \quad (4.6)$$

Menghitung *penalty* nutrisi dari setiap bahan menggunakan rumus (4.7)

$$penalty = |nutrisi\ yang\ dibutuhkan - nutrisi\ yang\ tersedia| \quad (4.7)$$

Menghitung nilai *fitness* menggunakan rumus (4.8) (Milah, 2015)

$$fitness = \frac{10000}{total\ cost + (total\ penalty \times 10000)} \quad (4.8)$$

total cost adalah total harga bahan pakan setiap individu

total *penalty* adalah total *penalty* setiap individu

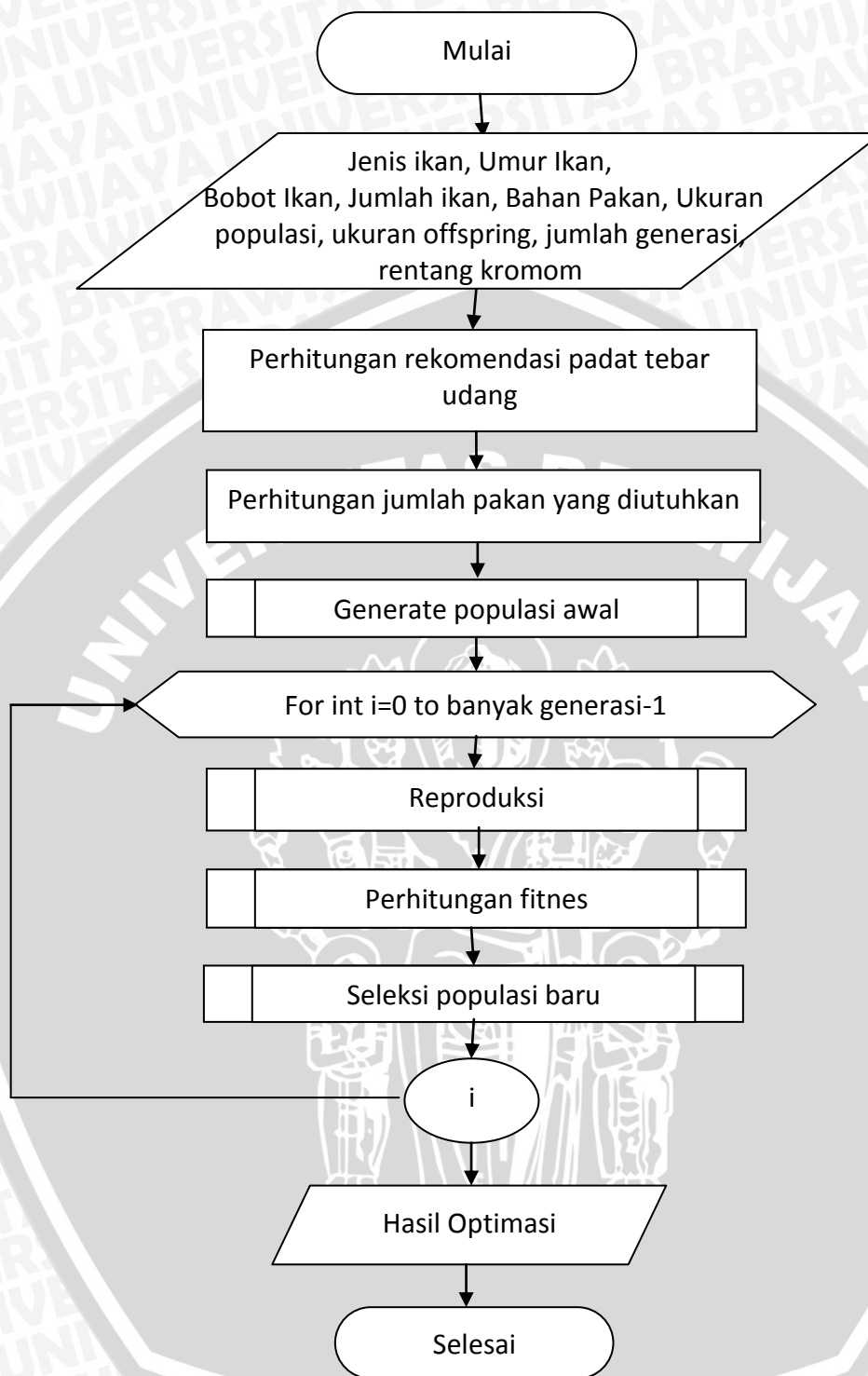
7. Melakukan proses seleksi dengan menggunakan metode *elitsm*.
8. Menampilkan optimasi komposisi pakan

4.2 Siklus Pemecahan Masalah

Pada bagian ini akan dibahas proses perancangan sistem untuk memperoleh optimasi yang terbaik. Langkah awal sistem akan menghitung rekomendasi padat tebar udang yang optimal dalam udidaya polikultur ikan dan udang berdasarkan jumlah ikan yang diinputkan oleh pengguna. Langkah selanjutnya adalah penentuan jumlah nutrisi yang di butuhkan, menghitung dosis pakan yang dibutuhkan ikan serta menghitung jumlah pakan yang di butuhkan dari inputan yang dilakukan oleh pengguna. Kemudian langkah selanjutnya yang merupakan tahapan dalam algoritma *evolution strategies* yaitu melakukan inialisasi awal, dengan cara membentuk nilai gen setiap individu sesuai dengan jumlah pakan yang diinputkan. Setiap gen memiliki nilai kromosom. Nilai kromosom dibentuk dari bilangan *random* dengan rentang yang telah ditentukan.

Setelah selesai pembentukan kromosom maka yang dilakukan adalah membentuk *offspring*. *Offspring* di dapatkan dengan cara melakukan proses reproduksi. Proses reproduksi yang dilakukan di penelitian ini menggunakan teknik $(\mu+\lambda)$. Sehingga proses reproduksi hanya menggunakan mutasi tanpa adanya rekombinasi. Pembentukan individu *offspring* juga dipengaruhi oleh nilai sigma. Nilai sigma dibangkitkan dengan cara *random* bilangan dengan rentang [0,1]. Setelah mendapatkan *offspring* maka langkah selanjutnya adalah menghitung nilai *fitness* dari setiap individu di dalam populasi. Nilai *fitness* di dapatkan dari total harga bahan pakan ikan dan nilai total *penalty* dari nutrisi bahan pakan.

Setelah proses sebelumnya selesai dilakukan maka langkah terakhir adalah melakukan proses seleksi individu yang melibatkan *parent* dan *offspring* dari hasil mutasi. Proses seleksi tersebut menggunakan metode *elitsm* untuk mencari nilai *fitness* yang tertinggi. Individu yang memiliki nilai *fitness* tertinggi merupakan hasil optimasi bahan pakan yang terbaik dengan menggunakan algoritma *evolution strategies*. Untuk Proses penyelesaian optimasi pakan ikan dapat dilihat pada Gambar 4.1.

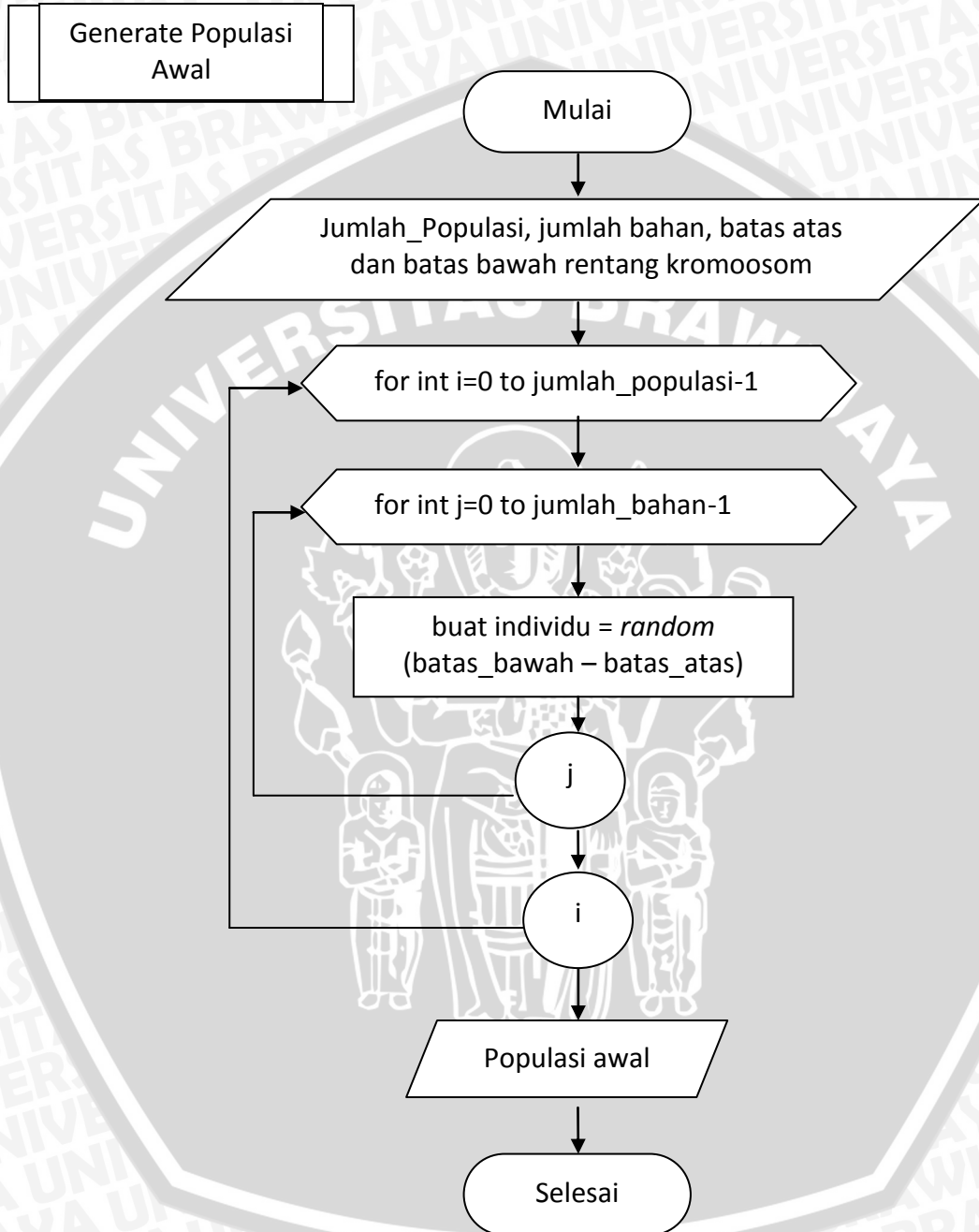


Gambar 4.1 Flowchart Proses algoritma *evolution strategies*

Gambar 4.1 menjelaskan tahapan proses algoritma *evolution strategies* dalam mengoptimasi komposisi pakan sistem polikultur ikan dan udang. tahapan proses tersebut adalah perhitungan rekomendasi padat tebar udang, perhitungan jumlah pakan, generate populasi awal, melakukan perulangan sebanyak generasi. Dalam satu kali perulangan terdapat proses reproduksi, perhitungan *fitness* dan seleksi populasi baru.

4.2.1 Generate Populasi Awal

Pada tahap ini akan membuat individu baru secara *random* dengan representasi *real-code*. Proses pembuatan populasi dapat dilihat pada Gambar 4.2 berikut



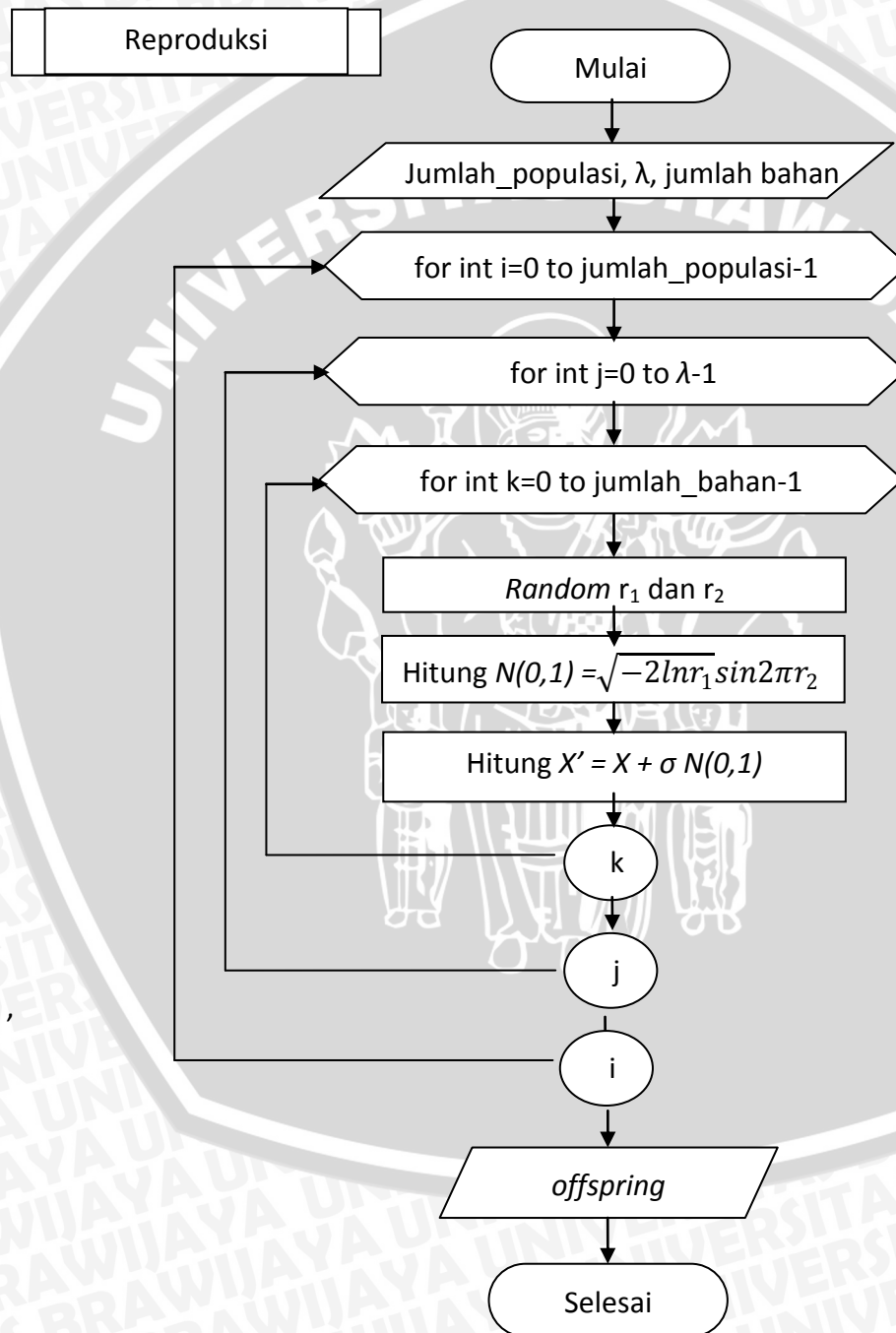
Gambar 4.2 Flowchart Generate Populasi Awal

Berdasarkan Gambar 4.2 langkah langkah dalam pembuatan populasi awal. Langkah pertama adalah sistem menerima masukan jumlah individu dalam populasi awal yang akan dibuat dan jumlah bahan pakan yang akan digunakan. Kemudian sistem akan melakukan perulangan untuk membuat individu. Individu baru dibuat dengan cara *random* dengan rentang yang telah ditentukan.

Perulangan ini dilakukan sebanyak jumlah bahan dan jumlah individu dalam populasi awal. Hasil dari proses ini adalah populasi awal yang akan digunakan untuk proses berikutnya.

4.2.2 Reproduksi

Pada penelitian ini menggunakan teknik $(\mu+\lambda)$ sehingga pada proses reproduksi hanya menggunakan mutasi. Proses reproduksi ini akan menghasilkan *offspring*. Proses Reproduksi dapat dilihat pada Gambar 4.3

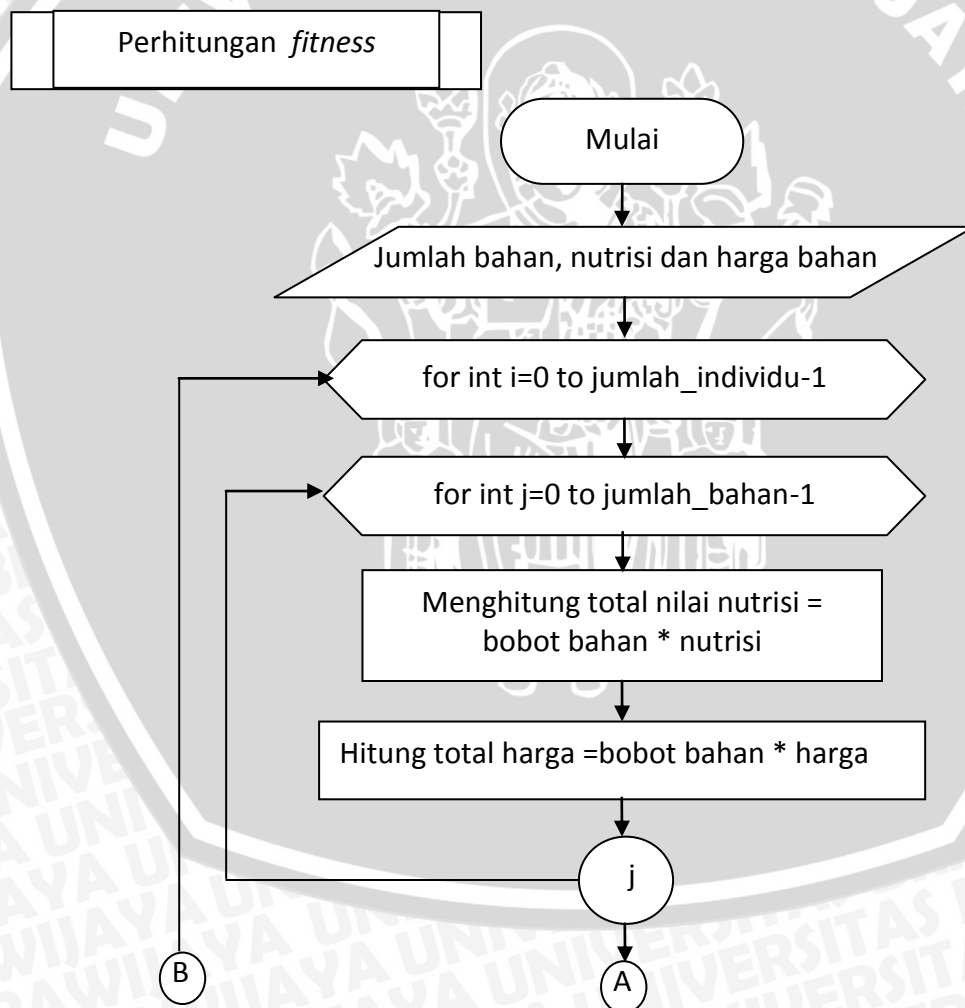


Gambar 4.3 Flowchart proses reproduksi

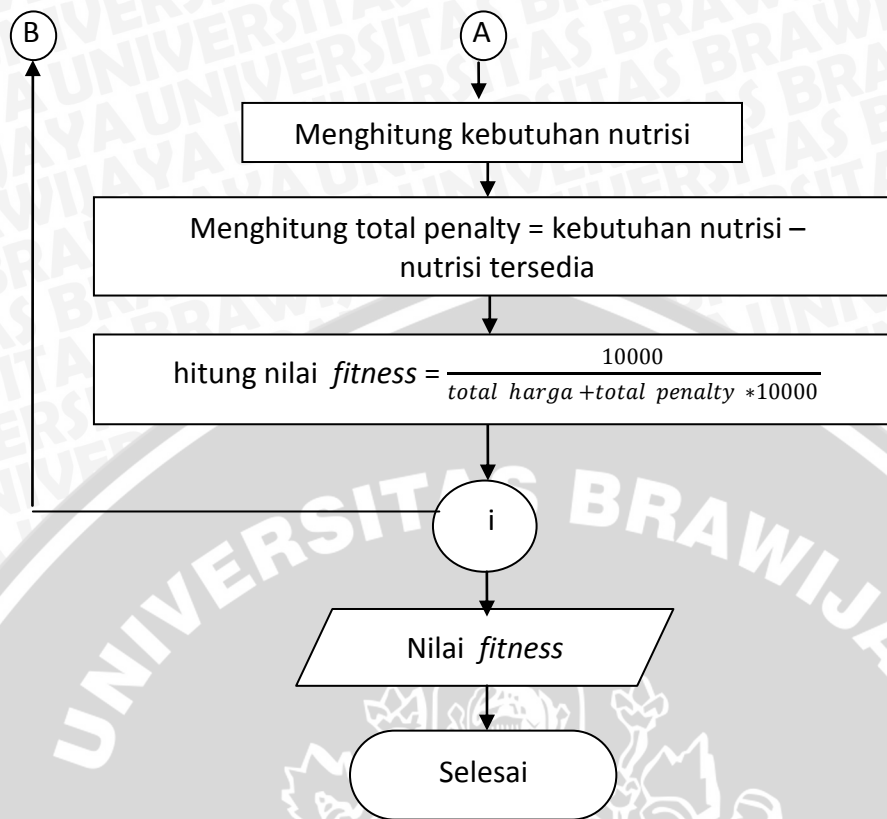
Gambar 4.3 merupakan langkah langkah dalam proses reproduksi. Reproduksi yang digunakan adalah mutasi. Langkah pertama adalah sistem menerima masukan untuk proses reproduksi yaitu jumlah populasi, λ dan jumlah bahan. kemudian proses selanjutnya sistem akan melakukan perulangan dalam pembentukan *offspring* sebanyak jumlah jumlah *populasi* kemudian melakukan perulangan sebanyak λ kemudian melakukan perulangan sebanyak jumlah bahan . Untuk membentuk *offspring* yang dilakukan adalah membangkitkan nilai r_1 dan r_2 secara acak kemudian menghitung nilai $N(0,1)$. Setelah itu menghitung nilai gen hasil mutasi (x'). hasil dari proses reproduksi dengan menggunakan mutasi adalah *offspring*.

4.2.3 Perhitungan nilai *fitness*

Pada proses ini akan menghitung nilai *fitness*. Semua individu yang berasal dari populasi awal atau *offspring* yang berasal dari reproduksi harus dicari nilai *fitness*nya. Proses pencarian nilai *fitness* dapat dilihat pada Gambar 4.4



Gambar 4.4 Flowchart menghitung nilai *fitness*



Gambar 4.5 Flowchart menghitung nilai *fitness* (lanjutan)

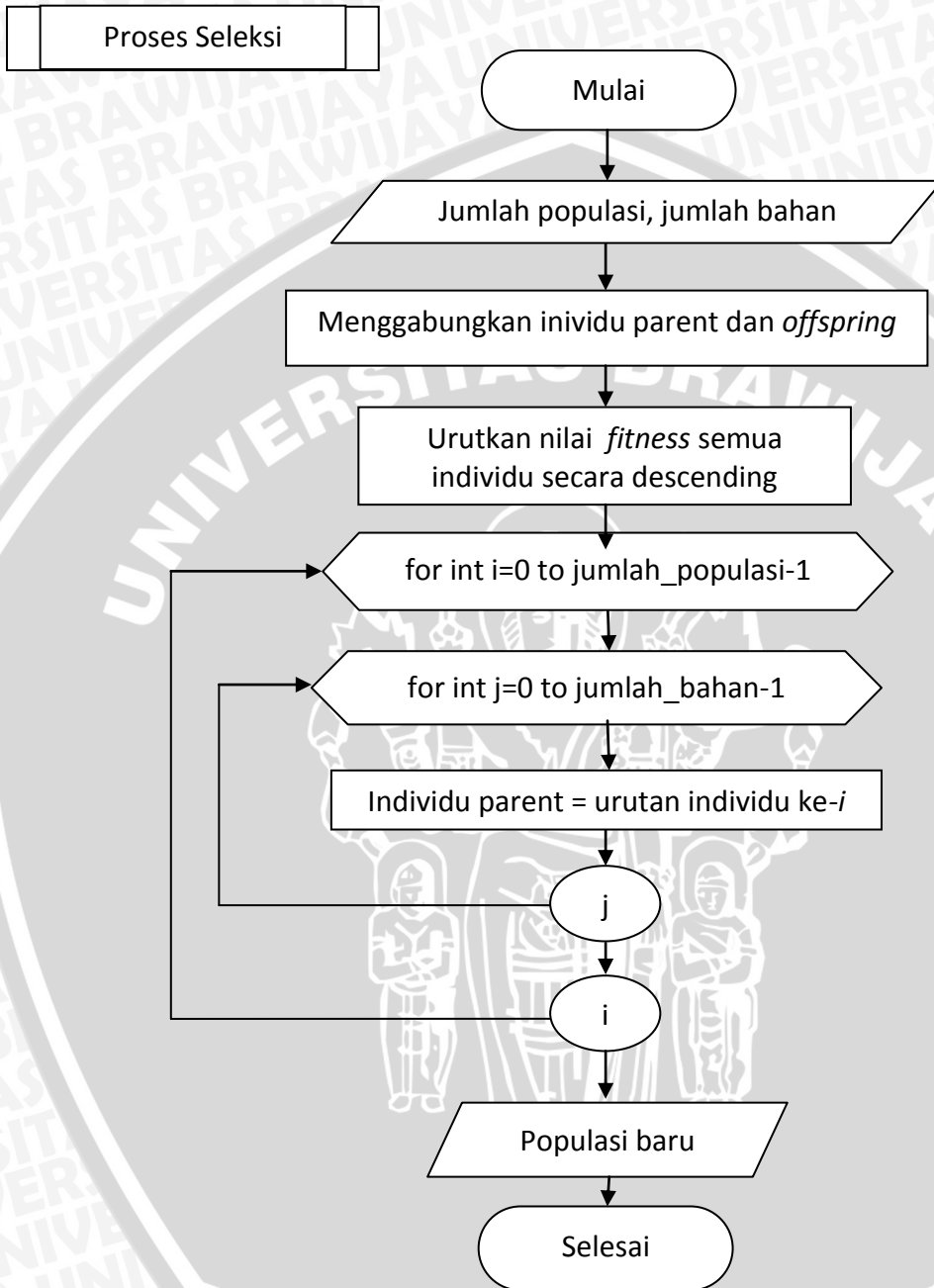
Gambar 4.5 dan 4.6 merupakan Gambar *Flowchart* menghitung nilai *fitness*. Berdasarkan Gambar 4.5 dan 4.6 langkah langkah menghitung nilai *fitness* yaitu langkah pertama adalah sistem menghitung total nutrisi setiap bahan baku dalam satu individu. Untuk mencari nutrisi bahan baku dengan cara mengkalikan bobot bahan dengan nutrisi bahan baku. kemudian sistem akan menghitung total harga dari setiap bahan baku dalam satu individu. Untuk mencari total harga bahan dengan cara mengkalikan bobot bahan dengan harga bahan baku. Setelah selesai menghitung total nutrisi dan harga dari bahan baku, langkah selanjutnya menghitung kebutuhan nutrisi kemudian menghitung nilai *penalty*.

Nilai *penalty* diperoleh dari kebutuhan nutrisi dikurang dengan nutrisi yang tersedia. Setelah selesai menghitung nilai *penalty* selanjutnya langkah terakhir menghitung nilai *fitness*. Cara untuk menghitung *fitness* yaitu $(10000 / (\text{total harga} + \text{total } \textit{penalty} * 10000))$. 10000 adalah bilangan konstanta yang digunakan dalam penelitian ini. Hasil akhir dari proses ini adalah nilai *fitness* untuk setiap individu pada populasi awal dan *offspring*.

4.2.4 Proses Seleksi

Pada proses metode seleksi ini menggunakan metode *elitism*. Pada penelitian ini menggunakan proses $(\mu + \lambda)$, sehingga proses seleksi ini melibatkan semua individu yaitu individu dari populasi awal yaitu *parent* dan *offspring*. dalam proses ini akan memilih individu yang memiliki jumlah *fitness* yang tertinggi

sebanyak jumlah populasi. Individu yang memiliki populasi tertinggi akan menjadi populasi baru pada generasi berikutnya. Proses seleksi dapat dilihat pada Gambar 4.6



Gambar 4.6 Flowchart Proses Seleksi

Berdasarkan Gambar 4.6 berisi diagram alir langkah langkah proses seleksi. Langkah pertama yang dilakukan adalah menggabungkan individu parent dengan *offspring* kemudian mengurutkan nilai *fitness* dari populasi awal dan *offspring* secara descending. Kemudian menyimpan individu dan kromosom terbaik dari setiap jumlah bahan sebanyak jumlah populasi ke dalam populasi baru. Hasil dari proses seleksi ini adalah populasi baru.

4.3 Contoh Perhitungan Manual

Contoh kasus, Misal terdapat pembudidaya yang ingin membudidayakan Ikan bandeng dan udang windu dengan menggunakan sistem polikultur. Pembudidaya tersebut ingin membuat pakan ikan. Bahan pakan yang tersedia dan yang akan digunakan adalah Tepung Ikan, Dedak Gandum dan Sorghum.

Karena yang dibudidayakan ikan bandeng dan udang maka data yang digunakan adalah data ikan bandeng. Ikan bandeng dalam tambak berumur 21 minggu dengan berat 0.6 Kg/ekor dan jumlah populasi ikan bandeng yaitu 500 ekor.

Perbandingan antara ikan bandeng dan udang adalah 2,5:20 ekor/m² sehingga rekomendasi padat tebar untuk udang yang akan dibudidayakan dengan sistem polikultur menggunakan persamaan (4-2)

$$\text{Padat tebar udang} = (\text{jumlah tebar ikan/perbandigan tebar ikan} \times \text{perbandingan tebar udang})$$

$$\text{padat tebar udang} = (500/2,5) \times 20 = 4000$$

Dari data ikan bandeng dapat diketahui nutrisi pakan ikan yang dibutuhkan ditunjukkan pada Tabel 2.2 yaitu:

Protein = 20%

Lemak = 8%

Serat = 4 %

Tahap selanjutnya menghitung jumlah pakan dengan menggunakan persamaan (4-1)

$$\text{Jumlah pakan} = (\text{dosis pakan} \times (\text{berat ikan}/\text{jumlah ikan}))/100$$

Dosis pakan yang dibutuhkan oleh ikan adalah 3% di ambil pada Tabel 2.6

$$\text{Jumlah pakn} = (3 \times (0.6 \text{ Kg/ekor} \times 500 \text{ ekor}))/100 = 9\text{Kg}$$

Nutrisi dari bahan pakan yang dimiliki ditunjukkan pada Tabel 2.5 dan harga bahan pakan adalah

Tepung Ikan	Dedak Gandum	Sorghum
Protein : 60.50 %	Protein : 11.99 %	Protein : 13.00 %
Lemak : 9.40 %	Lemak : 1.48 %	Lemak : 2.05 %
Serat : 0.70 %	Serat : 3.75 %	Serat : 13.5 %
Harga : 16000	Harga : 5000	Harga : 7000

Setelah mengetahui semua data yang dibutuhkan maka selanjutnya yaitu menyusun ransum optimal dari kombinasi pakan yang tersedia untuk mendapatkan hasil bahan pakan ikan yang memenuhi nutrisi ikan dengan biaya yang minimum

4.3.1 Inisialisasi

Dalam tahap inisialisasi harus diketahui nilai parameter awal untuk algoritma *evolution strategies* yaitu nilai populasi awal (μ), jumlah *offspring* (λ) dan jumlah generasi. Pada contoh perhitungan manual dalam penelitian ini siklus yang digunakan ($\mu/r+\lambda$) dan inisialisasi parameter awal yang digunakan adalah :

Jumlah populasi (μ) =10

Jumlah *offspring* (λ) =2 μ

Jumlah generasi = 2

4.3.2 Membuat populasi awal

Pada tahap ini membuat populasi awal sebanyak jumlah populasi yang di inputkan. Bahan pakan ikan akan jadi nilai gen setiap individu di dalam populasi. Nilai gen dari populasi dibangkitkan dengan cara *random* dengan rentang yang telah ditentukan. Dalam contoh perhitungan manual rentang yang digunakan yaitu 1-9. Nilai σ dibangkitkan secara *random* dengan rentang [0,1]. Maka representasi kromosom ditunjukkan pada Tabel 4.1

Taebel 4.1 Contoh Representasi Komosom

P	Kromosom			σ_1	σ_2	σ_3	total gen
	X_1	X_2	X_3				
	Tepung Ikan	Dedak Gandum	Sorgum				
1	8.16276	8.94391	6.78874	0.01790	0.63646	0.22619	23.89541
2	5.20865	2.40946	4.08231	0.69017	0.16930	0.46805	11.70041
3	4.69300	6.32285	6.61674	0.50272	0.56918	0.40372	17.63259
4	5.02396	6.68918	4.10275	0.51609	0.66797	0.91523	15.81590
5	4.61644	4.46120	6.88136	0.54456	0.93337	0.71777	15.95900
6	7.96523	7.86736	3.17406	0.24313	0.62246	0.57190	19.00665
7	4.57312	1.79200	6.62836	0.71401	0.12630	0.40519	12.99348
8	3.10345	6.23488	3.19139	0.45843	0.12890	0.78984	12.52973
9	6.82137	4.52822	2.61417	0.38927	0.41628	0.72322	13.96376
10	3.56733	5.99060	3.25251	0.57843	0.21028	0.10296	12.81044

4.3.3 Proses Generasi ke 1

Setelah populasi awal telah terbentuk tahap selanjutnya yaitu tahap proses generasi ke 1. Pada tahap ini yang dilakukan adalah membuat populasi baru dengan cara mutasi kemudian menghitung nilai *fitness* dan terakhir melakukan proses seleksi

4.3.3.1 Membuat populasi baru dengan mutasi

Pada tahap ini akan membuat populasi baru dengan mutasi. Pertama yang dilakukan adalah menghitung nilai $N(0,1)$ dengan dengan membangkitkan dua bilangan *random* r_1 dan r_2 dengan rentang [0,1]. Nilai $N(0,1)$ dihitung setiap jumlah gen perindividu

Contoh perhitungan manual untuk menghitung nilai $N_1(0,1)$, $N_2(0,1)$ dan $N_3(0,1)$ pada individu baru ke1 gen x_1 dan gen x_2 , jika $r_1 = 0.29941$, $r_2 = 0.54056$ untuk gen x_1 , $r_1 = 0.61270$ $r_2 = 0.19362$ untuk gen x_2 dan $r_1 = 0.34552$ $r_2 = 0.67913$ untuk gen x_3 maka:

$$\begin{aligned} N_1(0,1) &= \sqrt{-2\ln r_1 \sin 2\pi r_2} \\ &= \sqrt{-2 \ln 0.29941 \sin 2\pi 0.54056} \\ &= -0.39149 \end{aligned}$$

$$\begin{aligned} N_2(0,1) &= \sqrt{-2\ln r_1 \sin 2\pi r_2} \\ &= \sqrt{-2 \ln 0.61270 \sin 2\pi 0.19362} \\ &= 0.92838 \end{aligned}$$

$$\begin{aligned} N_3(0,1) &= \sqrt{-2\ln r_1 \sin 2\pi r_2} \\ &= \sqrt{-2 \ln 0.34552 \sin 2\pi 0.67913} \\ &= -1.31570 \end{aligned}$$

Untuk melihat hasil *random* pembangkitan nilai r yang digunakan pada perhitungan manual ini dapat dilihat pada Tabel 4.1 dan hasil lengkap perhitungan $N(0,1)$ dapat dilihat pada Tabel 4.2. Setelah menghitung nilai $N(0,1)$ yang dilakukan adalah menghitung nilai gen kromosom yang telah mutasi

Contoh perhitungan nilai x'_1 , x'_2 dan x'_3

$$\begin{aligned} x'_1 &= x_1 + \sigma_1 N_1(0,1) \\ &= 8.16276 + (0.01790x - 0.39149) \\ &= 8.15575 \end{aligned}$$

$$\begin{aligned} x'_2 &= x_2 + \sigma_2 N_2(0,1) \\ &= 8.94391 + (0.63646x 0.92838) \\ &= 9.53478 \end{aligned}$$

$$\begin{aligned} x'_3 &= x_3 + \sigma_3 N_3(0,1) \\ &= 6.78874 + (0.22619x - 1.31570) \\ &= 6.49115 \end{aligned}$$

Untuk hasil perhitungan x'_1 , x'_2 , x'_3 setiap individu dapat dilihat di Tabel 4.2

Tabel 4.1 Hasil *Random* Pembangkitan Nilai r

C(t)	parent	1		2		3	
		r1	r2	r1	r2	r1	r1
c1	p1	0.29941	0.54056	0.61270	0.19362	0.34552	0.67913
c2		0.08870	0.98319	0.60182	0.84967	0.26949	0.84500



c3	p2	0.90306	0.90095	0.11108	0.00681	0.82966	0.11687
c4		0.64994	0.70967	0.07418	0.18403	0.62375	0.01113
c5	p3	0.98787	0.34058	0.89620	0.34482	0.70487	0.69549
c6		0.46456	0.11719	0.48055	0.33002	0.64612	0.94766
c7	p4	0.52613	0.07096	0.76317	0.63957	0.67353	0.76010
c8		0.27621	0.60738	0.66236	0.43518	0.85110	0.27914
c9	p5	0.70513	0.08446	0.28831	0.94370	0.25739	0.54059
c10		0.59098	0.61849	0.59623	0.63887	0.70349	0.77641
c11	p6	0.89361	0.98093	0.61334	0.18929	0.96010	0.51681
c12		0.57635	0.72056	0.73032	0.68686	0.19924	0.63137
c13	p7	0.86303	0.04939	0.99298	0.85176	0.53678	0.16071
c14		0.79696	0.71291	0.63639	0.97164	0.16599	0.83614
c15	p8	0.04596	0.15727	0.41110	0.23130	0.69094	0.60712
c16		0.48974	0.16951	0.30467	0.76777	0.35230	0.06424
c17	p9	0.63424	0.65298	0.66912	0.42023	0.94586	0.11397
c18		0.08591	0.89052	0.24180	0.39742	0.28521	0.62038
c19	p10	0.55888	0.10955	0.16792	0.35013	0.08288	0.45163
c20		0.41098	0.19961	0.58351	0.99562	0.50630	0.66312

Tabel 4.2 Hasil Perhitungan Individu Baru

C(t)	P	$N_1(0.1)$	$N_2(0.1)$	$N_3(0.1)$	x'_1	x'_2	x'_3	total gen
c1	p1	-0.39149	0.92838	-1.31570	8.15575	9.53478	6.49115	24.18169
c2		-0.23210	-0.81653	-1.33939	8.15861	8.42423	6.48579	23.06862
c3	p2	-0.26324	0.08969	0.40950	5.02697	2.42464	4.27397	11.72558
c4		-0.89865	2.08779	0.06791	4.58842	2.76292	4.11409	11.46544
c5	p3	0.13162	0.38752	-0.78777	4.75916	6.54342	6.29870	17.60129
c6		0.83162	1.06079	-0.30187	5.11107	6.92664	6.49487	18.53258
c7	p4	0.48872	-0.56522	-0.88728	5.27619	6.31163	3.29068	14.87850
c8		-1.00203	0.35955	0.55835	4.50682	6.92936	4.61377	16.04995
c9	p5	0.42306	-0.54633	-0.41562	4.84682	3.95127	6.58304	15.38114
c10		-0.69499	-0.77897	-0.82716	4.23798	3.73413	6.28765	14.25976
c11	p6	-0.05671	0.91771	-0.03008	7.95144	8.43860	3.15686	19.54689
c12		-1.03190	-0.73123	-1.31994	7.71434	7.41220	2.41919	17.54573
c13	p7	0.16574	-0.09526	0.94452	4.69146	1.77997	7.01107	13.48250
c14		-0.65551	-0.16854	-1.62430	4.10509	1.77071	5.97020	11.84600
c15	p8	2.07249	1.32416	-0.53603	4.05355	6.40557	2.76802	13.22713
c16		1.04530	-1.53217	0.56738	3.58265	6.03738	3.63953	13.25956



c17	p9	-0.78240	0.43071	0.21902	6.51681	4.70752	2.77257	13.99690
c18		-1.40665	1.01239	-1.08706	6.27381	4.94966	1.82799	13.05146
c19	p10	0.68526	1.52740	0.66787	3.96370	6.31178	3.32128	13.59675
c20		1.26730	-0.02859	-0.99716	4.30037	5.98458	3.14985	13.43480

4.3.3.2 Menghitung nilai *fitness*

Pada tahap ini akan dihitung nilai *fitness* dari setiap individu *parent* dan individu hasil *offspring* hal yang pertama dilakukan adalah menghitung total cost. Setelah itu menghitung nutrisi kemudian menghitung nilai *penalty* dan terakhir menghitung nilai *fitness*. Untuk menghitung biaya bahan dapat menggunakan persamaan (4-2).

Tabel 4.3 Contoh perhitungan total harga p1

P	Gen Kromosom			Total Gen	Total Cost
	T. Ikan	D. Gandum	Sorghum		
	16000/Kg	5000/Kg	7000/Kg		
P1	8.16276	8.94391	6.78874	23.89541	83933

$$\begin{aligned}
 \text{harga tepung ikan} &= (\text{bobot bahan} \times \text{harga}) \\
 &= \left(\left(\frac{8.16276}{23.89541} \right) \times 9 \right) \times 16000 \\
 &= 49191
 \end{aligned}$$

$$\begin{aligned}
 \text{harga dedak gandum} &= (\text{bobot bahan} \times \text{harga}) \\
 &= \left(\left(\frac{8.94391}{23.89541} \right) \times 9 \right) \times 5000 \\
 &= 16843
 \end{aligned}$$

$$\begin{aligned}
 \text{harga sorghum} &= (\text{bobot bahan} \times \text{harga}) \\
 &= \left(\left(\frac{6.78874}{23.89541} \right) \times 9 \right) \times 7000 = 17898
 \end{aligned}$$

$$\begin{aligned}
 \text{Total cost} &= \text{harga tepung gandum} \times \text{harga dedak gandum} \times \text{harga sorghum} \\
 &= 49191 + 16843 + 17898 \\
 &= 83933
 \end{aligned}$$

Hasil lengkap perhitungan total cost tiap individu dalam populasi awal pada Tabel 4.5 dan individu *offspring* pada Tabel 4.6.

Tabel 4.4 Hasil perhitungan total cost *parent*

P	Harga			Total cost
	x1 16000/Kg	x2 5000/Kg	x3 7000/Kg	
p1	49191	16843	17898	83933
p2	64104	9267	21981	95352



p3	38326	16137	23641	78104
p4	45742	19032	16343	81117
p5	41655	12579	27165	81399
p6	60347	18627	10521	89494
p7	50682	6206	32138	89026
p8	35667	22392	16046	74106
p9	70345	14593	11794	96732
p10	40100	21044	15995	77139

Tabel 4.5 Hasil perhitungan total cost offspring

C(t)	P	Harga			Total cost
		x1	x2	x3	
		16000/Kg	5000/Kg	7000/Kg	
c1	p1	48567	17743	16911	83221
c2		50928	16433	17713	85074
c3	p2	61735	9305	22963	94004
c4		57628	10844	22606	91078
c5	p3	38936	16729	22545	78210
c6		39714	16819	22079	78611
c7	p4	51065	19090	13934	84088
c8		40435	19428	18110	77974
c9	p5	45377	11560	26964	83900
c10		42797	11784	27779	82360
c11	p6	58577	19427	10175	88179
c12		63313	19010	8686	91009
c13	p7	50107	5941	32761	88809
c14		49901	6726	31751	88379
c15	p8	44130	21792	13184	79106
c16		38908	20490	17292	76690
c17	p9	67045	15135	12479	94659
c18		69221	17066	8824	95110
c19	p10	41979	20890	15389	78257
c20		46093	20045	14771	80909

Setelah menghitung total cost yang dilakukan adalah menghitung total nilai yang terkandung pada masing masing bahan. Untuk menghitung total nutrisi bahan bisa menggunakan persamaan (4-3). Contoh perhitungan total nutrisi yang terkandung pada setiap bahan pakan ikan

Tabel 4.6 Contoh Perhitungan Total Nutrisi

P	Tepung Ikan			Dedak Gandum			Sorghum			Total gen	Total Nutrisi		
	P	L	S	P	L	S	P	L	S		P	L	S
	60.50 %	9.40 %	0.70 %	11.99 %	1.48 %	3.75 %	13.00 %	2.05 %	13.50 %				
p1	8.16276			8.94391			6.78874			23.89541	2.60	0.39	0.49



Nutrisi Tepung Ikan

$$\begin{aligned} \text{protein} &= \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right) \\ &= \left(\left(\frac{8.16276}{23.89541} \right) \times 9 \right) \times \left(\frac{60.5}{100} \right) \\ &= 1.86 \end{aligned}$$

$$\begin{aligned} \text{lemak} &= \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right) \\ &= \left(\left(\frac{8.16276}{23.89541} \right) \times 9 \right) \times \left(\frac{9.4}{100} \right) \\ &= 0.289 \end{aligned}$$

$$\begin{aligned} \text{Serat} &= \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right) \\ &= \left(\left(\frac{8.16276}{23.89541} \right) \times 9 \right) \times \left(\frac{0.7}{100} \right) \\ &= 0.022 \end{aligned}$$

Nutrisi Dedak Gandum

$$\begin{aligned} \text{protein} &= \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right) \\ &= \left(\left(\frac{8.94391}{23.89541} \right) \times 9 \right) \times \left(\frac{11.99}{100} \right) \\ &= 0.404 \end{aligned}$$

$$\begin{aligned} \text{lemak} &= \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right) \\ &= \left(\left(\frac{8.94391}{23.89541} \right) \times 9 \right) \times \left(\frac{1.48}{100} \right) \\ &= 0.05 \end{aligned}$$

$$\begin{aligned} \text{Serat} &= \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right) \\ &= \left(\left(\frac{8.94391}{23.89541} \right) \times 9 \right) \times \left(\frac{3.75}{100} \right) \\ &= 0.126 \end{aligned}$$

Nutrisi Sorghum

$$\begin{aligned} \text{protein} &= \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right) \\ &= \left(\left(\frac{6.78874}{23.89541} \right) \times 9 \right) \times \left(\frac{13}{100} \right) \\ &= 0.332 \end{aligned}$$

$$\text{lemak} = \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right)$$

$$= \left(\left(\frac{6.78874}{23.89541} \right) \times 9 \right) \times \left(\frac{2.05}{100} \right)$$

$$= 0.052$$

$$\text{Serat} = \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right)$$

$$= \left(\left(\frac{6.78874}{23.89541} \right) \times 9 \right) \times \left(\frac{13.5}{100} \right)$$

$$= 0.345$$

$$\begin{aligned} \text{Total nutrisi protein p1} &= \text{protein tepung ikan} + \text{protein dedak gandum} \\ &\quad + \text{protein sorghum} \\ &= 1.86 + 0.404 + 0.332 \\ &= 2.6 \end{aligned}$$

$$\begin{aligned} \text{Total nutrisi lemak p1} &= \text{lemak tepung ikan} + \text{lemak dedak gandum} \\ &\quad + \text{lemak sorghum} \\ &= 0.289 + 0.05 + 0.052 \\ &= 0.39 \end{aligned}$$

$$\begin{aligned} \text{Total nutrisi serat p1} &= \text{serat tepung ikan} + \text{serat dedak gandum} \\ &\quad + \text{serat sorghum} \\ &= 0.022 + 0.126 + 0.345 \\ &= 0.49 \end{aligned}$$

Untuk melihat hasil perhitungan total nutrisi secara lengkap dapat dilihat pada Tabel 4.7 dan 4.8.

Tabel 4.7 Total Nutrisi Parent

Parent	Tepung Ikan			Dedak Gandum			Sorghum			Total Nutrisi		
	P	L	S	P	L	S	P	L	S	P	L	S
	60.50%	9.40%	0.70%	11.99%	1.48%	3.75%	13.00%	2.05%	13.50%			
p1	1.860	0.289	0.022	0.404	0.050	0.126	0.332	0.052	0.345	2.6	0.39	0.49
p2	2.424	0.377	0.028	0.222	0.027	0.070	0.408	0.064	0.424	3.05	0.47	0.52
p3	1.449	0.225	0.017	0.387	0.048	0.121	0.439	0.069	0.456	2.28	0.34	0.59
p4	1.730	0.269	0.020	0.456	0.056	0.143	0.304	0.048	0.315	2.49	0.37	0.48
p5	1.575	0.245	0.018	0.302	0.037	0.094	0.504	0.080	0.524	2.38	0.36	0.64
p6	2.282	0.355	0.026	0.447	0.055	0.140	0.195	0.031	0.203	2.92	0.44	0.37
p7	1.916	0.298	0.022	0.149	0.018	0.047	0.597	0.094	0.620	2.66	0.41	0.69
p8	1.349	0.210	0.016	0.537	0.066	0.168	0.298	0.047	0.309	2.18	0.32	0.49

p9	2.660	0.413	0.031	0.350	0.043	0.109	0.219	0.035	0.227	3.23	0.49	0.37
p10	1.516	0.236	0.018	0.505	0.062	0.158	0.297	0.047	0.308	2.32	0.34	0.48

Tabel 4.8 Total Nutrisi *Offspring*

C(t)	Tepung Ikan			Dedak Gandum			Sorghum			Total Nutrisi		
	P	L	S	P	L	S	P	L	S	P	L	S
	60.50 %	9.40 %	0.70 %	11.99 %	1.48 %	3.75 %	13.00 %	2.05 %	13.50 %			
c1	1.836	0.285	0.021	0.425	0.053	0.133	0.314	0.050	0.326	2.58	0.39	0.48
c2	1.926	0.299	0.022	0.394	0.049	0.123	0.329	0.052	0.342	2.65	0.4	0.49
c3	2.334	0.363	0.027	0.223	0.028	0.070	0.426	0.067	0.443	2.98	0.46	0.54
c4	2.179	0.339	0.025	0.260	0.032	0.081	0.420	0.066	0.436	2.86	0.44	0.54
c5	1.472	0.229	0.017	0.401	0.050	0.125	0.419	0.066	0.435	2.29	0.34	0.58
c6	1.502	0.233	0.017	0.403	0.050	0.126	0.410	0.065	0.426	2.32	0.35	0.57
c7	1.931	0.300	0.022	0.458	0.057	0.143	0.259	0.041	0.269	2.65	0.4	0.43
c8	1.529	0.238	0.018	0.466	0.058	0.146	0.336	0.053	0.349	2.33	0.35	0.51
c9	1.716	0.267	0.020	0.277	0.034	0.087	0.501	0.079	0.520	2.49	0.38	0.63
c10	1.618	0.251	0.019	0.283	0.035	0.088	0.516	0.081	0.536	2.42	0.37	0.64
c11	2.215	0.344	0.026	0.466	0.058	0.146	0.189	0.030	0.196	2.87	0.43	0.37
c12	2.394	0.372	0.028	0.456	0.056	0.143	0.161	0.025	0.168	3.01	0.45	0.34
c13	1.895	0.294	0.022	0.142	0.018	0.045	0.608	0.096	0.632	2.65	0.41	0.70
c14	1.887	0.293	0.022	0.161	0.020	0.050	0.590	0.093	0.612	2.64	0.41	0.68
c15	1.669	0.259	0.019	0.523	0.065	0.163	0.245	0.039	0.254	2.44	0.36	0.44
c16	1.471	0.229	0.017	0.491	0.061	0.154	0.321	0.051	0.333	2.28	0.34	0.50
c17	2.535	0.394	0.029	0.363	0.045	0.114	0.232	0.037	0.241	3.13	0.48	0.38
c18	2.617	0.407	0.030	0.409	0.051	0.128	0.164	0.026	0.170	3.19	0.48	0.33
c19	1.587	0.247	0.018	0.501	0.062	0.157	0.286	0.045	0.297	2.37	0.35	0.47
c20	1.743	0.271	0.020	0.481	0.059	0.150	0.274	0.043	0.285	2.50	0.37	0.46

Setelah selesai menghitung total nutrisi sekarang menghitung nilai *penalty*. Untuk menghitung total *penalty* yang dilakukan adalah menghitung kebutuhan nutrisi. Dari data yang diketahui nutrisi yang dibutuhkan adalah protein 20%, Lemak, 8% dan serat 4%. Contoh perhitungan mencari jumlah nutrisi protein, lemak, serat yang dibutuhkan dalam 9Kg.

$$\begin{aligned} \text{protein} &= ((\text{nutrisi yang dibutuhkan}/100) \times \text{jumlah pakan}) \\ &= ((20/100) \times 9) \\ &= 1.8 \end{aligned}$$

$$\begin{aligned} \text{lemak} &= ((\text{nutrisi yang dibutuhkan}/100) \times \text{jumlah pakan}) \\ &= ((8/100) \times 9) \\ &= 0.72 \end{aligned}$$

$$\begin{aligned} \text{serat} &= ((\text{nutrisi yang dibutuhkan}/100) \times \text{jumlah pakan}) \\ &= ((4/100) \times 9) \\ &= 0.36 \end{aligned}$$

Hasil perhitungan kebutuhan nutrisi ditunjukkan pada Tabel 4.10 . setelah selesai menghitung kebutuhan nutrisi, yang dilakukan adalah menghitung total *penalty*. Nilai total *penalty* dihitung setiap individu. Contoh perhitungan nilai *penalty* pada individu p1

$$\begin{aligned} \text{Penalty protein} &= |\text{total nutrisi protein yang dibutuhkan} - \text{total nutrisi protein yang tersedia}| \\ &= |1.8 - 2.6| \\ &= 0.80 \end{aligned}$$

$$\begin{aligned} \text{Penalty lemak} &= |\text{total nutrisi lemak yang dibutuhkan} - \text{total nutrisi lemak yang tersedia}| \\ &= |0.72 - 0.39| \\ &= 0.33 \end{aligned}$$

$$\begin{aligned} \text{Penalty serat} &= |\text{total nutrisi serat yang dibutuhkan} - \text{total nutrisi serat yang tersedia}| \\ &= |0.36 - 0.49| \\ &= 0.13 \end{aligned}$$

$$\begin{aligned} \text{Total penalty} &= \text{penalt protein} + \text{penalty lemak} + \text{penalty serat} \\ &= 0.80 + 0.33 + 0.13 \\ &= 1.26 \end{aligned}$$

Hasil lengkap perhitungan *penalty* terdapat pada Tabel 4.9 dan 4.10.

Tabel 4.9 Perhitungan *Penalty Parent*

Parent	Kebutuhan			Tersedia			<i>penalty</i>			Total <i>Penalty</i>
	P	L	S	P	L	S	P	L	S	
p1	1.8	0.72	0.36	2.60	0.39	0.49	0.80	0.33	0.13	1.26



p2	1.8	0.72	0.36	3.05	0.47	0.52	1.25	0.25	0.16	1.67
p3	1.8	0.72	0.36	2.28	0.34	0.59	0.48	0.38	0.23	1.09
p4	1.8	0.72	0.36	2.49	0.37	0.48	0.69	0.35	0.12	1.15
p5	1.8	0.72	0.36	2.38	0.36	0.64	0.58	0.36	0.28	1.22
p6	1.8	0.72	0.36	2.92	0.44	0.37	1.12	0.28	0.01	1.41
p7	1.8	0.72	0.36	2.66	0.41	0.69	0.86	0.31	0.33	1.50
p8	1.8	0.72	0.36	2.18	0.32	0.49	0.38	0.40	0.13	0.91
p9	1.8	0.72	0.36	3.23	0.49	0.37	1.43	0.23	0.01	1.67
p10	1.8	0.72	0.36	2.32	0.34	0.48	0.52	0.38	0.12	1.02

Tabel 4.10 Perhitungan *Penalty Offspring*

C(t)	Kebutuhan			Tersedia			<i>penalty</i>			Total <i>Penalty</i>
	P	L	S	P	L	S	P	L	S	
c1	1.8	0.72	0.36	2.58	0.39	0.48	0.78	0.33	0.12	1.23
c2	1.8	0.72	0.36	2.65	0.40	0.49	0.85	0.32	0.13	1.30
c3	1.8	0.72	0.36	2.98	0.46	0.54	1.18	0.26	0.18	1.63
c4	1.8	0.72	0.36	2.86	0.44	0.54	1.06	0.28	0.18	1.52
c5	1.8	0.72	0.36	2.29	0.34	0.58	0.49	0.38	0.22	1.09
c6	1.8	0.72	0.36	2.32	0.35	0.57	0.52	0.37	0.21	1.10
c7	1.8	0.72	0.36	2.65	0.40	0.43	0.85	0.32	0.07	1.24
c8	1.8	0.72	0.36	2.33	0.35	0.51	0.53	0.37	0.15	1.06
c9	1.8	0.72	0.36	2.49	0.38	0.63	0.69	0.34	0.27	1.30
c10	1.8	0.72	0.36	2.42	0.37	0.64	0.62	0.35	0.28	1.25
c11	1.8	0.72	0.36	2.87	0.43	0.37	1.07	0.29	0.01	1.37
c12	1.8	0.72	0.36	3.01	0.45	0.34	1.21	0.27	0.02	1.50
c13	1.8	0.72	0.36	2.65	0.41	0.70	0.85	0.31	0.34	1.50
c14	1.8	0.72	0.36	2.64	0.41	0.68	0.84	0.31	0.32	1.48
c15	1.8	0.72	0.36	2.44	0.36	0.44	0.64	0.36	0.08	1.07
c16	1.8	0.72	0.36	2.28	0.34	0.50	0.48	0.38	0.14	1.01



c17	1.8	0.72	0.36	3.13	0.48	0.38	1.33	0.24	0.02	1.60
c18	1.8	0.72	0.36	3.19	0.48	0.33	1.39	0.24	0.03	1.66
c19	1.8	0.72	0.36	2.37	0.35	0.47	0.57	0.37	0.11	1.05
c20	1.8	0.72	0.36	2.50	0.37	0.46	0.70	0.35	0.10	1.14

Setelah selesai menghitung *penalty* maka selanjutnya menghitung nilai *fitness* dengan menggunakan persamaan (4.8). Contoh perhitungan nilai *fitness* pada individu p1

$$\begin{aligned}
 fitness\ p1 &= \frac{10000}{total\ cost + (total\ penalty \times 10000)} \\
 &= \frac{10000}{83933 + (1.26 \times 10000)} \\
 &= 0.10361
 \end{aligned}$$

Fitness dihitung setiap individu. Hasil perhitungan nilai *fitness* setiap individu pada populasi awal dan *offspring* dapat dilihat setiap individu ditunjukkan pada Tabel 4.11 dan 4.12

Tabel 4.11 Hasil perhitungan nilai *fitness* parent

Parent	Kromosom			σ_1	σ_2	σ_3	Total cost	total <i>penalty</i>	fitness
	x1	x2	X3						
1	8.16276	8.94391	6.78874	0.01790	0.63646	0.22619	83933	1.26	0.10361
2	5.20865	2.40946	4.08231	0.69017	0.16930	0.46805	95352	1.67	0.08926
3	4.69300	6.32285	6.61674	0.50272	0.56918	0.40372	78104	1.09	0.11240
4	5.02396	6.68918	4.10275	0.51609	0.66797	0.91523	81117	1.15	0.10792
5	4.61644	4.46120	6.88136	0.54456	0.93337	0.71777	81399	1.22	0.10688
6	7.96523	7.86736	3.17406	0.24313	0.62246	0.57190	89494	1.41	0.09651
7	4.57312	1.79200	6.62836	0.71401	0.12630	0.40519	89026	1.50	0.09613
8	3.10345	6.23488	3.19139	0.45843	0.12890	0.78984	74106	0.91	0.12013
9	6.82137	4.52822	2.61417	0.38927	0.41628	0.72322	96732	1.67	0.08819
10	3.56733	5.99060	3.25251	0.57843	0.21028	0.10296	77139	1.02	0.11453

Tabel 4.12 Hasil perhitungan nilai *fitness* offspring

C(t)	Parent	x1'	x2'	X3'	σ_1'	σ_2'	σ_3'	total cost	total <i>penalty</i>	<i>fitness</i>
c1	p1	8.15575	9.53478	6.49115	0.01969	0.70010	0.24881	83221	1.23	0.10470
c2		8.15861	8.42423	6.48579	0.01611	0.57281	0.20357	85074	1.30	0.10200
c3	p2	5.02697	2.42464	4.27397	0.75919	0.18623	0.51486	94004	1.63	0.09069



c4		4.58843	2.76293	4.11409	0.75919	0.18623	0.51486	91078	1.52	0.09405
c5	p3	4.75916	6.54342	6.29870	0.45245	0.51226	0.36335	78210	1.09	0.11228
c6		5.11107	6.92664	6.49487	0.45245	0.51226	0.36335	78611	1.10	0.11164
c7	p4	5.27619	6.31163	3.29068	0.46448	0.60118	0.82370	84088	1.24	0.10359
c8		4.50683	6.92936	4.61377	0.56770	0.73477	1.00675	77974	1.06	0.11295
c9	p5	4.84682	3.95128	6.58304	0.49010	0.84003	0.64600	83900	1.30	0.10319
c10		4.23798	3.73413	6.28765	0.49010	0.84003	0.64600	82360	1.25	0.10540
c11	p6	7.95144	8.43860	3.15686	0.26745	0.68470	0.62909	88179	1.37	0.09820
c12		7.71434	7.41220	2.41919	0.21882	0.56021	0.51471	91009	1.50	0.09433
c13	p7	4.69146	1.77997	7.01107	0.78541	0.13894	0.44571	88809	1.50	0.09637
c14		4.10509	1.77071	5.97020	0.78541	0.13894	0.44571	88379	1.48	0.09695
c15	p8	4.05355	6.40557	2.76802	0.41259	0.11601	0.71085	79106	1.07	0.11134
c16		3.58265	6.03738	3.63953	0.41259	0.11601	0.71085	76690	1.01	0.11525
c17	p9	6.51681	4.70752	2.77257	0.42819	0.45791	0.79554	94659	1.60	0.09038
c18		6.27381	4.94966	1.82799	0.42819	0.45791	0.79554	95110	1.66	0.08953
c19	p10	3.96370	6.31178	3.32128	0.52058	0.18925	0.09266	78257	1.05	0.11264
c20		4.30037	5.98458	3.14985	0.52058	0.18925	0.09266	80909	1.14	0.10833

Karena salah satu nilai *fitness offspring* ada yang lebih besar dari pada nilai *fitness* induknya maka nilai σ' untuk *offspring* dinaikan dengan cara $\sigma \times 1.1$.

4.3.3.3 Poses seleksi

Tahap ini merupakan tahapan terakhir dari algoritma *evolution strategies*. Dalam tahapan ini yang diambil adalah nilai *fitness* tertinggi dari populasi gabungan antara *parent* dan *offspring*. Hasil seleksi yang menghasilkan 10 individu ditunjukkan pada Tabel 4.13.

Tabel 4.13 hasil seleksi

P(t+1)	parent	x1	x2	X3	σ_1	σ_2	σ_3	<i>fitness</i>	total gen
p1	p8	3.10345	6.23488	3.19139	0.45843	0.12890	0.78984	0.12013	12.52973
p2	c16	3.58265	6.03738	3.63953	0.41259	0.11601	0.71085	0.11525	13.25957
p3	p10	3.56733	5.99060	3.25251	0.57843	0.21028	0.10296	0.11453	12.81044

p4	c8	4.50683	6.92936	4.61377	0.56770	0.73477	1.00675	0.11295	16.04995
p5	c19	3.96370	6.31178	3.32128	0.52058	0.18925	0.09266	0.11264	13.59675
p6	p3	4.69300	6.32285	6.61674	0.50272	0.56918	0.40372	0.11240	17.63259
p7	c5	4.75916	6.54342	6.29870	0.45245	0.51226	0.36335	0.11228	17.60129
p8	c6	5.11107	6.92664	6.49487	0.45245	0.51226	0.36335	0.11164	18.53258
p9	c15	4.05355	6.40557	2.76802	0.41259	0.11601	0.71085	0.11134	13.22713
p10	c20	4.30037	5.98458	3.14985	0.52058	0.18925	0.09266	0.10833	13.43480

Pada contoh perhitungan manual pada penelitian ini nilai *fitness* yang tertinggi adalah p1. Sehingga p1 adalah individu yang terbaik di generasinya ke 1. Individu hasil dari proses ini akan digunakan pada generasi selanjutnya.

4.3.4 Proses Generasi Ke 2

Setelah tahap proses generasi ke 1 selesai dan mendapatkan individu baru hasil seleksi maka tahap selanjutnya adalah populasi generasi ke 2. Pada tahap ini yang dilakukan adalah membuat populasi baru dengan cara mutasi kemudian menghitung nilai *fitness* dan terakhir melakukan proses seleksi. Individu parent yang digunakan pada populasi ini adalah individu hasil proses seleksi pada tahap generasi ke 1.

4.3.4.1 Membuat Populasi baru dengan mutasi

Pada tahap ini melakukan proses mutasi untuk membuat individu baru. Proses untuk mutasi pada generasi ke-2 sama dengan proses mutasi pada generasi ke-1.

Contoh perhitungan manual untuk menghitung nilai $N_1(0,1)$ $N_2(0,1)$ pada individu baru generasi ke-2, gen x_1 dan gen x_2 , jika $r_1 = 0.85168$, $r_2 = 0.32823$ untuk gen x_1 , $r_1 = 0.14044$, $r_2 = 0.10646$ untuk gen x_2 dan $r_1 = 0.40346$ $r_2 = 0.60753$ untuk gen x_3 maka:

$$\begin{aligned}
 N_1(0,1) &= \sqrt{-2\ln r_1} \sin 2\pi r_2 \\
 &= \sqrt{-2\ln 0.85168} \sin 2\pi 0.32823 \\
 &= 0.49957
 \end{aligned}$$

$$\begin{aligned}
 N_2(0,1) &= \sqrt{-2\ln r_1} \sin 2\pi r_2 \\
 &= \sqrt{-2\ln 0.14044} \sin 2\pi 0.10646 \\
 &= 1.22871
 \end{aligned}$$

$$\begin{aligned}
 N_3(0,1) &= \sqrt{-2\ln r_1} \sin 2\pi r_2 \\
 &= \sqrt{-2\ln 0.40346} \sin 2\pi 0.60753 \\
 &= -0.84262
 \end{aligned}$$

Untuk melihat hasil *random* pembangkitan nilai r yang digunakan pada perhitungan manual ini dapat dilihat pada Tabel 4.14 dan hasil lengkap perhitungan $N(0,1)$ dapat dilihat pada Tabel 4.15. Setelah menghitung nilai $N(0,1)$ yang dilakukan adalah menghitung nilai gen kromosom yang telah mutasi

Contoh perhitungan nilai x'_1 , x'_2 dan x'_3

$$\begin{aligned} x'_1 &= x_1 + \sigma_1 N1(0,1) \\ &= 3.10345 + (0.45843 \times 0.49957) \\ &= 3.33247 \end{aligned}$$

$$\begin{aligned} x'_2 &= x_2 + \sigma_2 N2(0,1) \\ &= 6.23488 + (0.12890 \times 1.22871) \\ &= 6.39326 \end{aligned}$$

$$\begin{aligned} x'_3 &= x_3 + \sigma_3 N3(0,1) \\ &= 3.19139 + (0.78984 \times -0.84262) \\ &= 2.52586 \end{aligned}$$

Untuk hasil perhitungan x'_1 , x'_2 , x'_3 setiap individu dapat dilihat di Tabel 4.15

Tabel 4.14 Hasil *Random* Pembangkitan Nilai r

C(t)	parent	1		2		3	
		r1	r2	r1	r2	r1	r1
c1	p1	0.85168	0.32823	0.14044	0.10646	0.40346	0.60753
c2		0.43624	0.97686	0.70465	0.44821	0.14443	0.01136
c3	p2	0.56524	0.58004	0.34734	0.96108	0.15482	0.40852
c4		0.61636	0.73616	0.29808	0.94770	0.85122	0.42037
c5	p3	0.32373	0.59207	0.22123	0.30446	0.76069	0.84769
c6		0.72772	0.94360	0.60206	0.45258	0.91111	0.25746
c7	p4	0.69076	0.28194	0.13803	0.85972	0.01935	0.05588
c8		0.72042	0.12972	0.38278	0.14581	0.84297	0.01969
c9	p5	0.87390	0.71004	0.90437	0.22947	0.17015	0.98008
c10		0.41479	0.17378	0.42355	0.76612	0.03362	0.11794
c11	p6	0.83724	0.15350	0.68068	0.96403	0.68796	0.92169
c12		0.90873	0.43225	0.63727	0.43282	0.88632	0.80126
c13	p7	0.66347	0.60921	0.40742	0.94640	0.84166	0.14742
c14		0.31778	0.14615	0.50105	0.30107	0.62810	0.01490
c15	p8	0.99685	0.40539	0.36994	0.96403	0.44821	0.81359
c16		0.94285	0.79355	0.31408	0.06839	0.29986	0.37283
c17	p9	0.60579	0.94184	0.82317	0.88194	0.86819	0.44198
c18		0.15326	0.75369	0.80013	0.00740	0.55179	0.28209

c19	p10	0.53105	0.88874	0.60246	0.84388	0.12170	0.15098
c20		0.18381	0.81798	0.81548	0.82101	0.68405	0.58141

Tabel 4.15 Hasil Perhitungan Individu Baru

C(t)	P	N1(0.1)	N2(0.1)	N3(0.1)	x'_1	x'_2	x'_3	total gen
c1	p1	0.49957	1.22871	-0.84262	3.33247	6.39326	2.52586	12.25160
c2		-0.18665	0.26747	0.14028	3.01789	6.26936	3.30220	12.58944
c3	p2	-0.51483	-0.35212	1.05016	3.37024	5.99653	4.38604	13.75281
c4		-0.98007	-0.50210	0.27228	3.17828	5.97913	3.83308	12.99050
c5	p3	-0.82118	1.63628	-0.60464	3.09234	6.33468	3.19026	12.61727
c6		-0.27667	0.29573	0.43102	3.40730	6.05278	3.29689	12.75697
c7	p4	0.84293	-1.53562	0.96615	4.98536	5.80103	5.58644	16.37282
c8		0.58935	1.09937	0.07213	4.84140	7.73714	4.68638	17.26493
c9	p5	-0.50292	0.44465	-0.23492	3.70189	6.39593	3.29951	13.39732
c10		1.17739	-1.30408	1.75847	4.57663	6.06498	3.48422	14.12583
c11	p6	0.48981	-0.19654	-0.40859	4.93924	6.21099	6.45179	17.60201
c12		0.18067	0.38888	-0.46602	4.78382	6.54419	6.42860	17.75662
c13	p7	-0.57393	-0.44283	0.46935	4.49949	6.31657	6.46924	17.28530
c14		1.20312	1.11563	0.09015	5.30351	7.11491	6.33146	18.74989
c15	p8	0.04448	-0.31598	-1.16710	5.13120	6.76477	6.07080	17.96677
c16		-0.33031	0.63405	1.11231	4.96162	7.25144	6.89903	19.11209
c17	p9	-0.35778	-0.42149	0.18955	3.90593	6.35667	2.90276	13.16536
c18		-1.93630	0.03104	1.06840	3.25465	6.40917	3.52750	13.19132
c19	p10	-0.72399	-0.83660	1.66784	3.92347	5.82625	3.30439	13.05412
c20		-1.67518	-0.57620	-0.42659	3.42830	5.87554	3.11032	12.41415

4.3.4.2 Menghitung nilai *fitness*

Pada tahap ini akan dihitung nilai *fitness* dari setiap individu *parent* dan individu hasil *offspring* untuk populasi generasi ke-2. Hal yang pertama dilakukan adalah menghitung total cost. Setelah itu menghitung nutrisi kemudian menghitung nilai *penalty* dan terakhir menghitung nilai *fitness*. Untuk menghitung biaya bahan dapat menggunakan persamaan (4-2).

Tabel 4.16 Contoh perhitungan total harga p1

P	Gen Kromosom			Total Gen	Total Cost
	T. Ikan	D. Gandum	Sorghum		
	16000/Kg	5000/Kg	7000/Kg		
P1	3.10345	6.23488	3.19139	12.52973	74106

$$\text{harga tepung ikan} = (\text{bobot bahan} \times \text{harga})$$

$$= \left(\left(\frac{3.10345}{12.52973} \right) \times 9 \right) \times 16000$$

$$= 35667$$

$$\text{harga dedak gandum} = (\text{bobot bahan} \times \text{harga})$$

$$= \left(\left(\frac{6.23488}{12.52973} \right) \times 9 \right) \times 5000$$

$$= 22392$$

$$\text{harga sorghum} = (\text{bobot bahan} \times \text{harga})$$

$$= \left(\left(\frac{3.19139}{12.52973} \right) \times 9 \right) \times 7000$$

$$= 16046$$

$$\text{Total cost} = \text{harga tepung gandum} \times \text{harga dedak gandum} \times \text{harga sorghum}$$

$$= 35667 + 22392 + 16046$$

$$= 74106$$

Hasil lengkap perhitungan total cost tiap individu dalam populasi awal pada Tabel 4.17 dan individu *offspring* pada Tabel 4.18.

Tabel 4.17 Hasil perhitungan total cost *parent*

P	Harga			Total cost
	x1	x2	x3	
	16000/Kg	5000/Kg	7000/Kg	
p1	35667	22392	16046	74106
p2	38908	20490	17292	76690
p3	40100	21044	15995	77139
p4	40435	19428	18110	77974
p5	41979	20890	15389	78257
p6	38326	16137	23641	78104
p7	38936	16729	22545	78210
p8	39714	16819	22079	78611
p9	44130	21792	13184	79106
p10	46093	20045	14771	80909

Tabel 4.18 Hasil perhitungan total cost *offspring*

C(t)	P	Harga			Total cost
		x1	x2	x3	
		16000/Kg	5000/Kg	7000/Kg	
c1	p1	39168	23482	12988	75639
c2		34519	22409	16525	73453
c3	p2	35288	19621	20092	75001
c4		35231	20712	18589	74533

c5	p3	35293	22593	15929	73815
c6		38461	21351	16282	76094
c7	p4	43847	15944	21496	81286
c8		40380	20166	17101	77647
c9	p5	39789	21483	15516	76788
c10		46655	19321	15539	81515
c11	p6	40407	15879	23092	79378
c12		38795	16585	22809	78188
c13	p7	37484	16444	23579	77507
c14		40731	17076	21274	79081
c15	p8	41125	16943	21287	79356
c16		37383	17074	22742	77199
c17	p9	42722	21727	13891	78340
c18		35529	21864	16847	74239
c19	p10	43280	20084	15947	79311
c20		39767	21298	15784	76850

Setelah menghitung total cost yang dilakukan adalah menghitung total nilai yang terkandung pada masing masing bahan. Untuk menghitung total nutrisi bahan bisa menggunakan persamaan (4-3) contoh perhitungan total nutrisi yang terkandung pada setiap bahan pakan ikan.

Tabel 4.19 Contoh Perhitungan Total Nutrisi

P	Tepung Ikan			Dedak Gandum			Sorghum			Total gen	Total Nutrisi		
	P	L	S	P	L	S	P	L	S		P	L	S
	60.50 %	9.40 %	0.70 %	11.99 %	1.48 %	3.75 %	13.00 %	2.05 %	13.50 %				
p1	3.10345			6.23488			3.19139			12.52973	2.18	0.32	0.49

Nutrisi Tepung Ikan

$$\begin{aligned}
 \text{protein} &= \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right) \\
 &= \left(\left(\frac{3.10345}{12.52973} \right) \times 9 \right) \times \left(\frac{60.5}{100} \right) \\
 &= 1.349
 \end{aligned}$$

$$\begin{aligned}
 \text{lemak} &= \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right) \\
 &= \left(\left(\frac{3.10345}{12.52973} \right) \times 9 \right) \times \left(\frac{9.4}{100} \right) \\
 &= 0.21
 \end{aligned}$$

$$\text{Serat} = \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right)$$



$$= \left(\left(\frac{3.10345}{12.52973} \right) \times 9 \right) \times \left(\frac{0.7}{100} \right)$$

$$= 0.016$$

Nutrisi Dedak Gandum

$$\text{protein} = \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right)$$

$$= \left(\left(\frac{6.23488}{12.52973} \right) \times 9 \right) \times \left(\frac{11.99}{100} \right)$$

$$= 0.537$$

$$\text{lemak} = \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right)$$

$$= \left(\left(\frac{6.23488}{12.52973} \right) \times 9 \right) \times \left(\frac{1.48}{100} \right)$$

$$= 0.066$$

$$\text{Serat} = \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right)$$

$$= \left(\left(\frac{6.23488}{12.52973} \right) \times 9 \right) \times \left(\frac{3.75}{100} \right)$$

$$= 0.168$$

Nutrisi Sorghum

$$\text{protein} = \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right)$$

$$= \left(\left(\frac{3.19139}{12.52973} \right) \times 9 \right) \times \left(\frac{13}{100} \right)$$

$$= 0.298$$

$$\text{lemak} = \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right)$$

$$= \left(\left(\frac{3.19139}{12.52973} \right) \times 9 \right) \times \left(\frac{2.05}{100} \right)$$

$$= 0.047$$

$$\text{Serat} = \left(\text{bobot bahan} \times \left(\frac{\text{nutrisi}}{100} \right) \right)$$

$$= \left(\left(\frac{3.19139}{12.52973} \right) \times 9 \right) \times \left(\frac{13.5}{100} \right)$$

$$= 0.309$$

Total nutrisi protein p1 = protein tepung ikan + protein dedak gandum

$$+ \text{protein sorghum}$$

$$= 1.349 + 0.537 + 0.298$$

$$= 2.18$$

$$\begin{aligned}
 \text{Total nutrisi lemak p1} &= \text{lemak tepung ikan} + \text{lemak dedak gandum} \\
 &\quad + \text{lemak sorghum} \\
 &= 0.21+0.066+0.047 \\
 &= 0.32
 \end{aligned}$$

$$\begin{aligned}
 \text{Total nutrisi serat p1} &= \text{serat tepung ikan} + \text{serat dedak gandum} \\
 &\quad + \text{serat sorghum} \\
 &= 0.016+0.168+0.309 \\
 &= 0.49
 \end{aligned}$$

Untuk melihat hasil perhitungan total nutrisi secara lengkap dapat dilihat pada Tabel 4.20 dan 4.21.

Tabel 4.20 Total Nutrisi Parent

Parent	Tepung Ikan			Dedak Gandum			Sorghum			Total Nutrisi		
	P	L	S	P	L	S	P	L	S	P	L	S
	60.50%	9.40%	0.70%	11.99%	1.48%	3.75%	13.00%	2.05%	13.50%			
p1	1.349	0.210	0.016	0.537	0.066	0.168	0.298	0.047	0.309	2.18	0.32	0.49
p2	1.471	0.229	0.017	0.491	0.061	0.154	0.321	0.051	0.333	2.28	0.34	0.50
p3	1.516	0.236	0.018	0.505	0.062	0.158	0.297	0.047	0.308	2.32	0.34	0.48
p4	1.529	0.238	0.018	0.466	0.058	0.146	0.336	0.053	0.349	2.33	0.35	0.51
p5	1.587	0.247	0.018	0.501	0.062	0.157	0.286	0.045	0.297	2.37	0.35	0.47
p6	1.449	0.225	0.017	0.387	0.048	0.121	0.439	0.069	0.456	2.28	0.34	0.59
p7	1.472	0.229	0.017	0.401	0.050	0.125	0.419	0.066	0.435	2.29	0.34	0.58
p8	1.502	0.233	0.017	0.403	0.050	0.126	0.410	0.065	0.426	2.32	0.35	0.57
p9	1.669	0.259	0.019	0.523	0.065	0.163	0.245	0.039	0.254	2.44	0.36	0.44
p10	1.743	0.271	0.020	0.481	0.059	0.150	0.274	0.043	0.285	2.50	0.37	0.46

Tabel 4.21 Total Nutrisi Offspring

C(t)	Tepung Ikan			Dedak Gandum			Sorghum			Total Nutrisi		
	P	L	S	P	L	S	P	L	S	P	L	S
	60.50%	9.40%	0.70%	11.99%	1.48%	3.75%	13.00%	2.05%	13.50%			
c1	1.481	0.230	0.017	0.563	0.070	0.176	0.241	0.038	0.250	2.29	0.34	0.44
c2	1.305	0.203	0.015	0.537	0.066	0.168	0.307	0.048	0.319	2.15	0.32	0.50



c3	1.334	0.207	0.015	0.471	0.058	0.147	0.373	0.059	0.387	2.18	0.32	0.55
c4	1.332	0.207	0.015	0.497	0.061	0.155	0.345	0.054	0.359	2.17	0.32	0.53
c5	1.335	0.207	0.015	0.542	0.067	0.169	0.296	0.047	0.307	2.17	0.32	0.49
c6	1.454	0.226	0.017	0.512	0.063	0.160	0.302	0.048	0.314	2.27	0.34	0.49
c7	1.658	0.258	0.019	0.382	0.047	0.120	0.399	0.063	0.415	2.44	0.37	0.55
c8	1.527	0.237	0.018	0.484	0.060	0.151	0.318	0.050	0.330	2.33	0.35	0.50
c9	1.505	0.234	0.017	0.515	0.064	0.161	0.288	0.045	0.299	2.31	0.34	0.48
c10	1.764	0.274	0.020	0.463	0.057	0.145	0.289	0.046	0.300	2.52	0.38	0.47
c11	1.528	0.237	0.018	0.381	0.047	0.119	0.429	0.068	0.445	2.34	0.35	0.58
c12	1.467	0.228	0.017	0.398	0.049	0.124	0.424	0.067	0.440	2.29	0.34	0.58
c13	1.417	0.220	0.016	0.394	0.049	0.123	0.438	0.069	0.455	2.25	0.34	0.59
c14	1.540	0.239	0.018	0.409	0.051	0.128	0.395	0.062	0.410	2.34	0.35	0.56
c15	1.555	0.242	0.018	0.406	0.050	0.127	0.395	0.062	0.411	2.36	0.35	0.56
c16	1.414	0.220	0.016	0.409	0.051	0.128	0.422	0.067	0.439	2.25	0.34	0.58
c17	1.615	0.251	0.019	0.521	0.064	0.163	0.258	0.041	0.268	2.39	0.36	0.45
c18	1.343	0.209	0.016	0.524	0.065	0.164	0.313	0.049	0.325	2.18	0.32	0.50
c19	1.637	0.254	0.019	0.482	0.059	0.151	0.296	0.047	0.308	2.41	0.36	0.48
c20	1.504	0.234	0.017	0.511	0.063	0.160	0.293	0.046	0.304	2.31	0.34	0.48

Setelah selesai menghitung total nutrisi sekarang menghitung nilai *penalty*. Proses perhitungan *penalty* digenerasi ke-2 sama dengan digenerasi ke-1 untuk menghitung total *penalty* yang dilakukan adalah menghitung kebutuhan nutrisi. Dari data yang diketahui nutrisi yang dibutuhkan adalah protein 20%, Lemak, 8% dan serat 4%. Contoh perhitungan mencari jumlah nutrisi protein , lemak, serat yang dibutuhkan dalam 9Kg.

$$\begin{aligned}
 \text{protein} &= ((\text{nutrisi yang dibutuhkan}/100) \times \text{jumlah pakan}) \\
 &= ((20/100) \times 9) \\
 &= 1.8
 \end{aligned}$$

$$\begin{aligned}
 \text{lemak} &= ((\text{nutrisi yang dibutuhkan}/100) \times \text{jumlah pakan}) \\
 &= ((8/100) \times 9) \\
 &= 0.72
 \end{aligned}$$



$$\begin{aligned}
 \text{serat} &= ((\text{nutrisi yang dibutuhkan}/100) \times \text{jumlah pakan}) \\
 &= ((4/100) \times 9) \\
 &= 0.36
 \end{aligned}$$

Hasil perhitungan kebutuhan nutrisi ditunjukkan pada Tabel 4.22. Setelah selesai menghitung kebutuhan nutrisi, yang dilakukan adalah menghitung total *penalty*. Nilai total *penalty* dihitung setiap individu. Contoh perhitungan nilai *penalty* pada individu p1

$$\begin{aligned}
 \text{Penalty protein} &= |\text{total nutrisi protein yang dibutuhkan} - \text{total nutrisi protein yang tersedia}| \\
 &= |1.8 - 2.18| \\
 &= 0.38
 \end{aligned}$$

$$\begin{aligned}
 \text{Penalty lemak} &= |\text{total nutrisi lemak yang dibutuhkan} - \text{total nutrisi lemak yang tersedia}| \\
 &= |0.72 - 0.32| \\
 &= 0.40
 \end{aligned}$$

$$\begin{aligned}
 \text{Penalty serat} &= |\text{total nutrisi serat yang dibutuhkan} - \text{total nutrisi serat yang tersedia}| \\
 &= |0.36 - 0.49| \\
 &= 0.13
 \end{aligned}$$

$$\begin{aligned}
 \text{Total penalty} &= \text{penalt protein} + \text{penalty lemak} + \text{penalty serat} \\
 &= 0.38 + 0.40 + 0.13 \\
 &= 0.91
 \end{aligned}$$

Hasil lengkap perhitungan *penalty* terdapat pada Tabel 4.22 dan 4.23.

Tabel 4.22 Perhitungan *Penalty Parent*

Parent	Kebutuhan			Tersedia			<i>penalty</i>			Total <i>Penalty</i>
	P	L	S	P	L	S	P	L	S	
p1	1.8	0.72	0.36	2.18	0.32	0.49	0.38	0.40	0.13	0.91
p2	1.8	0.72	0.36	2.28	0.34	0.5	0.48	0.38	0.14	1.01
p3	1.8	0.72	0.36	2.32	0.34	0.48	0.52	0.38	0.12	1.02
p4	1.8	0.72	0.36	2.33	0.35	0.51	0.53	0.37	0.15	1.06
p5	1.8	0.72	0.36	2.37	0.35	0.47	0.57	0.37	0.11	1.05
p6	1.8	0.72	0.36	2.28	0.34	0.59	0.48	0.38	0.23	1.09

p7	1.8	0.72	0.36	2.29	0.34	0.58	0.49	0.38	0.22	1.09
p8	1.8	0.72	0.36	2.32	0.35	0.57	0.52	0.37	0.21	1.10
p9	1.8	0.72	0.36	2.44	0.36	0.44	0.64	0.36	0.08	1.07
p10	1.8	0.72	0.36	2.5	0.37	0.46	0.70	0.35	0.10	1.14

Tabel 4.23 Perhitungan *Penalty Offspring*

C(t)	Kebutuhan			Tersedia			<i>penalty</i>			Total <i>Penalty</i>
	P	L	S	P	L	S	P	L	S	
c1	1.8	0.72	0.36	2.29	0.34	0.44	0.49	0.38	0.08	0.95
c2	1.8	0.72	0.36	2.15	0.32	0.5	0.35	0.40	0.14	0.89
c3	1.8	0.72	0.36	2.18	0.32	0.55	0.38	0.40	0.19	0.96
c4	1.8	0.72	0.36	2.17	0.32	0.53	0.37	0.40	0.17	0.94
c5	1.8	0.72	0.36	2.17	0.32	0.49	0.37	0.40	0.13	0.90
c6	1.8	0.72	0.36	2.27	0.34	0.49	0.47	0.38	0.13	0.98
c7	1.8	0.72	0.36	2.44	0.37	0.55	0.64	0.35	0.19	1.19
c8	1.8	0.72	0.36	2.33	0.35	0.5	0.53	0.37	0.14	1.04
c9	1.8	0.72	0.36	2.31	0.34	0.48	0.51	0.38	0.12	1.00
c10	1.8	0.72	0.36	2.52	0.38	0.47	0.72	0.34	0.11	1.16
c11	1.8	0.72	0.36	2.34	0.35	0.58	0.54	0.37	0.22	1.13
c12	1.8	0.72	0.36	2.29	0.34	0.58	0.49	0.38	0.22	1.09
c13	1.8	0.72	0.36	2.25	0.34	0.59	0.45	0.38	0.23	1.07
c14	1.8	0.72	0.36	2.34	0.35	0.56	0.54	0.37	0.20	1.11
c15	1.8	0.72	0.36	2.36	0.35	0.56	0.56	0.37	0.20	1.12
c16	1.8	0.72	0.36	2.25	0.34	0.58	0.45	0.38	0.22	1.05
c17	1.8	0.72	0.36	2.39	0.36	0.45	0.59	0.36	0.09	1.05
c18	1.8	0.72	0.36	2.18	0.32	0.5	0.38	0.40	0.14	0.92
c19	1.8	0.72	0.36	2.41	0.36	0.48	0.61	0.36	0.12	1.09
c20	1.8	0.72	0.36	2.31	0.34	0.48	0.51	0.38	0.12	1.01

Setelah selesai menghitung *penalty* maka selanjutnya menghitung nilai *fitness* nilai constanta yang di gunakan adalah 10000. Contoh perhitungan nilai *fitness* pada individu p1



$$\begin{aligned}
 \text{fitness } p1 &= \frac{10000}{\text{total cost} + (\text{total penalty} \times 10000)} \\
 &= \frac{10000}{74106 + (0.91 \times 10000)} \\
 &= 0.12013
 \end{aligned}$$

Fitness dihitung setiap individu. Hasil perhitungan nilai *fitness* setiap individu pada populasi awal dan *offspring* dapat dilihat setiap individu ditunjukkan pada Tabel 4.24 dan 4.25

Tabel 4.24 Hasil perhitungan nilai *fitness* parent

Parent	Kromosom			σ_1	σ_2	σ_3	Total cost	total penalty	fitnes
	x1	x2	X3						
1	3.10345	6.23488	3.19139	0.45843	0.12890	0.78984	74106	0.91	0.12013
2	3.58265	6.03738	3.63953	0.41259	0.11601	0.71085	76690	1.01	0.11525
3	3.56733	5.99060	3.25251	0.57843	0.21028	0.10296	77139	1.02	0.11453
4	4.50682	6.92936	4.61377	0.56770	0.73477	1.00675	77974	1.06	0.11295
5	3.96370	6.31178	3.32128	0.52058	0.18925	0.09266	78257	1.05	0.11264
6	4.69300	6.32285	6.61674	0.50272	0.56918	0.40372	78104	1.09	0.11240
7	4.75916	6.54342	6.29870	0.45245	0.51226	0.36335	78210	1.09	0.11228
8	5.11107	6.92664	6.49487	0.45245	0.51226	0.36335	78611	1.1	0.11164
9	4.05355	6.40557	2.76802	0.41259	0.11601	0.71085	79106	1.07	0.11134
10	4.30037	5.98458	3.14985	0.52058	0.18925	0.09266	80909	1.14	0.10833

Tabel 4.25 Hasil perhitungan nilai *fitness* offspring

C(t)	Parent	x1'	x2'	X3'	σ_1'	σ_2'	σ_3'	total cost	total penalty	<i>fitness</i>
c1	p1	3.33247	6.39326	2.52586	0.50427	0.14179	0.86882	75639	0.95	0.11743
c2		3.01789	6.26936	3.3022	0.50427	0.14179	0.86882	73453	0.89	0.12137
c3	p2	3.37024	5.99653	4.38604	0.45385	0.12761	0.78194	75001	0.96	0.11815
c4		3.17828	5.97913	3.83308	0.45385	0.12761	0.78194	74533	0.94	0.11913
c5	p3	3.09234	6.33468	3.19026	0.63627	0.23131	0.11325	73815	0.9	0.12070
c6		3.4073	6.05278	3.29689	0.63627	0.23131	0.11325	76094	0.98	0.11638
c7	p4	4.98536	5.80103	5.58644	0.62447	0.80825	1.10742	81286	1.19	0.10737



c8		4.8414	7.73714	4.68638	0.62447	0.80825	1.10742	77647	1.04	0.11358
c9	p5	3.70189	6.39593	3.29951	0.57264	0.20818	0.10193	76788	1	0.11519
c10		4.57663	6.06498	3.48422	0.57264	0.20818	0.10193	81515	1.16	0.10735
c11	p6	4.93924	6.21099	6.45179	0.45245	0.51226	0.36335	79378	1.13	0.11031
c12		4.78382	6.54419	6.4286	0.45245	0.51226	0.36335	78188	1.09	0.11230
c13	p7	4.49949	6.31657	6.46924	0.49769	0.56349	0.39968	77507	1.07	0.11342
c14		5.30351	7.11491	6.33146	0.40720	0.46104	0.32701	79081	1.11	0.11090
c15	p8	5.1312	6.76477	6.0708	0.40720	0.46104	0.32701	79356	1.12	0.11045
c16		4.96162	7.25144	6.89903	0.49769	0.56349	0.39968	77199	1.05	0.11401
c17	p9	3.90593	6.35667	2.90276	0.45385	0.12761	0.78194	78340	1.05	0.11259
c18		3.25465	6.40917	3.5275	0.45385	0.12761	0.78194	74239	0.92	0.11982
c19	p10	3.92347	5.82625	3.30439	0.57264	0.20818	0.10193	79311	1.09	0.11084
c20		3.4283	5.87554	3.11032	0.57264	0.20818	0.10193	76850	1.01	0.11506

Karena salah satu nilai *fitness offspring* ada yang lebih besar dari pada nilai *fitness* induknya maka nilai σ' untuk *offspring* dinaikan dengan cara $\sigma \times 1.1$.

4.3.4.3 Poses seleksi

Tahap ini merupakan tahapan terakhir dari algoritma *evolution strategies*. Dalam tahapan ini yang diambil adalah nilai *fitness* tertinggi dari populasi gabungan antara *parent* dan *offspring*. Hasil seleksi yang menghasilkan 10 individu ditunjukkan pada Tabel 4.26.

Tabel 4.26 hasil seleksi

P(t+1)	parent	x1	x2	X3	σ_1	σ_2	σ_3	<i>fitness</i>	total gen
p1	c2	3.01789	6.26936	3.3022	0.50427	0.14179	0.86882	0.12137	12.58945
p2	c5	3.09234	6.33468	3.19026	0.63627	0.23131	0.11325	0.12070	12.61728
p3	p1	3.103454	6.234881	3.191395	0.45843	0.128901	0.789838	0.12013	12.52973
p4	c18	3.25465	6.40917	3.5275	0.45385	0.12761	0.78194	0.11982	13.19132
p5	c4	3.17828	5.97913	3.83308	0.45385	0.12761	0.78194	0.11913	12.99049
p6	c3	3.37024	5.99653	4.38604	0.45385	0.12761	0.78194	0.11815	13.75281

p7	c1	3.33247	6.39326	2.52586	0.50427	0.14179	0.86882	0.11743	12.25159
p8	c6	3.4073	6.05278	3.29689	0.63627	0.23131	0.11325	0.11638	12.75697
p9	p2	3.582649	6.037384	3.639532	0.41258	0.116011	0.710854	0.11525	13.25956
p10	c9	3.70189	6.39593	3.29951	0.57264	0.20818	0.10193	0.11519	13.39733

Pada contoh perhitungan manual pada generasi ke-2 nilai *fitness* yang tertinggi adalah p1. Sehingga p1 adalah individu yang terbaik di generasinya ke 1.

4.3.5 Hasil Rekomendasi Komposisi Pakan

Pada pakan ikan bandeng membutuhkan pakan ikan sebanyak 9kg. Komposisi pakan yang dihasilkan sebagai berikut

$$\text{Tepung ikan} = \left(\left(\frac{3.01789}{12.58945} \right) \times 9 \right) = 2.15744 \text{ Kg}$$

$$\text{Dedak gandum} = \left(\left(\frac{6.26936}{12.58945} \right) \times 9 \right) = 4.48187 \text{ Kg}$$

$$\text{Sorghum} = \left(\left(\frac{3.3022}{12.58945} \right) \times 9 \right) = 2.36069 \text{ Kg}$$

$$\begin{aligned} \text{Total Harga} &= (2.15744 \times 16000) + (4.48187 \times 5000) + (2.36069 \times 7000) \\ &= 73453 \end{aligned}$$

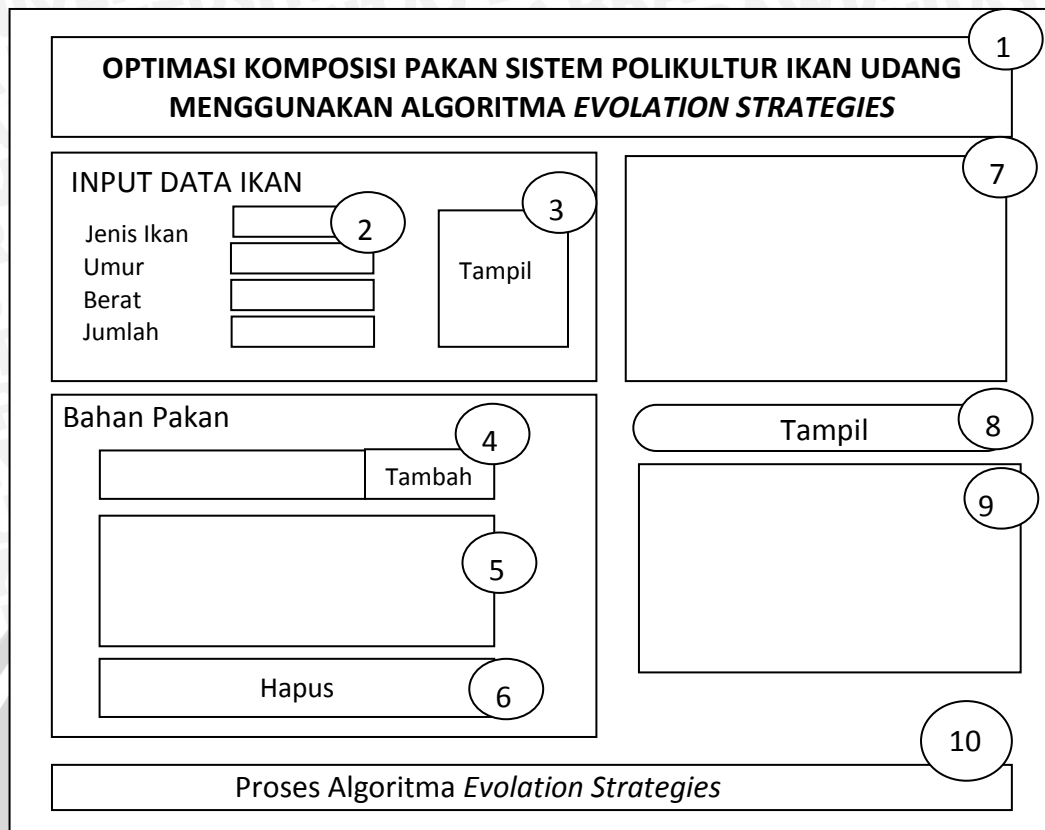
Sehingga untuk menghasilkan pakan ikan bandeng sebanyak 9kg dibutuhkan tepung ikan sebanyak 2.15744 Kg, dedak gandum sebanyak 4.48187 Kg dan Sorghum sebanyak 2.3069Kg dengan total harga Rp 73453.

4.4 Perancangan Antarmuka

Pada perancangan antarmuka aplikasi optimasi komposisi pakan ikan sistem polikultur dengan algoritma *evolution strategies* terdapat dua halaman antarmuka yaitu halaman input data ikan dan bahan dan halaman proses *evolution strategies*.

4.4.1 Rancangan Tampilan Halaman Input Data Ikan dan Bahan

Rancangan tampilan halaman input data ikan dan bahan digunakan untuk menerima informasi ikan, dan bahan baku yang digunakan. Pada halaman ini akan menampilkan kandungan nutrisi dan kebutuhan nutrisi ikan berdasarkan data yang di inputkan. Rancangan tampilan halaman input ditunjukkan pada Gambar 4.7



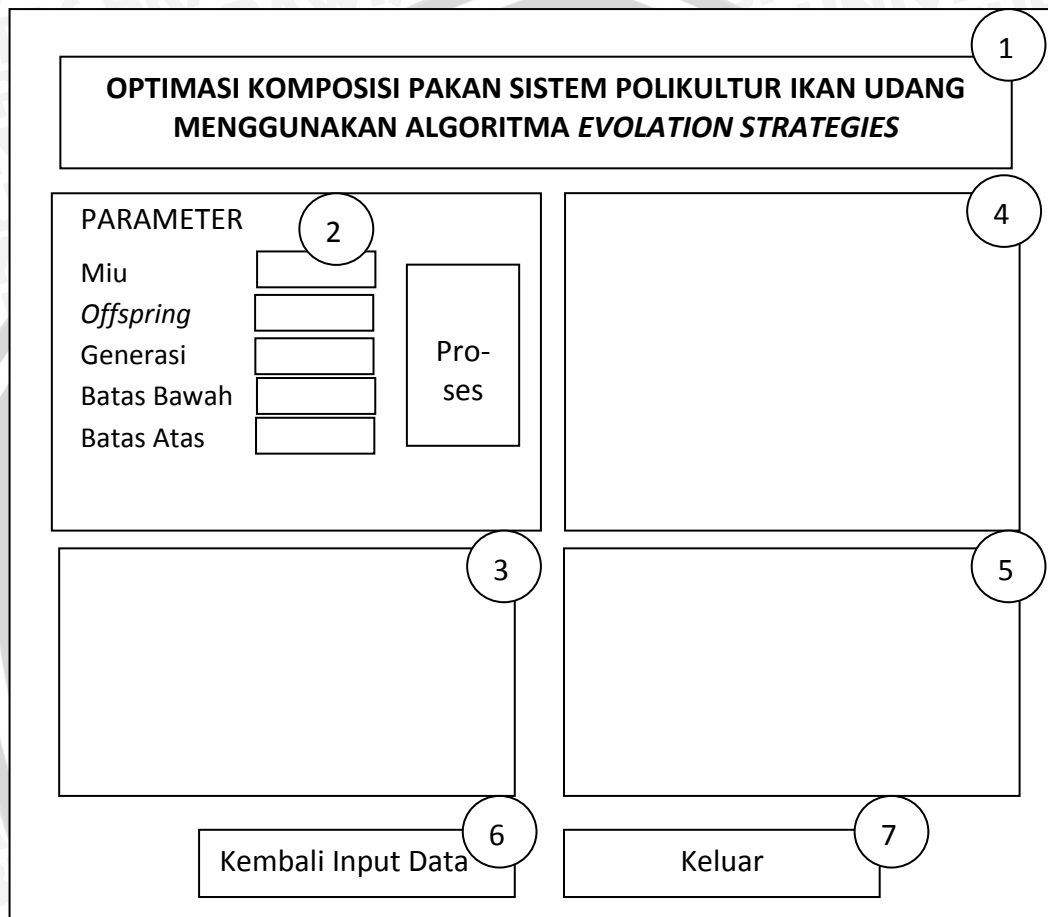
Gambar 4.7 Rancangan Tampilan Halaman Inout Data Ikan Bahan

Keterangan Gambar 4.7 :

1. Judul sistem
2. Tempat untuk menginputkan data ikan berupa jenis ikan, umur, berat, jumlah dan ikan.
3. Tomol untuk menampilkan data ikan yang diinputkan dan kebutuhan nutrisi
4. Tombol untuk menambah ahan pakan
5. Tempat untuk menampilkan ahan pakan yang dipilih
6. Tomol untuk menghapus bahan pakan yang telah dipilih
7. Tempat untuk menampilkan rekomendasi padat tebar udang dan nutrisi yang dibutuhkan ikan.
8. Tomol untuk menampilkan bahan pakan yang dipilih beserta nutrisinya dan harga.
9. Menampilkan data bahan baku yang digunakan beserta kandungan nutrisinya.
10. *Button* untuk menuju halaman algoritma *evolution strategies*.

4.4.2 Rancangan Tampilan Halaman Proses Algoritma *Evolution Strategies*

Rancangan tampilan halaman antar muka algoritma *evolution strategies* untuk permasalahan optimasi komposisi pakan ikan. Halaman antar muka algoritma *evolution strategies* digunakan untuk proses perhitungan menggunakan algoritma *evolution strategies*. Rancangan optimasi komposisi pakan ikan dapat dilihat pada Gambar 4.8 berikut :



Gambar 4.8 Rancangan Antarmuka Halaman Algoritma Genetika

Keterangan Gambar 4.8 :

1. Judul sistem
2. Tempat untuk menginputkan parameter algoritma *evolution strategies*
3. Tempat untuk menampilkan individu awal
4. Tempat untuk menampilkan individu hasil proses seleksi
5. Tempat untuk menampilkan rekomendasi komposisi bahan pakan
6. Tombol untuk kembali kehalaman input data ikan
7. Tomol untuk keluar.

4.5 Perancangan Uji Coba dan Evaluasi

Perancangan uji coba untuk mendapatkan nilai parameter optimal. Perancangan uji coba akan digunakan untuk mengevaluasi program. Uji coba akan menggunakan data ikan bandeng dengan usia 22 minggu, berat 0.6 kg, jumlah ikan 100 ekor dan bahan yang digunakan tepung ikan, dedak gandum, dan sorghum. Uji coba tersebut antara lain

1. Uji coba untuk menentukan rentang kromosom yang optimal
2. Uji coba untuk menentukan ukuran populasi yang optimal.
3. Uji coba untuk menentukan ukuran *offspring* yang optimal.
4. Uji coba untuk menentukan banyaknya generasi yang optimal.
5. Uji coba untuk menentukan siklus algoritma ES yang optimal
6. Uji coba untuk menentukan persamaan fitness yang optimal

4.5.1 Uji Coba Rentang Kromosom

Uji coba rentang kromosom digunakan untuk mengetahui rentang kromosom yang tepat untuk dapat menghasilkan komposisi pakan ikan yang optimal. Rentang kromosom yang digunakan adalah 1-10, 1-20, 1-30, 1-40, 1-50, 1-60, 1-70, 1-80, 1-90, 1-100. Rancangan uji coba rentang kromosom ditunjukkan pada Tabel 4.27

Tabel 4.27 Rancangan Uji Coba Rentang Kromosom

Rentang Kromosom	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
1-10											
1-20											
1-30											
1-40											
1-50											
1-60											
1-70											
1-80											
1-90											
1-100											

4.5.2 Uji Coba Ukuran Populasi

Uji coba ukuran populasi digunakan untuk mengetahui ukuran populasi yang tepat untuk dapat menghasilkan komposisi pakan ikan yang optimal. Ukuran

populasi yang digunakan adalah 10, 20,30,40,50,60,70,80,90,100. Rancangan uji coba ukuran populasi ditunjukkan pada Tabel 4.28

Tabel 4.28 Rancangan Uji Coba Ukuran Populasi

Banyak Populasi	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
10											
20											
30											
40											
50											
60											
70											
80											
90											
100											

4.5.3 Uji Coba Ukuran *Offspring*

Uji coba ukuran *offspring* digunakan untuk mengetahui ukuran *offspring* yang tepat untuk menghasilkan komposisi pakan ikan yang optimal. Ukuran *offspring* yang digunakan adalah 1 μ , 2 μ , 3 μ , 4 μ , 5 μ , 6 μ , 7 μ , 8 μ , 9 μ , 10 μ Rancangan uji coba ukuran populasi ditunjukkan pada Tabel 4.29

Tabel 4.29 Rancangan Uji Coba Ukuran *Offspring*

Banyak <i>offspring</i>	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
1 μ											
2 μ											
3 μ											
4 μ											
5 μ											
6 μ											
7 μ											
8 μ											
9 μ											
10 μ											

4.5.4 Uji Coba Banyaknya Generasi

Uji coba generasi ini digunakan untuk mengetahui banyaknya generasi yang dibutuhkan untuk mendapatkan hasil komposisi bahan pakan ikan yang optimal. Banyak generasi yang digunakan adalah 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000. Rancangan uji coba banyaknya generasi ditunjukkan pada Tabel 4.30

Tabel 4.30 Rancangan Uji Coba Banyaknya Generasi

Banyak Generasi	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
100											
200											
300											
400											
500											
600											
700											
800											
900											
1000											

4.5.5 Uji Coba Siklus Algoritma *Evolution Strategies*

Uji coba siklus algoritma ini digunakan untuk mengetahui siklus algoritma yang akan mendapatkan hasil lebih optimal dalam melakukan optimasi komposisi pakan sistem polikultur ikan dan udang. Rancangan uji coba siklus algoritma *evolution strategies* ditunjukkan pada Tabel 4.31

Tabel 4.31 Rancangan Uji Coba Tipe Algoritma *Evolution Strategies*

Tipe ES	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
(μ, λ)											
$(\mu + \lambda)$											
$(\mu/r, \lambda)$											
$(\mu/r + \lambda)$											

4.5.6 Uji Coba Persamaan *Fitness*

Uji Coba persamaan *fitness* ini digunakan untuk mengetahui persamaan *fitness* yang terbaik didalam melakukan optimasi komposisi pakan sistem

polikultur ikan dan udang . Persamaan yang digunakan dalam uji coa persamaan *fitness* sebanyak 5 persamaan yaitu :

1. Persamaan A

$$Fitness = \frac{10000}{total\ harga + penalty} \quad (4.9)$$

Persamaan A dilakukan pengujian karena pada persamaan A terdapat fungsi yang diminimalkan yaitu total harga dan *penalty*. Pengujian dilakukan untuk melihat solusi yang dihasilkan oleh persamaan A dalam mengoptimasi komposisi pakan sistem polikultur ikan dan udang.

2. Persamaan B

$$Fitness = \frac{10000}{total\ harga + (penalty \times 10000)} \quad (4.10)$$

Persamaan B dilakukan pengujian karena persamaan B terdapat pada penelitian sebelumnya yang dilakukan oleh Milah(2015) yang meminimalkan fungsi total harga dan *penalty* serta terdapat pengali 10000. Pengali 10000 digunakan agar rentang total harga dan *penalty* sama. Pengujian dilakukan untuk melihat solusi yang dihasilkan oleh persamaan B dalam mengoptimasi komposisi pakan sistem polikultur ikan dan udang.

3. Persamaan C

$$Fitness = \frac{10000}{(total\ harga \times \alpha) + (penalty \times 10000 \times \beta)} \quad (4.11)$$

Persamaan C dilakukan pengujian karena persamaan C terdapat pembobotan untuk tingkat prioritas fungsi yang diminimalkan. α dan β adalah bilangan bobot dari tingkat prioritas. Pada penelitian ini untuk α adalah 0.7 dan β adalah 0.3. Pengujian dilakukan untuk melihat solusi yang dihasilkan oleh persamaan C dalam mengoptimasi komposisi pakan sistem polikultur ikan dan udang.

4. Persamaan D

$$Fitness = \frac{10000}{total\ harga} + \frac{10000}{penalty \times 10000 + c} \quad (4.12)$$

Persamaan D dilakukan pengujian karena persamaan D membedakan fungsi untuk total harga dan *penalty* walaupun total harga dan *penalty* merupakan fungsi yang akan diminimalkan. Untuk mendapatkan *fitness* individu dengan cara menamakan *fitness* total harga dan *fitness penalty*. Untuk mencari *fitness penalty* di tambah dengan c. c adalah bilangan konstanta yang mendekati dengan bilangan 0. pada penelitian ini c yang digunakan adalah 0.01. Tujuan penamaan 0.01 agar jika nilai *penalty* sama dengan 0 maka hasil perhitungan *fitness* dari *penalty* tidak mendapatkan hasil tak hingga. Pengujian dilakukan untuk melihat solusi yang dihasilkan oleh persamaan D dalam mengoptimasi komposisi pakan sistem polikultur ikan dan udang.

5. Persamaan E

$$Fitness = \frac{10000}{total\ harga * \alpha} + \frac{10000}{penalty\ x\ 10000 * \beta + c} \quad (4.13)$$

Pada persamaan E mirip dengan persamaan D tetapi pada persamaan E terdapat pembobotan untuk tingkat prioritas fungsi yang diminimalkan. Pada penelitian ini α yang digunakan 0.7, β yang digunakan 0.3, dan c yang digunakan adalah 0.01. Pengujian dilakukan untuk melihat solusi yang dihasilkan oleh persamaan E dalam mengoptimasi komposisi pakan sistem polikultur ikan dan udang.

Uji coba persamaan fitness akan dilakukan sebanyak 10 kali uji coba. Rancangan uji coba Persamaan *fitness* ditunjukkan pada Tabel 4.32

Tabel 4.32 Rancangan Uji Coba Pengaruh Persmaan Fitness

Pesamaan	Nilai <i>fitness</i>										Rata-rata <i>fitness</i>	
	Percobaan ke- <i>i</i>											
	1	2	3	4	5	6	7	8	9	10		
A												
B												
C												
D												
E												

Setiap uji coba akan memperoleh komposisi bahan pakan, dari setiap komposisi pakan yang diperoleh akan dihitung nutrisinya yang tersedia. Setelah menghitung nutrisi yang tersedia selanjutnya membandingkan nutrisi yang tersedia dengan nutrisi yang dibutuhkan. Rancangan perbandingan nutrisi tersedia dan nutrisi yang dibutuhkan ditunjukkan pada Tabel 4.33.

Tabel 4.33 Rancangan Uji Coba Perbandingan nutrisi

Persamaan	Kandungan Nutrisi Yang Tersedia %			Kandungan Nutrisi Yang Dibutuhkan %			Harga
	Protein	Lemak	Serat	Protein	Lemak	Serat (max)	
A							
B							
C							
D							
E							

BAB 5 IMPLEMENTASI SISTEM

Pada bab ini akan dibahas tentang implementasi sistem sesuai dengan perancangan sistem pada bab sebelumnya. Bab implementasi sistem terdiri dari spesifikasi sistem, implementasi program dan implementasi antarmuka.

5.1 Spesifikasi Sistem

Spesifikasi sistem yang digunakan untuk mengimplementasikan sistem ini terdiri dari spesifikasi perangkat keras dan spesifikasi perangkat lunak.

5.1.1 Spesifikasi Perangkat Keras

Perangkat keras yang digunakan untuk mengimplementasikan sistem ini adalah laptop dengan minimal spesifikasi perangkat keras yang ditunjukkan pada Tabel 5.1.

Tabel 5.1 Spesifikasi Perangkat Keras

Komponen	Spesifikasi
Processor	Intel® Core i3
Memori (RAM)	2 GB RAM
Hardisk	500 Gb HDD

5.1.2 Spesifikasi Perangkat Lunak

Perangkat lunak yang digunakan untuk mengimplemntasikan sistem ini menggunakan laptop yang dengan minimal spesifikasi perangkat lunak yang ditunjukkan pada Tabel 5.2

Tabel 5.2 Spesifikasi Perangkat Lunak

Sistem Operasi	Microsoft Wondows 7 (32 bit)
Bahasa Pemrograman	Java
Tools Pemrograman	Netbeans 7.3
Server	Localhost
Database	MySQL

5.2 Implementasi Program

Pada subab ini akan diahas tentang implementasi program sesuai dengan perancangan yang telah dibuat. Sistem optimasi komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma evolution strategies menggunakan bahasa pemrograman java berbasis desktop.

3. 13-26 digunakan untuk menampilkan populasi awal.

5.2.2 Implementasi Reproduksi

Pada implementasi reproduksi hanya menggunakan mutasi karena siklus ES yang digunakan adalah $(\mu+\lambda)$. Pada proses reproduksi juga terdapat proses *self adaption*. Implementasi proses mutasi dapat dilihat pada *Source Code 5.2*.

```

1     public void Mutasi() {
2         for (int i = 0; i < miu; i++) {
3             for (int j = 0; j < lamda; j++) {
4                 individu_c[c] = "C" + (c + 1);
5                 individu[c] = individu_p[i];
6                 for (int k = 0; k < jumlah_bahan; k++) {
7                     r1 = 0 + Math.random() * 1;
8                     r2 = 0 + Math.random() * 1;
9                     N = (Math.sqrt(-2 * Math.log(r1)) *
10 Math.sin(2 * Math.PI * r2));
11                     kromosom_m[c][k] = kromosom[i][k] +
12 (sigma[i][k] * N);
13                     if (kromosom_m[c][k] < batas_bawah) {
14                         kromosom_m[c][k] = batas_bawah;
15                     }
16                     if (kromosom_m[c][k] > batas_atas) {
17                         kromosom_m[c][k] = batas_atas;
18                     }
19                     total_gen[c] = total_gen[c] +
20 kromosom_m[c][k];
21                 }
22                 c++;
23             }
24         }
25         //menghitung fitness
26         fitness_c = new double[miu * lamda];
27         fitness_c = nilai_fitness1(miu * lamda,
28 jumlah_bahan, kromosom_m, total_gen);
29         //update nilai sigma
30         double kenaikan_sigma = Math.round((20.0 / 100.0) *
31 lamda);
32         int jumlah_offspring = 0;
33         int jumlah_offspring1 = 0;
34         for (int i = 0; i < miu; i++) {
35             int fitness_baik = 0;
36             //perulangan untuk menghitung jumlah fitness
37             yang lebih baik dari
38             for (int j = 0; j < lamda; j++) {
39                 if (fitness_p[i] <
40 fitness_c[jumlah_offspring]) {
41                     fitness_baik++;

```

```

35     }
36     jumlah_offspring++;
37 }
38 for (int j = 0; j < lamda; j++) {
39     for (int k = 0; k < jumlah_bahan; k++) {
40         if (fitness_baik >= kenaikan_sigma) {
41             sigma_m[jumlah_offspring1][k] =
42                 sigma[i][k] * 1.1;
43         } else {
44             sigma_m[jumlah_offspring1][k] =
45                 sigma[i][k] * 0.9;
46         }
47     }
48     jumlah_offspring1++;
49 }

```

Source Code 5.2 Implementasi Proses Mutasi

Keterangan *Source Code 5.2* :

1. Baris 2-21 digunakan untuk proses mutasi
2. Baris 22-24 digunakan untuk menghitung nilai *fitness*
3. Baris 25-49 digunakan untuk update nilai sigma.

5.2.3 Implementasi Proses Perhitungan *fitness*

Pada proses perhitungan nilai *fitness* yang pertama dilakukan adalah menghitung total harga setiap bahan kemudian menghitung nutrisi yang tersedia dari bahan. Setelah selesai menghitung total harga dan nutrisi yang dilakukan adalah menghitung nilai *penalty* dan terakhir menghitung nilai *fitness*. Implementasi proses perhitungan *fitness* ditunjukkan pada *Source Code 5.3*

```

public double[] nilai_fitness1(int jumlah_individu, int
1 jumlah_bahan, double[][] kromosom2, double[] total_gen) {
2     for (int i = 0; i < jumlah_individu; i++) {
3         for (int j = 0; j < jumlah_bahan; j++) {
4             bobot_bahan[i][j] = (kromosom2[i][j] /
5 total_gen[i]) * jumlah_pakan;
6             harga_bahan1[i][j] = bobot_bahan[i][j] *
7 harga[j];
8             total_harga[i] = total_harga[i] +
9 Math.round(harga_bahan1[i][j]);
10        }
11    }
12    for (int i = 0; i < jumlah_individu; i++) {
13        for (int j = 0; j < jumlah_bahan; j++) {
14            bobot_bahan[i][j] = (kromosom2[i][j] /
15 total_gen[i]) * jumlah_pakan;
16            nutrisi_p[i][j] = bobot_bahan[i][j] *

```



```

        (protein[j] / 100); //nutrisi_p adalah nutrisi protein
                nutrisi_l[i][j] = bobot_bahan[i][j] *
13 (lemak[j] / 100); //nutrisi_l adalah nutrisi lemak
                nutrisi_s[i][j] = bobot_bahan[i][j] *
14 (serat[j] / 100); //nutrisi_s adalah nutrisi seral
15         total_p[i] = total_p[i] + nutrisi_p[i][j];
16         total_l[i] = total_l[i] + nutrisi_l[i][j];
17         total_s[i] = total_s[i] + nutrisi_s[i][j];
18     }
19 }
    double[] kebutuhan_nutrisi_p = new
20 double[jumlah_individu];
    double[] kebutuhan_nutrisi_l = new
21 double[jumlah_individu];
    double[] kebutuhan_nutrisi_s = new
22 double[jumlah_individu];
23     for (int i = 0; i < jumlah_individu; i++) {
        kebutuhan_nutrisi_p[i] = ((kebutuhan_p / 100) *
24 jumlah_pakan);
        kebutuhan_nutrisi_l[i] = ((kebutuhan_l / 100) *
25 jumlah_pakan);
        kebutuhan_nutrisi_s[i] = ((kebutuhan_s / 100) *
26 jumlah_pakan);
27     }
28     double[] penalty_p = new double[jumlah_individu];
29     double[] penalty_l = new double[jumlah_individu];
30     double[] penalty_s = new double[jumlah_individu];
    double[] total_penalty = new
31 double[jumlah_individu];
32     for (int i = 0; i < jumlah_individu; i++) {
        penalty_p[i] = Math.abs(kebutuhan_nutrisi_p[i]
33 - total_p[i]);
        penalty_l[i] = Math.abs(kebutuhan_nutrisi_l[i]
34 - total_l[i]);
        penalty_s[i] = Math.abs(kebutuhan_nutrisi_s[i]
35 - total_s[i]);
        total_penalty[i] = penalty_p[i] + penalty_l[i]
36 + penalty_s[i];
37     }
    double[] nilai_fitness = new
38 double[jumlah_individu];
39     for (int i = 0; i < jumlah_individu; i++) {
        nilai_fitness[i] = (10000 / (total_harga[i] +
40 (total_penalty[i] * 10000)));
41     }
42     return nilai_fitness;
43 }

```

Source Code 5.3 Implementasi Proses Perhitungan *fitness*

Keterangan *Source Code* 5.3 :

1. Baris 2-8 digunakan untuk menghitung total harga

2. Baris 9-19 digunakan untuk menghitung nutrisi bahan pakan
3. Baris 20-27 digunakan untuk menghitung nutrisi yang dibutuhkan
4. Baris 28-38 digunakan untuk menghitung nilai *penalty*
5. Baris 39-43 digunakan untuk menghitung *fitness*

5.2.4 Implementasi Proses Seleksi

Proses seleksi adalah proses terakhir pada algoritma *Evolution Strategies*. Proses dilakukan dengan cara menggabungkan nilai *fitness* dari *parent* dan *offspring*, kemudian diurutkan. Setelah selesai proses pengurutan yang dilakukan adalah mengambil individu yang memiliki *fitness* tertinggi sebanyak jumlah *miu*. Implementasi proses seleksi ditunjukkan pada *Source Code 5.4*.

```

1     public void ProsesSeleksi() {
2         for (int i = 0; i < miu; i++) {
3             for (int j = 0; j < jumlah_bahan; j++) {
4                 kromosom_t[no][j] = kromosom[i][j];
5                 sigma_t[no][j] = sigma[i][j];
6             }
7             fitness_t[no] = fitness_p[i];
8             individu_t[no] = individu_p[i];
9             no++;
10        }
11        for (int i = 0; i < miu * lamda; i++) {
12            for (int j = 0; j < jumlah_bahan; j++) {
13                kromosom_t[no][j] = kromosom_m[i][j];
14                sigma_t[no][j] = sigma_m[i][j];
15            }
16            fitness_t[no] = fitness_c[i];
17            individu_t[no] = individu_c[i];
18            no++;
19        }
20        int[] Indeks_gabung = new int[fitness_t.length];
21        for (int i = 0; i < fitness_t.length; i++) {
22            Indeks_gabung[i] = i;
23        }
24        for (int i = 0; i < fitness_t.length; i++) {
25            double fitness_temp = fitness_t[i];
26            int Indeks_gabung_Temp = Indeks_gabung[i];
27            for (int j = (i + 1); j < fitness_t.length;
28 j++) {
29                if (fitness_t[j] > fitness_temp) {
30                    fitness_temp = fitness_t[j];
31                    fitness_t[j] = fitness_t[i];

```


5.3 Implementasi Antarmuka

Implementasi antarmuka terdiri dari dua yaitu implementasi antarmuka untuk menerima input data ikan dan bahan serta implementasi Proses perhitungan algoritma *evolution Strategies*.

5.3.1 Tampilan Halaman Input Data Ikan dan Bahan

Halaman ini merupakan halaman awal ketika sistem dijalankan. Pada halaman ini user diminta untuk memasukan data ikan dan bahan pakan ikan. Tampilan Halaman Input Data Ikan dan Bahan ditunjukkan pada Gambar 5.1

Optimasi Komposisi Pakan Sistem Polikultur Ikan dan Udang By Nenovli Widerti

OPTIMASI KOMPOSISI PAKAN SISTEM POLIKULTUR IKAN DAN UDANG
MENGUNAKAN ALGORITMA EVOLUTION STRATEGIES

INPUT DATA IKAN

Jenis Ikan: Ikan Bandeng (1) (5)
Umur Ikan: 22 (2)
Berat Ikan: 0.6 (3)
Jumlah Ikan: 100 (4) Tampil

Bahan Pakan

Cental/Sorgum (8) Tambah (9)

No	Nama
1	Tepung Ikan (10)
2	Dedak Gandum
3	Cental/Sorgum

Hapus Bahan yang Dipilih (11)

Rekomendasi Padat Tebar Udang dan Nutrisi yang dibutuhkan

Padat Tebar Udang : 80.0 (6)

Jumlah Pakan yang dibutuhkan : 1.8 Kg (7)
Nutrisi yang dibutuhkan Tepung Ikan :
Protein 20.0%
Lemak 8.0%
Serat 4.0%

Nutrisi Dari Bahan Pakan dan Harga

Tampil (12)

Nama Bahan	Protein	Lemak	Serat	Harga
Tepung Ikan	60.5	9.4	0.7	16000 (13)
Dedak Gandum	11.99	1.48	3.75	5000
Cental/Sorgum	13.0	2.05	13.5	7000

Proses Algoritma Evolution Strategies (14)

Created By Nenovli Widerti (125150201111007)

Gambar 5.1 Tampilan Halaman Input Data Ikan

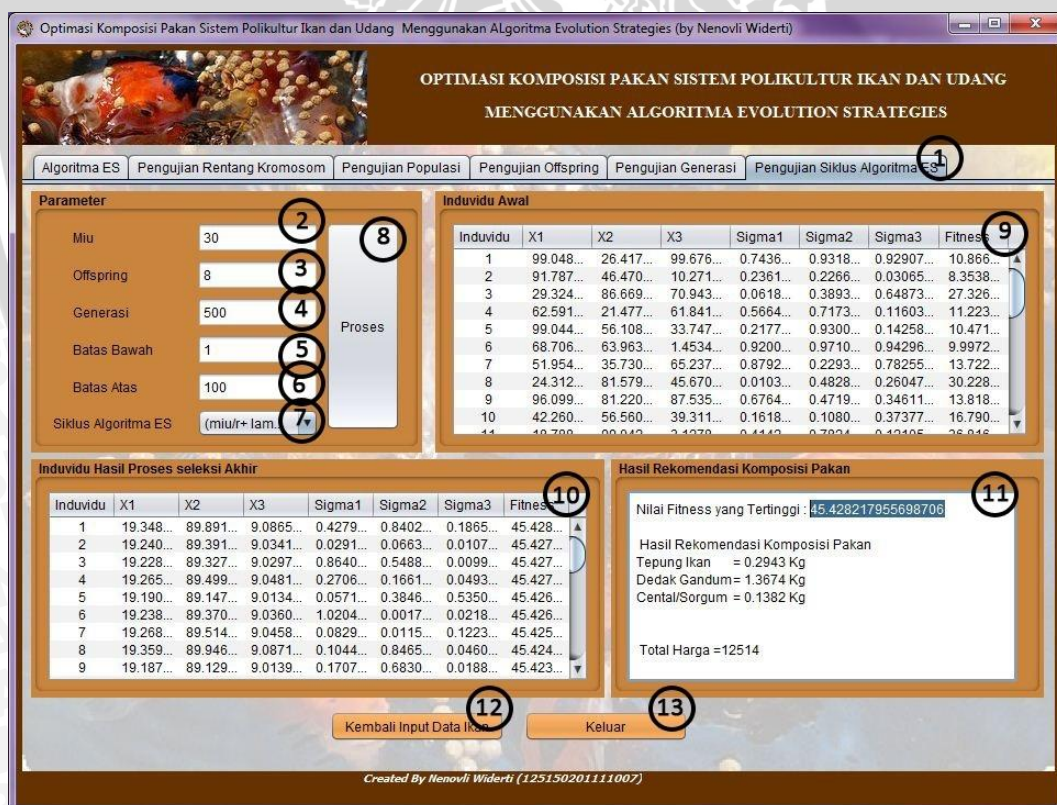
Keterangan Gambar 5.1:

1. *Combo Box* digunakan untuk memilih jenis ikan.
2. *Text Filed* digunakan untuk menginputkan umur ikan
3. *Text Fild* digunakan untuk menginputkan berat ikan
4. *Text Fild* digunakan untuk menginputkan jumlah ikan
5. Tombol untuk menampilkan rekomendasi padat tebar udang dan nutrisi yang dibutuhkan ikan
6. *Text Fild* untuk menampilkan hasil perhitungan rekomendasi padat tebar udang

7. *Text Area* untuk menampilkan nutrisi yang dibutuhkan oleh ikan
8. *Combo box* digunakan untuk memilih bahan pakan ikan
9. Tombol untuk menambah bahan pakan yang dipilih
10. Tabel digunakan untuk menampung ahan pakan yang dipilih
11. Tomol yang digunakan untuk menghapus bahan pakan yang dipilih
12. Tombol yang digunakan untuk manmpilkan bahan pakan yang dipilih beserta nutrisi dan harganya
13. Tabel untuk menampilkan nutrisi ahan pakan yang dipilih beserta harganya
14. Tombol yang digunakan untuk masuk kehalaman algoritma *evolution strategies*.

5.3.2 Tampilan Halaman Proses Algoritma

Halaman ini merupakan halaman proses perhitungan algoritma *evolution strategies*. pada halaman ini user akan mengimputkan parameter parameter yang dibutuhkan oleh algoritma *evolution strategies* yaitu nilai *miu*, *offspring*, generasi, batas bawah dan batas atas. Tampilan alaman Proses Algoritma ditunjukkan pada Gambar 5.2



Keterangan Gambar 5.2:

1. Tab pengujian digunakan untuk melakukan pengujian

2. *Text Filed* digunakan untuk menginputkan nilai miu
3. *Text Fild* digunakan untuk menginputkan nilai *offspring*
4. *Text Fild* digunakan untuk menginputkan jumlah generasi
5. *Text Field* digunakan untuk menginputkan bawah bawah kromosom
6. *Text Field* digunakan untuk menginputkan atas bawah kromosom
7. *Combo box* digunakan untuk memilih siklus ES
8. Tombol untuk melakukan proses algoritma *evolution strategies*.
9. Tabel untuk menampilkan individu awal
10. Tabel untuk menampilkan individu hasil proses akhir
11. Text Area untuk menampilkan nilai *fitness* tertinggi, komposisi pakan yang direkomendasikan dan total harga.
12. Tombol untuk kembali ke halaman input data ikan
13. Tombol untuk keluar



BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini akan dibahas hasil pengujian sistem optimasi komposisi pakan sistem polikultur ikan dan udang yang menggunakan algoritma *evolution strategies* telah dilakukan dan kemudian hasil uji coba akan dianalisis.

6.1 Sistematika Pengujian

Pada bab perancangan sebelumnya telah dibahas bahwa pada penelitian ini terdapat 6 jenis pengujian. Setiap pengujian akan dilihat pengaruhnya dengan melihat nilai *fitness*. Pengujian yang dilakukan adalah, pengujian rentang kromosom, pengujian banyak populasi, pengujian banyak *offspring*, pengujian banyak generasi dan pengujian siklus algoritma *evolution strategies*. Pada pengujian ke 1 bertujuan untuk mengetahui pengaruh rentang kromosom. Pengujian ke 2 bertujuan untuk mengetahui pengaruh ukuran populasi. Pengujian ke 3 untuk mengetahui pengaruh ukuran *offspring*. Pengujian ke 4 untuk mengetahui pengaruh generasi. Pengujian ke 5 untuk mengetahui pengaruh siklus yang digunakan dalam algoritma *evolution strategies*. Setelah dilakukan beberapa pengujian yang dilakukan adalah menghitung rata-rata dari *fitness*. Pengujian ke 6 bertujuan untuk mengetahui pengaruh persamaan *fitness* yang digunakan terhadap hasil komposisi yang didapatkan.

6.2 Analisis dan Hasil Pembahasan

Berikut ini akan ditampilkan hasil pengujian berdasarkan pengujian yang telah dilakukan yang sesuai dengan perancangan uji coba.

6.2.1 Hasil Uji Coba Rentang Kromosom

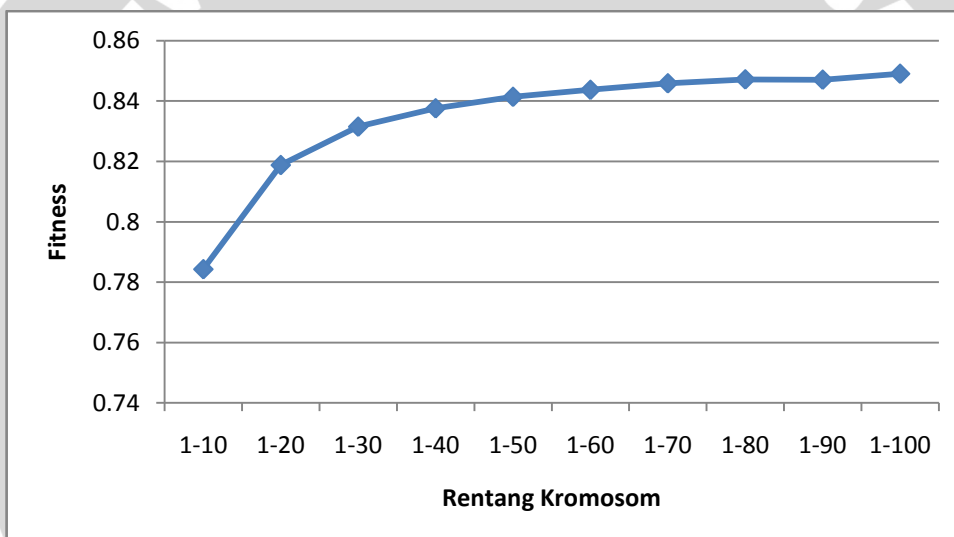
Uji coba rentang kromosom dilakukan terhadap sistem untuk mengetahui rentang kromosom yang akan menghasilkan solusi paling optimal untuk permasalahan komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *evolution strategies*. Uji coba dilakukan pada jumlah ikan bandeng dengan usia 22 minggu dan berat 0.6 Kg dengan jumlah ikan yang digunakan adalah 100. Uji coba dilakukan 10 kali menggunakan rentang kromosom berturut-turut adalah 1-10, 1-20, 1-30, 1-40, 1-50, 1-60, 1-70, 1-80, 1-90, 1-100, ukuran populasi 50, *offspring* 4 μ , generasi 500 dan menggunakan siklus ES ($\mu+\lambda$). Hasil uji coba rentang kromosom ditunjukkan pada Tabel 6.1

Tabel 6.1 Hasil Uji Coba Rentang Kromosom

Rentang kromosom	Nilai <i>fitness</i>						Rata-Rata <i>fitness</i>
	Percobaan ke- <i>i</i>						
	1	2	3	4	10	
1-10	0.7842999	0.7842901	0.7842997	0.7842998	0.784300348	0.784298078
1-20	0.8187622	0.8187550	0.8187646	0.8187547	0.818754927	0.818758472

1-30	0.8314753	0.8314506	0.8314593	0.8314582	0.831459223	0.83146209
1-40	0.8375611	0.8375611	0.8375612	0.8373162	0.837560948	0.837536701
1-50	0.8413451	0.8413362	0.8413441	0.8413490	0.841348295	0.84134326
1-60	0.8439526	0.8439517	0.8439500	0.8439514	0.843949975	0.843654984
1-70	0.8457990	0.8458142	0.8458121	0.8458147	0.845815151	0.845808628
1-80	0.8472478	0.8459606	0.8472350	0.8472550	0.847254178	0.847115498
1-90	0.8449892	0.8422396	0.8483743	0.8483755	0.845493756	0.847013998
1-100	0.8492748	0.8492798	0.8462788	0.8492795	0.84927925	0.84897838

Dari data hasil uji coba rentang kromosom pada Tabel 6.1 dapat dibuat grafik untuk melihat pengaruh rentang kromosom terhadap nilai *fitness* yang didapatkan. Grafik hasil uji coba rentang kromosom ditunjukkan pada Gambar 6.1



Gambar 6.1 Hasil Uji Coba Rentang Kromosom

Berdasarkan grafik pada Gambar 6.1 dapat dilihat bahwa rentang kromosom berpengaruh terhadap hasil *fitness* algoritma *evolution strategies*. Nilai *fitness* tertinggi saat rentang kromosom 1-100. Nilai *fitness* terendah terjadi saat rentang kromosom 1-10. Berdasarkan grafik tersebut dapat dilihat bahwa semakin besar rentang kromosom maka nilai *fitness* yang dihasilkan semakin besar, tetapi semakin kecil rentang kromosom menghasilkan nilai *fitness* semakin kecil. Hal itu terjadi karena semakin besar rentang kromosom maka variasi dari individu yang terbentuk lebih beragam sehingga peluang untuk mendapatkan nilai *fitness* yang tertinggi semakin besar.

6.2.2 Hasil Uji Coba Ukuran Populasi

Uji coba ukuran populasi dilakukan terhadap sistem untuk mengetahui ukuran populasi yang akan menghasilkan solusi paling optimal untuk permasalahan komposisi pakan sistem polikultur ikan dan udang menggunakan

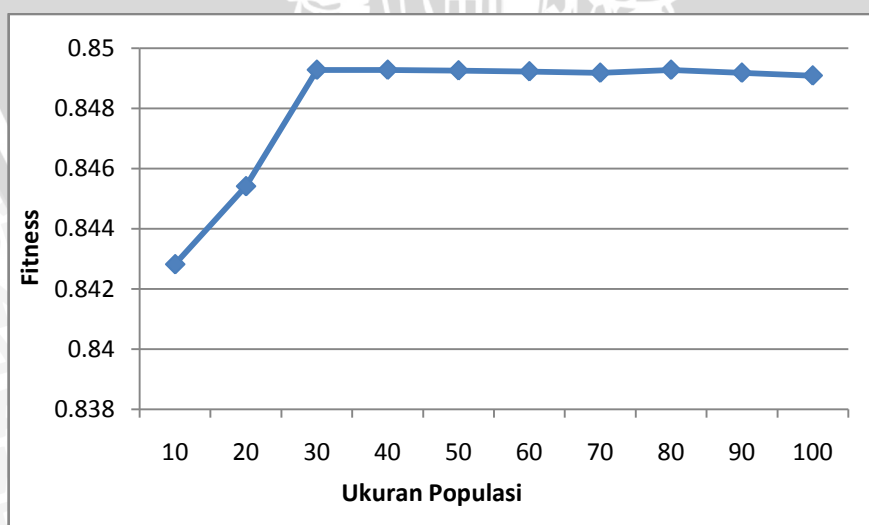


algoritma *evolution strategies*. Uji coba dilakukan pada jumlah ikan bandeng dengan usia 22 minggu dan berat 0.6 Kg dengan jumlah ikan yang digunakan adalah 100. Uji coba dilakukan 10 kali menggunakan rentang kromosom 1-100, ukuran populasi berturut-turut adalah 10, 20, 30, 40, 50, 60, 70, 80, 90 dan 100 *offspring* 4 μ , jumlah generasi 500 dan menggunakan siklus ES ($\mu+\lambda$). Hasil uji coba ukuran populasi ditunjukkan pada Tabel 6.2

Tabel 6.2 Hasil Uji Coba Ukuran Populasi

Ukuran Populasi	Nilai <i>fitness</i>						Rata-Rata <i>fitness</i>
	Percobaan ke- <i>i</i>						
	1	2	3	4	10	
10	0.84928016	0.8492791	0.84927862	0.84927491	0.849274904	0.842824092
20	0.84927858	0.8323108	0.84927494	0.84928021	0.849276949	0.845413131
30	0.84928073	0.8492749	0.84927628	0.84927874	0.849277453	0.84927749
40	0.84927814	0.8492768	0.84927875	0.84927858	0.849279328	0.849278289
50	0.8492792	0.8492769	0.84927959	0.84927844	0.849280399	0.849257126
60	0.84927978	0.8487619	0.84928007	0.84928003	0.849277942	0.849226683
70	0.84927583	0.8492776	0.84830361	0.84927461	0.849277298	0.849180059
80	0.8492799	0.8492751	0.8492767	0.84927935	0.849279237	0.84927848
90	0.84927936	0.849276	0.84927968	0.84927565	0.849280117	0.849180807
100	0.84927942	0.8492793	0.84928026	0.84927808	0.849279238	0.849087981

Dari data hasil uji coba ukuran populasi pada Tabel 6.2 dapat dibuat grafik untuk melihat pengaruh ukuran populasi terhadap nilai *fitness* yang didapatkan. Grafik hasil uji coba ukuran populasi ditunjukkan pada Gambar 6.2



Gambar 6.2 Grafik Hasil Uji Coba Ukuran Populasi

Berdasarkan grafik pada Gambar 6.2 nilai *fitness* tertinggi saat jumlah populasi 30 dan nilai *fitness* terendah saat populasi 10. Hal itu menunjukkan bahwa semakin besar jumlah populasi maka nilai *fitness* semakin tinggi. Nilai *fitness* semakin besar karena jumlah dan keberagaman individu yang dihasilkan lebih banyak sehingga peluang untuk menghasilkan nilai *fitness* yang besar semakin besar. Grafik tersebut juga menunjukkan bahwa pada saat populasi 30 sampai dengan populasi 100 terjadi konvergensi. Konvergensi dapat terjadi karena nilai kromosom dari individu *offspring* yang dihasilkan dan individu *parent* mirip. Jumlah populasi juga mempengaruhi waktu eksekusi program, semakin besar jumlah populasi maka semakin lama eksekusi program.

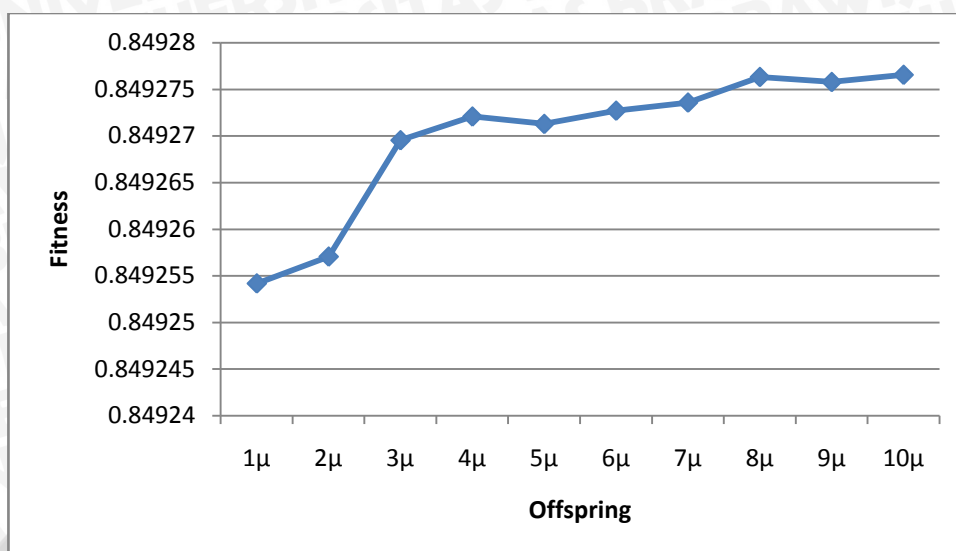
6.2.3 Hasil Uji Coba Ukuran *Offspring*

Uji coba ukuran *offspring* dilakukan terhadap sistem untuk mengetahui ukuran *offspring* yang akan menghasilkan solusi paling optimal untuk permasalahan komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *evolution strategies*. Uji coba dilakukan pada jumlah ikan bandeng dengan usia 22 minggu dan berat 0.6 Kg dengan jumlah ikan yang digunakan adalah 100. Uji coba dilakukan 10 kali menggunakan rentang kromosom 1-100, populasi 50, *offspring* yang digunakan berturut-turut 1μ , 2μ , 3μ , 4μ , 5μ , 6μ , 7μ , 8μ , 9μ , 10μ , jumlah generasi 500 dan menggunakan siklus ES ($\mu+\lambda$). Hasil uji coba ukuran *offspring* ditunjukkan pada Tabel 6.3

Tabel 6.3 Hasil Uji Coba Ukuran *offspring*

Ukuran <i>offspring</i>	Nilai <i>fitness</i>						Rata-Rata <i>fitness</i>
	Percobaan ke- <i>i</i>						
	1	2	3	4	10	
1μ	0.84926163	0.8492621	0.84926372	0.84924607	0.849247382	0.849254188
2μ	0.84924633	0.8492637	0.84927433	0.84924868	0.849260722	0.849257065
3μ	0.84927441	0.8492748	0.84927411	0.8492751	0.849250272	0.849269535
4μ	0.84927417	0.8492744	0.84927421	0.84926347	0.849274201	0.849272075
5μ	0.84926376	0.8492747	0.84927422	0.84927485	0.849274668	0.849271284
6μ	0.84927489	0.849275	0.84927477	0.84927506	0.849263864	0.849272702
7μ	0.84927492	0.8492749	0.84927498	0.84927503	0.849274405	0.849273571
8μ	0.84927496	0.8492745	0.84927802	0.84927723	0.849274833	0.849276315
9μ	0.84927494	0.8492751	0.84927494	0.84927919	0.84927733	0.849275777
10μ	0.84928031	0.8492783	0.84927475	0.849275	0.849274701	0.849276548

Dari data hasil uji coba ukuran *offspring* pada Tabel 6.3 dapat dibuat grafik untuk melihat pengaruh ukuran *offspring* terhadap nilai *fitness* yang didapatkan. Grafik hasil uji coba ukuran *offspring* ditunjukkan pada Gambar 6.3



Gambar 6.3 Grafik Hasil Uji Coba Ukuran Offspring

Berdasarkan grafik pada Gambar 6.3 menunjukkan bahwa nilai *fitness* terbesar saat ukuran *offspring* 8μ. Nilai *fitness* terendah terjadi saat ukuran *offspring* 1μ. Hal itu menunjukkan bahwa semakin tinggi ukuran *offspring* maka cenderung semakin tinggi nilai *fitness* yang dihasilkan. Semakin tinggi nilai *fitness* karena jumlah individu yang dihasilkan saat proses mutasi semakin banyak dan memungkinkan keberagaman individu sangat tinggi. Tetapi, tidak semua ukuran *offspring* yang tinggi menghasilkan nilai *fitness* yang tinggi seperti pada ukuran *offspring* 5μ lebih rendah dari pada ukuran *offspring* 4μ dan ukuran *offspring* 9μ lebih rendah dari pada ukuran *offspring* 10μ. Hal itu terjadi karena dalam pembentukan individu *offspring* dihasilkan dipengaruhi oleh individu awal dan nilai sigma. Nilai individu awal dalam proses algoritma *evolution strategies* dibangkitkan secara acak dan nilai sigma dibangkitkan secara acak dengan rentang yang sempit yaitu [0,1]. Semakin besar ukuran *offspring* juga dapat membuat waktu eksekusi program semakin lama.

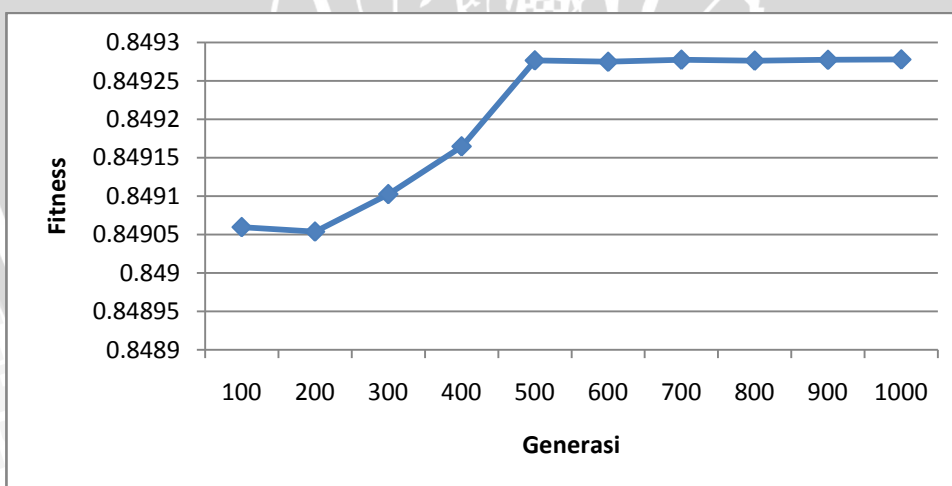
6.2.4 Hasil Uji Coba Jumlah Generasi

Uji coba jumlah generasi dilakukan terhadap sistem untuk mengetahui jumlah generasi yang akan menghasilkan solusi paling optimal untuk permasalahan komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *evolution strategies*. Uji coba dilakukan pada jumlah ikan bandeng dengan usia 22 minggu dan berat 0.6 Kg dengan jumlah ikan yang digunakan adalah 100. Uji coba dilakukan 10 kali menggunakan rentang kromosom 1-100, ukuran populasi 50, *offspring* 5μ dan generasi yang digunakan berturut-turut yang digunakan 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000 dan menggunakan siklus ES ($\mu+\lambda$). Hasil uji coba jumlah generasi ditunjukkan pada Tabel 6.4

Tabel 6.4 Hasil Uji Coba Jumlah Generasi

Jumlah generasi	Nilai <i>fitness</i>						Rata-Rata <i>fitness</i>
	Percobaan ke- <i>i</i>						
	1	2	3	4	10	
100	0.84717058	0.8492742	0.849275	0.84927437	0.849251469	0.849059354
200	0.84861726	0.8492751	0.84927457	0.84870093	0.849275037	0.849053513
300	0.84754331	0.8492751	0.8492751	0.84927512	0.849274895	0.849102319
400	0.8492793	0.849275	0.84928089	0.84927549	0.84926367	0.849164542
500	0.84927794	0.849279	0.8492743	0.84927497	0.849279401	0.8492765
600	0.84927511	0.8492749	0.84927505	0.84927543	0.849266356	0.849274675
700	0.84927582	0.8492796	0.84928005	0.84927944	0.849274994	0.849277372
800	0.84927485	0.8492751	0.84927505	0.84927912	0.849274826	0.849276151
900	0.84927511	0.8492782	0.84927705	0.84927806	0.849274915	0.849277196
1000	0.84927984	0.849275	0.84928058	0.84927757	0.849276352	0.849277724

Dari data hasil uji coba jumlah generasi pada Tabel 6.4 dapat dibuat grafik untuk melihat pengaruh jumlah generasi terhadap nilai *fitness* yang didapatkan. Grafik hasil uji coba jumlah generasi ditunjukkan pada Gambar 6.4.



Gambar 6.4 Hasil Uji Coba Jumlah Generasi

Berdasarkan grafik pada Gambar 6.4 dapat dilihat bahwa jumlah generasi mempengaruhi hasil *fitness* yang diperoleh. Nilai *fitness* tertinggi terjadi saat jumlah generasi 500, dan nilai *fitness* terendah saat generasi ke 200. Grafik pada Gambar 6.4 cenderung menunjukkan peningkatan nilai *fitness*. Peningkatan nilai *fitness* terjadi karena semakin banyak jumlah generasi maka proses mutasi akan sering dilakukan sehingga dalam proses seleksi dapat mendapatkan individu yang memiliki nilai *fitness* yang terbaik. Hal itu terjadi karena individu-individu yang

terbentuk dari proses mutasi lebih bervariasi sehingga terdapat peluang untuk menghasilkan individu-individu baru yang memiliki *fitness* yang terbaik. Akan tetapi tidak semua jumlah generasi yang banyak dapat menghasilkan nilai *fitness* yang tinggi seperti saat jumlah generasi 200 nilai *fitness*nya lebih rendah dari pada nilai *fitness* dengan jumlah generasi 100. Hal tersebut terjadi karena konsep dari algoritma *evolution strategies* dalam pembentukan individu awal dan nilai sigma dibentuk secara acak.

Grafik tersebut juga menunjukkan bahwa pada saat generasi 500 sampai dengan generasi 1000 terjadi konvergensi. Konvergensi dapat terjadi karena nilai kromosom dari individu *offspring* yang dihasilkan dan individu *parent* mirip. Jumlah generasi juga mempengaruhi waktu eksekusi program. Semakin banyak jumlah generasi maka waktu eksekusi program semakin lama.

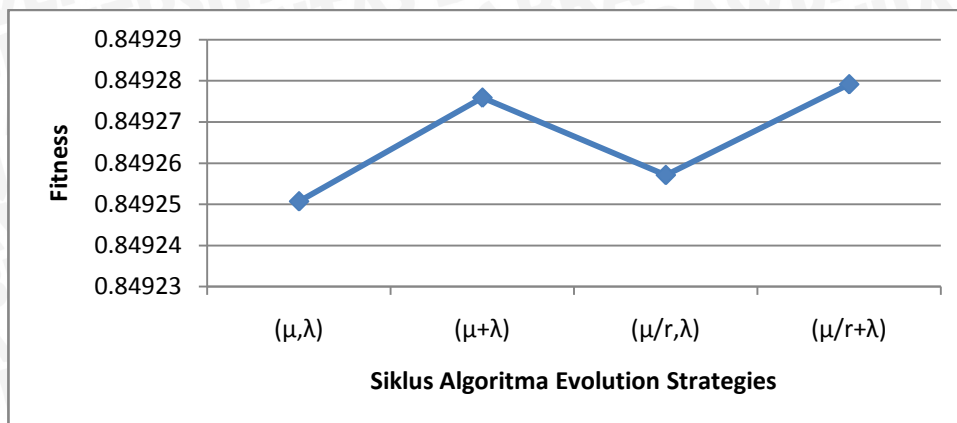
6.2.5 Hasil Uji Coba Siklus Algoritma Evolution Strategies

Uji coba siklus algoritma *evolution strategies* dilakukan terhadap sistem untuk mengetahui tahapan siklus yang akan menghasilkan solusi paling optimal untuk permasalahan komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *evolution strategies*. Uji coba dilakukan pada jumlah ikan bandeng dengan usia 22 minggu dan berat 0.6 Kg dengan jumlah ikan yang digunakan adalah 100. Uji coba dilakukan 10 kali menggunakan rentang kromosom 1-100, ukuran populasi 30, *offspring* 8 μ , dan jumlah generasi 500. Hasil uji coba jumlah tahapan siklus algoritma *evolution strategies* ditunjukkan pada Tabel 6.5

Tabel 6.5 Hasil Uji Coba Tipe Algoritma Evolution Strategies

Jumlah generasi	Nilai <i>fitness</i>						Rata-Rata <i>fitness</i>
	Percobaan ke- <i>i</i>						
	1	2	3	4	10	
(μ, λ)	0.84923292	0.8492329	0.84923292	0.849268594	0.849280268	0.849250693
($\mu + \lambda$)	0.84927672	0.8492749	0.84927495	0.84927854	0.849274732	0.8492759
($\mu/r, \lambda$)	0.84926172	0.8492616	0.84924934	0.84924764	0.849260373	0.849257088
($\mu/r + \lambda$)	0.8492797	0.8492792	0.84927977	0.84927953	0.849278387	0.849279153

Dari data hasil uji coba siklus algoritma *evolution strategies* pada Tabel 6.5 dapat dibuat grafik untuk melihat pengaruh siklus algoritma *evolution strategies* terhadap nilai *fitness* yang didapatkan. Grafik hasil uji coba siklus algoritma *evolution strategies* ditunjukkan pada Gambar 6.5



Gambar 6.5 Hasil Uji Coba Tipe Algoritma Evolution Strategies

Berdasarkan grafik pada Gambar 6.5 menunjukkan bahwa siklus dari algoritma *evolution strategies* mempengaruhi jumlah *fitness* yang dihasilkan. *Fitness* terbaik terjadi saat menggunakan siklus ES $(\mu/r+\lambda)$ dan nilai *fitness* yang terendah saat siklus ES (μ,λ) . Hal itu terjadi karena dalam proses reproduksi tipe $(\mu/r+\lambda)$ melibatkan proses rekombinasi dan mutasi sedangkan Siklus ES (μ,λ) hanya melibatkan proses mutasi. Proses rekombinasi akan menghasilkan individu *offspring* dari kombinasi individu *parent*. Proses mutasi akan menghasilkan individu *offspring* dari perubahan kromosom.

Siklus ES $(\mu/r+\lambda)$ menghasilkan nilai *fitness* terbaik karena siklus ES $(\mu/r+\lambda)$ juga melibatkan individu awal dan individu *offspring* dalam proses seleksi sedangkan tipe (μ,λ) hanya melibatkan individu *offspring*. Pada siklus ES $(\mu/r,\lambda)$ lebih rendah diandingkan siklus ES $(\mu + \lambda)$ dan $(\mu/r+\lambda)$ karena dalam proses seleksi hanya melibatkan individu *offspring*.

6.2.6 Hasil Uji Coba Persamaan Fitness

Uji coba persamaan fitness dilakukan terhadap sistem untuk mengetahui persamaan fitness yang akan menghasilkan komposisi bahan pakan yang layak digunakan. Uji coba dilakukan pada jumlah ikan bandeng dengan usia 22 minggu dan berat 0.6 Kg dengan jumlah ikan yang digunakan adalah 100. Uji coba persamaan fitness dilakukan sebanyak 10 kali menggunakan rentang kromosom 1-100, ukuran populasi 30, *offspring* 8μ , dan jumlah generasi 500 dan siklus yang digunakan $(\mu/r+\lambda)$. Hasil uji coba persamaan fitness terhadap kombinasi bahan yang digunakan ditunjukkan pada Tabel 6.6

Tabel 6.6 Uji Coba Persamaan Fitness

Persamaan	Nilai <i>fitness</i>						Rata-Rata <i>fitness</i>
	Percobaan ke- <i>i</i>						
	1	2	3	4	...	10	
A	1.083513	1.083513	1.083513	1.083513	1.083513	1.08351324
B	0.862464	0.862464	0.862463	0.862463	0.862464	0.86246375

C	1.394700	1.394699	1.394699	1.394699	1.394699	1.39469893
D	14.085039	14.085106	14.085086	14.085147	14.085311	14.0852196
E	45.428098	45.428730	45.427315	45.428927	45.428218	45.4275679

Dari data hasil uji persamaan fitness pada Tabel 6.6 diperoleh rekomendasi komposisi bahan pakan sistem polikultur ikan dan udang. Rekomendasi komposisi bahan pakan ditunjukkan pada Tabel 6.7

Tabel 6.7 Rekomendasi Komposisi Bahan Pakan

Persamaan	Bahan Pakan			Harga
	Tepung Ikan	Dedak Gandum	Tepung Kedelai	
A	0.0177	1.7643	0.0181	9230
B	0.0177	1.7643	0.0181	9230
C	0.0177	1.7643	0.0181	9230
D	0.2943	1.3674	0.1383	12514
E	0.2943	1.3674	0.1382	12514

Tabel 6.7 menunjukkan rekomendasi komposisi bahan pakan yang diperoleh dari perhitungan setiap persamaan fitness oleh sistem. Setelah mendapatkan rekomendasi komposisi bahan selanjutnya adalah menghitung nutrisi dari komposisi bahan yang direkomendasikan oleh sistem. Perhitungan nutrisi dilakukan dengan cara metode yang digunakan pakar. Nutrisi yang tersedia dari bahan tersebut dibandingkan dengan nutrisi yang di butuhkan oleh ikan. Tabel perbandingan nutrisi yang tersedia dengan nutrisi yang diutuhkan ditunjukkan pada Tabel 6.8

Tabel 6.8 Perbandingan Nutrisi yang Tersedia dan Dibutuhkan

Persamaan	Kandungan Nutrisi Yang Tersedia %			Kandungan Nutrisi Yang Dibutuhkan %			Harga
	Protein	Lemak	Serat	Protein	Lemak	Serat (max)	
A	12.48	1.56	3.82	20	7-8	4	9230
B	12.48	1.56	3.82	20	7-8	4	9230
C	12.48	1.56	3.82	20	7-8	4	9230
D	20	2.82	4	20	7-8	4	12514
E	20	2.82	4	20	7-8	4	12514

Berdasarkan perbandingan nutrisi yang tersedia dan dibutuhkan pada Tabel 6.8 diketahui bahwa rekomendasi komposisi bahan pakan menggunakan persamaan D dan E mendekati layak untuk digunakan. Hal itu ditunjukkan dari perbandingan nutrisi protein yang tersedia yaitu 20 % dan yang di butuhkan adalah 20% sehingga kebutuhan protein mencukupi, kebutuhan serat juga mencukupi karena yang tersedia adalah 4% dan yang dibutuhkan maksimal 4%. Tetapi, untuk nutrisi lemak memiliki perbedaan yang jauh antara nutrisi yang dibutuhkan dengan nutrisi yang tersedia. Hal ini terjadi karena pemilihan

kombinasi bahan pakan tidak tepat. Pada Persamaan A, B dan C rekomendasi komposisi pakan tidak layak digunakan karena nutrisi yang tersedia yaitu protein dan lemak memiliki perbedaan yang jauh dari nutrisi yang dibutuhkan walaupun nutrisi serat yang tersedia mencukupi kebutuhan. Dari analisis kelayakan rekomendasi komposisi pakan persamaan fitness D dan persamaan fitness E yang mampu memberikan solusi terbaik.

Dari nilai *fitness* yang dihasilkan pada Tabel 6.6 persamaan E dapat menghasilkan nilai *fitness* yang tertinggi. Nilai *fitness* digunakan sebagai nilai untuk membandingkan solusi yang terbaik antara individu-individu yang ada. Jika dilihat dari kualitas solusi dan nilai *fitness* maka persamaan E lebih tepat untuk digunakan pada permasalahan yang diangkat dalam penelitian ini. Hal itu terjadi karena pada persamaan E terdapat bobot yang membedakan tingkat prioritas dari nilai yang akan diminimalkan.

Untuk membuktikan bahwa persamaan E adalah persamaan yang terbaik untuk permasalahan dalam penelitian ini maka persamaan *fitness* E akan digunakan untuk merekomendasikan komposisi pakan dari bahan yang telah direkomendasikan oleh pakar. Karena kombinasi bahan juga mempengaruhi hasil *fitness* dan komposisi pakan. Bahan pakan yang direkomendasikan adalah tepung ikan, dedak gandum dan tepung kedelai. Hasil rekomendasi komposisi pakan dari bahan yang direkomendasikan ditunjukkan pada Tabel 6.9

Tabel 6.9 Hasil Rekomendasi dari Bahan yang direkomendasikan

Persamaan	Bahan Pakan			Kandungan Nutrisi %			Harga
	Tepung Ikan	Dedak Gandum	Tepung Kedelai	Protein	Lemak	Serat	
E	0.0129	1.2875	0.4996	23.9	5.7	3.51	9391

Dari hasil rekomendasi komposisi pakan bahwa persamaan *fitness* E dapat dinyatakan layak digunakan karena protein yang tersedia masih berada sekitar nutrisi protein yang dibutuhkan yaitu 20%. Untuk nutrisi lemak tidak terlalu berbeda jauh dari lemak yang dibutuhkan yaitu 7%. Menurut Juhrani (2014) kebutuhan lemak ikan bisa mencapai 4%-18% sehingga lemak yang tersedia masih dalam rentang yang dibutuhkan ikan. Untuk serat yang tersedia mencukupi kandungan nutrisi serat yang di butuhkan yaitu 4 %.

BAB 7 PENUTUP

Pada bab ini akan dibahas hasil pengujian sistem optimasi komposisi pakan sistem polikultur ikan dan udang yang menggunakan algoritma *evolution strategies* telah dilakukan dan kemudian hasil uji coba akan dianalisis.

7.1 Kesimpulan

Berdasarkan hasil penelitian tentang optimasi komposisi pakan sistem polikultur ikan dan udang menggunakan algoritma *evolution strategies* ditarik beberapa kesimpulan yaitu sebagai berikut:

1. Untuk mengimplementasi algoritma *evolution strategies* dalam penentuan komposisi pakan sistem polikultur ikan dan udang yang harus dilakukan adalah menentukan kebutuhan nutrisi dan jumlah pakan dari ikan yang akan dibudidayakan dengan sistem polikultur. Dalam proses algoritma *evolution strategies* untuk mengoptimasi komposisi pakan diawali dengan membangkitkan individu sebanyak jumlah populasi yang diinputkan. Masing-masing individu memiliki nilai gen sebanyak jumlah bahan pakan yang diinputkan. Masing-masing gen merepresentasikan nilai kromosom. Representasi kromosom yang digunakan adalah representasi *real code* yang nilainya dibangkitkan secara *random* pada rentang tertentu. Kemudian akan dilakukan proses mutasi untuk menghasilkan keturunan baru sehingga individu lebih beragam. Setelah proses mutasi selesai dilakukan maka akan dilakukan proses seleksi dengan menggunakan seleksi *elitism* yang akan dipilih individu dengan nilai *fitness* tertinggi sebagai individu yang lolos ke generasi selanjutnya sebanyak jumlah populasi.
2. Solusi yang dihasilkan dari implementasi algoritma *evolution strategies* dalam mengoptimasi komposisi pakan sistem polikultur ikan dan udang dapat diketahui berdasarkan nilai *fitness* yang dihasilkan. Dengan menggunakan parameter hasil pengujian yang diperoleh rata-rata nilai *fitness* yang diperoleh adalah 77.08264553. *Fitness* tersebut diperoleh dengan menggunakan parameter pengujian yaitu rentang kromosom 1-100, populasi 30, *Offspring* 8μ jumlah generasi 500 dan menggunakan siklus ES ($\mu/r+\lambda$) serta menggunakan persamaan *fitness* E. Dari hasil *fitness* tersebut juga diperoleh komposisi terbaik pada ikan bandeng usia 22 minggu dengan berat 0.6 Kg dan jumlah ikan 100 adalah tepung ikan 0.0129 Kg, dedak gandum 1.2875 Kg dan Tepung Kedelai 0.4996 Kg dengan total harga Rp. 9391,00.

7.2 Saran

Pada penelitian yang telah dilakukan dapat di jadikan untuk bahan penelitian atau pengembangan lebih lanjut adalah sebagai berikut :

1. Dapat menambahkan jenis polikultur lain selain ikan bandeng dan udang, ikan nila dan udang.

2. Dapat menambahkan nutrisi yang lain yang dibutuhkan ikan untuk membuat komposisi pakan ikan supaya kebutuhan keutuhan nutrisi lain yang dibutuhkan oleh ikan terpenuhi.
3. Perbaikan solusi bisa dilakukan dengan melakukan hibridisasi *evolution strategies* dengan algoritma atau metode lain untuk merekomendasikan kombinasi bahan pakan yang tepat. Karena komposisi pakan juga di pengaruhi oleh kombinasi bahan pakan. Jika kombinasi bahan pakan tepat maka komposisi pakan yang dihasilkan juga lebih bagus.



DAFTAR PUSTAKA

- Abad, A & Elipe,A. 2014. *Evolution strategies for computing periodic orbits*. Tersedia di : <http://ac.els-cdn.com/S0378475414002018/1-s2.0-S0378475414002018-main.pdf?_tid=9bbb4c9c-df8a-11e5-b85f-00000aab0f02&acdnat=1456822391_4bc476060a7b6264ced1bd95844eb93f> [Diskes 11 Desember 2015]
- Afrianto, E & Liviawaty, E. 1998. *Beberapa Metode Budidaya Ikan*. [e-book]. Tersedia di : <<https://books.google.co.id/books?id=iEHXUEt-5FcC&printsec=frontcover&dq=Beberapa+Metode+Budidaya+Ikan&hl=id&sa=X&ved=0ahUKEwi7ydmHqZ3LAhUNC44KHdJ4AYMQ6AEIGTAA#v=onepage&q=Beberapa%20Metode%20Budidaya%20Ikan&f=false>> [Diakses 29 Februari 2016]
- Afrianto, E & Liviawaty, E. 2005. *Pakan Ikan*. [e-book]. Tersedia di : <<https://books.google.co.id/books?id=hz1-LkilerYC&printsec=frontcover&dq=pakan+ikan&hl=id&sa=X&ved=0ahUKEwiznIXglZ7LAhXBVo4KHZTSD8YQ6AEIGjAA#v=onepage&q=pakan%20ikan&f=false>> [Diakses 29 Februari 2016]
- Atjo, A.S. dan Yusuf, M.,2011. *Tahapan Polikultur Udang Windu*. [Online]. Tersedia di : <<http://pusluh.kkp.go.id/arsip/c/51/>> [Diakses Tanggal 19 oktober 2015]
- Dharmawan, B., 2007. *Usaha Pembuatan Pakan Ikan Konsumsi Sukses Bisnis Pembuatan Pakan Ikan Konsumsi*.Pustaka Baru Press. Yogyakarta
- FAO. *Nile tilapia – Nutritional requirements*. [Online]. Tersedia di : <<http://www.fao.org/fishery/affris/species-profiles/nile-tilapia/nutritional-requirements/en/>> [11 Desember 2015]
- Ghufran, H.M. dan Kordi, K., 2013. *Budidaya Nila Unggul*. Jakarta Selatan: AgroMedia Pustaka.
- Juhrani.2014.*Cara Membuat Pakan Buatan Ikan/Udang*. [Online]. Tersedia di : <<http://setbakorluh.kalselprov.go.id/?p=228>> [diakses tanggal 30 september 2015].
- Junior *et al*, 2012. *Polyculture of Nile tilapia and shrimp at different stocking densities*. *Sociedade Brasileira de Zootecnia*, v.41, n.7, p.1561-1569.
- Kementerian Kelautan dan Perikanan. 2013. *Data Statistik Komsumsi Ikan*. [Online]. Tersedia di : <http://statistik.kkp.go.id/index.php/guest/infografik_detail/angka_konsumsi_ikan> [Diakses februari 2016].
- Kholifah, U., Trisyani, N., Yuniar, I., 2008. *Pengaruh Padat Tebar yang Berbeda terhadap Kelangsungan Hidup dan Pertumbuhan pada Polikultur Udang Windu (Penaeus Monodon Fab) dan Ikan Bandeng (Chanos Chanos) pada*

Hapa di Tambak Brebes - Jawa Tengah. Department of Fisheries, Faculty of Engineering and Marine Science, Hang Tuah University

Mahmudy,W.F.,2013. *Algoritma Evolusi*. Program Teknologi Informasi dan Ilmu Komputer. Universitas Brawijaya. Malang

Milah,H., Mahmudy,W.F.,2015.*Implementasi Algoritma Evolution Strategies Untuk Optimasi Komposisi Pakan Ternak Sapi Potong*. Prog. Teknologi Informasi dan Ilmu Kompute. Universitas Brawijaya. Malang.

Murachman, et al., 2010. *Model polikultur udang windu (penaeus monodon Fab), Ikan Bandeng (Chanos-chanos Forskal) dan Rumput Laut (Gracillaria Sp.) Secara tradisional*. Jurnal Pembangunan dan Alam Lestari. Vol.1, No.1

Murtidjo, B.A., 2002. *Budidaya dan Pembenihan Bandeng*. [e-book]. Tersedia di : <<https://books.google.co.id/books?id=M1cXfkKCSx0C&pg=PA88&dq=pakan+ikan+bandeng&hl=en&sa=X&ved=0CCwQ6AEwAWoVChMI3oCkjsXOyAIVTQOOCh2aiwjg#v=onepage&q=pakan%20ikan%20bandeng&f=false>> [Diakses 19 Oktober 2015]

Purwakusuma, W., 2012. *Kebutuhan Nutrisi Ikan*. [Online]. Tersedia di: <http://o-fish.com/PakanIkan/kebutuhan_nutrisi.php> [Diakses 28 September 2015]

Rifqi,N., Maharani,W., dan Shaufiah. 2011. *Analisis dan Implementasi klasifikasi data mining menggunakan jaringan syarf tiruan dan Evolution Strategies*. Fak. Informatika, IT Telkom Bandung. [online]. Ersedia di:<<https://id.scribd.com/doc/115301485/183-191-Knsi2011-029-Analisis-Dan-implementasi-Klasifikasi-Data-Mining-Menggunakan-Jaringan-Syaraf-Tiruan-Dan-Evolution-Strategies>> [Diakses 30 September 2015].

Schwefel, H-P 1995. *Evolution and Optimum Seeking*. Wiley. New York.

Simão et al.2013. *Stocking densities and feeding strategies in shrimp and tilapia polyculture in tanks*. *Pesq. agropec. bras. vol.48 no.8 Brasília*.

Suyanto, 2008. *Evolutionary Computation Komputasi Berbasis "Evolusi" dan "Genitika"*. Informatika. Bandung.

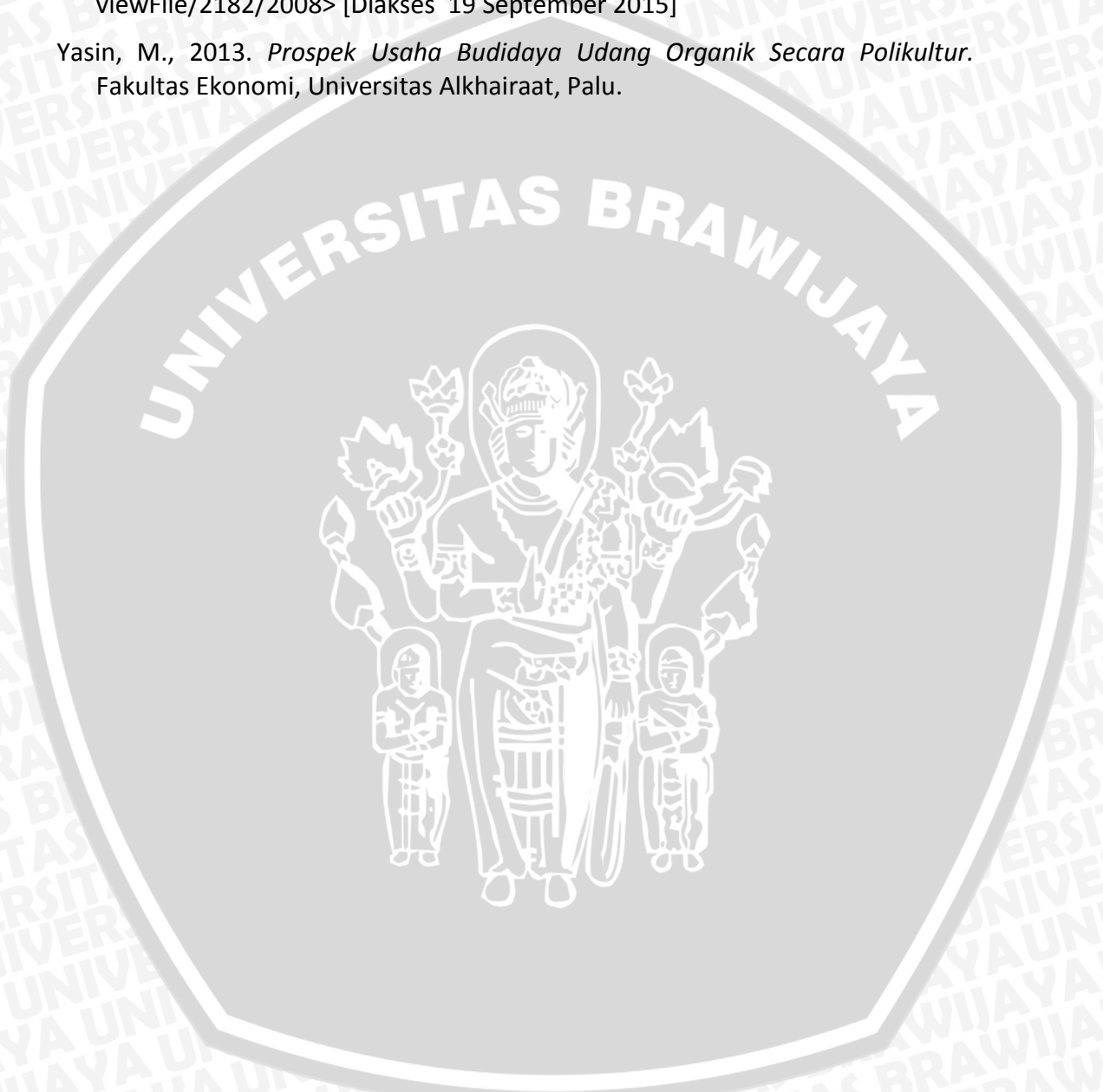
Suyanto, R.S. dan Takarina, E.P., 2009. *Panduan Budidaya Udang Windu*. [e-book]. Tersedia di : <<https://books.google.co.id/books?id=HRRyVAhNwNsC&printsec=frontcover&dq=udang+windu&hl=en&sa=X&ved=0CB4Q6AEwAGoVChMIyBCEs8LcyAVTAqOCh3SaQ96#v=onepage&q=udang%20windu&f=false>> [Diakses 19 Oktober 2015].

Tim Penulis CMK & B. Prasetya W.,2015. *Panduan Praktis Pakan Ikan Konsumsi*. [e-book]. Tersedia di : <https://books.google.co.id/books?id=_UxcCwAAQBAJ&pg=PA1&dq=Pakan+ikan+adalah&hl=id&sa=X&ved=0ahUKEwjI2_bn45vLahWCKl4KHaDDCCKEQ6AEINTAG#v=onepage&q=Pakan%20ikan%20adalah&f=false> [Diakses 29 Februari 2016]

Tristian. 2011, *Budidaya Ikan Bandeng* . [Online]. Tersedia di : <
<http://pusluh.kkp.go.id/arsip/file/55/ikan-bandeng.pdf/>> [Diakses 21
Oktober 2015]

Wardhani, L.K., Safrizal, M., Chairi, A., 2011. *Optimasi Komposisi Bahan Pakan Ikan Air Tawar Menggunakan Metode Multi Objective Genetic Algorithm*. Tersedia di: <[http:// journal.uii.ac.id/index.php/Snati/article/viewFile/2182/2008](http://journal.uii.ac.id/index.php/Snati/article/viewFile/2182/2008)> [Diakses 19 September 2015]

Yasin, M., 2013. *Prospek Usaha Budidaya Udang Organik Secara Polikultur*. Fakultas Ekonomi, Universitas Alkhairaat, Palu.



LAMPIRAN A DATA YANG DIGUNAKAN

A.1 Kebutuhan Nutri Ikan Bandeng

Karakteristik Nutrisi	Diet I (Usia 1-4 Minggu)	Diet II (Usia 5-24 Minggu)
Protein, %	30	20
Lemak, %	8-10	7-8
<i>Cholesterol</i> , %	0.8-1.0	0.8-1.0
Abu, maks, %	13	13
Serat Kasar, maks, %	4	4
Energi, K.cal/kg	3.500	3.500

Sumber : Murtijo, 2012

A.2 Kebutuhan Nutrisi Ikan Nila

Berat	umur	protein	lemak	serat	Berat
0-1 g	0-2 minggu (larva)	40%	10-15%	6-8%	0-1 g
1-10 g	2-4 minggu	35-40%	10-15%	6-8%	1-10 g
10-25 g	4-7 minggu	30-35%	10-15%	6-8%	10-25 g
25-200g	8-12 minggu	30-32%	10-15%	6-8%	25-200
>200g	> 12 minggu	28-30	10-15%	6-%	>200

Sumber : FAO

A.3 Dosis Pakan Ikan Bandeng

Umur Ikan	Dosis pakan (%bb)
≤2 minggu	5%
>2 minggu	3%

Sumber :Tristian, 2011

A.4 Dosis Pakan Ikan Nila

umur	berat	Dosis pakan
0-3 minggu	1-5 g	10-7 %
4-7 minggu	5-20g	6-4 %
8-10 minggu	20-100g	4-2.5 %
11-12minggu	100-200g	2.5-2%
13-17minggu	200-400g	2-1.5%

Sumber : Ghufuran dan Kordi, 2013

A.5 Daftar Harga Bahan Pakan

No	Bahan Baku Pakan	Nutrisi			Harga
		Protein	Lemak	Serat	
1	Tepung Ikan	60.50%	9.40%	0.70%	16000
2	Tepung Kepala Udang	53.74%	6.65%	14.61%	4500
3	Tepung Darah	71.45%	0.42%	7.95%	6500
4	Arang Bulu Ayam	25.54%	3.80%	1.80%	5000
5	Tepung Tulang	25.54%	3.80%	1.80%	4500
6	Dedak Padi	11.35%	12.15%	24.46%	3500
7	Dedak Gandum	11.99%	1.48%	3.75%	5000
8	Jagung	8.50%	3.80%	2.2	5000
9	Cental/Sorgum	13.00%	2.05%	13.50%	7000
10	Tepung Kedelai	39.60%	14.30%	2.80%	5500
11	Bungkil kedelai	44.00%	0.80%	7.30%	5500
12	Tepung Ampas Tahu	23.55%	5.54%	16.53%	2500
13	Tepung Bungkil Kacang Tanah	50.70%	10.90%	3.60%	3500
14	Bungkil Kelapa	17.09%	9.44%	3.60%	3000
15	Biji Kapuk/Randu	27.40%	5.60%	25.30%	4650
16	Tepung Daun Turi	27.54%	4.73%	14.01%	2000
17	Tepung Daun Lamtoro	36.82%	5.40%	18.14%	3000
18	Bekatul	12.90%	13	11.40%	3700

Sumer: Dharmawan, 2012

Data yang digunakan diatas sudah dikonsultasikan degan dosen Fakultas Perikanan dan Ilmu Kelautan Universitas Brawijaya.

Malang, 1 Maret 2016

Muhammad Fakhri, S.Pi., M.Sc

LAMPIRAN B HASIL WAWANCARA

Tempat	:	Fakultas Perikanan dan Ilmu Kelautan Universitas Brawijaya
Tanggal	:	29-Februari-2016
Nama Responden	:	Muhammad Fakhri, S.Pi., M.Sc
Tujuan	:	1. Verifikasi Kelayakan Hasil Komposisi yang diperoleh sistem

Soal Tanya Jawab	Responden
<p>1. Pertanyaan: Apakah ada aturan untuk mengkombinasikan bahan pakan?</p> <p>Jawaban: Iya, karena setiap bahan memiliki kandungan nutrisi yang berbeda dan komposisi bahan tidak boleh hanya dari sumber hewani saja.</p>	
<p>2. Pertanyaan: Bagaimanakah cara mengkombinasikan bahan yang bagus?</p> <p>Jawaban: misalkan membuat komposisi pakan dari 3 bahan, komposisi bahan pertama harus memiliki kandungan protein yang tinggi, bahan kedua bisa dipilih bahan yang memiliki lemak tinggi dan serat rendah dan bahan ketiga dipilih dengan protein rendah dan serat rendah.</p>	
<p>3. Pertanyaan: bagaimana cara menentukan suatu komposisi dikatakan</p>	

layak atau tidak?

Jawaban:

suatu komposisi dikatakan layak dilihat dari kebutuhan nutrisi yang ingin dicapai misalkan protein yang ingin dicapai 20%, lemak 7%, serat max 4% maka komposisi dikatakan layak jika berkisar diangka tersebut. Jika ada ketentuan maksimum maka komposisi yang baik tidak melebihi ketentuan maksimum seperti serat dengan maksimum 4% maka komposisi yang bagus tidak mengandung serat lebih dari 4%.

Malang, 1 Maret 2016

Muhammad Fakhri, S.Pi., M.Sc

LAMPIRAN C NILAI FITNESS HASIL UJI COBA

C.1 Hasil Uji Coba Rentang Kromosom

Rentang Kromosom	Nilai <i>fitness</i>										Rata-rata <i>Fitness</i>
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
1-10	0.7842999	0.7842901	0.7842997	0.7842998	0.7843003	0.7843002	0.7842900	0.7843001	0.7843002	0.7843003	0.7842981
1-20	0.8187622	0.8187550	0.8187646	0.8187547	0.8187550	0.8187549	0.8187548	0.8187638	0.8187649	0.8187549	0.8187585
1-30	0.8314753	0.8314506	0.8314593	0.8314582	0.8314579	0.8314708	0.8314626	0.8314515	0.8314755	0.8314592	0.8314621
1-40	0.8375611	0.8375611	0.8375612	0.8373162	0.8375611	0.8375622	0.8375611	0.8375610	0.8375609	0.8375609	0.8375367
1-50	0.8413451	0.8413362	0.8413441	0.8413490	0.8413429	0.8413471	0.8413473	0.8413358	0.8413368	0.8413483	0.8413433
1-60	0.8439526	0.8439517	0.8439500	0.8439514	0.8439499	0.8439509	0.8439498	0.8439497	0.8409939	0.8439500	0.8436550
1-70	0.8457990	0.8458142	0.8458121	0.8458147	0.8458164	0.8458000	0.8458012	0.8458004	0.8458132	0.8458152	0.8458086
1-80	0.8472478	0.8459606	0.8472350	0.8472550	0.8471839	0.8472527	0.8472555	0.8472562	0.8472540	0.8472542	0.8471155
1-90	0.8449892	0.8422396	0.8483743	0.8483755	0.8483615	0.8483735	0.8471848	0.8483724	0.8483753	0.8454938	0.8470140
1-100	0.8492748	0.8492798	0.8462788	0.8492795	0.8492766	0.8492780	0.8492771	0.8492797	0.8492801	0.8492792	0.8489784

C.2 Hasil Uji Coba Ukuran Populasi

Ukuran Populasi	Nilai <i>fitness</i>										Rata-rata <i>Fitness</i>
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
10	0.84928016	0.84927911	0.84927862	0.84927491	0.84926331	0.84823016	0.78581004	0.84927463	0.849275075	0.849274904	0.842824092
20	0.84927858	0.832310798	0.84927494	0.84928021	0.84927502	0.84927539	0.84927932	0.84927476	0.827605357	0.849276949	0.845413131
30	0.84928073	0.849274925	0.84927628	0.84927874	0.84927855	0.849276	0.84927705	0.84927627	0.849278899	0.849277453	0.84927749
40	0.84927814	0.849276805	0.84927875	0.84927858	0.84927945	0.84927418	0.84927816	0.84927909	0.849280418	0.849279328	0.849278289
50	0.8492792	0.849276898	0.84927959	0.84927844	0.84927867	0.84927767	0.84927941	0.84906308	0.849277909	0.849280399	0.849257126
60	0.84927978	0.848761947	0.84928007	0.84928003	0.84927507	0.84927879	0.84927895	0.84927502	0.849279228	0.849277942	0.849226683
70	0.84927583	0.849277628	0.84830361	0.84927461	0.84927878	0.84927951	0.8492797	0.84927507	0.849278555	0.849277298	0.849180059
80	0.8492799	0.849275061	0.8492767	0.84927935	0.84927931	0.84927984	0.84927883	0.84928058	0.849275996	0.849279237	0.84927848
90	0.84927936	0.84927596	0.84927968	0.84927565	0.84928003	0.84927862	0.84927489	0.84830369	0.849280075	0.849280117	0.849180807
100	0.84927942	0.849279333	0.84928026	0.84927808	0.84928002	0.84927883	0.84902695	0.84761744	0.849280234	0.849279238	0.849087981

C.3 Hasil Uji Coba Ukuran Offspring

Ukuran Offspring	Nilai fitness										Rata-rata Fitness
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
1 μ	0.8492616	0.8492621	0.8492637	0.8492461	0.8492449	0.8492460	0.8492602	0.8492618	0.8492481	0.8492474	0.8492542
2 μ	0.8492463	0.8492637	0.8492743	0.8492487	0.8492608	0.8492629	0.8492616	0.8492452	0.8492463	0.8492607	0.8492571
3 μ	0.8492744	0.8492748	0.8492741	0.8492751	0.8492742	0.8492596	0.8492637	0.8492742	0.8492750	0.8492503	0.8492695
4 μ	0.8492742	0.8492744	0.8492742	0.8492635	0.8492743	0.8492619	0.8492750	0.8492750	0.8492741	0.8492742	0.8492721
5 μ	0.8492638	0.8492747	0.8492742	0.8492748	0.8492751	0.8492627	0.8492634	0.8492748	0.8492747	0.8492747	0.8492713
6 μ	0.8492749	0.8492750	0.8492748	0.8492751	0.8492639	0.8492745	0.8492750	0.8492759	0.8492741	0.8492639	0.8492727
7 μ	0.8492749	0.8492749	0.8492750	0.8492750	0.8492745	0.8492627	0.8492750	0.8492749	0.8492744	0.8492744	0.8492736
8 μ	0.8492750	0.8492745	0.8492780	0.8492772	0.8492773	0.8492778	0.8492749	0.8492786	0.8492750	0.8492748	0.8492763
9 μ	0.8492749	0.8492751	0.8492749	0.8492792	0.8492789	0.8492635	0.8492782	0.8492776	0.8492782	0.8492773	0.8492758
10 μ	0.8492803	0.8492783	0.8492748	0.8492750	0.8492751	0.8492780	0.8492791	0.8492751	0.8492751	0.8492747	0.8492765

C.4 Hasil Uji Coba Jumlah Generasi

Jumlah Generasi	Nilai <i>fitness</i>										Rata-rata <i>Fitness</i>
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
100	0.8471706	0.8492742	0.8492750	0.8492744	0.8492751	0.8492742	0.8492635	0.8492748	0.8492604	0.8492515	0.8490594
200	0.8486173	0.8492751	0.8492746	0.8487009	0.8492761	0.8482919	0.8492748	0.8492747	0.8492748	0.8492750	0.8490535
300	0.8475433	0.8492751	0.8492751	0.8492751	0.8492750	0.8492749	0.8492751	0.8492780	0.8492767	0.8492749	0.8491023
400	0.8492793	0.8492750	0.8492809	0.8492755	0.8481581	0.8492793	0.8492750	0.8492799	0.8492788	0.8492637	0.8491645
500	0.8492779	0.8492790	0.8492743	0.8492750	0.8492772	0.8492749	0.8492750	0.8492750	0.8492772	0.8492794	0.8492765
600	0.8492751	0.8492749	0.8492751	0.8492754	0.8492789	0.8492759	0.8492751	0.8492750	0.8492750	0.8492664	0.8492747
700	0.8492758	0.8492796	0.8492800	0.8492794	0.8492797	0.8492792	0.8492748	0.8492751	0.8492750	0.8492750	0.8492774
800	0.8492748	0.8492751	0.8492751	0.8492791	0.8492751	0.8492783	0.8492751	0.8492790	0.8492751	0.8492748	0.8492762
900	0.8492751	0.8492782	0.8492770	0.8492781	0.8492804	0.8492755	0.8492788	0.8492788	0.8492751	0.8492749	0.8492772
1000	0.8492798	0.8492750	0.8492806	0.8492776	0.8492799	0.8492754	0.8492751	0.8492797	0.8492778	0.8492764	0.8492777

C.5 Hasil Uji Coba Siklus Algoritma Evolution Strategies

Siklus ES	Nilai <i>fitness</i>										Rata-rata <i>Fitness</i>
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
(μ, λ)	0.84923292	0.849232922	0.84923292	0.849268594	0.84923292	0.84923292	0.84928027	0.84928027	0.849232922	0.849280268	0.849250693
($\mu + \lambda$)	0.84927672	0.84927492	0.84927495	0.84927854	0.84927505	0.84927545	0.84927495	0.84927505	0.849278633	0.849274732	0.8492759
($\mu/r, \lambda$)	0.84926172	0.849261624	0.84924934	0.84924764	0.84925111	0.84924612	0.849258745	0.8492741	0.849260103	0.849260373	0.849257088
($\mu/r + \lambda$)	0.8492797	0.84927925	0.84927977	0.84927953	0.84927847	0.84927904	0.8492789	0.84927871	0.84927976	0.849278387	0.849279153

C.6 Hasil Uji Coba Persamaan Fitness

Persamaan	Nilai <i>fitness</i>										Rata-rata <i>Fitness</i>
	Percobaan ke- <i>i</i>										
	1	2	3	4	5	6	7	8	9	10	
A	1.083513	1.083513	1.083513	1.083513	1.083513	1.083513	1.083513	1.083513	1.083513	1.083513	1.083513238
B	0.862464	0.862464	0.862463	0.862463	0.862464	0.862465	0.862463	0.862464	0.862464	0.862464	0.862463754
C	1.394700	1.394699	1.394699	1.394699	1.394698	1.394699	1.394699	1.394698	1.394699	1.394699	1.394698931
D	14.085039	14.085106	14.085086	14.085147	14.084719	14.085875	14.085434	14.085113	14.085367	14.085311	14.08521959
E	45.428098	45.428730	45.427315	45.428927	45.426489	45.425365	45.426260	45.428452	45.427825	45.428218	45.42756792

LAMPIRAN D INDIVIDU HASIL UJI COBA KE-10

D.1 Hasil Uji Coba ke-10 Rentang Kromosom 1-100

Populasi Awal

P	x_1	x_2	x_3	σ_1	σ_2	σ_3	<i>fitness</i>	total gen
P1	3.250723	22.62351	80.06479	0.045766	0.491353	0.320443	0.637662	105.939
P2	11.98962	20.37171	98.18023	0.108929	0.240183	0.643001	0.616595	130.5416
P3	85.65635	58.8021	48.19274	0.942644	0.717044	0.325291	0.457582	192.6512
P4	15.82284	17.30141	62.08192	0.008884	0.162765	0.165925	0.600822	95.20617
P5	94.66288	49.05096	69.32541	0.361451	0.422588	0.752026	0.449333	213.0393
P6	40.5749	68.84885	79.23832	0.881796	0.294415	0.224061	0.599506	188.6621
P7	81.79908	29.38344	62.54829	0.949451	0.81579	0.757679	0.432027	173.7308
P8	17.65664	52.04695	81.27972	0.802636	0.85812	0.794254	0.650914	150.9833
P9	10.91131	95.03033	81.81843	0.278454	0.173574	0.33294	0.698859	187.7601
P10	59.17267	14.81635	55.30534	0.89671	0.375555	0.888882	0.43167	129.2944
P11	15.74247	1.287254	58.49649	0.473205	0.787237	0.287465	0.542568	75.52622
P12	77.06602	41.43704	83.36466	0.83321	0.083249	0.175076	0.474633	201.8677
P13	58.66224	33.78336	81.39429	0.318877	0.705601	0.714444	0.495226	173.8399
P14	18.94212	18.11702	64.82842	0.830216	0.043984	0.23787	0.585442	101.8876
P15	88.51277	40.52072	81.61123	0.151848	0.829765	0.854801	0.455543	210.6447
P16	8.934287	15.89755	85.18004	0.450399	0.458774	0.960867	0.615932	110.0119
P17	19.43127	28.54044	38.29947	0.592799	0.496131	0.942561	0.585445	86.27119
P18	69.94803	64.96932	26.20288	0.455961	0.252133	0.893533	0.473942	161.1202
P19	43.31461	8.984812	8.270191	0.654829	0.486152	0.992751	0.347037	60.56962
P20	33.00373	9.214999	36.38903	0.199331	0.289504	0.088492	0.447369	78.60776
P21	6.301243	11.64389	40.03174	0.951649	0.421767	0.591887	0.622658	57.97687
P22	27.18518	61.44146	16.65233	0.452927	0.590573	0.725193	0.609616	105.279
P23	41.2616	18.80074	94.43582	0.642052	0.744716	0.568439	0.523517	154.4982
P24	94.8956	34.31816	86.0915	0.226755	0.625855	0.888088	0.443155	215.3053
P25	87.811	27.3212	53.59864	0.376749	0.935969	0.956835	0.41266	168.7308
P26	30.72382	97.49861	27.61831	0.986923	0.226294	0.137957	0.671771	155.8407
P27	79.78747	62.47229	54.16185	0.862283	0.308243	0.923415	0.476818	196.4216
P28	73.2296	91.76768	11.74154	0.771128	0.796771	0.004279	0.491083	176.7388
P29	92.47197	23.8976	47.26134	0.724115	0.600089	0.296409	0.395816	163.6309
P30	82.3706	92.33034	34.6793	0.99892	0.404288	0.234565	0.499129	209.3802
P31	23.06987	40.52859	11.78242	0.137124	0.714295	0.03436	0.566139	75.38089
P32	3.112507	77.53565	96.26021	0.295102	0.910234	0.5633	0.690121	176.9084



P33	42.11966	81.96763	16.59571	0.19662	0.276349	0.378599	0.578573	140.683
P34	26.9897	36.41535	56.76845	0.383827	0.633513	0.908687	0.580793	120.1735
P35	61.54886	47.04175	63.49071	0.226633	0.240488	0.565193	0.495285	172.0813
P36	65.01539	75.40714	21.98814	0.068995	0.06746	0.269614	0.497303	162.4107
P37	4.057072	10.79507	78.45282	0.164596	0.541323	0.820489	0.616205	93.30496
P38	86.37853	90.5304	3.209549	0.49331	0.92834	0.178272	0.453021	180.1185
P39	56.372	6.164051	59.76425	0.822692	0.778341	0.931704	0.424021	122.3003
P40	13.80892	59.08588	96.87236	0.263843	0.286868	0.365644	0.657735	169.7672
P41	57.69715	41.96091	29.15081	0.205996	0.128544	0.970179	0.45845	128.8089
P42	98.3426	24.66062	22.84095	0.646232	0.726571	0.317543	0.360671	145.8442
P43	11.34831	54.57909	56.84384	0.166219	0.109282	0.749786	0.677569	122.7712
P44	24.47646	33.8359	45.03202	0.862635	0.383927	0.638182	0.576553	103.3444
P45	7.161099	18.02884	64.98016	0.487176	0.458814	0.326603	0.627121	90.1701
P46	93.3641	78.51809	15.24671	0.580323	0.355135	0.596232	0.439919	187.1289
P47	78.74286	20.51145	29.22321	0.03431	0.266413	0.288368	0.381688	128.4775
P48	5.774236	88.24136	23.59754	0.948311	0.302962	0.905096	0.767235	117.6131
P49	80.03984	42.74777	31.85439	0.855419	0.566817	0.743349	0.42461	154.642
P50	15.00864	4.125859	57.86478	0.028661	0.164532	0.684646	0.557187	76.99928

Individu Hasil Seleksi Generasi ke-500

individu	x_1	x_2	x_3	σ_1	σ_2	σ_3	<i>fitness</i>	total gen
P1	1	99.92922	1.023115	0.582742	0.186172	0.556186	0.849279	101.9523
P2	1	99.92597	1.02217	0.281539	0.089945	0.268709	0.849278	101.9481
P3	1	99.97946	1.023509	0.347579	0.111043	0.33174	0.849276	102.003
P4	1	99.98555	1.023789	0.228047	0.072855	0.217654	0.849276	102.0093
P5	1	100	1.024289	0.641017	0.204789	0.611805	0.849275	102.0243
P6	1	100	1.024136	0.524468	0.167555	0.500568	0.849275	102.0241
P7	1	100	1.024099	0.309693	0.098939	0.29558	0.849275	102.0241
P8	1	100	1.024098	0.996848	0.318468	0.951421	0.849275	102.0241
P9	1	100	1.024098	0.205242	0.06557	0.195889	0.849275	102.0241
P10	1	100	1.023967	0.524468	0.167555	0.500568	0.849275	102.024
P11	1	100	1.023854	0.424819	0.135719	0.40546	0.849275	102.0239
P12	1	100	1.023843	0.309693	0.098939	0.29558	0.849275	102.0238
P13	1	100	1.023765	0.312821	0.099939	0.298566	0.849274	102.0238
P14	1	100	1.023756	0.386199	0.123381	0.3686	0.849274	102.0238
P15	1	100	1.023665	0.424819	0.135719	0.40546	0.849274	102.0237
P16	1	100	1.023474	0.312821	0.099939	0.298566	0.849274	102.0235
P17	1	99.95989	1.015089	0.654032	0.208947	0.624227	0.849266	101.975
P18	1	99.94627	1.014236	0.253385	0.08095	0.241838	0.849266	101.9605



P19	1	99.95021	1.014469	0.309693	0.098939	0.29558	0.849266	101.9647
P20	1	99.95805	1.014769	0.281539	0.089945	0.268709	0.849266	101.9728
P21	1	99.95722	1.014633	0.228047	0.072855	0.217654	0.849265	101.9719
P22	1	99.97237	1.015116	0.424819	0.135719	0.40546	0.849265	101.9875
P23	1	99.95567	1.014006	0.312821	0.099939	0.298566	0.849265	101.9697
P24	1	99.93814	1.012779	0.382337	0.122147	0.364914	0.849264	101.9509
P25	1	99.99146	1.015942	0.281539	0.089945	0.268709	0.849264	102.0074
P26	1	100	1.016063	0.382337	0.122147	0.364914	0.849264	102.0161
P27	1	100	1.016038	0.879309	0.280918	0.839239	0.849264	102.016
P28	1	100	1.016037	0.281539	0.089945	0.268709	0.849264	102.016
P29	1	100	1.01603	0.424819	0.135719	0.40546	0.849264	102.016
P30	1	100	1.016008	0.791379	0.252826	0.755315	0.849264	102.016
P31	1	99.94841	1.012888	0.888191	0.283755	0.847716	0.849264	101.9613
P32	1	99.98748	1.015072	0.996848	0.318468	0.951421	0.849264	102.0026
P33	1	99.93676	1.012052	0.253385	0.08095	0.241838	0.849264	101.9488
P34	1	100	1.015724	0.897163	0.286622	0.856278	0.849263	102.0157
P35	1	99.97874	1.014439	0.281539	0.089945	0.268709	0.849263	101.9932
P36	1	100	1.015701	0.228047	0.072855	0.217654	0.849263	102.0157
P37	1	100	1.015601	0.472021	0.150799	0.450511	0.849263	102.0156
P38	1	100	1.015601	0.807447	0.257959	0.770651	0.849263	102.0156
P39	1	100	1.015567	0.476789	0.152322	0.455061	0.849263	102.0156
P40	1	99.96503	1.01335	0.472021	0.150799	0.450511	0.849263	101.9784
P41	1	100	1.015407	0.347579	0.111043	0.33174	0.849263	102.0154
P42	1	100	1.015395	0.654032	0.208947	0.624227	0.849263	102.0154
P43	1	100	1.015376	0.386199	0.123381	0.3686	0.849263	102.0154
P44	1	100	1.015373	0.524468	0.167555	0.500568	0.849263	102.0154
P45	1	99.94395	1.012012	0.529766	0.169247	0.505624	0.849263	101.956
P46	1	100	1.015311	0.726702	0.232163	0.693586	0.849263	102.0153
P47	1	100	1.015304	0.582742	0.186172	0.556186	0.849263	102.0153
P48	1	100	1.015249	0.42911	0.13709	0.409555	0.849263	102.0152
P49	1	100	1.015237	0.382337	0.122147	0.364914	0.849263	102.0152
P50	1	100	1.01522	0.382337	0.122147	0.364914	0.849263	102.0152

Hasil Rekomendasi Komposisi Pakan

Tepung Ikan = 0.0177 Kg

Dedak Gandum = 1.7643 Kg

Cental/Sorgum = 0.0181 Kg

Total Harga = 9230

D.2 Hasil Uji Coba ke-10 Ukuran Populasi 30

Individu Populasi Awal



P	x_1	x_2	x_3	σ_1	σ_2	σ_3	<i>fitness</i>	total gen
P1	61.38323	35.00124	75.38971	0.605648	0.719116	0.900627	0.486323	171.7742
P2	61.77581	27.56305	83.80904	0.486137	0.498435	0.299803	0.480944	173.1479
P3	24.27231	45.13154	9.488749	0.93319	0.638055	0.720569	0.570874	78.8926
P4	70.60643	96.73449	10.64065	0.536776	0.835947	0.274731	0.503453	177.9816
P5	31.6483	58.73733	80.99981	0.106527	0.071059	0.443306	0.61887	171.3854
P6	94.43683	64.50213	23.25954	0.276275	0.706831	0.046576	0.428131	182.1985
P7	69.13515	62.67293	69.89775	0.400148	0.031267	0.312987	0.508351	201.7058
P8	43.85782	84.95713	66.14606	0.174402	0.703156	0.334294	0.60587	194.961
P9	94.05683	67.44174	42.78245	0.594289	0.058318	0.293461	0.453471	204.281
P10	87.72264	89.86516	66.84622	0.515484	0.28792	0.285547	0.5075	244.434
P11	28.12764	10.35643	65.94608	0.267032	0.430937	0.133735	0.518885	104.4302
P12	60.80279	25.21368	80.24252	0.036613	0.519212	0.192967	0.475709	166.259
P13	95.74765	79.09949	22.88096	0.49514	0.21819	0.889796	0.446615	197.7281
P14	79.40015	41.62425	82.59742	0.769058	0.857477	0.372886	0.470661	203.6218
P15	69.27004	1.727416	61.6429	0.125071	0.828435	0.64532	0.398707	132.6404
P16	70.52185	68.37369	2.320844	0.869837	0.22321	0.465165	0.442105	141.2164
P17	21.59486	38.19594	27.3157	0.050074	0.130036	0.979444	0.58913	87.10651
P18	87.60272	18.443	67.13496	0.676691	0.809311	0.890293	0.412781	173.1807
P19	2.961853	38.54921	88.18531	0.797401	0.726696	0.129221	0.656779	129.6964
P20	62.16902	11.21863	84.06218	0.759746	0.08303	0.771226	0.453018	157.4498
P21	66.8651	61.01231	88.12504	0.191604	0.572806	0.787504	0.52288	216.0024
P22	94.27782	4.208362	86.44672	0.376949	0.936798	0.292466	0.40378	184.9329
P23	23.5931	38.35482	69.1187	0.91553	0.810482	0.063957	0.612241	131.0666
P24	21.03792	95.40234	21.03525	0.467866	0.065442	0.227675	0.727907	137.4755
P25	99.01034	9.051763	96.80395	0.440075	0.500028	0.317514	0.415106	204.8661
P26	19.0211	58.70973	33.98913	0.141768	0.359177	0.21966	0.672889	111.72
P27	14.83988	56.95931	26.40603	0.036873	0.799322	0.343067	0.698911	98.20522
P28	17.90697	98.23251	3.578379	0.530941	0.407442	0.342832	0.755729	119.7179
P29	31.88128	31.44372	58.45021	0.425053	0.12519	0.763293	0.547998	121.7752
P30	74.26956	97.00029	30.21695	0.888933	0.108048	0.518327	0.51836	201.4868

Individu Hasil Seleksi Generasi ke 500

Individu	x_1	x_2	x_3	σ_1	σ_2	σ_3	<i>fitness</i>	total gen
P1	1	99.94486	1.022727	0.620129	0.475884	0.400421	0.849277	101.9676
P2	1	99.93756	1.022215	0.626393	0.480691	0.404466	0.849277	101.9598
P3	1	100	1.024253	0.247605	0.190011	0.15988	0.849275	102.0243
P4	1	100	1.024138	0.703022	0.539496	0.453946	0.849275	102.0241
P5	1	100	1.024131	0.275116	0.211123	0.177645	0.849275	102.0241



P6	1	99.99765	1.02393	0.275116	0.211123	0.177645	0.849275	102.0216
P7	1	99.98877	1.023399	0.563753	0.432622	0.364019	0.849275	102.0122
P8	1	100	1.023993	0.569448	0.436992	0.367696	0.849275	102.024
P9	1	100	1.023822	0.305685	0.234581	0.197383	0.849274	102.0238
P10	1	100	1.023803	0.512503	0.393293	0.330927	0.849274	102.0238
P11	1	100	1.023748	0.63272	0.485547	0.408552	0.849274	102.0237
P12	1	100	1.023656	0.613928	0.471125	0.396417	0.849274	102.0237
P13	1	100	1.023514	0.336253	0.258039	0.217121	0.849274	102.0235
P14	1	99.95994	1.015464	0.275116	0.211123	0.177645	0.849266	101.9754
P15	1	99.94466	1.014261	0.222844	0.17101	0.143892	0.849266	101.9589
P16	1	99.9406	1.013664	0.275116	0.211123	0.177645	0.849265	101.9543
P17	1	99.94994	1.013588	0.512503	0.393293	0.330927	0.849265	101.9635
P18	1	100	1.016107	0.512503	0.393293	0.330927	0.849264	102.0161
P19	1	100	1.016046	0.558116	0.428296	0.360379	0.849264	102.016
P20	1	100	1.015979	0.569448	0.436992	0.367696	0.849264	102.016
P21	1	100	1.01597	0.305685	0.234581	0.197383	0.849264	102.016
P22	1	100	1.015928	0.926365	0.710889	0.59816	0.849264	102.0159
P23	1	100	1.015863	0.461253	0.353963	0.297834	0.849264	102.0159
P24	1	100	1.015821	0.563753	0.432622	0.364019	0.849264	102.0158
P25	1	100	1.015818	0.954721	0.732649	0.61647	0.849264	102.0158
P26	1	100	1.015794	0.373615	0.28671	0.241246	0.849264	102.0158
P27	1	100	1.015655	0.275116	0.211123	0.177645	0.849263	102.0157
P28	1	100	1.015609	0.20056	0.153909	0.129503	0.849263	102.0156
P29	1	100	1.015589	0.569448	0.436992	0.367696	0.849263	102.0156
P30	1	100	1.015548	0.63272	0.485547	0.408552	0.849263	102.0155

Hasil Rekomendasi Komposisi Pakan

Tepung Ikan = 0.0177 Kg

Dedak Gandum = 1.7643 Kg

Cental/Sorgum = 0.0181 Kg

Total Harga = 9230

D.3 Hasil Uji Coba ke-10 Ukuran Offspring 8μ

Individu Populasi Awal

P	x ₁	x ₂	x ₃	σ ₁	σ ₂	σ ₃	fitness	total gen
P1	10.75946	30.47829	50.64324	0.284725	0.935328	0.459735	0.648079	91.881
P2	39.01541	69.33502	91.94495	0.980973	0.382428	0.660631	0.611471	200.2954
P3	62.82315	33.2776	65.25319	0.5747	0.082201	0.607244	0.471154	161.3539
P4	38.58248	65.21391	63.46226	0.24505	0.525811	0.375744	0.592611	167.2586
P5	21.74823	77.08686	10.31641	0.788891	0.401981	0.416918	0.690059	109.1515

P6	68.66094	19.72717	33.34815	0.989935	0.479855	0.575083	0.397545	121.7363
P7	11.30918	41.79137	76.32091	0.88304	0.727738	0.6504	0.651251	129.4215
P8	63.95295	64.79116	37.55276	0.011898	0.54969	0.219619	0.49683	166.2969
P9	86.64581	70.90428	82.10792	0.956138	0.493056	0.308896	0.496273	239.658
P10	35.4403	3.959184	5.679578	0.382394	0.378961	0.330328	0.325095	45.07906
P11	70.31985	24.81463	62.14238	0.837727	0.422322	0.292099	0.440379	157.2769
P12	6.498261	31.62106	10.77263	0.496831	0.035089	0.870731	0.719609	48.89195
P13	77.19508	30.4197	65.76741	0.250097	0.002648	0.302569	0.443026	173.3822
P14	60.31544	57.17427	52.64285	0.390458	0.054482	0.879155	0.505461	170.1326
P15	95.82243	38.0599	18.76168	0.84513	0.341382	0.895918	0.379162	152.644
P16	49.12851	48.27955	91.48947	0.385332	0.41741	0.343335	0.548679	188.8975
P17	2.393065	92.41545	56.36944	0.623603	0.299343	0.191428	0.73501	151.178
P18	99.49079	15.63312	24.25136	0.324665	0.442725	0.408657	0.346602	139.3753
P19	82.11083	84.67181	55.22948	0.671465	0.794744	0.59072	0.503519	222.0121
P20	74.40976	60.41359	50.70778	0.562002	0.831914	0.328684	0.480211	185.5311
P21	86.02945	57.81405	10.29257	0.793146	0.896431	0.903862	0.410843	154.1361
P22	7.711097	77.69313	53.83969	0.50257	0.549934	0.102772	0.712499	139.2439
P23	78.00082	82.37105	26.88633	0.486205	0.767457	0.552803	0.486093	187.2582
P24	51.41555	6.640706	40.35968	0.842759	0.907063	4.44E-04	0.403402	98.41593
P25	16.38716	28.29576	26.56172	0.784622	0.681109	0.011286	0.594499	71.24464
P26	48.24976	90.28957	76.62289	0.721081	0.101504	0.145838	0.603235	215.1622
P27	37.60333	71.94101	46.2377	0.799821	0.211413	0.348399	0.598434	155.782
P28	13.6537	17.9916	85.34068	0.780095	0.596738	0.027922	0.612346	116.986
P29	62.3745	20.26856	65.44821	0.308981	0.298786	0.66967	0.448944	148.0913
P30	20.79448	32.43818	13.63549	0.643658	0.040611	0.213283	0.55392	66.86815

Individu Hasil Seleksi Generasi ke-500

Individu	x_1	x_2	x_3	σ_1	σ_2	σ_3	<i>fitness</i>	total gen
P1	1	100	1.024086	9.295033	0.656461	16.2902	0.849275	102.0241
P2	1	100	1.023884	9.295033	0.656461	16.2902	0.849275	102.0239
P3	1	100	1.015675	10.32781	0.729401	18.10022	0.849263	102.0157
P4	1	100	1.014944	5.488624	0.387634	9.619199	0.849262	102.0149
P5	1	100	1.01365	5.488624	0.387634	9.619199	0.849261	102.0136
P6	1	100	1.004995	4.445786	0.313983	7.791552	0.849249	102.005
P7	1	100	1.00379	10.32781	0.729401	18.10022	0.849247	102.0038
P8	1	99.94693	1	7.528977	0.531734	13.19506	0.849246	101.9469
P9	1	99.94696	1	5.488624	0.387634	9.619199	0.849246	101.947
P10	1	99.947	1	10.32781	0.729401	18.10022	0.849246	101.947
P11	1	99.9471	1	4.445786	0.313983	7.791552	0.849246	101.9471
P12	1	99.94715	1	6.098471	0.430704	10.688	0.849246	101.9471
P13	1	99.94716	1	4.939762	0.34887	8.657279	0.849246	101.9472



P14	1	99.94736	1	3.601086	0.254327	6.311157	0.849246	101.9474
P15	1	99.94737	1	3.601086	0.254327	6.311157	0.849246	101.9474
P16	1	99.94746	1	6.098471	0.430704	10.688	0.849246	101.9475
P17	1	99.94748	1	3.240978	0.228894	5.680041	0.849246	101.9475
P18	1	99.94751	1	5.488624	0.387634	9.619199	0.849246	101.9475
P19	1	99.94751	1	4.939762	0.34887	8.657279	0.849246	101.9475
P20	1	99.94755	1	4.445786	0.313983	7.791552	0.849246	101.9475
P21	1	99.94771	1	6.098471	0.430704	10.688	0.849246	101.9477
P22	1	99.94773	1	4.939762	0.34887	8.657279	0.849246	101.9477
P23	1	99.94779	1	10.32781	0.729401	18.10022	0.849246	101.9478
P24	1	99.9479	1	4.939762	0.34887	8.657279	0.849246	101.9479
P25	1	99.9479	1	4.001207	0.282585	7.012396	0.849246	101.9479
P26	1	99.9479	1	4.445786	0.313983	7.791552	0.849246	101.9479
P27	1	99.94794	1	15.74122	1.111723	27.58759	0.849246	101.9479
P28	1	99.948	1	5.488624	0.387634	9.619199	0.849246	101.948
P29	1	99.94805	1	7.528977	0.531734	13.19506	0.849246	101.948
P30	1	99.94809	1	10.32781	0.729401	18.10022	0.849246	101.9481

Hasil Rekomendasi Komposisi Pakan

Tepung Ikan = 0.0176 Kg

Dedak Gandum = 1.7643 Kg

Cental/Sorgum = 0.0181 Kg

Total Harga = 9230

D.4 Hasil Uji Coba ke-10 Jumlah Generasi 500

Individu Populasi Awal

P	x ₁	x ₂	x ₃	σ ₁	σ ₂	σ ₃	fitness	total gen
P1	7.892843	33.26038	20.50798	0.896531	0.733289	0.548719	0.693431	61.66121
P2	88.55595	52.11074	81.81024	0.52271	0.691174	0.41876	0.470456	222.4769
P3	2.592595	13.7065	60.30963	0.795305	0.815513	0.243997	0.630128	76.60872
P4	7.752405	95.69289	74.08388	0.464093	2.50E-04	0.447644	0.710048	177.5292
P5	56.1847	70.62007	91.60122	0.161743	0.693365	0.805415	0.56185	218.406
P6	47.87443	12.32881	46.20613	0.19572	0.26148	0.375884	0.434889	106.4094
P7	44.16374	22.36936	80.09912	0.972933	0.03475	0.394418	0.508804	146.6322
P8	64.8849	65.95256	98.01935	0.170266	0.963454	0.524617	0.539253	228.8568
P9	92.41014	40.01168	50.83889	0.037364	0.102742	0.830845	0.423981	183.2607
P10	31.63515	97.09003	44.22442	0.060881	0.811971	0.302367	0.669621	172.9496
P11	3.479097	99.93375	40.18671	0.71853	0.837553	0.341939	0.757028	143.5996
P12	56.22148	97.47511	34.37902	0.465782	0.492592	0.619509	0.567601	188.0756



P13	10.36061	99.34009	4.721166	0.108327	0.27418	0.29683	0.790624	114.4219
P14	31.01625	5.250732	36.00931	0.67218	0.213123	0.235148	0.438762	72.27629
P15	33.19843	42.74977	86.45082	0.064888	0.446197	0.211241	0.587974	162.399
P16	88.90199	2.229285	21.26281	0.672019	0.274652	0.864627	0.322413	112.3941
P17	32.4822	16.35987	25.953	0.421091	0.833412	0.480036	0.45237	74.79507
P18	38.52883	98.90018	42.41602	0.310473	0.373881	0.345602	0.638651	179.845
P19	74.53448	34.6136	64.42061	0.405339	0.662081	0.311813	0.452319	173.5687
P20	33.4888	58.36784	52.2172	0.48038	0.637314	0.974116	0.594204	144.0738
P21	3.703373	71.57172	27.28654	0.895318	0.961062	0.200115	0.754828	102.5616
P22	61.15515	15.17955	54.49315	0.65177	0.529497	0.279303	0.428019	130.8279
P23	2.541651	84.72796	18.05887	0.548271	0.215841	0.96992	0.78994	105.3285
P24	37.57044	73.06027	77.67815	0.745362	0.762842	0.006357	0.616235	188.3089
P25	64.40434	85.73232	26.24547	0.117935	0.744496	0.985927	0.521022	176.3821
P26	20.18422	52.91056	13.93406	0.422394	0.165656	0.922114	0.636309	87.02885
P27	85.32973	24.09054	32.77264	0.378663	0.545738	0.500763	0.386451	142.1929
P28	12.43949	93.04344	74.32251	0.270365	0.137881	0.246059	0.699566	179.8055
P29	33.78229	10.29412	50.11035	0.031303	0.33881	0.356571	0.473838	94.18676
P30	43.18621	66.53793	11.06354	0.115245	0.840616	0.041834	0.530762	120.7877

Individu Hasil Seleksi Generasi ke-500

Individu	x_1	x_2	x_3	σ_1	σ_2	σ_3	<i>fitness</i>	total gen
P1	1	99.92769	1.023134	0.22105	0.087022	0.391049	0.849279	101.9508
P2	1	99.93227	1.023037	0.336915	0.132635	0.596021	0.849279	101.9553
P3	1	99.93567	1.023042	0.272901	0.107435	0.482777	0.849279	101.9587
P4	1	99.94661	1.023469	0.564863	0.222373	0.999273	0.849278	101.9701
P5	1	99.97373	1.023639	0.245611	0.096691	0.434499	0.849276	101.9974
P6	1	99.97084	1.023174	0.508377	0.200136	0.899346	0.849276	101.994
P7	1	100	1.024274	0.272901	0.107435	0.482777	0.849275	102.0243
P8	1	100	1.024221	0.415945	0.163747	0.735829	0.849275	102.0242
P9	1	100	1.024099	0.37435	0.147373	0.662246	0.849275	102.0241
P10	1	100	1.02405	0.303224	0.119372	0.536419	0.849275	102.0241
P11	1	100	1.024032	0.508377	0.200136	0.899346	0.849275	102.024
P12	1	100	1.023912	0.336915	0.132635	0.596021	0.849275	102.0239
P13	1	100	1.023903	0.336915	0.132635	0.596021	0.849275	102.0239
P14	1	100	1.023875	0.336915	0.132635	0.596021	0.849275	102.0239
P15	1	100	1.023855	0.415945	0.163747	0.735829	0.849275	102.0239
P16	1	100	1.023769	0.303224	0.119372	0.536419	0.849274	102.0238
P17	1	100	1.023697	0.333546	0.131309	0.590061	0.849274	102.0237
P18	1	100	1.023613	0.303224	0.119372	0.536419	0.849274	102.0236
P19	1	100	1.023562	0.570569	0.224619	1.009367	0.849274	102.0236



P20	1	100	1.02355	0.303224	0.119372	0.536419	0.849274	102.0236
P21	1	100	1.023496	0.457539	0.180122	0.809411	0.849274	102.0235
P22	1	99.93377	1.014583	0.272901	0.107435	0.482777	0.849267	101.9484
P23	1	99.9524	1.015569	0.336915	0.132635	0.596021	0.849267	101.968
P24	1	99.95051	1.015122	0.633966	0.249577	1.121519	0.849267	101.9656
P25	1	99.96467	1.015215	0.272901	0.107435	0.482777	0.849266	101.9799
P26	1	99.96483	1.01502	0.272901	0.107435	0.482777	0.849265	101.9799
P27	1	99.95068	1.013843	0.336915	0.132635	0.596021	0.849265	101.9645
P28	1	100	1.016087	0.415945	0.163747	0.735829	0.849264	102.0161
P29	1	100	1.016068	0.37435	0.147373	0.662246	0.849264	102.0161
P30	1	100	1.016051	0.37435	0.147373	0.662246	0.849264	102.0161

Hasil Rekomendasi Komposisi Pakan

- Tepung Ikan = 0.0177 Kg
- Dedak Gandum = 1.7643 Kg
- Cental/Sorgum = 0.0181 Kg
- Total Harga = 9230

D.5 Hasil Uji Coba ke -10 Siklus Algoritma Evolution Strategies ($\mu/r+\lambda$)

Individu Populasi Awal

P	x_1	x_2	x_3	σ_1	σ_2	σ_3	fitness	total gen
P1	1.22022	0.667299	0.074512	0.476429	0.428016	0.317813	0.383885	1.96203
P2	0.662288	0.071587	0.354351	0.233658	0.775016	0.432814	0.375665	1.088225
P3	1.239053	1.323727	1.400854	0.654053	0.207067	0.591239	0.528838	3.963634
P4	0.488675	0.657909	0.693073	0.123981	0.836473	0.594752	0.562153	1.839657
P5	1.520838	0.346977	0.45694	0.689189	0.086164	0.27194	0.36701	2.324755
P6	0.707041	0.13335	1.569058	0.129734	0.143971	0.81642	0.499547	2.409449
P7	0.13927	0.211447	1.252382	0.980759	0.517052	0.17539	0.612702	1.603099
P8	0.576408	1.778042	1.504536	0.026614	0.753761	0.194809	0.67089	3.858985
P9	0.934586	1.149053	0.684029	0.438005	0.375805	0.041136	0.526374	2.767668
P10	0.887405	0.214696	1.52084	0.706845	0.630823	0.308407	0.480228	2.622941
P11	1.070779	1.50667	1.093548	0.061309	0.987834	0.53991	0.55395	3.670997
P12	0.803281	1.289929	0.824541	0.923912	0.868113	0.11157	0.570306	2.91775
P13	1.476359	1.598141	1.659358	0.877216	0.05394	0.364571	0.529826	4.733858
P14	0.901455	0.008915	0.508913	0.303232	0.52887	0.972411	0.363593	1.419283
P15	0.15683	0.126431	0.609332	0.501546	0.26641	0.076257	0.586092	0.892593
P16	0.111337	1.101784	1.080295	0.414618	0.149546	0.125663	0.694427	2.293415
P17	1.543038	0.793118	1.197924	0.703132	0.830974	0.381973	0.451981	3.534081

P18	1.266932	1.138643	1.523011	0.53541	0.852062	0.051226	0.516555	3.928586
P19	0.559274	0.497733	0.492337	0.158089	0.755908	0.243488	0.500004	1.549344
P20	1.259408	0.392346	1.564376	0.784913	0.64398	0.747128	0.460192	3.21613
P21	0.15849	0.358605	1.657078	0.120667	0.63384	0.659803	0.621242	2.174172
P22	1.085109	0.874096	1.792892	0.581286	0.833793	0.542363	0.527303	3.752097
P23	1.479204	0.259685	0.814733	0.991597	0.724819	0.126568	0.387534	2.553622
P24	1.638626	0.736708	0.918114	0.861263	0.555939	0.9906	0.427075	3.293448
P25	0.329669	0.144441	0.693011	0.738508	0.450981	0.655724	0.515051	1.167122
P26	1.298376	0.353808	0.339762	0.211144	0.186271	0.500003	0.368581	1.991946
P27	1.199571	1.038261	0.944091	0.81535	0.933618	0.837274	0.492357	3.181922
P28	1.193755	0.840517	0.116486	0.984183	0.933595	0.687416	0.412712	2.150758
P29	1.43804	1.784903	1.642011	0.638892	0.942767	0.742459	0.544329	4.864953
P30	0.328253	0.58209	0.991332	0.625083	0.116522	0.996915	0.620989	1.901676

Individu Hasil Seleksi Generasi ke 500

Individu	x_1	x_2	x_3	σ_1	σ_2	σ_3	fitness	total gen
P1	1.006401	197.2207	1	0.786861	0.174061	0.912011	0.853342	199.2271
P2	1.001678	196.516	1	0.133645	0.088502	0.752093	0.853341	198.5176
P3	1.000336	196.4383	1	0.005976	0.217401	0.579112	0.853338	198.4387
P4	1	196.4286	1	0.471188	0.988774	0.127073	0.853337	198.4286
P5	1	196.4295	1	0.48741	1.067651	0.50494	0.853337	198.4295
P6	1	196.4304	1	0.066298	0.227699	1.099136	0.853337	198.4304
P7	1	196.4311	1	0.241435	0.30116	0.177742	0.853337	198.4311
P8	1	196.4324	1	0.420337	0.142069	0.579395	0.853337	198.4324
P9	1.001609	196.6972	1	0.018352	0.390698	0.977127	0.853337	198.6988
P10	1	196.4425	1	0.329259	0.074115	0.63366	0.853337	198.4425
P11	1	196.4475	1	0.269687	0.367609	0.290595	0.853337	198.4475
P12	1	196.4548	1	0.20473	0.156419	1.082069	0.853336	198.4548
P13	1	196.4556	1	0.152126	0.169263	0.300018	0.853336	198.4556
P14	1	196.4674	1	0.15455	0.154969	1.01465	0.853336	198.4674
P15	1	196.4691	1	0.940895	0.926622	0.236826	0.853336	198.4691
P16	1	196.475	1	0.578424	0.42735	0.429443	0.853336	198.475
P17	1	196.4814	1	0.723103	0.216745	0.38416	0.853336	198.4814
P18	1	196.4825	1	0.515125	0.792157	0.308997	0.853336	198.4825
P19	1	196.4828	1	0.549504	0.032673	0.431374	0.853336	198.4828
P20	1	196.4852	1	1.054532	0.058174	1.003602	0.853336	198.4852
P21	1	196.4878	1	1.06911	0.124539	0.721582	0.853336	198.4878
P22	1	196.493	1	0.177233	0.114986	0.147083	0.853336	198.493



P23	1	196.494	1	0.591518	0.749346	0.131083	0.853336	198.494
P24	1	196.5035	1	0.38205	0.079646	0.372756	0.853335	198.5035
P25	1	196.5046	1	0.546755	0.258735	1.057894	0.853335	198.5046
P26	1	196.5103	1	0.034156	0.326835	0.37806	0.853335	198.5103
P27	1	196.5109	1	0.399138	0.744614	1.030119	0.853335	198.5109
P28	1	196.511	1	0.69014	0.065271	0.853358	0.853335	198.511
P29	1	196.5111	1	0.10439	0.876981	0.883547	0.853335	198.5111
P30	1	196.5142	1	0.08071	0.086923	0.696679	0.853335	198.5142

Hasil Rekomendasi Komposisi Pakan

Tepung Ikan = 0.0091 Kg
 Dedak Gandum = 1.7819 Kg
 Sorghum = 0.009 Kg
 Total Harga = 9118

D.6 Hasil Uji Coba ke-10 Persamaan Fitness

Individu Awal

P	x_1	x_2	x_3	σ_1	σ_2	σ_3	fitness	total gen
P1	99.04878	26.41724	99.67685	0.743698	0.931817	0.929071	10.86674	225.1429
P2	91.78719	46.47068	10.27133	0.236146	0.226617	0.030659	8.353813	148.5292
P3	29.32418	86.66911	70.94354	0.061833	0.389365	0.648735	27.32657	186.9368
P4	62.59102	21.47781	61.84109	0.566479	0.717383	0.116037	11.22389	145.9099
P5	99.04462	56.10851	33.74741	0.217791	0.930045	0.14258	10.47115	188.9005
P6	68.7068	63.96301	1.453402	0.920046	0.971092	0.942965	9.997251	134.1232
P7	51.95488	35.73081	65.23728	0.87928	0.229323	0.782554	13.72227	152.923
P8	24.31224	81.57983	45.67083	0.010376	0.482834	0.260479	30.22852	151.5629
P9	96.09915	81.22054	87.53582	0.676452	0.471935	0.346116	13.81871	264.8555
P10	42.26057	56.56018	39.31152	0.161822	0.108016	0.373772	16.79047	138.1323
P11	18.78897	99.04294	3.127845	0.414242	0.782401	0.121054	36.81688	120.9597
P12	77.09582	79.08083	31.80229	0.134051	0.825122	0.664759	13.68333	187.9789
P13	16.78808	2.096333	43.76718	0.542732	0.824815	0.76102	13.67564	62.6516
P14	31.79544	85.64073	65.91249	0.109024	0.625693	0.042747	25.95314	183.3487
P15	11.5344	61.73023	27.32478	0.04626	0.406015	0.797654	22.50333	100.5894
P16	41.03805	54.50516	38.19114	0.591526	0.276254	0.587001	16.72869	133.7343
P17	95.12001	80.50682	16.99528	0.428635	0.448414	0.512986	10.76411	192.6221
P18	76.96111	65.36187	75.19032	0.287787	0.776	0.73775	13.99437	217.5133
P19	44.93327	3.438661	73.5562	0.832126	0.574775	0.517432	11.51396	121.9281
P20	89.32175	76.70437	28.09974	0.683986	0.256923	0.830648	11.9953	194.1259

P21	56.33054	24.57041	15.93094	0.532026	0.766945	0.86403	9.243966	96.83189
P22	56.70332	27.10887	53.74088	0.047354	0.839645	0.551932	11.84738	137.5531
P23	54.45055	79.69008	72.14626	0.09643	0.12402	0.858696	18.17647	206.2869
P24	23.92099	44.504	50.9734	0.917379	0.568601	0.01748	21.26483	119.3984
P25	67.31037	13.65308	11.8945	0.392479	0.652657	0.89909	7.032365	92.85795
P26	67.07177	31.74646	95.45245	0.843048	0.004274	0.648733	12.97264	194.2707
P27	65.63002	98.28195	79.31248	0.355651	0.27321	0.691314	18.20739	243.2245
P28	52.7475	63.3105	33.3421	0.154309	0.348094	0.831885	15.33151	149.4001
P29	42.42123	23.06561	10.27152	0.017807	0.820111	0.054013	9.540331	75.75835
P30	31.78399	55.67605	94.03902	0.931902	0.117152	0.503987	21.31587	181.4991

Individu Hasil seleksi pada generai ke 500

Individu	x_1	x_2	x_3	σ_1	σ_2	σ_3	fitness	total gen
P1	19.34849	89.89105	9.086595	0.427974	0.840295	0.186502	45.42822	118.3261
P2	19.2408	89.3913	9.034168	0.029151	0.06637	0.010728	45.42751	117.6663
P3	19.22844	89.32702	9.029731	0.864086	0.548857	0.009969	45.42729	117.5852
P4	19.26585	89.49999	9.048138	0.270627	0.166163	0.049304	45.4271	117.814
P5	19.19002	89.14742	9.013435	0.057195	0.384623	0.535079	45.42665	117.3509
P6	19.23804	89.37031	9.036066	1.020485	0.001746	0.021865	45.4265	117.6444
P7	19.26813	89.5143	9.04586	0.082924	0.011584	0.122351	45.42553	117.8283
P8	19.35966	89.94642	9.087153	0.104485	0.846509	0.046082	45.42449	118.3932
P9	19.18711	89.12932	9.01396	0.170791	0.683048	0.018822	45.42324	117.3304
P10	19.24381	89.39189	9.039962	0.353149	0.072258	0.208134	45.42293	117.6757
P11	19.1679	89.04255	8.999418	0.4113	0.138063	0.062002	45.42284	117.2099
P12	19.15831	88.99565	9.002496	0.336895	0.161254	0.177763	45.4227	117.1565
P13	19.12872	88.87627	8.983904	0.093562	0.568188	0.026939	45.42228	116.9889
P14	19.34759	89.89387	9.086777	0.155263	0.552811	1.050802	45.42159	118.3282
P15	19.1594	88.99752	8.996819	0.397319	0.635204	0.093501	45.42152	117.1537
P16	19.30346	89.67166	9.062373	0.398817	0.381714	0.492518	45.42149	118.0375
P17	19.36134	89.95753	9.083829	0.209852	0.561868	0.155017	45.42103	118.4027
P18	19.19061	89.16457	9.004414	0.283298	0.243524	0.016496	45.42089	117.3596
P19	19.2478	89.43224	9.039399	0.023724	0.207319	0.014864	45.42047	117.7194
P20	19.23184	89.35069	9.036083	0.837539	0.558512	0.001319	45.4203	117.6186
P21	19.28469	89.57906	9.054356	0.01682	0.088344	0.654595	45.41951	117.9181
P22	19.28585	89.60935	9.058178	0.154166	0.665678	0.014933	45.41905	117.9534
P23	19.2548	89.44494	9.054493	0.083323	0.116396	0.073899	45.41861	117.7542
P24	19.30708	89.68665	9.079731	1.024218	0.598934	0.870241	45.41855	118.0735
P25	19.34755	89.86532	9.085406	0.302746	0.750495	0.070619	45.4183	118.2983
P26	19.26822	89.50468	9.044661	0.106245	0.441361	0.097381	45.41797	117.8176



P27	19.03051	88.39595	8.945687	0.344122	0.78273	0.667998	45.41795	116.3721
P28	19.23996	89.36607	9.038811	0.421281	0.759731	0.033927	45.41776	117.6448
P29	19.17187	89.08264	9.002065	0.020428	0.032165	0.011637	45.41741	117.2566
P30	19.24216	89.40901	9.032581	0.44037	0.152124	0.121495	45.41725	117.6838

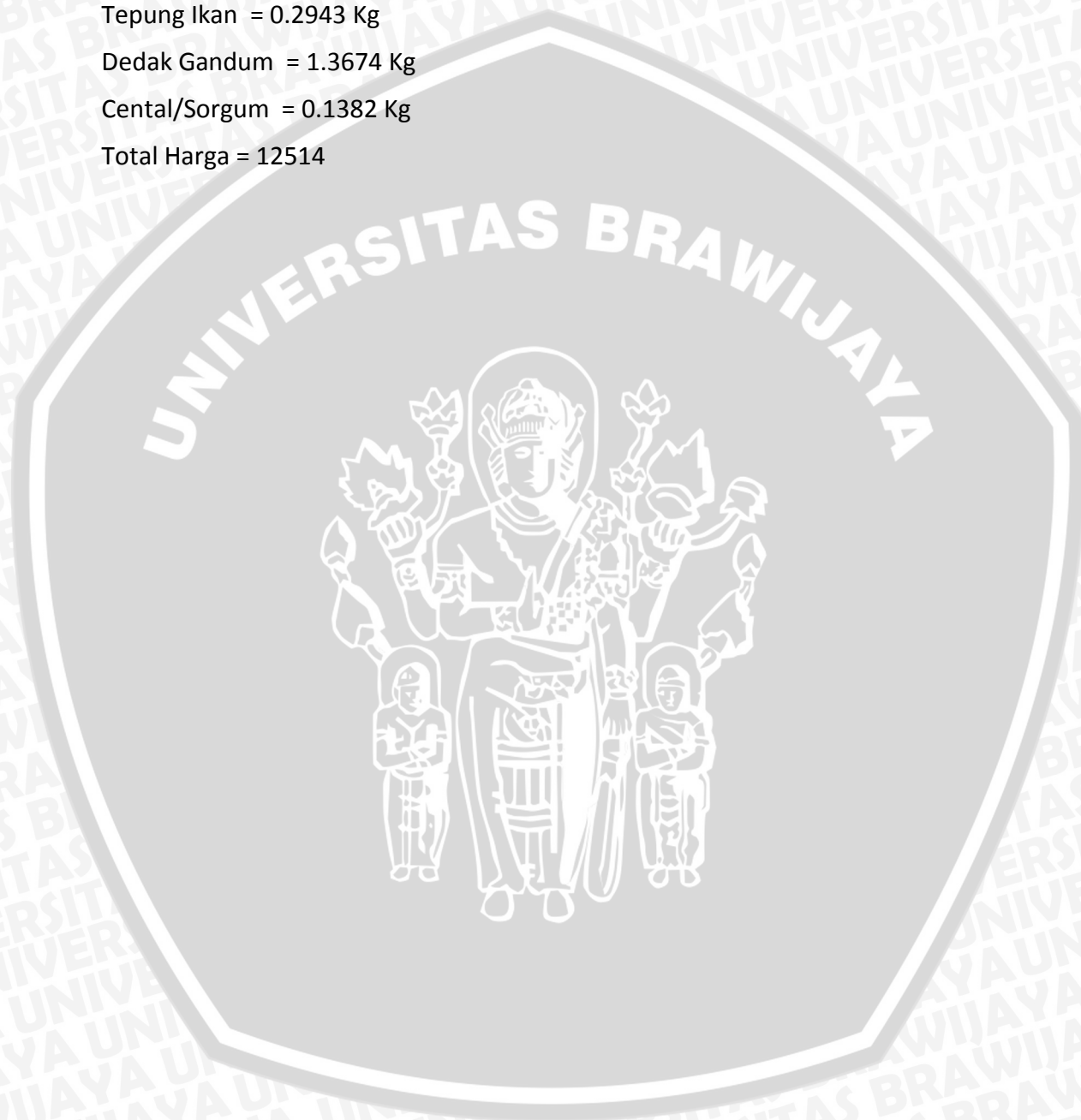
Hasil Rekomendasi Komposisi Pakan

Tepung Ikan = 0.2943 Kg

Dedak Gandum = 1.3674 Kg

Cental/Sorgum = 0.1382 Kg

Total Harga = 12514



LAMPIRAN E HASIL PERHITUNGAN SISTEM

E.1 Individu awal

P	x_1	x_2	x_3	σ_1	σ_2	σ_3	fitness	total gen
P1	22.4402	82.1006	76.8097	0.6481	0.3329	0.6399	17.1629	181.3505
P2	13.0566	53.0247	82.6891	0.4631	0.5365	0.5279	13.6663	148.7705
P3	88.7668	50.8245	81.3969	0.1268	0.4069	0.7952	7.9893	220.9882
P4	58.8807	8.0908	69.4862	0.7878	0.5003	0.3598	6.3286	136.4577
P5	16.6272	62.2943	97.9192	0.9028	0.1535	0.3633	13.4205	176.8407
P6	75.5132	10.7657	6.4838	0.7002	0.4989	0.9128	5.4365	92.7628
P7	51.2790	10.5333	66.0178	0.7408	0.1120	0.8636	6.6165	127.8302
P8	98.3338	75.0227	48.0785	0.4166	0.7748	0.7960	9.1667	221.4350
P9	10.7100	99.9747	71.9367	0.9447	0.4447	0.7338	29.7626	182.6215
P10	76.8274	11.4961	69.8559	0.7791	0.5318	0.2798	6.2166	158.1793
P11	70.8451	41.3548	32.0024	0.0618	0.6068	0.4217	8.1252	144.2022
P12	9.3614	30.2832	64.1489	0.9014	0.4364	0.9528	11.7459	103.7935
P13	38.8613	24.1068	37.0662	0.6762	0.9585	0.0618	8.1995	100.0343
P14	89.4086	8.8325	59.8026	0.9328	0.3180	0.4325	5.8356	158.0438
P15	64.8253	1.8827	3.4609	0.6676	0.1560	0.0622	4.7591	70.1688
P16	79.3221	66.3370	10.7622	0.1490	0.9353	0.8004	9.5774	156.4213
P17	8.4703	87.3587	10.2192	0.9328	0.5501	0.3280	33.7585	106.0482
P18	24.3881	51.8433	27.8117	0.2463	0.8595	0.3861	16.2119	104.0431
P19	91.3183	33.2857	50.2844	0.5427	0.0546	0.6044	6.9693	174.8884
P20	98.7849	81.8620	78.7878	0.2376	0.0599	0.3227	9.2921	259.4347
P21	46.5175	74.8370	82.2008	0.6156	0.0285	0.5130	11.9070	203.5553
P22	92.2576	39.6164	12.8681	0.7058	0.6546	0.9380	7.1994	144.7421
P23	4.7144	13.6746	21.3168	0.8489	0.6366	0.2967	12.7825	39.7057
P24	89.7816	99.2049	29.0928	0.3746	0.5041	0.5299	11.2155	218.0792
P25	64.0104	17.8060	1.8532	0.6503	0.9205	0.0675	6.1789	83.6695
P26	98.2358	33.8689	92.1147	0.0067	0.8942	0.4061	7.0102	224.2194
P27	51.5611	47.2383	12.2982	0.1539	0.4228	0.4801	10.1546	111.0977
P28	10.1299	29.2547	41.2962	0.5269	0.4596	0.2377	13.2456	80.6808
P29	58.6713	41.6475	4.4717	0.0976	0.6606	0.2235	8.8088	104.7904
P30	88.7380	51.0396	45.6699	0.0084	0.7491	0.7961	8.0628	185.4474

E.2 Individu hasil rekombinasi

C	Parent	x_1	x_2	x_3	σ_1	σ_2	σ_3	fitness	total gen
C1	P4 dan P21	52.699	41.464	75.844	0.781	0.567	0.240	8.760	170.007
C2	P9 dan P10	43.769	55.735	70.896	0.043	0.690	0.534	10.626	170.400
C3	P21 dan P5	31.572	68.566	90.060	0.713	0.734	0.935	12.564	190.198
C4	P22 dan P20	95.521	60.739	45.828	0.431	0.616	0.001	8.416	202.088



C5	P7 dan P23	27.997	12.104	43.667	0.271	0.988	0.698	7.462	83.768
C6	P5 dan P21	31.572	68.566	90.060	0.968	0.107	0.095	12.564	190.198
C7	P28 dan P9	10.420	64.615	56.616	0.385	0.971	0.617	21.382	131.651
C8	P21 dan P15	55.671	38.360	42.831	0.075	0.251	0.910	8.584	136.862
C9	P23 dan P3	46.741	32.250	51.357	0.962	0.478	0.788	8.472	130.347
C10	P13 dan P14	64.135	16.470	48.434	0.820	0.282	0.181	6.566	129.039
C11	P7 dan P8	74.806	42.778	57.048	0.670	0.365	0.699	8.005	174.633
C12	P2 dan P29	35.864	47.336	43.580	0.724	0.443	0.836	11.379	126.780
C13	P29 dan P7	54.975	26.090	35.245	0.708	0.708	0.326	7.532	116.310
C14	P14 dan P20	94.097	45.347	69.295	0.421	0.957	0.766	7.575	208.739
C15	P2 dan P24	51.419	76.115	55.891	0.589	0.962	0.613	12.536	183.425
C16	P15 dan P30	76.782	26.461	24.565	0.108	0.817	0.156	6.768	127.808
C17	P10 dan P1	49.634	46.798	73.333	0.230	0.860	0.200	9.365	169.765
C18	P8 dan P16	88.828	70.680	29.420	0.152	0.251	0.474	9.539	188.928
C19	P25 dan P3	76.389	34.315	41.625	0.888	0.835	0.090	7.393	152.329
C20	P23 dan P10	40.771	12.585	45.586	0.304	0.351	0.624	6.926	98.943
C21	P15 dan P13	51.843	12.995	20.264	0.997	0.005	0.644	6.316	85.102
C22	P10 dan P25	70.419	14.651	35.855	0.494	0.802	0.820	6.194	120.924
C23	P12 dan P14	49.385	19.558	61.976	0.451	0.071	0.943	7.288	130.919
C24	P2 dan P11	41.951	47.190	57.346	0.976	0.783	0.076	10.232	146.486
C25	P11 dan P3	79.806	46.090	56.700	0.135	0.311	0.244	8.040	182.595
C26	P24 dan P9	50.246	99.590	50.515	0.079	0.437	0.036	15.619	200.350
C27	P29 dan P27	55.116	44.443	8.385	0.881	0.536	0.435	9.453	107.944
C28	P1 dan P4	40.660	45.096	73.148	0.975	0.681	0.326	9.810	158.904
C29	P11 dan P29	64.758	41.501	18.237	0.776	0.947	0.965	8.542	124.496
C30	P12 dan P4	34.121	19.187	66.818	0.565	0.287	0.004	7.898	120.126
C31	P20 dan P1	60.613	81.981	77.799	0.066	0.310	0.478	11.439	220.393
C32	P18 dan P24	57.085	75.524	28.452	0.094	0.876	0.845	12.451	161.061
C33	P16 dan P7	65.301	38.435	38.390	0.325	0.317	0.087	8.120	142.126
C34	P16 dan P17	43.896	76.848	10.491	0.676	0.828	0.390	14.511	131.235
C35	P27 dan P4	55.221	27.665	40.892	0.193	0.002	0.003	7.665	123.778
C36	P27 dan P9	31.136	73.607	42.117	0.083	0.625	0.395	17.128	146.860
C37	P27 dan P3	70.164	49.031	46.848	0.645	0.398	0.704	8.681	166.043
C38	P17 dan P7	29.875	48.946	38.119	0.661	0.402	0.977	13.017	116.939
C39	P12 dan P11	40.103	35.819	48.076	0.454	0.357	0.458	9.324	123.998
C40	P7 dan P18	37.834	31.188	46.915	0.783	0.688	0.263	8.998	115.937
C41	P2 dan P6	44.285	31.895	44.586	0.385	0.406	0.227	8.639	120.767
C42	P29 dan P6	67.092	26.207	5.478	0.146	0.532	0.430	6.930	98.777
C43	P16 dan P27	65.442	56.788	11.530	0.509	0.560	0.664	9.809	133.759
C44	P17 dan P1	15.455	84.730	43.514	0.182	0.371	0.036	26.901	143.699
C45	P25 dan P21	55.264	46.321	42.027	0.427	0.847	0.272	9.373	143.612
C46	P26 dan P2	55.646	43.447	87.402	0.939	0.601	0.351	8.693	186.495
C47	P20 dan P5	57.706	72.078	88.353	0.757	0.501	0.059	10.616	218.138

C48	P13 dan P20	68.823	52.984	57.927	0.667	0.558	0.992	8.957	179.734
C49	P7 dan P1	36.860	46.317	71.414	0.801	0.157	0.303	10.262	154.590
C50	P20 dan P24	94.283	90.533	53.940	0.765	0.412	0.385	10.308	238.757
C51	P20 dan P9	54.747	90.918	75.362	0.472	0.993	0.831	12.905	221.028
C52	P2 dan P13	25.959	38.566	59.878	0.375	0.856	0.176	10.775	124.402
C53	P7 dan P25	57.645	14.170	33.935	0.386	0.644	0.526	6.423	105.750
C54	P24 dan P13	64.321	61.656	33.079	0.187	0.601	0.460	10.387	159.057
C55	P20 dan P18	61.587	66.853	53.300	0.005	0.450	0.439	10.635	181.739
C56	P9 dan P28	10.420	64.615	56.616	0.688	0.104	0.055	21.382	131.651
C57	P9 dan P20	54.747	90.918	75.362	0.260	0.263	0.698	12.905	221.028
C58	P13 dan P27	45.211	35.673	24.682	0.060	0.985	0.826	9.276	105.566
C59	P19 dan P25	77.664	25.546	26.069	0.502	0.958	0.642	6.691	129.279
C60	P22 dan P27	71.909	43.427	12.583	0.266	0.784	0.309	8.350	127.920
C61	P14 dan P6	82.461	9.799	33.143	0.728	0.998	0.825	5.678	125.403
C62	P15 dan P3	76.796	26.354	42.429	0.839	0.597	0.567	6.866	145.579
C63	P17 dan P8	53.402	81.191	29.149	0.869	0.874	0.378	13.432	163.742
C64	P10 dan P14	83.118	10.164	64.829	0.800	0.211	0.864	6.020	158.112
C65	P13 dan P20	68.823	52.984	57.927	0.770	0.091	0.111	8.957	179.734
C66	P16 dan P19	85.320	49.811	30.523	0.848	0.442	0.554	8.151	165.655
C67	P19 dan P12	50.340	31.784	57.217	0.161	0.192	0.635	8.213	139.341
C68	P27 dan P13	45.211	35.673	24.682	0.737	0.833	0.580	9.276	105.566
C69	P15 dan P17	36.648	44.621	6.840	0.027	0.284	0.967	11.634	88.109
C70	P5 dan P29	37.649	51.971	51.195	0.263	0.864	0.454	11.459	140.816
C71	P7 dan P18	37.834	31.188	46.915	0.080	0.829	0.850	8.998	115.937
C72	P9 dan P11	40.778	70.665	51.970	0.686	0.499	0.274	13.582	163.412
C73	P3 dan P16	84.044	58.581	46.080	0.339	0.443	0.506	8.735	188.705
C74	P11 dan P30	79.792	46.197	38.836	0.040	0.893	0.444	8.090	164.825
C75	P3 dan P10	82.797	31.160	75.626	0.846	0.746	0.378	7.135	189.584
C76	P25 dan P3	76.389	34.315	41.625	0.984	0.213	0.751	7.393	152.329
C77	P17 dan P8	53.402	81.191	29.149	0.399	0.488	0.568	13.432	163.742
C78	P23 dan P11	37.780	27.515	26.660	0.359	0.982	0.590	8.819	91.954
C79	P1 dan P18	23.414	66.972	52.311	0.423	0.439	0.147	17.033	142.697
C80	P4 dan P1	40.660	45.096	73.148	0.041	0.578	0.438	9.810	158.904
C81	P1 dan P16	50.881	74.219	43.786	0.183	0.469	0.116	12.990	168.886
C82	P14 dan P11	80.127	25.094	45.903	0.328	0.198	0.556	6.730	151.123
C83	P18 dan P19	57.853	42.564	39.048	0.874	0.777	0.801	8.877	139.466
C84	P26 dan P30	93.487	42.454	68.892	0.523	0.002	0.949	7.444	204.833
C85	P21 dan P3	67.642	62.831	81.799	0.551	0.885	0.668	9.478	212.272
C86	P8 dan P10	87.581	43.259	58.967	0.691	0.176	0.340	7.629	189.807
C87	P12 dan P11	40.103	35.819	48.076	0.476	0.480	0.408	9.324	123.998
C88	P13 dan P27	45.211	35.673	24.682	0.286	0.623	0.472	9.276	105.566
C89	P17 dan P29	33.571	64.503	7.345	0.697	0.923	0.406	15.449	105.419
C90	P26 dan P4	78.558	20.980	80.800	0.161	0.984	0.106	6.733	180.339

C91	P20 dan P16	89.054	74.099	44.775	0.236	0.647	0.188	9.583	207.928
C92	P27 dan P19	71.440	40.262	31.291	0.106	0.915	0.314	8.013	142.993
C93	P15 dan P6	70.169	6.324	4.972	0.047	0.244	0.603	5.122	81.466
C94	P26 dan P12	53.799	32.076	78.132	0.702	0.472	0.897	8.035	164.006
C95	P22 dan P5	54.442	50.955	55.394	0.507	0.754	0.646	9.662	160.791
C96	P12 dan P18	16.875	41.063	45.980	0.743	0.076	0.186	13.740	103.918
C97	P14 dan P21	67.963	41.835	71.002	0.674	0.220	0.186	8.162	180.800
C98	P4 dan P9	34.795	54.033	70.711	0.623	0.702	0.101	11.311	159.540
C99	P1 dan P7	36.860	46.317	71.414	0.556	0.352	0.093	10.262	154.590
C100	P5 dan P4	37.754	35.193	83.703	0.756	0.440	0.938	8.996	156.649
C101	P16 dan P12	44.342	48.310	37.456	0.411	0.154	0.448	10.689	130.107
C102	P7 dan P24	70.530	54.869	47.555	0.745	0.684	0.897	9.109	172.955
C103	P24 dan P3	89.274	75.015	55.245	0.214	0.382	0.652	9.512	219.534
C104	P1 dan P27	37.001	64.669	44.554	0.634	0.503	0.395	13.879	146.224
C105	P30 dan P27	70.150	49.139	28.984	0.498	0.986	0.262	8.840	148.273
C106	P30 dan P16	84.030	58.688	28.216	0.131	0.838	0.733	8.882	170.934
C107	P13 dan P9	24.786	62.041	54.501	0.081	0.529	0.619	15.241	141.328
C108	P12 dan P7	30.320	20.408	65.083	0.365	0.493	0.745	8.221	115.812
C109	P7 dan P22	71.768	25.075	39.443	0.901	0.293	0.556	6.895	136.286
C110	P28 dan P8	54.232	52.139	44.687	0.964	0.434	0.911	9.988	151.058
C111	P20 dan P11	84.815	61.608	55.395	0.203	0.407	0.354	8.839	201.818
C112	P28 dan P19	50.724	31.270	45.790	0.450	0.906	0.583	8.192	127.785
C113	P23 dan P5	10.671	37.984	59.618	0.284	0.063	0.319	13.299	108.273
C114	P7 dan P27	51.420	28.886	39.158	0.055	0.623	0.839	7.956	119.464
C115	P6 dan P24	82.647	54.985	17.788	0.697	0.564	0.833	8.749	155.421
C116	P28 dan P19	50.724	31.270	45.790	0.546	0.020	0.182	8.192	127.785
C117	P3 dan P9	49.738	75.400	76.667	0.085	0.708	0.010	11.834	201.805
C118	P12 dan P30	49.050	40.661	54.909	0.261	0.707	0.274	9.082	144.620
C119	P2 dan P22	52.657	46.321	47.779	0.387	0.347	0.512	9.471	146.756
C120	P5 dan P19	53.973	47.790	74.102	0.538	0.029	0.911	9.188	175.865

E.3 Individu Hasil mutasi

C	x_1	x_2	x_3	σ_1	σ_2	σ_3	fitness	total gen
C1	53.772	41.293	75.852	0.703	0.510	0.216	8.696	170.917
C2	43.744	54.641	70.206	0.039	0.621	0.481	10.537	168.592
C3	31.898	67.060	89.697	0.642	0.660	0.841	12.375	188.655
C4	95.349	60.624	45.827	0.388	0.554	0.001	8.415	201.800
C5	27.962	12.271	44.203	0.298	1.087	0.767	7.486	84.437
C6	33.381	68.547	89.988	0.871	0.096	0.085	12.355	191.916
C7	10.468	65.067	57.206	0.346	0.874	0.555	21.338	132.741
C8	55.646	38.297	42.385	0.068	0.226	0.819	8.583	136.328
C9	44.878	32.169	50.424	1.058	0.526	0.867	8.582	127.471



C10	63.585	16.573	48.553	0.902	0.310	0.199	6.586	128.711
C11	73.645	43.106	57.400	0.737	0.402	0.769	8.067	174.152
C12	37.136	47.429	44.153	0.652	0.399	0.752	11.197	128.717
C13	55.745	26.935	35.456	0.779	0.778	0.359	7.574	118.136
C14	94.304	44.788	68.495	0.379	0.861	0.689	7.541	207.586
C15	51.404	74.850	56.781	0.530	0.866	0.552	12.351	183.035
C16	76.825	26.956	24.306	0.119	0.898	0.171	6.801	128.087
C17	49.419	45.865	73.661	0.207	0.774	0.180	9.297	168.945
C18	88.899	70.588	29.280	0.137	0.226	0.426	9.529	188.766
C19	76.039	34.945	41.593	0.977	0.919	0.099	7.447	152.577
C20	40.921	12.510	45.736	0.274	0.316	0.562	6.914	99.167
C21	49.811	13.000	20.981	1.097	0.005	0.708	6.389	83.791
C22	70.832	15.293	35.757	0.543	0.883	0.902	6.228	121.882
C23	49.064	19.643	62.345	0.497	0.078	1.037	7.307	131.052
C24	41.595	45.605	57.394	0.878	0.704	0.068	10.091	144.594
C25	79.900	45.988	56.858	0.122	0.280	0.220	8.030	182.746
C26	50.309	99.321	50.500	0.071	0.393	0.033	15.585	200.129
C27	56.427	45.272	8.039	0.793	0.483	0.392	9.415	109.739
C28	39.613	45.699	72.551	1.073	0.749	0.358	9.955	157.863
C29	62.978	42.646	19.530	0.854	1.041	1.062	8.762	125.154
C30	33.868	19.391	66.816	0.622	0.316	0.004	7.929	120.076
C31	60.583	81.665	78.108	0.059	0.279	0.430	11.406	220.356
C32	57.101	75.909	28.273	0.103	0.964	0.930	12.480	161.284
C33	64.695	38.558	38.396	0.358	0.349	0.096	8.158	141.648
C34	42.909	75.144	10.205	0.744	0.910	0.429	14.514	128.258
C35	55.362	27.668	40.898	0.174	0.002	0.002	7.659	123.928
C36	31.101	73.977	42.122	0.092	0.687	0.435	17.187	147.200
C37	70.155	48.369	46.938	0.580	0.359	0.634	8.630	165.462
C38	28.773	48.573	38.676	0.727	0.442	1.075	13.137	116.022
C39	39.901	35.816	48.010	0.499	0.393	0.504	9.341	123.727
C40	38.258	31.267	46.722	0.704	0.620	0.237	8.977	116.247
C41	44.706	32.037	44.624	0.347	0.366	0.205	8.626	121.367
C42	67.053	25.636	6.407	0.131	0.478	0.387	6.883	99.096
C43	65.714	57.159	11.711	0.560	0.616	0.731	9.823	134.583
C44	15.377	84.931	43.498	0.201	0.408	0.039	27.019	143.806
C45	55.321	45.781	42.238	0.385	0.762	0.245	9.313	143.340
C46	56.470	43.369	87.886	0.845	0.541	0.316	8.650	187.726
C47	57.519	72.597	88.402	0.833	0.552	0.065	10.667	218.518
C48	68.774	53.348	56.826	0.734	0.613	1.091	8.996	178.948
C49	36.535	46.039	71.922	0.721	0.142	0.273	10.249	154.496
C50	93.711	90.361	53.544	0.842	0.453	0.424	10.335	237.616
C51	54.010	91.086	74.121	0.519	1.092	0.914	13.059	219.217
C52	25.999	39.245	60.017	0.412	0.941	0.194	10.857	125.261

C53	57.224	15.336	34.332	0.424	0.708	0.579	6.532	106.893
C54	64.298	61.733	32.505	0.206	0.661	0.506	10.411	158.535
C55	61.594	67.225	53.514	0.005	0.495	0.483	10.664	182.333
C56	10.911	64.764	56.609	0.619	0.093	0.049	21.136	132.284
C57	54.538	91.119	75.883	0.286	0.290	0.768	12.927	221.540
C58	45.271	36.127	25.127	0.066	1.084	0.908	9.321	106.525
C59	77.329	25.170	25.594	0.452	0.862	0.578	6.670	128.093
C60	71.804	43.647	11.989	0.293	0.862	0.340	8.382	127.440
C61	82.268	10.937	32.998	0.801	1.098	0.908	5.745	126.203
C62	76.288	25.780	42.988	0.755	0.538	0.511	6.845	145.056
C63	54.605	80.725	29.068	0.782	0.787	0.340	13.222	164.398
C64	83.189	9.948	65.676	0.720	0.190	0.777	6.016	158.813
C65	69.293	53.021	58.038	0.693	0.082	0.100	8.936	180.352
C66	84.236	49.601	30.435	0.933	0.486	0.609	8.180	164.271
C67	50.430	31.625	55.914	0.145	0.173	0.571	8.200	137.970
C68	45.397	36.624	24.366	0.811	0.916	0.637	9.391	106.387
C69	36.627	44.294	8.460	0.030	0.312	1.064	11.639	89.382
C70	37.360	51.113	50.801	0.237	0.777	0.408	11.397	139.273
C71	37.698	31.916	46.516	0.088	0.912	0.935	9.098	116.130
C72	40.857	70.218	52.232	0.617	0.449	0.246	13.485	163.306
C73	84.278	58.104	45.684	0.305	0.399	0.456	8.694	188.066
C74	79.805	46.425	38.191	0.044	0.982	0.488	8.107	164.420
C75	84.099	32.765	75.247	0.930	0.821	0.416	7.186	192.111
C76	76.938	34.232	42.221	0.886	0.192	0.675	7.372	153.390
C77	53.255	80.732	28.990	0.359	0.439	0.511	13.410	162.977
C78	37.675	28.399	27.386	0.395	1.080	0.649	8.942	93.460
C79	23.421	66.562	52.392	0.381	0.395	0.133	16.920	142.374
C80	40.686	44.310	73.284	0.037	0.520	0.394	9.733	158.279
C81	50.749	72.986	43.906	0.165	0.422	0.104	12.837	167.641
C82	79.674	25.236	45.741	0.361	0.218	0.612	6.748	150.651
C83	57.700	42.829	37.457	0.962	0.854	0.881	8.928	137.987
C84	93.933	42.456	69.173	0.471	0.002	0.854	7.434	205.562
C85	67.931	62.979	81.850	0.496	0.797	0.601	9.473	212.760
C86	88.619	43.371	59.198	0.622	0.158	0.306	7.607	191.188
C87	39.657	35.656	48.248	0.523	0.528	0.449	9.339	123.561
C88	45.438	36.286	24.931	0.315	0.685	0.520	9.331	106.655
C89	33.978	64.593	7.017	0.627	0.831	0.366	15.338	105.588
C90	78.537	21.911	80.716	0.177	1.082	0.116	6.779	181.164
C91	89.158	73.796	44.747	0.212	0.582	0.169	9.556	207.701
C92	71.432	38.387	31.422	0.095	0.824	0.283	7.864	141.240
C93	70.198	6.608	4.639	0.052	0.268	0.664	5.137	81.445
C94	54.364	32.043	79.278	0.631	0.425	0.808	8.010	165.685
C95	54.848	50.319	56.010	0.456	0.679	0.582	9.567	161.177

C96	18.503	41.020	46.000	0.669	0.068	0.167	13.303	105.524
C97	67.726	41.844	71.289	0.742	0.242	0.205	8.170	180.859
C98	34.343	55.572	70.589	0.686	0.772	0.111	11.542	160.504
C99	36.025	46.914	71.469	0.612	0.387	0.102	10.394	154.408
C100	37.202	35.559	82.954	0.832	0.484	1.032	9.066	155.715
C101	44.438	48.410	37.472	0.452	0.169	0.493	10.690	130.321
C102	69.917	54.412	46.115	0.820	0.752	0.986	9.121	170.445
C103	89.409	75.406	55.340	0.236	0.420	0.717	9.531	220.155
C104	36.859	64.691	44.773	0.697	0.554	0.434	13.893	146.323
C105	69.519	49.655	29.333	0.547	1.085	0.288	8.919	148.507
C106	84.131	58.850	27.269	0.145	0.922	0.806	8.898	170.251
C107	24.687	61.898	55.309	0.073	0.476	0.557	15.147	141.895
C108	29.513	20.038	65.105	0.402	0.542	0.820	8.232	114.656
C109	70.800	25.247	39.924	0.991	0.323	0.612	6.936	135.971
C110	53.028	51.543	44.298	1.060	0.477	1.002	10.034	148.868
C111	84.931	62.024	55.363	0.224	0.447	0.390	8.861	202.318
C112	51.383	31.489	45.616	0.405	0.815	0.525	8.178	128.488
C113	11.086	37.998	59.486	0.255	0.057	0.287	13.214	108.570
C114	51.471	27.976	40.495	0.049	0.560	0.755	7.866	119.942
C115	82.807	55.169	16.973	0.767	0.620	0.916	8.764	154.950
C116	49.964	31.260	45.914	0.601	0.022	0.201	8.231	127.138
C117	49.647	75.797	76.653	0.093	0.779	0.011	11.882	202.097
C118	49.052	41.697	55.095	0.287	0.777	0.302	9.176	145.844
C119	52.781	46.472	47.294	0.426	0.382	0.564	9.484	146.547
C120	53.932	47.813	74.294	0.591	0.032	1.002	9.190	176.039

E.4 Individu Hasil Seleksi Generasi ke 1

C	Parent	x_1	x_2	x_3	σ_1	σ_2	σ_3	fitness	total gen
P1	P17	8.470	87.359	10.219	0.933	0.550	0.328	33.758	106.048
P2	P9	10.710	99.975	71.937	0.945	0.445	0.734	29.763	182.621
P3	C44	15.377	84.931	43.498	0.201	0.408	0.039	27.019	143.806
P4	C7	10.468	65.067	57.206	0.346	0.874	0.555	21.338	132.741
P5	C56	10.911	64.764	56.609	0.619	0.093	0.049	21.136	132.284
P6	C36	31.101	73.977	42.122	0.092	0.687	0.435	17.187	147.200
P7	P1	22.440	82.101	76.810	0.648	0.333	0.640	17.163	181.350
P8	C79	23.421	66.562	52.392	0.381	0.395	0.133	16.920	142.374
P9	P18	24.388	51.843	27.812	0.246	0.859	0.386	16.212	104.043
P10	C26	50.309	99.321	50.500	0.071	0.393	0.033	15.585	200.129
P11	C89	33.978	64.593	7.017	0.627	0.831	0.366	15.338	105.588
P12	C107	24.687	61.898	55.309	0.073	0.476	0.557	15.147	141.895
P13	C34	42.909	75.144	10.205	0.744	0.910	0.429	14.514	128.258
P14	C104	36.859	64.691	44.773	0.697	0.554	0.434	13.893	146.323



P15	P2	13.057	53.025	82.689	0.463	0.537	0.528	13.666	148.770
P16	C72	40.857	70.218	52.232	0.617	0.449	0.246	13.485	163.306
P17	P5	16.627	62.294	97.919	0.903	0.154	0.363	13.420	176.841
P18	C77	53.255	80.732	28.990	0.359	0.439	0.511	13.410	162.977
P19	C96	18.503	41.020	46.000	0.669	0.068	0.167	13.303	105.524
P20	P28	10.130	29.255	41.296	0.527	0.460	0.238	13.246	80.681
P21	C63	54.605	80.725	29.068	0.782	0.787	0.340	13.222	164.398
P22	C113	11.086	37.998	59.486	0.255	0.057	0.287	13.214	108.570
P23	C38	28.773	48.573	38.676	0.727	0.442	1.075	13.137	116.022
P24	C51	54.010	91.086	74.121	0.519	1.092	0.914	13.059	219.217
P25	C57	54.538	91.119	75.883	0.286	0.290	0.768	12.927	221.540
P26	C81	50.749	72.986	43.906	0.165	0.422	0.104	12.837	167.641
P27	P23	4.714	13.675	21.317	0.849	0.637	0.297	12.783	39.706
P28	C32	57.101	75.909	28.273	0.103	0.964	0.930	12.480	161.284
P29	C3	31.898	67.060	89.697	0.642	0.660	0.841	12.375	188.655
P30	C6	33.381	68.547	89.988	0.871	0.096	0.085	12.355	191.916

E.5 Individu Hasil Seleksi Generasi ke 500

P+1	P	x_1	x_2	x_3	σ_1	σ_2	σ_3	fitness	total gen
P1	P1	1.000	87.649	33.758	0.744	0.966	0.868	76.872	122.407
P2	P2	1.000	87.518	33.703	0.486	0.025	0.003	76.869	122.221
P3	P3	1.000	88.307	34.016	0.669	0.777	0.238	76.867	123.323
P4	P4	1.000	87.329	33.624	0.774	0.165	0.129	76.867	121.953
P5	P5	1.000	87.388	33.646	0.390	0.852	0.311	76.865	122.033
P6	P6	1.000	87.055	33.511	0.507	0.141	0.912	76.856	121.566
P7	P7	1.000	87.032	33.501	0.216	1.053	1.092	76.855	121.533
P8	C89	1.000	87.072	33.519	0.188	0.120	0.112	76.854	121.591
P9	P8	1.000	87.000	33.488	0.738	0.031	0.016	76.850	121.488
P10	P9	1.000	87.427	33.659	0.091	0.454	0.101	76.849	122.086
P11	P10	1.000	86.848	33.427	0.381	0.418	0.595	76.845	121.274
P12	P11	1.000	87.173	33.556	0.666	0.282	0.112	76.842	121.729
P13	P12	1.000	87.089	33.531	0.234	0.023	0.057	76.842	121.620
P14	P13	1.000	87.204	33.579	0.858	0.145	0.704	76.841	121.783
P15	P14	1.000	86.829	33.421	0.227	0.002	0.044	76.838	121.250
P16	P15	1.000	86.759	33.392	0.989	0.212	0.993	76.836	121.152
P17	P16	1.000	86.715	33.374	0.664	0.095	0.139	76.835	121.089
P18	P17	1.000	87.619	33.759	0.380	0.769	0.519	76.835	122.378
P19	P18	1.000	86.796	33.408	0.410	0.760	0.044	76.834	121.204
P20	P19	1.000	87.270	33.611	0.157	0.741	0.382	76.833	121.881
P21	P20	1.000	87.013	33.501	0.467	0.126	0.136	76.833	121.515
P22	P21	1.000	86.651	33.345	0.617	0.053	0.833	76.832	120.996
P23	P22	1.000	86.679	33.360	0.345	0.217	0.602	76.831	121.038



P24	P23	1.000	87.102	33.526	0.675	0.022	0.086	76.831	121.628
P25	P24	1.000	87.182	33.558	1.021	1.095	0.656	76.828	121.741
P26	P25	1.000	87.206	33.585	0.504	0.534	0.197	76.828	121.791
P27	P26	1.000	87.435	33.683	0.313	0.051	0.116	76.828	122.117
P28	P27	1.000	86.449	33.263	0.652	0.056	0.603	76.824	120.712
P29	P28	1.000	86.387	33.238	0.646	0.747	0.397	76.824	120.625
P30	P29	1.000	86.471	33.274	1.010	0.724	0.310	76.823	120.744

