

OPTIMASI BIAYA DAN ASUPAN GIZI PASIEN DIET KHUSUS DENGAN MENGGUNAKAN ALGORITMA *EVOLUTION* *STRATEGIES*

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Melly Yansari
NIM: 125150200111121



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

PENGESAHAN

OPTIMASI BIAYA DAN ASUPAN GIZI PASIEN DIET KHUSUS DENGAN
MENGUNAKAN ALGORITMA *EVOLUTION STRATEGIES*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana
Komputer

Disusun Oleh :

Melly Yansari

NIM: 125150200111121

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

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

NIP: 197306192002122001

Drs. Marji, M.T

NIP: 196708011992031001

Mengetahui

Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D

NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS SKRIPSI

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

Apabila ternyata dalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang 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 Juli 2016

Melly Yansari

NIM: 125150200111121

KATA PENGANTAR

Dengan menyebut nama Alla SWT Yang Maha Pengasih dan Maha Penyayang. Puji syukur kehadiran Allah SWT karena limpahan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul “Optimasi Biaya Dan Asupan Gizi Pasien Diet Khusus Dengan Menggunakan Algoritma *Evolution Strategies*”. Shalawat serta salam senantiasa tercurahkan kepada junjungan Nabi besar kita Nabi Muhammad SAW beserta keluarga dan para sahabatnya. Skripsi ini disusun untuk memenuhi salah satu syarat memperoleh gelar Sarjana Komputer di Fakultas Ilmu Komputer di Universitas Brawijaya Malang (Filkom UB).

Pada kesempatan ini penulis juga ingin menyampaikan terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan bantuan dan dukungan dalam menyelesaikan skripsi ini, antara lain :

1. Dian Eka Ratnawati, S.Si., M.Kom dan Drs. Marji. M.T, selaku Dosen Pembimbing yang telah membimbing, mengarahkan dan memberikan banyak saran kepada penulis sehingga dapat menyelesaikan skripsi ini.
2. Agus Wahyu Widodo, S.T, M.Cs selaku Ketua Program Studi Informatika/Illmu Komputer Universitas Brawijaya.
3. Seluruh Dosen PTIIK UB yang telah membagikan ilmunya kepada penulis selama masa perkuliahan.
4. Seluruh Civitas Akademika PTIIK UB yang telah memberikan dukungan dan bantuan selama menempuh studi di Informatika/Illmu Komputer Universitas Brawijaya dan selama penyelesaian skripsi ini.
5. Kedua Orang Tua Penulis Syahrial dan Susi Yanti serta keluarga besar atas segala do’a, nasihat, dukungan baik moril maupun materil yang begitu besar terhadap kelancaran dalam menyelesaikan skripsi ini.
6. Mas Karid Nurvenus yang telah bersedia memberikan referensi/sumber data penelitian kepada penulis.

7. Sahabat serta teman-teman penulis, Ahmad Leo Yudanto, Ziya, Reza, Julio, Zata dan seluruh teman-teman TIF-F, serta teman-teman angkatan 2012, angkatan 2013, dan angkatan 2014 yang selalu memberikan bantuan, dukungan, motivasi dan berbagi informasi demi kelancaran skripsi.

Dengan kerendahan hati, penulis menyadari bahwa skripsi ini masih memiliki banyak kekurangan. Oleh karena itu kritik dan saran yang bersifat konstruktif sangat dibutuhkan sebagai pedoman untuk menyempurnakan skripsi ini agar lebih baik. Penulis berharap semoga skripsi ini dapat bermanfaat bagi diri sendiri maupun bagi semua pihak.

Malang, 28 Juli 2016

Penulis

mellyyansari@yahoo.com



ABSTRAK

Harga bahan makanan di Indonesia saat ini terus meningkat. Sedangkan masyarakat Indonesia harus pandai menyiasati makanan harian mereka dengan kecukupan gizi yang terpenuhi namun dengan biaya yang optimal juga. Pemenuhan gizi yang baik adalah gizi yang terpenuhi seimbang dimana tidak berlebihan maupun tidak kekurangan. Karena itu, diperlukan perhitungan gizi setiap individunya besar gizi yang dibutuhkan. Dalam optimasi kebutuhan gizi ini bukan hanya dialami orang sehat saja. Orang yang menjalani diet khusus juga memerlukan pemenuhan gizi harian mereka. Diet khusus disini adalah orang-orang penderita diabetes mellitus dimana harus memperhatikan kandungan karbohidratnya, penderita jantung yang harus memperhatikan asupan lemaknya dan penderita hati yang harus memperhatikan asupan proteinnya.

Untuk mengatasi permasalahan tersebut, dapat diatasi dengan mengganti atau mengkombinasikan bahan makanan yang mahal dengan bahan makanan yang murah dengan tetap menjaga asupan gizi yang sesuai kebutuhan. Pada penelitian ini digunakan metode algoritma *evolution strategies* dengan menerapkan pengkodean *floating point*, *mutation*, dan *elitism selection* untuk membantu menentukan kombinasi berat bahan makanan yang memenuhi kebutuhan gizi berdasarkan data pasien, komposisi bahan makanan, dan batasan biaya yang dimasukkan pengguna. Terdapat 118 bahan makanan yang digunakan dengan kandungan gizi berupa karbohidrat, protein, dan lemak. Dimana 118 bahan makanan ini dibagi-bagi berdasarkan jenis penyakit. Untuk metode reproduksi yang digunakan adalah *mutation*. Sedangkan metode seleksi yang digunakan adalah *elitism selection*. Dari hasil pengujian parameter algoritma *evolution strategies* diperoleh ukuran populasi terbaik sebesar 200 individu, jumlah generasi terbaik sebanyak 250 generasi, dan banyak *offspring* sebanyak 10. Berdasarkan uji coba terhadap 4 studi kasus, diketahui bahwa sistem dapat menghasilkan hasil komposisi bahan makanan yang masih dalam batas toleransi kecukupan gizi pasien diet khusus.

ABSTRACT

Food prices in Indonesia today continues to increase. While the Indonesian people must be good at their daily meals with nutritional adequacy are met but with an optimal cost as well. Good nutrition is balanced nutrition fulfilled which is not excessive and does not lack. Therefore, the necessary nutrients each individual calculation major nutrients needed. In the optimization of nutritional requirements is not only experienced healthy people only. People who undergo a special diet also requires the fulfillment of their daily nutrition. Special Diet here are people with diabetes mellitus which should memeperhatikan carbohydrate content, heart patients should pay attention to fat intake and heart patients should pay attention to the protein intake.

To overcome these problems, can be solved by replacing or combining groceries are expensive with inexpensive foodstuffs while maintaining the nutritional intake as needed. In this study used a method evolution algorithm coding strategies to implement floating point, uniform mutation, and elitism selection to help determine the combined weight of foodstuffs which meet the nutritional needs based on patient data, food composition, and cost limit entered by the user. There are 118 food ingredients that are used with the nutrients such as carbohydrates, protein, and fat. 118 where this food is broken down by type of disease. For reproduction method used is the mutation rate. While the selection method used is elitism selection. From the test results obtained by the genetic algorithm parameter best population size of 200 individuals, the number of the best generation of 250 generations, and many offspring as many as 7. Based on the trial against the four case studies, it is known that the system can produce food composition is still within the tolerance limits adequacy nutritional patient special diets.



DAFTAR ISI

LEMBAR PENGESAHAN	ii
PERNYATAAN ORISINALITAS SKRIPSI	iii
KATA PENGANTAR.....	iv
ABstrak	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xii
DAFTAR PERSAMAAN.....	xiii
DAFTAR LAMPIRAN	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan masalah	2
1.6 Sistematika pembahasan.....	3
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI.....	4
2.1 Tinjauan Penelitian Terdahulu.....	4
2.2 Diet Pada Penderita Penyakit Diabetes Melitus	5
2.3 Diet Pada Penyakit Jantung	6
2.4 Diet Pada Penyakit Hati	7
2.5 Diet Pada Orang Sehat.....	8
2.6 Gizi.....	8
2.6.1 Energi.....	9
2.6.2 Protein.....	9
2.6.3 Lemak	10
2.6.4 Karbohidrat	10
2.7 Angka Kebutuhan Gizi	10
2.7.1 Kebutuhan Energi.....	11
2.7.2 Kebutuhan Karbohidrat, Lemak, dan Protein	12

2.7.3 Kebutuhan Gizi Seimbang	12
2.8 <i>Evolution Strategies</i>	12
2.9 Algoritma Evolusi	13
2.9.1 Struktur Dasar <i>Evolution Startegies (ES)</i>	13
2.9.2 Representasi Kromosom	15
2.9.3 Inisialisasi	15
2.9.4 Reproduksi	15
2.9.5 Rekombinasi	15
2.9.6 Mutasi	16
2.9.7 Perhitungan <i>Fitness</i>	17
2.9.8 Seleksi	18
BAB III METODOLOGI PENELITIAN	19
3.1 Tahapan Penelitian	19
3.2 Data yang digunakan	20
3.3 Algoritma yang digunakan	20
3.4 Kebutuhan Sistem	20
BAB IV PERANCANGAN SISTEM	21
4.1 Formulasi Permasalahan	21
4.2 Siklus Penyelesaian Masalah	21
4.2.1 Generate Populasi Awal	24
4.2.2 Reproduksi	25
4.2.3 Penalti Gizi	27
4.2.4 Penalti Harga	30
4.2.5 Menghitung Nilai <i>Fitness</i>	32
4.2.6 Seleksi	32
4.3 Perhitungan Manual	33
4.3.1 Parameter Algoritma <i>Evolution Strategies</i>	34
4.3.2 Generate Populasi Awal	34
4.3.3 Reproduksi	35
4.3.4 Perhitungan Penalti Gizi	36
4.3.5 Penalti Harga	39
4.3.6 <i>Fitness</i>	39

4.3.7 Seleksi.....	40
4.2 Perancangan User Interface	41
4.2.1 Rancangan Halaman Beranda	42
4.2.2 Rancangan Halaman Data dan Proses Algoritma Evolution Startegies.....	43
4.3 Perancangan Uji Coba dan Evaluasi.....	44
4.3.1 Rancangan Uji Coba Ukuran Populasi (μ) dan Offspring (λ)	44
4.3.2 Rancangan Uji Coba Banyak Generasi	44
BAB V IMPLEMENTASI.....	46
5.1 Lingkungan Implentasi	46
5.1.1 Lingkungan Perangkat Keras	46
5.1.2 Lingkungan Perangkat Lunak	46
5.2 Implementasi Sistem	46
5.2.1 Mengambil Data Bahan Makanan	46
5.2.2 Implementasi Inisialisasi Populasi Awal.....	48
5.2.3 Implementasi Mutasi	48
5.2.4 Perhitungan Fitness.....	50
5.2.5 Implementasi Seleksi dengan Elitism.....	53
BAB VI PENGUJIAN DAN ANALISIS	54
6.1 Hasil dan Analisis Uji Coba Ukuran Populasi (μ) dan Offspring (λ).....	54
6.2 Hasil dan Analisis Uji Coba Banyak Generasi	56
6.3 Hasil dan Analisa Pengujian Sistem	58
6.3.1 Kasus 1.....	58
6.3.2 Kasus 2.....	60
6.3.3 Kasus 3.....	62
BAB VII PENUTUP	64
7.1 Kesimpulan.....	64
7.2 Saran	64
DAFTAR PUSTAKA.....	66
LAMPIRAN A DAFTAR BAHAN MAKANAN.....	69
LAMPIRAN B MANUALISASI	70

DAFTAR TABEL

Tabel 2. 1 Perbedaan Penelitian Sebelumnya dengan Penelitian Penulis.....	4
Tabel 2.2 Cara menaksir kebutuhan energi menurut aktivitas dengan menggunakan kelipatan AMB	12
Tabel 2. 3 Contoh Representasi Kromosom.....	15
Tabel 2. 4 Contoh Rekombinasi	16
Tabel 2. 5 Contoh Mutasi	17
Tabel 2. 6 Contoh Seleksi	18
Tabel 4. 1 Tabel bahan makanan yang dipilih.....	34
Tabel 4. 2 Individu pada populasi awal.....	35
Tabel 4. 3 Individu Hasil Mutasi	36
Tabel 4. 4 Nilai Penalti Gizi.....	38
Tabel 4. 5 Nilai Penalti Harga	39
Tabel 4. 6 Nilai Fitness.....	40
Tabel 4. 7 Individu Gabungan	40
Tabel 4. 8 Individu gabungan berdasarkan nilai fitness terbesar	40
Tabel 4. 9 Individu yang lolos ke generasi berikutnya	41
Tabel 4. 10 Populasi Baru	41
Tabel 4. 11 Rancangan Uji Coba Banyaknya Populasi.....	44
Tabel 4. 12 Tabel Rancangan Uji Coba Ukuran <i>Offspring</i>	44
Tabel 4. 13 Rancangan Uji Coba Ukuran Generasi.....	45
Tabel 6. 1 Hasil Coba Ukuran Populasi.....	54
Tabel 6. 2 Hasil Coba Ukuran <i>Offspring</i>	55
Tabel 6. 3 Hasil Coba Banyaknya Generasi	57
Tabel 6. 4 Menu Kasus 1	58
Tabel 6. 5 Hasil Pemenuhan Gizi dari Sistem untuk Menu Kasus 1	59
Tabel 6. 6 Menu Kasus 2	60
Tabel 6. 7 Hasil Pemenuhan Gizi dari Sistem untuk Menu Kasus 2	61
Tabel 6. 8 Menu Kasus 3	62
Tabel 6. 9 Hasil Pemenuhan Gizi dari Sistem untuk Menu Kasus 3	63
Tabel 6. 10 Menu Kasus 4	Error! Bookmark not defined.

Tabel 6. 11 Hasil Pemenuhan Gizi dari Sistem untuk Menu Kasus 4Error!
Bookmark not defined.

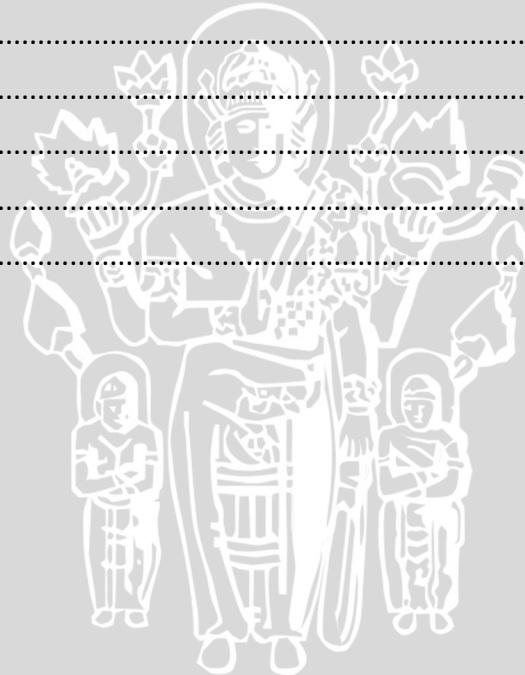


DAFTAR GAMBAR

Gambar 2. 1 <i>Pseudocode</i> Algoritma <i>Evolution Strategies</i>	14
Gambar 3. 1 Tahapan Penelitian.....	19
Gambar 4. 1 <i>Flowchart</i> Optimasi asupan gizi	22
Gambar 4. 2 <i>Flowchart</i> Proses Pembentukan Susunan kromosom optimal	23
Gambar 4. 3 <i>Flowchart</i> Proses <i>Generate</i> Populasi Awal.....	24
Gambar 4. 4 <i>Flowchart</i> Proses Reproduksi	26
Gambar 4. 5 <i>Flowchart</i> Perhitungan Penalti Gizi Jenis	29
Gambar 4. 6 <i>Flowchart</i> Penalti Harga	31
Gambar 4. 7 <i>Flowchart</i> Perhitungan <i>Fitness</i>	32
Gambar 4. 8 <i>Flowchart</i> Proses Seleksi	33
Gambar 4. 9 Gambar Halaman Utama.....	42
Gambar 4. 10 Halaman Data dan Proses ES.....	43
Gambar 5. 1 Source Code Mengambil Data Makanan.....	47
Gambar 5. 2 Source Code Inisialisasi Populasi Awal.....	48
Gambar 5. 3 Source Code Proses Mutasi.....	49
Gambar 5. 4 Source Code Proses Perhitungan <i>Fitness</i>	51
Gambar 5. 5 Source Code Proses Seleksi.....	53
Gambar 6. 1 Grafik Pengaruh Ukuran Populasi (<i>miu</i>).....	55
Gambar 6. 2 Grafik Pengaruh Ukuran <i>Offspring</i> (<i>lambda</i>)	56
Gambar 6. 3 Grafik Pengaruh Ukuran generasi	57

DAFTAR PERSAMAAN

Persamaan 2- 1.....	16
Persamaan 2- 2.....	16
Persamaan 2- 3.....	16
Persamaan 2- 4.....	17
Persamaan 2- 5.....	17
Persamaan 2- 6.....	39
Persamaan 4- 1.....	27
Persamaan 4- 2.....	27
Persamaan 4- 3.....	27
Persamaan 4- 4.....	27
Persamaan 4- 5.....	27
Persamaan 4- 6.....	27
Persamaan 4- 7.....	28
Persamaan 4- 8.....	30
Persamaan 4- 9.....	30



DAFTAR LAMPIRAN

LAMPIRAN A DAFTAR BAHAN MAKANAN.....	69
LAMPIRAN B MANUALISASI	70



BAB I PENDAHULUAN

Pada bab ini berisi latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

1.1 Latar belakang

Harga bahan makanan di Indonesia saat ini terus meningkat. Sedangkan masyarakat Indonesia harus pandai menyiasati makanan harian mereka dengan kecukupan gizi yang terpenuhi namun dengan biaya yang optimal juga. Pemenuhan gizi yang baik adalah gizi yang terpenuhi seimbang dimana tidak berlebihan maupun tidak kekurangan. Karena itu, diperlukan perhitungan gizi setiap individunya yang dibutuhkan. Dalam optimasi kebutuhan gizi ini bukan hanya dialami orang sehat saja. Orang yang menjalani diet khusus juga memerlukan pemenuhan gizi harian mereka. Diet khusus disini adalah orang-orang penderita diabetes mellitus dimana harus memperhatikan kandungan karbohidratnya, penderita jantung yang harus memperhatikan asupan lemaknya dan penderita hati yang harus memperhatikan asupan proteinnya.

Langkah optimasi ini diharapkan mampu mengatasi permasalahan-permasalahan yang dialami pasien diet khusus. Tujuan optimasi ini membantu menentukan kombinasi berat bahan makanan yang memenuhi kebutuhan gizi berdasarkan komposisi berat bahan makanan dan batasan yang diinginkan. Salah satu algoritma yang bisa dipakai untuk melakukan optimasi ialah dengan menggunakan Algoritma Evolusi. Terdapat beberapa algoritma yang termasuk dalam algoritma Evolusi antara lain *Genetic Algorithm*, *Genetic Programming*, *Evolution Strategies*, *Differential Evolution*, *Evolutionary Programming*, dan *Grammatical Evolution*.

Peneliti dalam penelitian ini adalah dengan menggunakan algoritma optimasi *Evolution Strategies*. Pemilihan algoritma ini disebabkan karena kecepatan proses yang dilakukan lebih baik jika dibandingkan dengan *Genetic Algorithm*. Algoritma *evolution strategies* (ES) merupakan cabang dari *evolution algorithm* yang baik digunakan untuk optimasi. ES merupakan algoritma berbasis populasi yang sangat sesuai untuk permasalahan kompleks yang sulit diselesaikan dengan pendekatan analisis berbasis model matematis (Mahmudy, WF, Marian, RM & Luong, 2013). Algoritma *evolution strategies* pernah digunakan untuk penyelesaian *Vehicle Routing Problem* pada distribusi minuman soda XYZ. Penelitian yang dilakukan oleh Harun (2014) menunjukkan ukuran populasi (μ) = 80, ukuran *offspring* (λ) = 2μ , dan jumlah generasi 2000 dapat menyelesaikan masalah VRPTW dengan waktu kurang lebih 30 detik.

Ada salah satu penelitian yang membahas optimasi asupan gizi pasien diet yaitu penelitian yang dilakukan oleh Karid (2015) menunjukkan bahwa algoritma genetik dapat digunakan untuk menyelesaikan permasalahan optimasi asupan gizi pasien diet. Dalam penelitian tersebut memakai data bahan makanan Daftar Komposisi Bahan Makanan (DKBM) Indonesia yang diterbitkan oleh Depkes 2005.

Berdasarkan permasalahan di atas, solusi yang ditawarkan adalah dengan mengoptimasi biaya dan asupan gizi pasien diet menggunakan algoritma *evolution strategies*. Penelitian ini bertujuan untuk memaksimalkan asupan gizi pada pasien diet dengan biaya yang minimal melalui optimasi komposisi berat bahan makanan.

1.2 Rumusan masalah

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

1. Bagaimana mengimplementasikan algoritma *evolution strategies* untuk optimasi biaya dan asupan gizi pasien diet khusus ?
2. Bagaimana menentukan nilai-nilai parameter *evolution strategies* yang tepat ?

1.3 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini adalah :

1. Untuk mengimplementasikan algoritma *evolution strategies* dalam menyelesaikan masalah optimasi biaya dan asupan gizi pasien diet khusus.
2. Untuk menentukan nilai-nilai parameter *evolution strategies* yang tepat.

1.4 Manfaat

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

- Bagi Penulis
Penulis mendapatkan pemahaman tentang implementasi algoritma *evolution strategies* untuk optimasi biaya dan asupan gizi pasien diet khusus.
- Bagi Pengguna
Memberikan solusi dalam menghemat biaya dan mengoptimalkan asupan gizi pasien diet khusus.

1.5 Batasan masalah

Untuk menghindari meluasnya masalah yang akan dibahas, maka diberikan batasan masalah sebagai berikut :

1. Sistem dirancang untuk diaplikasikan pada orang-orang sehat dan yang memiliki penyakit diabetes mellitus, jantung, hati dimana bahan makanan nya perlu di kontrol.

2. Faktor yang mempengaruhi dibatasi jenis kelamin, usia, tinggi badan, berat badan.
3. Kebutuhan nutrisi yang digunakan dalam perhitungan adalah jumlah kalori dan macronutrient (protein, lemak dan karbohidrat).
4. Terdapat 118 macam bahan makanan yang dapat dioptimasi.
5. Biaya pengeluaran optimum ditentukan dari komposisi jumlah makanan yang telah dipilih

1.6 Sistematika pembahasan

Sistematika yang digunakan untuk menyusun laporan penelitian ini adalah sebagai berikut :

BAB I PENDAHULUAN

Pada bab pendahuluan ini berisi latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Pada bab ini berisi pembahasan review tentang penelitian-penelitian yang terdahulu yang berhubungan dengan permasalahan dalam penelitian ini, teori-teori pendukung seperti algoritma *Evolution Strategies*, penyakit yang perlu dilakukan diet dan gizi.

BAB III METODOLOGI PENELITIAN

Pada bab ini berisi penjelasan metode atau langkah-langkah yang akan digunakan dalam penelitian meliputi studi literatur, metode pengambilan data, analisis kebutuhan, perancangan sistem, implementasi sistem, pengujian dan analisis.

BAB IV PERANCANGAN SISTEM

Pada bab ini berisi perancangan yang terdiri perancangan sistem untuk hasil optimasi model dengan algoritma *Evolution Strategies*, perancangan antarmuka, serta perancangan uji coba dan evaluasi.

BAB V IMPLEMENTASI

Pada bab ini berisi pembahasan proses implementasi, batasan – batasan implementasi, serta algoritma yang digunakan dalam system.

BAB VI PENGUJIAN DAN ANALISIS

Pada bab ini memuat pengujian sistem dan analisis hasil pengujian tentang optimasi dengan metode yang digunakan.

BAB VII PENUTUP

Pada bab ini memuat kesimpulan dari hasil penelitian dan saran untuk pengembangan sistem selanjutnya agar lebih baik.

BAB II TINJAUAN PUSTAKA DAN DASAR TEORI

Pada bab ini berisi pembahasan review tentang penelitian-penelitian yang terdahulu yang berhubungan dengan permasalahan yang ada dalam penelitian ini. Selain itu pada bab ini juga diuraikan teori-teori pendukung yang digunakan dalam penelitian seperti algoritma *Evolution Strategies*, tentang penderita penyakit diabetes mellitus, jantung, dan hati dan tentang gizi.

2.1 Tinjauan Penelitian Terdahulu

Telah ada penelitian yang membahas optimasi biaya dan asupan gizi pasien diet khusus. Penelitian dilakukan dengan metode yang berbeda. Penelitian tersebut yang dilakukan oleh Karid pada tahun 2015 menunjukkan bahwa algoritma genetika dapat digunakan untuk menyelesaikan permasalahan optimasi biaya dan asupan gizi pasien diet khusus. Lewat penelitiannya Karid membuktikan bahwa algoritma genetika dapat digunakan untuk menyelesaikan permasalahan optimasi asupan gizi pasien diet. Lewat penelitiannya Karid juga membuktikan semakin besar nilai probabilitas mutasi (semakin kecil nilai probabilitas *crossover*) maka nilai rata-rata *fitness* yang didapat semakin rendah. Hal ini disebabkan kemampuan algoritma genetika yang semakin menurun untuk belajar dari generasi sebelumnya dan tidak mampu untuk mengeksplorasi daerah *optimum local*. Namun jika probabilitas *crossover* terlalu besar dan probabilitas mutasi terlalu kecil akan menurunkan kemampuan algoritma genetika untuk mengeksplorasi daerah pencarian.

Algoritma *evolution strategies* merupakan cabang dari *evolution algorithm* dan merupakan salah satu algoritma yang baik digunakan untuk optimasi (Mahmudy, 2013). Algoritma *evolution strategies* pernah digunakan untuk penyelesaian *Vehicle Routing Problem* pada distribusi minuman soda XYZ. Penelitian yang dilakukan oleh Harun pada tahun 2014 menunjukkan ukuran populasi (μ) = 80, ukuran *offspring* (λ) = 2μ , dan jumlah generasi 2000 dapat menyelesaikan masalah VRPTW dengan waktu kurang lebih 30 detik. Perbandingan penelitian-penelitian yang dilakukan sebelumnya dengan penelitian penulis, terinci di Tabel 2.1.

Tabel 2. 1 Perbedaan Penelitian Sebelumnya dengan Penelitian Penulis

No	Judul	Objek	Metode	Perbandingan	
				Studi Literatur	Skripsi Penulis
1	Penerapan Algoritma Genetika Untuk Optimasi Asupan Gizi Pasien Diet	Asupan gizi pasien diet	Algoritma Genetik, <i>Elitism Selection</i>	Gizi, Algoritma Genetik, <i>Elitism Selection</i>	Gizi, Algoritma <i>Evolution Strategies</i> ,

	Khusus Dengan Biaya Minimal (Karid, 2015)				<i>Elitism Selection</i>
2	Implementasi <i>Evolution Strategies</i> untuk Penyelesaian <i>Vehicle Routing Problem with Time Windows</i> pada Distribusi Minuman Soda XYZ (Harun, 2014)	Minuman Soda XYZ	<i>Evolution Strategies</i> , teknik ES ($\mu+\lambda$)	Minuman Soda XYZ, teknik ES ($\mu+\lambda$) reproduksi hanya melibatkan mutasi tanpa rekombinasi	Asupan gizi pasien diet, teknik ES ($\mu + \lambda$) reproduksi hanya melibatkan mutasi tanpa rekombinasi

Sumber: Diolah oleh penulis

Pada penelitian ini peneliti akan mencoba menyelesaikan masalah optimasi biaya dan asupan gizi pasien diet khusus menggunakan salah satu cabang algoritma evolusi yaitu *evolution strategies*. Tipe proses ES yang digunakan adalah ($\mu+\lambda$) di mana pada proses seleksi melibatkan *parent* dan *offspring* hasil dari mutasi. Metode seleksi menggunakan *elitism selection*.

2.2 Diet Pada Penderita Penyakit Diabetes Melitus

Diabetes mellitus, yang lebih dikenal di Indonesia dengan sebutan penyakit “kencing manis” merupakan salah satu penyakit yang prevalensinya kian meningkat. Diabetes mellitus merupakan kelainan pengolahan karbohidrat dalam tubuh yang disebabkan oleh kurang nya hormone insulin, sehingga karbohidrat tidak dapat dipergunakan oleh sel untuk diubah menjadi tenaga. Akibatnya karbohidrat yang ada di dalam tubuh dalam bentuk glukosa akan tertumpuk dalam darah sehingga terjadi peningkatan glukosa dalam darah. (Dewi, et al,. 2013)

Peningkatan prevalensi diabetes mellitus, selain faktor keturunan juga berkaitan dengan gaya hidup, yaitu asupan makanan yang berlebihan dan kurang nya olahraga.

Penyakit ini sebenarnya bisa dihindari, yaitu dengan menjaga pola makan yang sehat dan seimbang serta berolahraga secara teratur. Jika seseorang sudah divonis menderita diabetes, maka dia akan hidup selamanya dengan penyakit itu. Namun, jangan khawatir dan putus asa. Penderita diabetes mellitus bisa ‘hidup nyaman dan berumur panjang’, karena penyakit ini bisa dikendalikan. (Ayu, 2013)

Tujuan terapi diet pada diabetes mellitus (Dewi, et al,. 2013):

1. Memberikan zat gizi yang cukup untuk mempertahankan status gizi pasien

2. Mempertahankan berat badan untuk pasien usi dewasa, mencapai pertumbuhan pada pasien usia muda
3. Mencegah munculnya komplikasi akut/kronik
4. Membatasi asupan bahan makanan dengan indeks glikemik tinggi

Makanan yang dianjurkan (Dewi, et al., 2013):

1. Sumber karbohidrat kompleks, seperti nasi, roti, mi, kentang singkong, ubu dan sagu
2. Sumber protein rendah lemak, seperti ikan, ayam tanpa kulit, susu skim, tempe, tahu dan kacang-kacangan
3. Sumber lemak dalam jumlah terbatas yaitu brntuk makanan yang mudah cerna. Makanan terutama diolah dengan cara dipanggang, dikukus, disetup, direbus, dibakar.

Makanan yang tidak dianjurkan (Dewi, et al., 2013):

1. Mengandung banyak gula sederhana seperti gula pasir, gula merah, sirup, jam, jeli, buah-buahan yang diawetkan dengan gula, susu kental manis, softdrink, dan es krim
2. Mengandung banyak lemak, seperti cake, fast food, goreng-gorengan
3. Mengandung banyak natrium, seperti ikan asin, telur asin, makanan yang diawetkan

2.3 Diet Pada Penyakit Jantung

Salah satu penyakit jantung yang semakin banyak diidap, yaitu jantung koroner dapat disebabkan penyumbatan dan penyempitan pembuluh arteri korener akibat penumpukan lemak (kolesterol ataupun trigliserida) di bawah lapisan terdalam (endothelium) dinding pembuluh nadi. Faktor risiko utama penyakit yang satu ini adalah penambahan berat badan atau kegemukan, hipertensi, hyperlipidemia. Faktor risiko lainnya adalah stress, genetik, umur dan diabetes mellitus. (Dewi, et al., 2013)

Tujuan diet pada penyakit jantung (Dewi, et al., 2013):

1. Memberikan makanan yang tepat tanpa memberatkan kerja jantung
2. Mencegah atau menghilangkan penimbunan garam dan air
3. Menurunkan berat badan bila terlalu gemuk

Makanan yang dianjurkan (Dewi, et al., 2013):

1. Sumber Karbohidrat : beras ditim atau disaring, roti, mie, kentang, macaroni, biscuit, tepung beras, terigu, sagu, kentang, gula pasir, gula merah, madu, sirup.
2. Sumber protein hewani : daging sapi, ayam dibuang kulitnya, ikan, putih telur, susu rendah lemak

3. Sumber protein nabati : kacang-kacangan kering, seperti : kacang kedelai dan hasil olahannya yaitu tahu, tempe
4. Sayuran : semua sayuran segar yang tidak mengandung gas, seperti bayam, kangkung, buncis, labu siam, wortel
5. Buah-buahan : semua buah-buahan segar, seperti pisang, papaya, jeruk, apel, melon
6. Lemak : minyak jagung, minyak kedelai, margarin, mentega dalam jumlah terbatas dan tidak untuk menggoreng tetapi untuk menumis, santan encer
7. Bumbu-bumbu : semua bumbu yang segar
8. Minuman : teh encer, cokelat, sirup

2.4 Diet Pada Penyakit Hati

Hati (liver) merupakan organ terbesar dalam tubuh manusia. Di dalam hati terjadi proses-proses penting bagi kehidupan yaitu proses penyimpanan energi, pembentukan protein dan asam empedu, pengaturan metabolisme kolesterol, dan penetralan racun/obat yang masuk dalam tubuh. Hati yang sehat bisa menyaring racun dan melakukan proses detoksifikasi secara optimal. Bila hati sakit, otomatis racun bakal tertumpuk dan tubuh rentan terkena penyakit serius.(Pramono, 2010)

Tujuan diet sehat untuk penderita penyakit hati adalah memberikan makanan cukup untuk mempercepat perbaikan fungsi tanpa memperberat kerja hati. Syaratnya diet untuk penderita penyakit hati adalah: (Aryanto, 2015)

1. Kalori tinggi, kandungan karbohidrat tinggi, lemak sedang dan protein disesuaikan dengan keadaan penderita
2. Diet diberikan secara bertahap, disesuaikan dengan nafsu makan dan toleransi penderita
3. Cukup vitamin dan mineral
4. Rendah garam atau cairan dibatasi bila terjadi penimbunan garam atau air
5. Mudah dicerna
6. Hindari bahan makanan yang mengandung gas
7. Bila berat badan berlebihan, harus diturunkan secara bertahap sesuai kebutuhan penderita
8. Bahan makanan yang mengandung lemak dan kolesterol.

Pantangan makanan penderita penyakit hati beberapa diantaranya adalah (Anonim, 2015) :

1. Makanan yang mengandung lemak sangat tinggi misalnya adalah daging kambing, daging babi, jeroan, eskrim, susu yang full cream, keju, jenis makanan yang digoreng dan juga makanan bersantan.

2. Makanan di dalam kemasan atau kalengan seperti sarden dan juga kornet
3. Makanan yang menimbulkan gas seperti ubi, kacang merah, lobak, buah mentimun, buah durian, sayur kol, sayur sawi dan juga buah nangka.
4. Masakan bumbu yang mempunyai aroma serta rasa yang sangat tinggi misalnya adalah seperti cabai, bawang-bawangan, merica, jahe dan juga cuka.
5. Dan segala jenis makanan serta minuman yang mengandung alkohol dan juga bersoda atau berkarbonasi

Makanan yang sangat dianjurkan pada penderita hati ialah (Anonim, 2015) :

1. Makanan mengandung karbohidrat kompleks seperti nasi, havermout, roti putih dan juga jenis umbi-umbian.
2. Makanan yang mengandung protein tinggi misalnya adalah telur, ikan, daging ayam tanpa kulit, tempe, tahu, kacang hijau, sayuran dan juga buah-buahan yang mengandung serat tinggi.
3. Makanan yang mengandung hidrat arang tinggi yang bisa dicerna dengan baik. Misalnya adalah gula, sari buah, selai, sirup, manisan dan juga madu murni.

2.5 Diet Pada Orang Sehat

Diet pada orang sehat biasanya dilakukan untuk menurunkan berat badan atau sebaliknya, meningkatkan berat badan. Diet ini juga dapat bermanfaat untuk menjaga kesehatan tubuh dan terhindar dari penyakit tergantung jenis diet yang diterapkan. Diet tersebut dilakukan untuk mengatur asupan gizi tertentu dengan cara mengonsumsi makanan sesuai kebutuhan gizi. (Karid, 2015)

2.6 Gizi

Dalam bidang kesehatan, istilah gizi (sering disebut pula nutrisi) diartikan sebagai sebuah proses dalam tubuh makhluk hidup untuk memanfaatkan makanan guna pembentukan energi, tumbuh-kembang dan pemeliharaan tubuh. Ilmu gizi mempelajari proses tersebut. Nutrient atau zat-zat gizi merupakan substansi biokimia yang digunakan tubuh dan harus diperoleh dengan jumlah yang sama kuat dari makanan yang kita makan. Nutrien terdiri atas kelompok makronutrien dan mikronutrien. Hidrat arang, lemak, dan protein digolongkan ke dalam kelompok makronutrien karena dikonsumsi dengan jumlah relative besar (ukuran gram), sedangkan vitamin dan mineral digolongkan ke dalam mikronutrien karena diperlukan tubuh dengan jumlah relative kecil (microgram hingga milligram) dan sebagian besar didaur ulang dalam tubuh sendiri. (Andry, 2006)

Makanan merupakan substansi yang diambil tubuh untuk memberikan nutrient. Umumnya makanan diambil lewat tubuh lewat mulut, dicernakan dan diserap dalam saluran cerna untuk kemudian menjalani proses metabolisme.

Setiap makanan mengandung nutrient dengan jenis-jenis dan jumlah yang berbeda. Tidak ada satu pun jenis makanan yang mengandung semua nutrient esensial dengan jumlah yang diperlukan bagi kesehatan yang optimal. (Andry, 2006)

2.6.1 Energi

Energi basal adalah energi minimal yang diperlukan tubuh dalam keadaan istirahat, paling sedikit dua belas jam setelah makan, mental dan fisik dalam keadaan istirahat total, berbaring tapi tidak tidur, dan suhu ruangan sekitar 25 derajat celcius. Energi basal dipengaruhi oleh luas badan (ditentukan oleh faktor berat badan dan tinggi badan) umur, jenis kelamin, cuaca, ras, status gizi, penyakit, dan hormon. (Karyadi & Muhilal, 1988)

Energi yang diperlukan untuk melakukan kegiatan fisik didapat dari data empiris masing-masing kegiatan. Cara mendapatkan angka kebutuhan energi untuk masing-masing kegiatan fisik ialah dengan mengukur konsumsi oksigen selama kegiatan. Pemakaian satu liter oksigen setara dengan pemakaian energi sebanyak 4.95 Kalori. (Karyadi & Muhilal, 1988)

2.6.2 Protein

Protein merupakan zat gizi yang sangat penting bagi tubuh manusia karena diperlukan sebagai bahan pembentuk jaringan tubuh dan pengatur metabolisme. (Susanto & Dewanti, 2004)

Protein terbentuk dari unsur-unsur yang hampir sama dengan karbohidrat dan lemak yaitu unsur karbon, hydrogen dan oksigen tetapi ditambah dengan unsur nitrogen.

Molekul protein tersusun dari satuan dasar kimia yaitu asam amino. Asam-asam amino dalam molekul protein saling berhubungan dengan satu ikatan yang disebut ikatan peptide (-CONH-) satu molekul protein dapat terdiri dari 12 sampai 18 asam amino dan dapat mencapai jumlah ratusan asam amino. Oleh karena itu sifat dan kualitas protein dipengaruhi oleh jumlah dan jenis asam amino penyusunnya. (Susanto & Dewanti, 2004)

Fungsi protein antara lain :

1. Pertumbuhan dan pemeliharaan jaringan tubuh
2. Sebagai pengatur
3. Sebagai sumber energi
4. Pertahanan tubuh/ imunitas
5. Penunjang mekanis
6. Alat pengangkat dan alat penyimpanan

2.6.3 Lemak

Lemak adalah sumber energi lain di samping Karbohidrat dan protein. Setiap gram lemak menghasilkan 9 kalori lebih besar dari karbohidrat dan protein. Lemak adalah senyawa organik yang mengandung Carbon (C), Hidrogen (H) dan Oksigen (O) dan larut dalam alcohol dan larutan organik lainnya akan tetapi tidak larut dalam air. (Dewi, et al., 2013)

Fungsi lemak antara lain (Susanto & Dewanti, 2004) :

1. Penghasil Energi
2. Pembangun/ Pembentuk Struktur Tubuh
3. Protein Sparer
4. Penghasil Asam Lemak Essensial
5. Pelarut Vitamin
6. Pelumas Antara Persendian

2.6.4 Karbohidrat

Karbohidrat merupakan sumber energi utama yang disediakan melalui makanan sehari-hari. Sekitar 60-80% dari seluruh kalori makanan berasal dari karbohidrat. Bahan makanan pokok seperti padi-padian atau biji-bijian, sagu, dan ketela merupakan bahan yang mengandung karbohidrat dalam jumlah yang tinggi. Sumber karbohidrat dalam bentuk gula diantaranya gula pasir dan madu. (Susanto & Dewanti, 2004)

Fungsi Karbohidrat antara lain (Susanto & Dewanti, 2004) :

1. Karbohidrat Sebagai Sumber Energi Utama
2. Pengatur Metabolism Lemak
3. Penghemat Fungsi Energi
4. Sumber Energi Utama Bagi Otak Dan Syaraf
5. Simpanan Sebagai Glikogen
6. Pengatur Peristaltik Usus

2.7 Angka Kebutuhan Gizi

Angka kebutuhan gizi (Dietary Requirements) adalah banyaknya zat-zat gizi yang dibutuhkan seseorang (individu) untuk mempertahankan dan mencapai status gizi (Almatsier, 2004b). Angka kebutuhan gizi berbeda dengan angka kecukupan gizi (AKG). Angka kecukupan gizi atau Recommended Dietary Allowance (RDA) adalah tingkat konsumsi zat-zat gizi esensial yang dinilai cukup untuk memenuhi kebutuhan gizi hampir semua orang sehat di suatu negara (Almatsier, 2004a).

Kebutuhan gizi adalah jumlah zat gizi minimal yang diperlukan seseorang untuk hidup sehat, yang dipengaruhi oleh berbagai faktor (Susanto & Dewanti, 2004) :

1. Tahap-Tahap Perkembangan Kehidupan Manusia
2. Jenis Kegiatan Yang Dilakukan Dan Ukuran Tubuh
3. Keadaan Kesehatan, Sakit/Penyembuhan
4. Keadaan Fisiologi Khusus(Hamil/Menyusui)
5. Mutu Gizi Konsumsi Bahan Pangan

2.7.1 Kebutuhan Energi

Angka Metabolisme Basal (AMB) atau Basal Metabolic Rate (BMR) adalah kebutuhan energi minimal yang dibutuhkan tubuh untuk menjalankan proses tubuh yang vital. Kebutuhan energi metabolisme basal termasuk jumlah energi yang diperlukan untuk pernapasan, peredaran darah, pekerjaan ginjal, pankreas, dan lain-lain alat tubuh, serta untuk proses metabolisme di dalam sel-sel dan untuk mempertahankan suhu tubuh. Kurang lebih dua pertiga energi yang dikeluarkan seseorang sehari digunakan untuk kebutuhan aktivitas metabolisme basal tubuh. Angka metabolisme basal dinyatakan dalam kilokalori per kilogram berat badan per jam. Angka ini berbeda antar orang dan mungkin pada orang yang sama bila terjadi perubahan dalam keadaan fisik dan lingkungan. (Ilma, 2011)

Penentuan kebutuhan energi dapat dilakukan dengan langkah-langkah sebagai berikut (Almatsier, 2004b):

$$keEnerg = AMB \times AF \dots\dots\dots (2.1)$$

Keterangan:

AMB = Nilai AMB

AF = Nilai aktivitas fisik pasien

1. Menentukan nilai AMB

$$AMB \text{ (Laki - laki)} = 66 + (13,7 * BB) + (5 * TB) - (6,8 * U) \quad (2.2)$$

$$AMB \text{ (Perempuan)} = 655 + (9,6 * BB) + (1,8 * TB) - (4,7 * U) \quad (2.3)$$

Keterangan:

BB = Berat Badan (kg)

TB = Tinggi Badan (cm)

U = Usia (tahun)

2. Menentukan kebutuhan energi untuk aktivitas fisik

Aktivitas fisik dapat dibagi dalam empat golongan, yaitu sangat ringan, ringan, sedang, dan berat. Kebutuhan energi untuk berbagai aktivitas fisik dapat dinyatakan dalam kelipatan AMB (Tabel 2.2).

Tabel 2.2 Cara menaksir kebutuhan energi menurut aktivitas dengan menggunakan kelipatan AMB

Ativitas	Gender	
	Laki-laki	Perempuan
Sangat Ringan	1,30	1,30
Ringan	1,65	1,55
Sedang	1,76	1,70
Berat	2,10	2,00

Sumber: Amatsier, 2004, Penuntun Diet

2.7.2 Kebutuhan Karbohidrat, Lemak, dan Protein

Jumlah kebutuhan karbohidrat adalah 65% dari kebutuhan energi total, kebutuhan lemak adalah 20%, dan kebutuhan protein adalah 15%. Sedangkan tiap 1 g karbohidrat sama dengan 4 Kal, 1 g lemak sama dengan 9 Kal, dan 1 g protein sama dengan 4 Kal.

2.7.3 Kebutuhan Gizi Seimbang

Badan Kesehatan Dunia (WHO) secara umum menganjurkan konsumsi sayuran dan buah-buahan untuk hidup sehat sejumlah 400 g perorang perhari, yang terdiri dari 250 g sayur (setara dengan 2 ½ porsi atau 2 ½ gelas sayur setelah dimasak dan ditiriskan) dan 150 g buah, (setara dengan 3 buah pisang ambon ukuran sedang atau 1 ½ potong pepaya ukuran sedang atau 3 buah jeruk ukuran sedang). Bagi orang Indonesia dianjurkan konsumsi sayuran dan buah-buahan 300-400 g perorang perhari bagi anak balita dan anak usia sekolah, dan 350-600 g perorang perhari bagi remaja dan orang dewasa. Sekitar dua-pertiga dari jumlah anjuran konsumsi sayuran dan buah-buahan tersebut adalah porsi sayur. (PMK No.41, 2014)

2.8 Evolution Strategies

Evolution Strategies pertama kali dikembangkan pada tahun 1960-an oleh mahasiswa *Technical University of Berlin (TUB)* yaitu Ingo Rechenberg dan Hans-Paul Schwefel. Konsep yang mendasari timbulnya ES adalah konsep evolusi dan seleksi alam yang dikemukakan oleh Charles Darwin. Fungsi dari ES adalah untuk menemukan (atau mendekati) solusi optimal dari suatu permasalahan (Beyer, 2002).

Evolution Strategies (ES) memiliki ciri utama berupa penggunaan vektor bilangan pecahan (*real vector*) sebagai representasi solusi. Berbeda dengan *Genetic Algorithms (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).

2.9 Algoritma Evolusi

Algoritma evolusi atau dalam bahasa inggris *Evolutionary algorithm* (EA) adalah program komputer yang biasa digunakan untuk menyelesaikan masalah optimasi yang kompleks dengan meniru teori evolusi Darwin. EA akan mencari solusi optimal pada suatu permasalahan. Untuk menyelesaikan masalah tersebut, EA akan membangkitkan makhluk buatan untuk merepresentasikan solusi dari masalah yang ada. Setiap makhluk buatan ini akan saling berkompetisi satu sama lain dan diharapkan dari kompetisi tersebut akan didapatkan satu individu terbaik yang merupakan solusi dari masalah yang ada (Jones, 2004).

Algoritma evolusi merupakan teknik optimasi yang meniru evolusi biologi. Di mana akan terdapat sejumlah individu dalam sebuah populasi awal. Individu-individu ini akan diseleksi oleh alam dan individu yang baik akan mempunyai peluang yang lebih besar untuk melewatinya. Setiap individu pada algoritma evolusi merepresentasikan solusi dari masalah yang akan diselesaikan. Untuk mengetahui seberapa baik suatu individu maka digunakan sebuah fungsi *fitness* (Mahmudy, 2013).

Secara sederhana dapat diartikan EA merupakan algoritma *generate and test*. EA akan membentuk beberapa solusi dan mengkombinasikan atau merubah solusi tersebut ke solusi yang baru. Kualitas solusi dari masalah yang dicari direpresentasikan melalui fungsi evaluasi (*fitness*). Sebuah solusi dengan *fitness* yang baik kemungkinan besar akan menghasilkan solusi lain yang lebih baik tapi tidak menutup kemungkinan bahwa solusi dengan nilai *fitness* yang buruk dapat membawa solusi baru yang lebih baik. Beberapa sifat dari EA antara lain :

- EA merupakan algoritma berbasis populasi
- EA menggunakan proses rekombinasi dan mutasi untuk menggabungkan suatu informasi dari beberapa individu untuk mendapatkan informasi terbaru
- EA bersifat stokastik

Proses umum algoritma evolusi akan diterapkan pada beberapa tipe EA yang telah dikembangkan. Beberapa tipe EA yaitu algoritma genetik (*genetic algorithms*), *evolution strategies*, *evolutionary programming* (EP), dan *genetic programming* (GP). Perbedaan pada tipe-tipe EA ini dapat dilihat pada representasi individu yang ada. Algoritma genetik menggunakan representasi biner, ES menggunakan representasi vektor bilangan pecahan, EP menggunakan representasi *finite state machines*, dan GP menggunakan representasi *trees* (Eiben, 2004).

2.9.1 Struktur Dasar *Evolution Strategies* (ES)

Prosedur umum dalam ES dapat dinyatakan dengan istilah (μ, λ) . Di mana μ adalah jumlah solusi awal atau populasi awal, sedangkan λ merupakan jumlah

solusi yang dihasilkan dari generasi awal (*offspring*). (μ, λ) juga mengartikan bahwa generasi awal atau populasi awal (μ) tidak diikutsertakan pada proses seleksi untuk generasi berikutnya dan hanya melibatkan hasil *offspring* (λ). Selain bentuk (μ, λ) terdapat pula bentuk $(\mu + \lambda)$ di mana populasi awal (μ) dan hasil *offspring* (λ) akan dilibatkan pada seleksi (Brownlee, 2011).

Karena ES lebih mengandalkan mutasi, maka proses rekombinasi tidak selalu digunakan. Secara umum terdapat empat tipe proses dari ES, yaitu :

- (μ, λ)
- $(\mu/r, \lambda)$
- $(\mu + \lambda)$
- $(\mu/r + \lambda)$

ES(μ, λ) tidak menggunakan rekombinasi dalam proses reproduksi. Seleksi menggunakan *elitism selection* hanya melibatkan individu dalam *offspring*, individu induk dalam populasi tidak dilibatkan. ES($\mu/r, \lambda$) serupa dengan ES(μ, λ) dengan tambahan melibatkan proses rekombinasi. ES($\mu + \lambda$) tidak menggunakan rekombinasi dan proses seleksi menggunakan *elitism selection* melibatkan individu *offspring* dan induk (Mahmudy, 2013).

Apabila $P(t)$ dan $C(t)$ merupakan populasi (*parent*) dan *offspring* pada generasi ke- t maka siklus ES dapat dideskripsikan sebagai berikut :

```

procedure EvolutionStrategies
begin
    t = 0
    inisialisasi P(t): generate  $\mu$  individu
    while (bukan kondisi berhenti) do
        rekombinasi: produksi C(t) sebanyak  $\lambda$  dari P(t)
        mutasi C(t)
        seleksi P(t+1) dari P(t) dan C(t)
        t = t + 1
    end while
end

```

Gambar 2. 1 Pseudocode Algoritma Evolution Strategies

Sumber: [Mahmudy, 2013]

Jika kita perhatikan siklus di atas serupa dengan siklus algoritma genetika (GAs). Perbedaan nyata ES dan GAs adalah pada operator yang digunakan. Perbedaan yang lain adalah mutasi pada GAs digunakan untuk menghasilkan individu baru (*offspring*) sebagai tambahan dari *offspring* yang diproduksi oleh operator *crossover*. Pada ES, mutasi diterapkan pada *offspring* yang dihasilkan proses rekombinasi. Rekombinasi pada ES mirip dengan operator *crossover* pada GAs tapi bisa menggunakan lebih dari dua induk (Mahmudy, 2013).

2.9.2 Representasi Kromosom

Representasi kromosom merupakan proses pengkodean dari penyelesaian asli suatu permasalahan. Solusi dari suatu permasalahan harus dipetakan (*encoding*) menjadi string kromosom. String kromosom tersusun atas sejumlah gen yang menggambarkan variable-variabel keputusan yang digunakan dalam solusi (Mahmudy, 2013).

Seperti halnya untuk *real-coded* GA, variable keputusan (x_1 dan x_2) langsung menjadi gen string kromosom. Selain gen yang menyatakan variable keputusan, parameter tambahan yang melekat pada setiap kromosom adalah *sigma* (σ). Nilai ini menyatakan level mutasi untuk kromosom tersebut. Nilai ini akan berubah sepanjang generasi secara adaptif. Jika P adalah satu kromosom maka $P = (x_1, x_2, \sigma_1, \sigma_2)$ dengan panjang string sebesar 4 (Mahmudy, 2013).

2.9.3 Inisialisasi

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 *miu* (μ) yang menyatakan ukuran populasi harus ditentukan. Panjang string kromosom (*stringLen*) dihitung berdasarkan presisi variable dan solusi yang dicari (Mahmudy, 2013).

Misalkan populasi inisialisasi dibangkitkan secara random dengan $\mu = 4$ seperti pada Tabel 2.3.

Tabel 2. 3 Contoh Representasi Kromosom

P(t)	x1	x2	σ_1	σ_2	f(x1,x2)
1	1.48980	2.09440	0.14197	0.91090	19.82128
2	8.49170	2.57540	0.53801	0.86904	34.70609
3	1.84610	1.70970	0.99835	0.49351	11.58639
4	5.81140	5.07790	0.40521	0.98911	14.56208

Sumber : Mahmudy, 2013

2.9.4 Reproduksi

Reproduksi bertujuan untuk menghasilkan keturunan dari individu-individu yang ada di populasi. Berbeda pada algoritma genetik yang menggunakan dua operator genetika yaitu tukar silang (*crossover*) dan mutasi (*mutation*) pada algoritma *evolution strategies* menggunakan rekombinasi dan mutasi. (Mahmudy, 2013).

2.9.5 Rekombinasi

Rekombinasi dilakukan untuk menghasilkan *offspring* sebanyak 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 nilai rata-rata nilai elemen induk. 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 diberikan sebagai berikut pada Tabel 2.4.

Tabel 2. 4 Contoh Rekombinasi

C(t)	Induk	x1	x2	σ_1	σ_2	f(x1,x2)
c1	P1 dan P3	0.17815	1.90205	0.57016	0.70221	16.64183
c2	P2 dan P3	3.3228	2.14255	0.76818	0.68128	19.5813
c3	P1 dan P4	3.6506	3.58615	0.27359	0.95001	9.57005
c4	P2 dan P4	7.15155	3.82665	0.47161	0.92907	12.52404
c5	P1 dan P3	0.17815	1.90205	0.57016	0.70221	16.64183
c6	P3 dan P4	1.98265	1.98265	1.70178	0.74131	12.65007

Sumber : Mahmudy, 2013

2.9.6 Mutasi

Mutasi digunakan sebagai operator untuk menjaga keragaman populasi. Misalkan $P = (x_1, x_2, \sigma_1, \sigma_2)$ adalah individu yang terpilih untuk melakukan mutasi, maka dihasilkan *offspring* $P' = (x'_1, x'_2, \sigma'_1, \sigma'_2)$ sebagai berikut (Mahmudy, 2013).
 $x' = x + \sigma N(0,1)$ 2- 1

Rumusan ini bisa didetailkan sebagai berikut:

$$\begin{aligned} x'_1 &= x_1 + \sigma_1 N(0,1) \\ x'_2 &= x_2 + \sigma_2 N(0,1) \end{aligned} \dots\dots\dots 2- 2$$

dimana :

x' = offspring

$N(0,1)$ 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 r_1 dan r_2 pada interval $[0,1]$. Rumus yang digunakan adalah (Schwefel, 1995):

$$N(0,1) = \sqrt{-2 \cdot \ln r_1} * (\sin 2\pi r_2) \dots\dots\dots 2- 3$$

Misalkan $r_1 = 0,4749$ dan $r_2 = 0.3296$ maka didapatkan $N(0,1) = 1,0709$

Dimana :

r = bilangan random untuk membangkitkan nilai $N(0,1)$

Nilai σ dinaikkan jika *fitness* hasil mutasi lebih baik dari induknya. Sebaliknya jika *fitness* hasil mutasi lebih buruk dari induknya maka nilai σ diturunkan. Nilai σ dinaikkan dengan rumusan $\sigma' = \sigma \times 1,1$ dan diturunkan dengan rumusan $\sigma' = \sigma \times 0,9$ (Mahmudy, 2013).

Tabel 2. 5 Contoh Mutasi

C(t)	N1(0.1)	N2(0.1)	x1'	x2'	$\sigma 1'$	$\sigma 2'$	f(x1,x2)
C1	0.18850	0.27470	0.07068	2.09495	0.62717	0.77243	21.33870
C2	1.79470	0.13590	1.94415	2.04997	0.84499	0.74940	19.90344
C3	0.56030	1.86570	3.49731	1.81372	0.30095	1.04501	10.98098
C4	0.61890	0.46130	7.44343	3.39807	0.42445	0.83617	5.40751
C5	0.13710	0.42010	0.25632	1.60705	0.51314	0.63199	11.26206
C6	1.11250	0.21530	2.76338	3.23420	0.77195	0.81544	16.32937

Sumber : Mahmudy, 2013

2.9.7 Perhitungan *Fitness*

Sama halnya pada algoritma genetika, pada algoritma ES suatu individu dievaluasi berdasarkan suatu nilai tertentu berdasarkan suatu fungsi tertentu sebagai ukuran nilai *fitness*-nya. Pada masalah optimasi, jika solusi yang dicari adalah memaksimalkan fungsi h , maka nilai *fitness* yang digunakan adalah nilai dari fungsi h tersebut, yakni $f = h$. Sedangkan jika masalahnya adalah meminimalkan fungsi h , maka fungsi h tidak bisa digunakan secara langsung. Sehingga nilai *fitness* untuk masalah minimasi adalah $f = 1/h$, yang artinya semakin kecil nilai h , maka semakin besar nilai f . Namun fungsi ini akan bermasalah jika h bernilai 0, yang akan mengakibatkan f bernilai tak hingga. Maka, h perlu ditambah dengan sebuah bilangan. Sehingga rumus *fitness* ditunjukkan pada persamaan (2-3) (Suyanto, 2011).

$$f = \frac{1}{(h+a)} \dots\dots\dots 2- 4$$

Di mana :

- a adalah suatu bilangan konstanta sesuai dengan permasalahan yang diselesaikan.
- h adalah sebuah fungsi yang akan diminimalkan
- f adalah fungsi *fitness*

Dalam penelitian ini perhitungan *fitness* bertujuan untuk meminimalkan harga dengan memperhatikan nilai penalti. Sehingga fungsi *fitness* yang digunakan dalam penelitian ini ditunjukkan pada persamaan (2-4) (Tyas, 2013)

$$f = \frac{1}{\text{penaltiGizi} + \text{penaltiHarga} + 1} \dots\dots\dots 2- 5$$

Di mana :

penaltiGizi = Nilai penalti gizi
 penaltiHarga = Nilai penalti harga

2.9.8 Seleksi

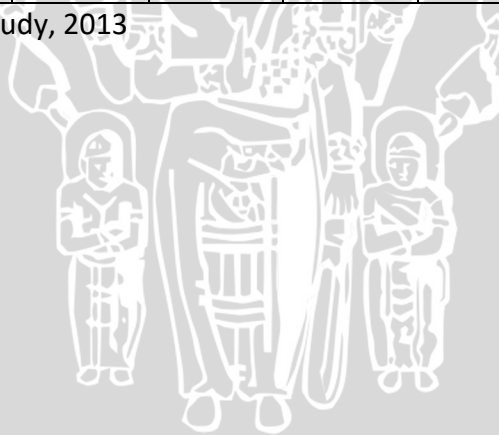
Seleksi merupakan tahapan terakhir yang dilakukan untuk memilih individu dari himpunan populasi dan *offspring* yang akan dipertahankan hidup pada generasi berikutnya. Semakin besar nilai *fitness* kromosom, maka semakin besar peluang kromosom tersebut terpilih. Hal ini dilakukan agar terbentuk generasi berikutnya yang lebih baik dari generasi sekarang (Mahmudy, 2013), Metode seleksi yang digunakan dalam penelitian ini adalah metode seleksi *elitism*.

Metode seleksi *elitism* 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).

Tabel 2. 6 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	34.70609
P2	C1	0.07068	2.09495	0.62717	0.77243	21.33870
P3	C2	1.94415	2.04996	0.84499	0.74940	19.90344
P4	P1	1.48980	2.09440	0.14197	0.91090	19.82128

Sumber : Mahmudy, 2013

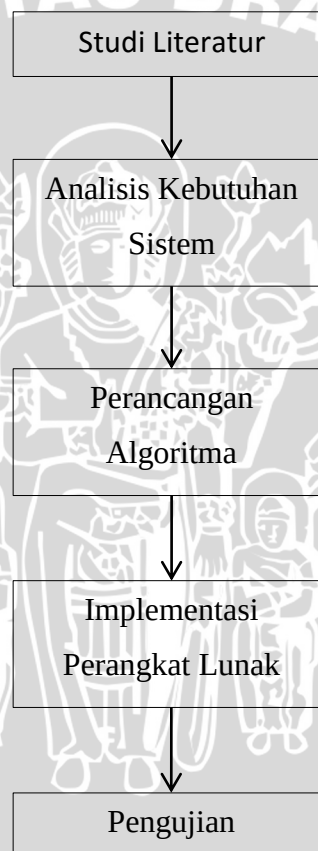


BAB III METODOLOGI PENELITIAN

Demi terselesaikannya penelitian ini, maka dibutuhkan susunan tahapan kegiatan penelitian yang terstruktur dan tepat serta perancangan yang baik. Pada metodologi penelitian ini akan dibahas langkah-langkah dan rancangan yang digunakan dalam pembuatan aplikasi optimasi biaya dan asupan gizi pasien diet khusus menggunakan algoritma *evolution strategies*.

3.1 Tahapan Penelitian

Tahapan penelitian ini meliputi studi literatur, pengumpulan dan analisis data, analisis dan perancangan sistem, implementasi sistem, pengujian sistem, dan evaluasi hasil yang ditunjukkan pada Gambar 3.1 berikut ini.



Gambar 3. 1 Tahapan Penelitian

Berdasarkan Gambar 3.1 , tahapan penelitian skripsi dapat dijelaskan sebagai berikut :

1. Mengumpulkan literatur yang terkait dan mempelajari metode-metode yang akan digunakan dalam penelitian skripsi ini
2. Melakukan analisa terhadap kebutuhan sistem
3. Melakukan perancangan sistem
4. Melakukan implementasi berdasarkan dari hasil analisa dan perancangan yang telah dilakukan sebelumnya

5. Melakukan pengujian terhadap sistem, apakah sistem tersebut sudah baik atau masih membutuhkan perbaikan

3.2 Data yang digunakan

Data yang digunakan dalam penelitian adalah sebagai berikut:

1. Data DKBM (Daftar Komposisi Bahan Makanan) Indonesia yang dikeluarkan oleh Depkes dan telah disesuaikan untuk masing-masing jenis diet.
2. Data harga bahan makanan didapatkan dengan melakukan survey ke beberapa tempat yang menjual bahan makanan di Kota Blitar pada Mei - Juni 2015.

3.3 Algoritma yang digunakan

Penelitian ini menggunakan algoritma Evolution Strategies seperti yang telah diuraikan pada Bab 2 telah terbukti efektif digunakan untuk melakukan suatu optimasi pada permasalahan tertentu. Karena fungsi dari Evolution Strategies itu sendiri adalah untuk menemukan (atau mendekati) solusi optimal dari suatu permasalahan (Beyer & Schwefel, 2002). Implementasi pada algoritma akan diterapkan pada aplikasi berbasis desktop. Sehingga implementasinya menggunakan program dengan bahasa Java.

3.4 Kebutuhan Sistem

Analisa kebutuhan berguna untuk mendapatkan apa saja yang diperlukan dalam pembuatan sistem ini. Dimana kebutuhan-kebutuhan tersebut antara lain:

1. Dalam sistem ini Hardware yang bisa digunakan, yaitu:
 - Komputer/PC
2. Dalam sistem ini Software yang digunakan, yaitu:
 - OS Windows 10 64-bit sebagai operating sistem yang digunakan untuk menjalankan semua aplikasi yang dibutuhkan.
 - Microsoft office 2013 sebagai aplikasi penunjang dokumen
 - Netbeans 8.0 3.
 - JDK 1.8
 - Microsoft excel 2013 sebagai aplikasi penunjang penyimpanan data dan perhitungan.

BAB IV PERANCANGAN SISTEM

4.1 Formulasi Permasalahan

Pada bagian ini dibahas proses perancangan algoritma untuk memperoleh solusi keluaran yang terbaik. Langkah awal proses ini adalah melakukan perhitungan jumlah kalori dan menentukan kebutuhan protein, lemak dan karbohidrat dari user. Langkah selanjutnya adalah tahapan-tahapan yang ada pada metode *Evolution Strategies*. Selain itu akan dijelaskan pula perancangan user interface sistem yang akan dibuat.

Optimasi biaya dan asupan gizi pasien diet dengan algoritma *evolution strategies* terdiri dari beberapa proses sebagai berikut :

1. Melakukan proses perhitungan nutrisi (karbohidrat, lemak dan protein) dengan memasukkan data berupa ;
 - Usia
 - Berat Badan
 - Tinggi Badan
 - Jenis Kelamin
 - Jenis Aktivitas
 - Bahan makanan yang dipilih
2. Inisialisasi parameter awal, yaitu:
 - Jumlah populasi (μ),
 - Ukuran offspring (λ),
 - Jumlah Generasi
3. Membuat populasi awal sesuai dengan jumlah populasi yang dimasukkan.
4. Membuat populasi baru menggunakan mutasi untuk pembentukan offspring. Hasil mutasi ini nantinya akan diikuti bersama parent dalam proses seleksi.
5. Menghitung nilai fitness masing-masing individu lalu diseleksi.
6. Proses seleksi dengan metode elitism untuk mendapatkan individu yang bertahan pada generasi selanjutnya.

Proses selesai apabila kondisi akhir sudah terpenuhi. Hasil akhir adalah kromosom terbaik pada akhir generasi.

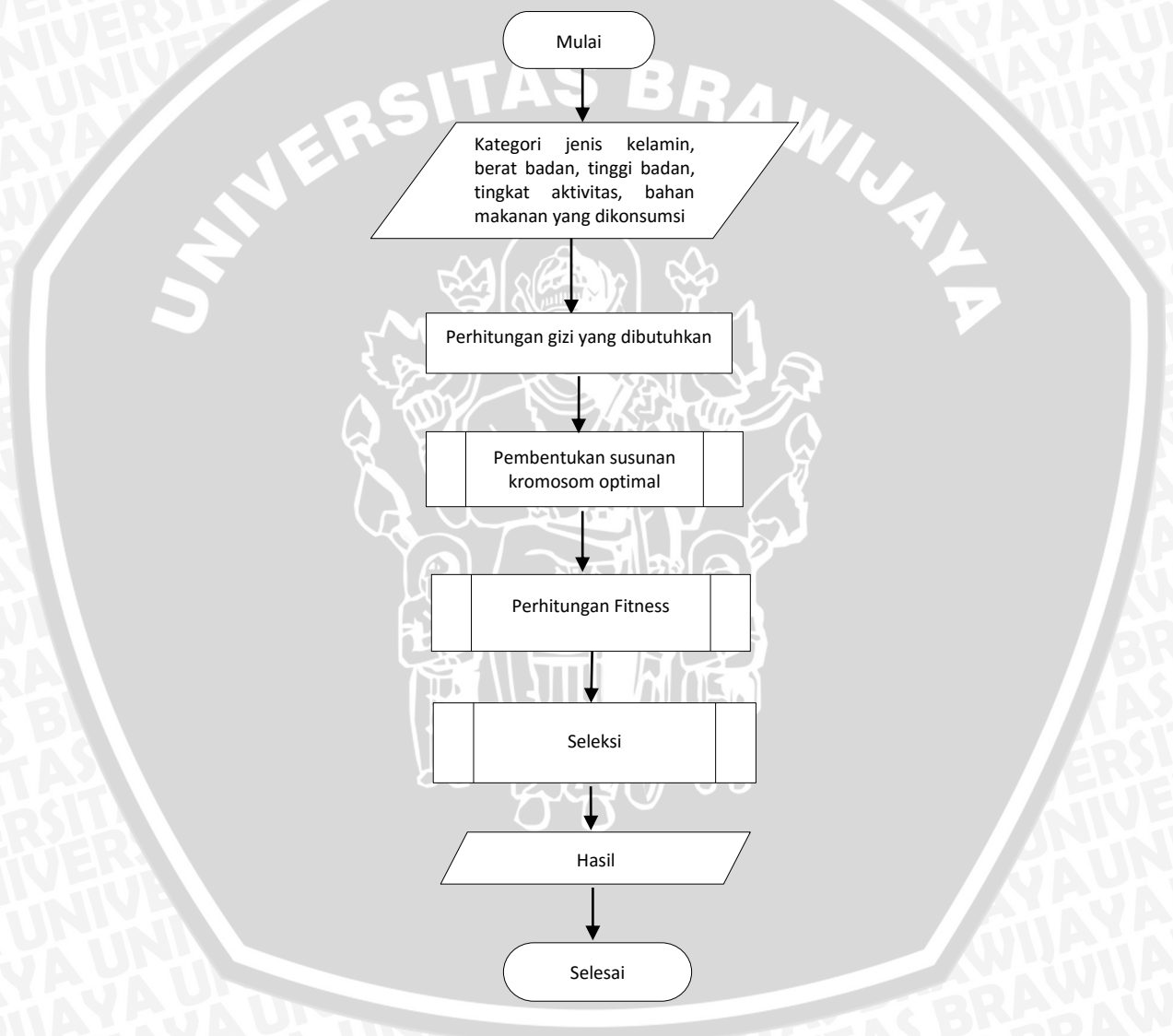
4.2 Siklus Penyelesaian Masalah

Pada bagian ini awalnya akan dibahas proses perancangan sistem untuk memperoleh solusi keluaran yang baik. Langkah awal proses ini adalah penentuan jenis kelamin, berat badan, tinggi badan, tingkat aktivitas, bahan makanan yang dikonsumsi. Langkah selanjutnya adalah melakukan inisialisasi populasi awal. Representasi kromosom didapat dari susunan bahan makanan yang tersedia yang bobotnya dibangkitkan secara acak.

Populasi awal dibangkitkan dari nilai parameter yang dimasukkan. Tipe proses ES yang digunakan dalam penelitian ini adalah $(\mu + \lambda)$ dimana proses

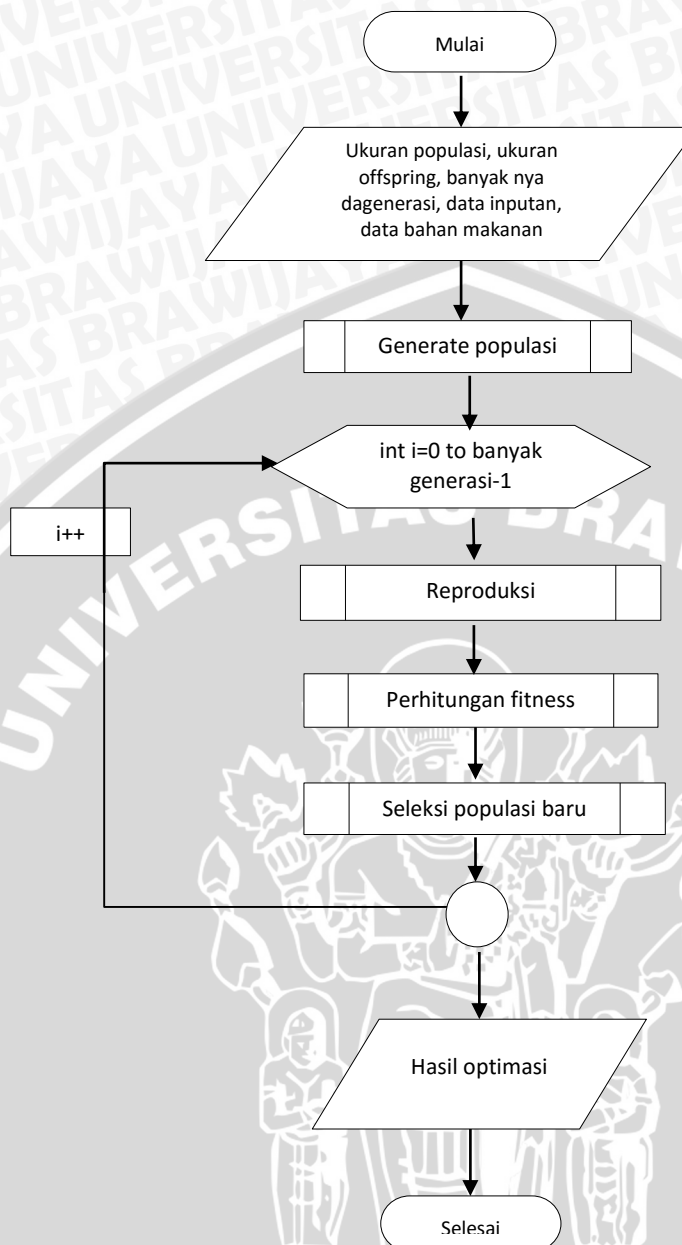
reproduksi melibatkan rekombinasi dan hasil mutasi nantinya akan diikuti bersama *parent* dalam proses seleksi.

Offspring dihasilkan dari jumlah kombinasi dua induk. Selanjutnya individu dari populasi awal dan *offspring* hasil *mutasi* digabungkan untuk proses seleksi. Proses seleksi dilakukan dengan menghitung nilai *fitness* masing-masing individu. Individu yang mampu bertahan adalah individu yang memiliki nilai *fitness* terbaik setelah n generasi. Nilai *fitness* didapat dari nilai penalti gizi dan nilai penalti harga yang didapat. Proses penyelesaian masalah optimasi biaya dan asupan gizi pasien diet khusus menggunakan algoritma *evolution strategies* dapat dilihat pada Gambar 4.1.



Gambar 4. 1 Flowchart Optimasi asupan gizi

Flowchart pada Gambar 4.1 menjelaskan tentang optimasi asupan menggunakan algoritma *evolution strategies* di mana terdapat subproses di dalamnya yaitu proses pembentukan susunan kromosom optimal. Subproses pembentukan susunan kromosom optimal dapat dilihat pada Gambar 4.2.



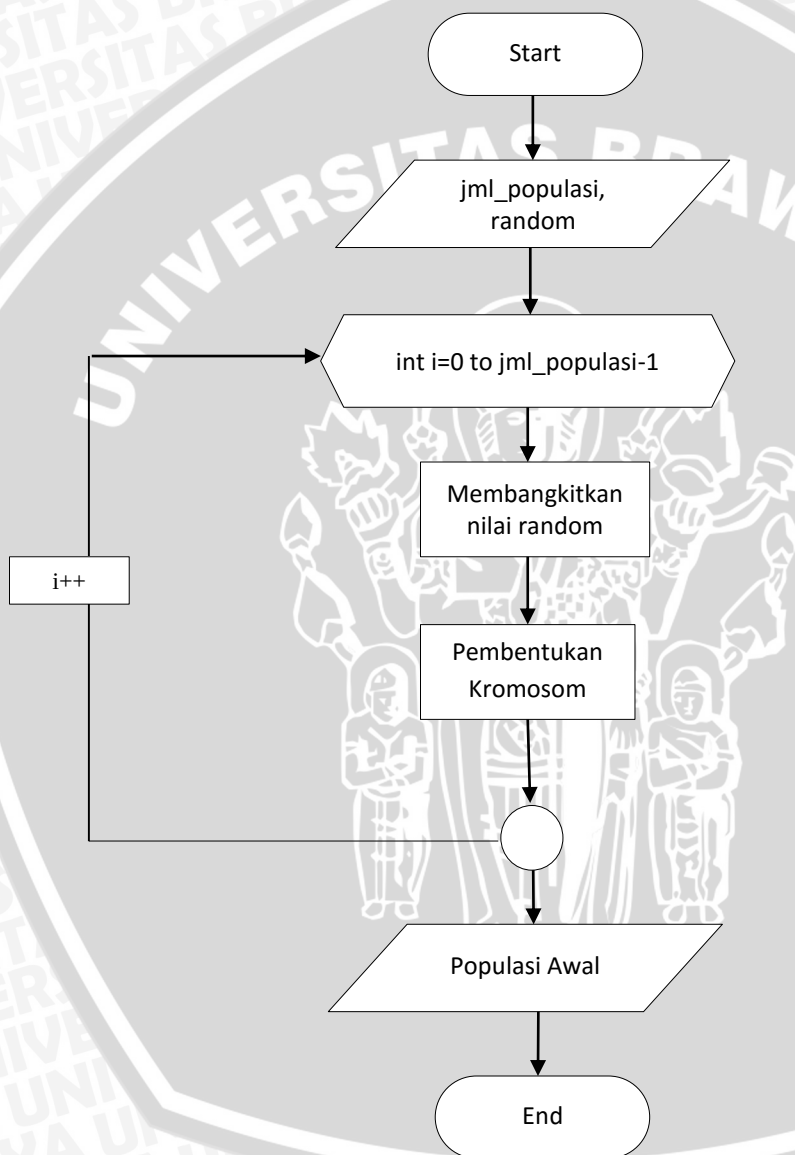
Gambar 4. 2 Flowchart Proses Pembentukan Susunan kromosom optimal

Seperti yang terlihat pada Gambar 4.2 terdapat empat proses utama untuk menyelesaikan masalah optimasi asupan gizi dengan bahan makanan, yaitu proses *Generate Populasi Awal*, *Reproduksi*, *Perhitungan Fitness* dan *Seleksi Populasi baru*. Langkah kerjanya adalah setelah memasukan kategori jenis kelamin, memasukkan usia, berat badan, tingkat aktivitas, bahan makanan yang dikonsumsi, bobot per jenis yang diinginkan, maka sistem akan membuat populasi awal sesuai dengan jumlah populasi yang dimasukkan. Setelah populasi awal terbentuk akan dilakukan pengulangan untuk membentuk *offspring* dengan cara mutasi. Hasil mutasi dan populasi awal akan dihitung nilai *fitness* masing-masing individu lalu diseleksi. Proses pengulangan dilakukan sampai banyak generasi

tercapai. Hasil keluaran berupa hasil optimasi yang didapatkan dengan algoritma *evolution strategies*.

4.2.1 Generate Populasi Awal

Pada proses ini sistem akan membuat populasi awal secara acak yang terdiri dari sejumlah individu yang diinputkan. Setiap individu akan dibentuk dengan representasi kromosom berjenis *real*. Proses pembuatan populasi ini dapat dilihat pada Gambar 4.3 berikut :



Gambar 4. 3 Flowchart Proses Generate Populasi Awal

Berdasarkan Gambar 4.3 dapat dijelaskan langkah-langkah dalam pembuatan populasi awal adalah sebagai berikut :

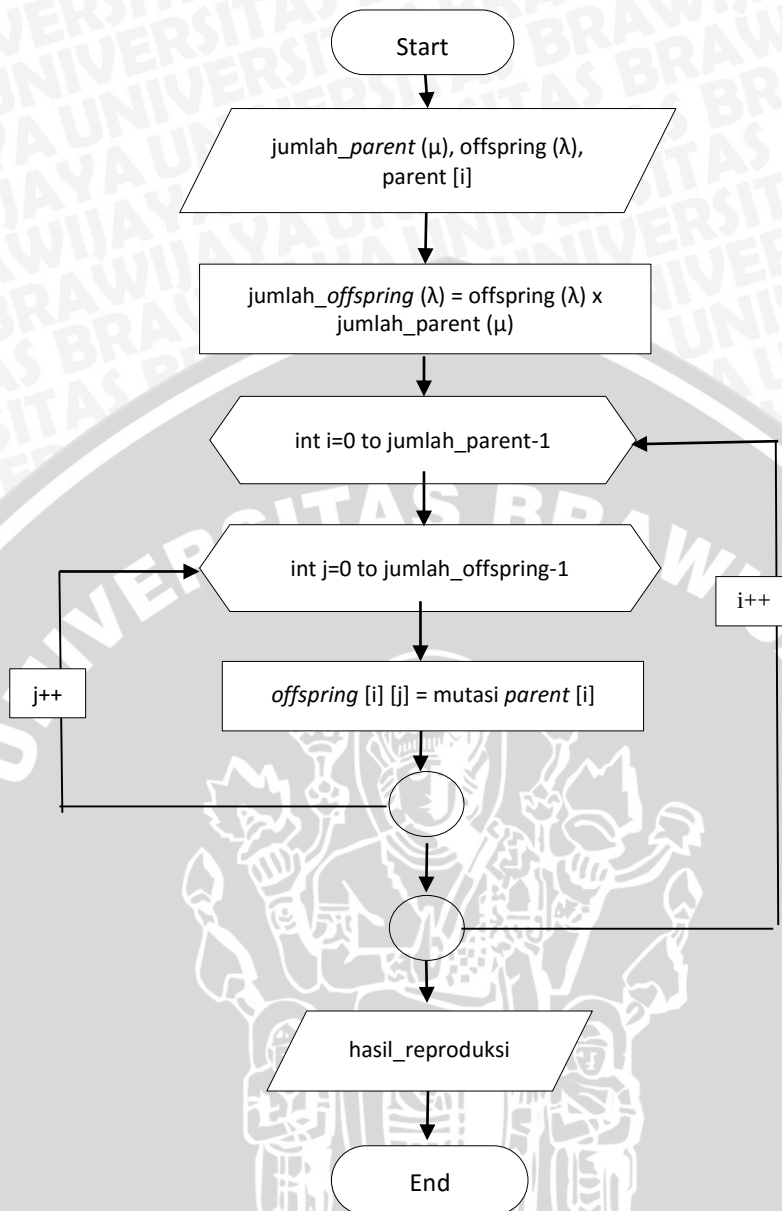
1. Menentukan panjang kromosom, dimana panjang kromosom sesuai dengan banyak bahan makanan yang diinginkan oleh user.

2. Mengisi tiap gen pada kromosom dengan cara membangkitkan bilangan random $[0..1]$. Dimana, bilangan itu merepresentasikan berat dari suatu bahan makanan.
3. Sistem menerima masukan berupa jumlah populasi yang akan dibuat.
4. Sistem akan melakukan proses pembuatan individu sebanyak jumlah populasi yang ditentukan.
5. Hasil proses ini berupa populasi awal yang akan digunakan untuk proses berikutnya.

4.2.2 Reproduksi

Pada proses reproduksi sistem akan melakukan pembuatan *offspring* menggunakan mutasi. Pembentukan *offspring* (λ) dilakukan sejumlah populasi (μ) atau ($\lambda=\mu$). Hasil mutasi ini nantinya akan diikuti bersama *parent* dalam proses seleksi. Berdasarkan literatur yang telah dijelaskan sebelumnya proses ini berupa proses ($\mu+\lambda$). Pada proses reproduksi ini juga melibatkan *self adaptation* dimana nilai sigma dari *offspring* akan berubah sesuai dengan nilai *fitness* yang dihasilkan. Langkah-langkah yang dilakukan dalam proses reproduksi ini dapat dilihat pada Gambar 4.4 berikut :





Gambar 4. 4 Flowchart Proses Reproduksi

Berdasarkan diagram alir di atas dapat dijelaskan langkah-langkah dalam proses reproduksi sebagai berikut:

1. *Input* dalam proses reproduksi berupa *parent* yang akan digunakan dalam proses mutasi. Jumlah *child* sama dengan jumlah *parent*.
2. Untuk pembentukan *offspring*, sistem akan melakukan perulangan sesuai dengan jumlah *parent*.
3. Proses mutasi untuk membentuk *offspring* dilakukan sebanyak jumlah anak.
4. Setelah proses mutasi selesai, sistem akan mengganti nilai *strategy parameters* dari *parent* asal sesuai dengan nilai *fitness* hasil *offspring* untuk melakukan proses *self adaptation*.
5. Hasil mutasi berupa *offspring* yang akan diseleksi pada proses berikutnya.

6. Kemudian dilakukan evaluasi dengan cara mengambil individu terbaik sebanyak populasi awal dari urutan nilai *fitness* secara *descending* dari populasi gabungan (*parent* dan *offspring*)

4.2.3 Penalti Gizi

Penalti merupakan suatu bobot yang digunakan ketika individu melakukan pelanggaran terhadap aturan. Langkah pertama adalah menghitung berat bahan makanan sebenarnya berdasarkan nilai yang ada pada gen kromosom. Berdasarkan dasar teori tentang Kebutuhan Gizi Seimbang pada Bab 2, peneliti mengambil angka 350 g sebagai acuan rata-rata berat bahan makanan maksimal per hari untuk dikalikan dengan nilai gen yang sudah di acak sebelum nya. Seperti yang dapat dilihat pada Persamaan 4.1

$$\text{berat} = \text{gen} \times 350 \text{ g} \dots\dots\dots 4- 1$$

Keterangan:

berat = berat makanan sebenarnya

gen = nilai gen

Setelah didapatkan nilai berat bahan makanan yang sebenarnya, maka dilanjutkan dengan menghitung kandungan gizi bahan makanan.

$$\text{eg} = \frac{\text{berat}}{100} \times \text{nilaiGiziEgi} \dots\dots\dots 4- 2$$

$$\text{kb} = \frac{\text{berat}}{100} \times \text{nilaiGiziKbi} \dots\dots\dots 4- 3$$

$$\text{lm} = \frac{\text{berat}}{100} \times \text{nilaiGiziLmi} \dots\dots\dots 4- 4$$

$$\text{pt} = \frac{\text{berat}}{100} \times \text{nilaiGiziPti} \dots\dots\dots 4- 5$$

Keterangan:

eg= jumlah kandungan energi

kb= jumlah kandungan karbohidrat

lm= jumlah kandungan lemak

pt= jumlah kandungan protein

i = gen ke-

berat = berat bahan makanan sebenarnya

nilaiGiziEg = nilai gizi energi bahan makanan per 100 g

nilaiGiziKb = nilai gizi karbohidrat bahan makanan per 100 g

nilaiGiziLm = nilai gizi bahan makanan per 100 g

nilaiGiziPt = nilai gizi protein bahan makanan per 100 g

Selanjutnya, dilanjutkan dengan menghitung nilai penalti per jenis bahan makan untuk penghitungan penalti gizi dengan pembobotan jenis bahan makan.

$$\text{penaltiMenu} = |(egi - (\text{kebEnergi} * BJM))| + |(kbi - (\text{kebKarbohidrat} * BJM))| + |(lmi - (\text{kebLemak} * BJM))| + |(pti - (\text{kebProtein} * BJM))| \dots\dots\dots 4- 6$$

Keterangan:

penaltiMenu = nilai penalti per jenis bahan makan

i = jenis bahan makan (pagi, snack 1, siang, snack 2, sore)

j = jenis prioritas gizi (energi, karbohidrat, lemak, dan protein)

eg = jumlah kandungan energi per jenis bahan makan

kb = jumlah kandungan karbohidrat per jenis bahan makan

lm = jumlah kandungan lemak per jenis bahan makan

pt = jumlah kandungan protein per jenis bahan makan

kebEnergi = nilai kebutuhan energi

kebKarbohidrat = nilai kebutuhan karbohidrat

kebLemak = nilai kebutuhan lemak

kebProtein = nilai kebutuhan protein

BJM = bobot jenis menu [0..1] yang ditentukan

Setelah diketahui nilai penalti per jenis bahan makan, maka dilakukan penghitungan nilai total penalti.

$$\text{penalti} = \text{penaltiMenu}_{pg} + \text{penaltiMenu}_{sn1} + \text{penaltiMenu}_{si} + \text{penaltiMenu}_{sn2} + \text{penaltiMenu}_{sr} \dots\dots\dots 4- 7$$

Keterangan:

penalti = nilai penalti

penaltiMenu_{pg} = penalti makan pagi

penaltiMenu_{sn1} = penalti snack 1

penaltiMenu_{si} = penalti makan siang

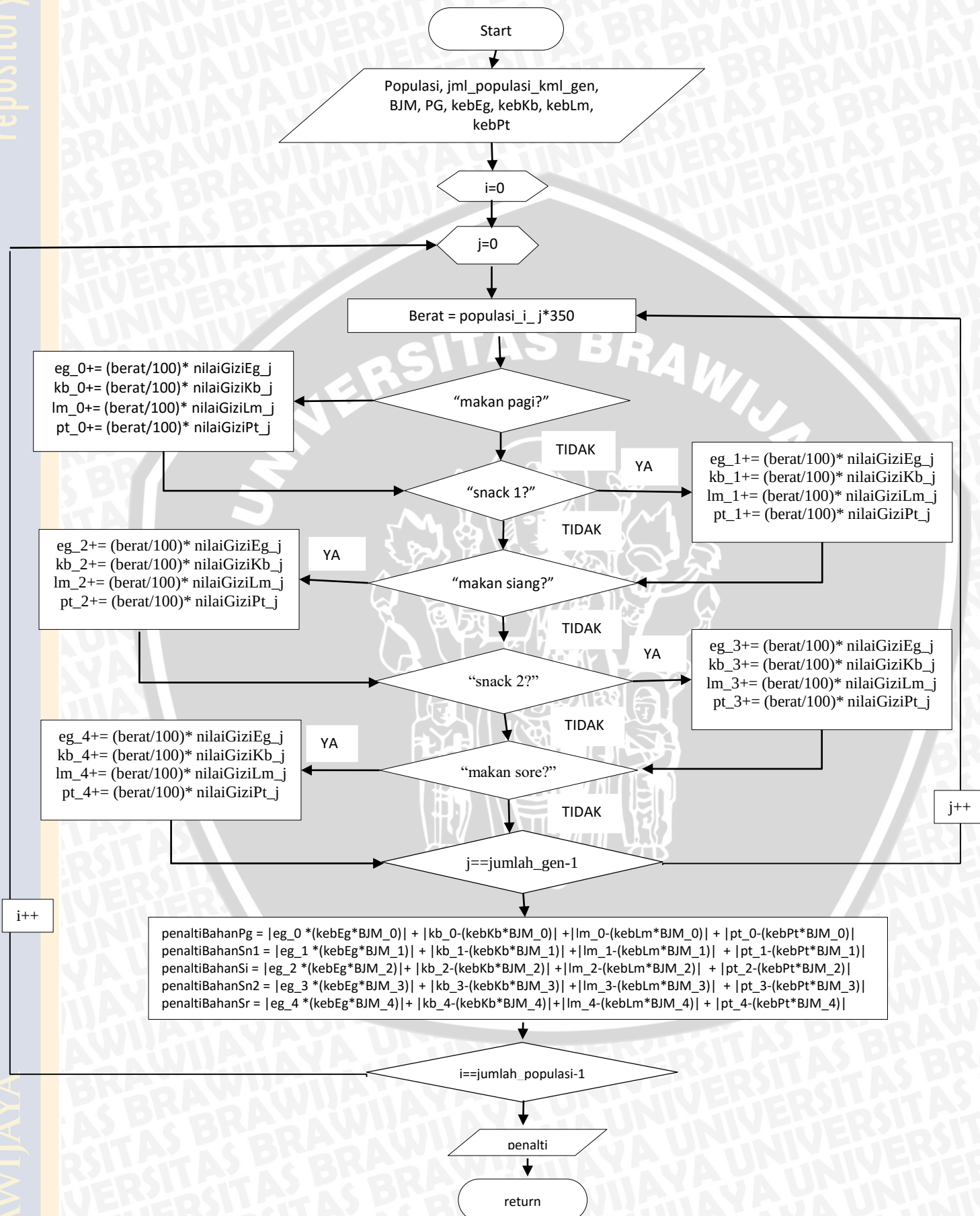
penaltiMenu_{sn2} = penalti snack 2

penaltiMenu_{sr} = penalti makan sore

Berikut adalah langkah-langkah untuk perhitungan penalti gizi adalah sebagai berikut:

1. Menghitung berat bahan makanan sebenarnya berdasarkan nilai gen dengan menggunakan Persamaan (4.1).
2. Menghitung nilai gizi (energi, karbohidrat, lemak, dan protein) berdasarkan berat bahan makanan sebenarnya yang sudah didapat menggunakan Persamaan (4.2), (4.3), (4.4), dan (4.5).
3. Menghitung nilai penalti per jenis bahan makan menggunakan Persamaan (4.6).
4. Menghitung nilai total penalti menggunakan Persamaan (4.7).

Flowchart penghitungan penalti gizi dapat dilihat pada Gambar 4.5 berikut:



Gambar 4. 5 Flowchart Perhitungan Penalti Gizi

4.2.4 Penalti Harga

Penalti harga berguna untuk memberikan bobot ketika individu melakukan suatu pelanggaran seperti total harga yang melebihi anggaran. Penghitungan penalti harga dilakukan menggunakan Persamaan 4.8 dan 4.9.

$$totHarga = \sum_{i=0}^n \frac{berat}{100} \times harga_i \dots\dots\dots 4-8$$

Keterangan:

totHarga = totalHarga

i = gen ke- *n* = jumlah gen

berat = berat bahan makanan sebenarnya

harga = harga bahan makanan per 100 gr

$$\text{penalti} \begin{cases} 0 & totHarga - anggaran < 0 \\ totHarga - anggaran & totHarga - anggaran > 0 \end{cases} \dots\dots\dots 4-9$$

Keterangan:

penalti = nilai hasil pembobotan

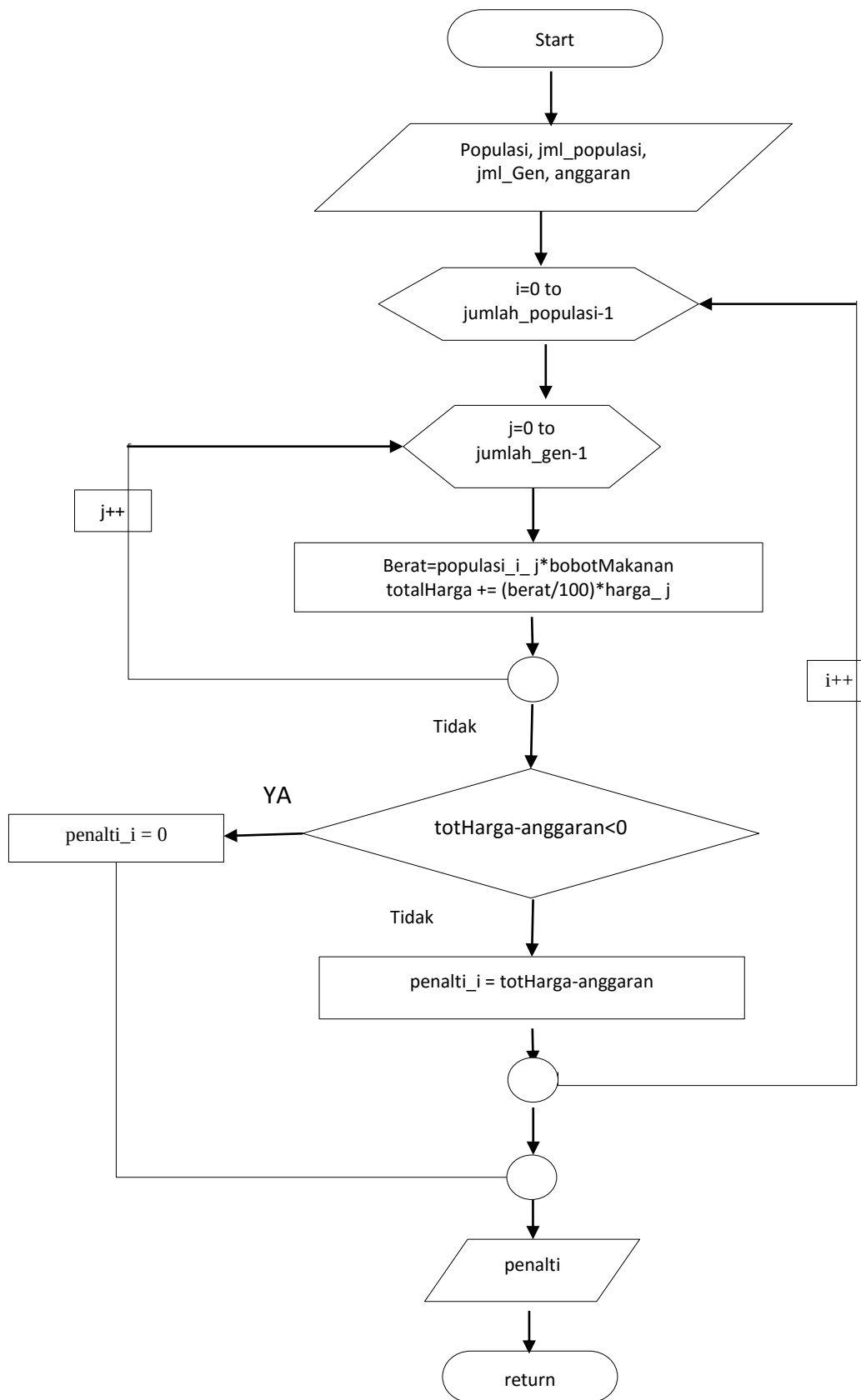
totHarga = total harga bahan makanan pada suatu kromosom

anggaran = anggaran bahan makanan (pagi, snack 1, siang, snack 2, dan sore)

Berikut langkah-langkah perhitungan penalti harga:

1. Menghitung berat bahan makanan sebenarnya berdasarkan nilai gen dengan menggunakan Persamaan 4.1
2. Menghitung total harga berdasarkan berat bahan makanan sebenarnya yang sudah didapat menggunakan Persamaan (4.8).
3. Menghitung nilai penalti menggunakan Persamaan (4.9).

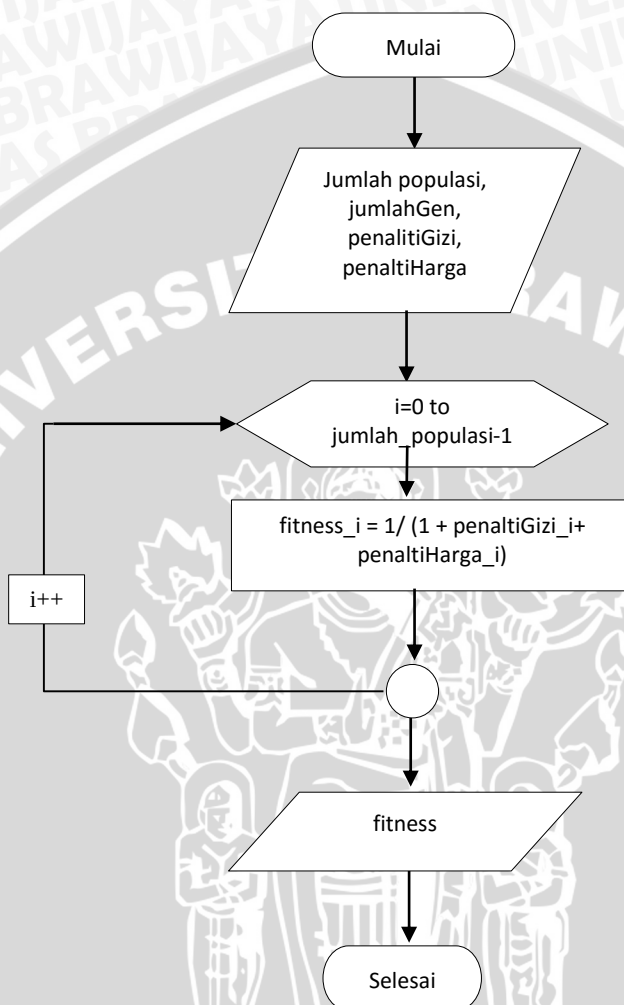
Flowchart penghitungan penalti harga dapat dilihat pada Gambar 4.7 berikut:



Gambar 4. 6 Flowchart Penalti Harga

4.2.5 Menghitung Nilai *Fitness*

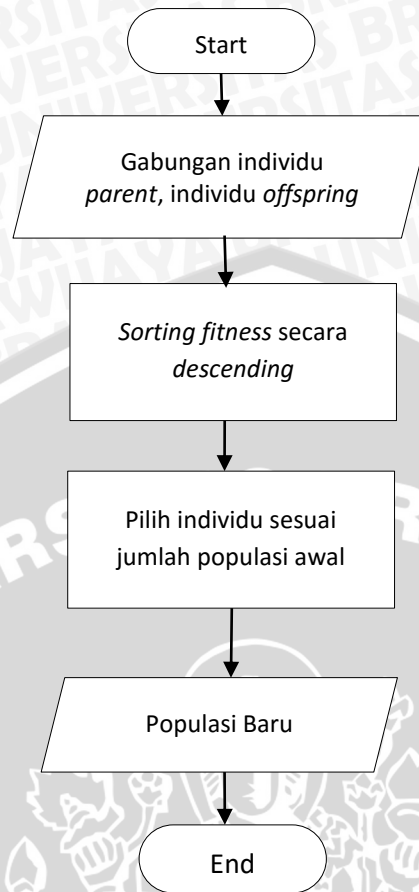
Sebelum melakukan proses seleksi, terlebih dahulu dilakukan proses perhitungan nilai *fitness*. Semua individu, baik *parent* maupun *offspring* hasil reproduksi dihitung nilai *fitness*-nya. *Flowchart* proses perhitungan nilai *fitness* ditunjukkan pada Gambar 4.5.



Gambar 4. 7 *Flowchart* Perhitungan *Fitness*

4.2.6 Seleksi

Proses terakhir dalam *evolution strategies* adalah proses seleksi di mana pada proses ini sistem akan melakukan seleksi tiap individu yang dibawa ke generasi berikutnya. Proses seleksi yang akan diterapkan pada tipe proses ES ($\mu+\lambda$) melibatkan *parent* dan *offspring*. Metode seleksi menggunakan metode *elitism*, dengan mengurutkan semua individu berdasarkan nilai *fitness* terbesar untuk selanjutnya akan terpilih sebanyak populasi berdasarkan urutan tertinggi. *Flowchart* proses seleksi dapat dilihat pada Gambar 4.9 berikut :



Gambar 4. 8 Flowchart Proses Seleksi

Langkah-langkah yang digunakan dalam proses seleksi seperti yang termuat dalam Gambar 4.9 adalah sebagai berikut :

1. Individu *parent* dengan individu *offspring* digabung menjadi satu beserta nilai *fitness* dari masing – masing individu tersebut
2. Kemudian sistem akan melakukan *sorting fitness* secara *descending* (dari terbesar ke terkecil)
3. Memilih individu sesuai dengan jumlah populasi yang telah di tentukan di awal untuk menjadi *parent* di generasi berikutnya
4. Terbentuk suatu populasi baru

4.3 Perhitungan Manual

Dimisalkan data pasien sebagai berikut:

Jenis kelamin : Perempuan

Usia : 45 tahun

Berat badan : 65 kg

Tinggi badan : 150 cm

Penyakit : Diabetes melitus

Aktivitas : Sedang

Perhitungan kebutuhan energi pasien dengan menggunakan persamaan (2.3) pada kondisi diatas adalah:

$$\text{Nilai AMB} = 655 + (9,6 * 65) + (1,8 * 150) - (4,7 * 45) = 1337.5$$

$$\text{Keb. Energi} = 1337.5 \times 1,7 = 2273.75 \text{ kkal}$$

Pasien diet diabetes melitus dengan 2273.75 kkal membutuhkan jumlah gizi sebagai berikut:

$$\text{Karbohidrat} = 2273.75 * 0,65 = 1477.9375 \text{ Kal} \approx 369.484375 \text{ gr}$$

$$\text{Lemak} = 2273.75 * 0,2 = 454.7500 \text{ Kal} \approx 50.52777778 \text{ gr}$$

$$\text{Protein} = 2273.75 * 0,15 = 341.0625 \text{ Kal} \approx 85.265625 \text{ gr}$$

4.3.1 Parameter Algoritma Evolution Strategies

Parameter yang diinisialisasi adalah ukuran Populasi (miu), ukuran Offspring (lambda), ukuran Generasi, dan prioritas gizi. Parameter parameter tersebut diperlukan pada proses algoritma Evolution Strategies.

ukuran Populasi (miu) = 3

ukuran Offspring (lambda) = 1

ukuran Generasi = 1

4.3.2 Generate Populasi Awal

Dimisalkan total anggaran yang disiapkan untuk membeli bahan makanan adalah Rp. 45000 dan bahan makanan yang dikonsumsi adalah sebagai berikut:

Tabel 4. 1 Tabel bahan makanan yang dipilih

Pilihan bahan makanan		Kalori	Karbo	Lemak	Protein	Harga
Makan Pagi	roti putih	248	50	1.2	8	3300
	susu skim	36	5.1	0.1	3.5	2000
Snack 1	agar agar	0	0	0.2	0	5000
Makan Siang	bayam	36	6.5	0.5	3.5	800
	beras giling	360	78.9	0.7	6.8	1000
	daun bawang	29	5.2	0.7	1.8	500
	ayam	302	0	25	18.2	2780
Snack 2	apel	58	14.9	0.4	0.3	1800
	jeruk manis	45	11.2	0.2	0.9	1300
Makan Sore	beras giling	360	78.9	0.7	6.8	1000
	brokoli	34	6.64	0.37	2.82	1500
	daging sapi	207	0	14	18	9500
	tahu	68	1.6	4.6	7.8	600
	seledri	20	4.6	0.1	1	1500

Proses generate populasi awal merupakan proses untuk membuat populasi awal sejumlah ukuran populasi yang telah ditentukan, yaitu 3. Jumlah gen dari setiap kromosom nilainya adalah sama, sesuai dengan jumlah bahan makanan yang dikonsumsi, yaitu 14 (2 bahan makan pagi, 1 bahan snack 1, 4 bahan makan siang, 2 bahan snack 2, dan 4 bahan makan sore). Nilai gen merupakan hasil pembangkitkan bilangan acak antara 0 sampai 1. Dimana, bilangan tersebut merepresentasikan berat bahan makanan. Hasil generate populasi awal dapat dilihat pada Tabel 4.2.

Tabel 4. 2 Individu pada populasi awal

Parent	Kromosom	Fitness
P1	[0.79 ,0.37, 0.18, 0.49, 0.81, 0.19, 0.07, 0.62, 0.48, 0.54, 0.30, 0.70, 0.79, 0.58]	0.000064120
P2	[0.73, 0.79, 0.33, 0.95, 0.19, 0.24, 0.15, 0.61, 0.14, 0.05, 0.65, 0.75, 0.55, 0.40]	0.000052014
P3	[0.63,0.94,0.17,0.94,0.16,0.17,0.08,0.24,0.85,0.27,0.80, 0.03, 0.96, 0.80]	0.000787525

4.3.3 Reproduksi

Setelah proses inialisasi awal pada Tabel 4.2 kemudian dilakukan proses reproduksi. Oleh karena pada penelitian ini menggunakan tipe proses ES ($\mu + \lambda$), maka pada proses reproduksi hanya akan melakukan mutasi dan tanpa proses rekombinasi. Pada setiap gen yang ada pada Tabel 4.2 memiliki nilai level mutasi (σ). Di mana σ ini dibangkitkan dengan nilai random antara 0 sampai dengan 1. Oleh karena pada inialisasi awal terdapat 3 parent maka jumlah offspring juga ada 3. Proses mutasi pada individu Tabel 4.2 akan dijelaskan di bawah ini :

P1 = [0.79 ,0.37, 0.18, 0.49, 0.81, 0.19, 0.07, 0.62, 0.48, 0.54, 0.30, 0.70, 0.79, 0.58]
dan nilai sigma mutasi nya σ_1 = [0.1946, 0.9814, 0.2367, 0.7349, 0.5047, 0.8653, 0.7424, 0.1487, 0.2944, 0.7910, 0.4842]

Maka akan dihasilkan offspring C1 = (x_1' , x_2' x_{14}') dan sigma mutasi $\sigma' = (\sigma_1', \sigma_2'..... \sigma_{14}')$

Dari persamaan 2-2 diperoleh :

$$x_1' = x_1 + \sigma_1 N(0,1)$$

$$x_2' = x_2 + \sigma_2 N(0,1)$$

Sedangkan $N(0,1)$ didapatkan dari hasil persamaan 2-3 sebagai berikut :

$$N(0,1) = \sqrt{-2 \cdot \ln r_1} \cdot \sin 2\pi r_2 \text{ dengan } r_1 = 0.3921 \text{ dan } r_2 = 0.0620$$

$$N(0,1) = \sqrt{-2 \cdot \ln 0.686} \cdot (\sin(2 \cdot 180 \cdot 0.473))$$

= 0.5194 Sehingga didapatkan :

$$x_1' = x_1 + \sigma_1 N(0,1)$$

$$= 0.79 + 0.0.1946 * 0.5194 = 0.8921$$

Jadi dari 3 parent pada Tabel 4.2 memiliki offspring yang dapat ditunjukkan pada Tabel 4.3

Tabel 4. 3 Individu Hasil Mutasi

Offspring	Kromosom	Fitness
C1	[0.89, 0.61, 0.09, 0.18, 0.21, 0.06, 1.2126, 0.72, 0.51, 0.64, 0.07, 0.55, 0.89, 1.12]	4.304905E-05
C2	[0.18, 0.07, 0.28, 0.42, 0.71, 0.12, 0.86, 0.61, 0.37, 0.25, 0.85, 0.40, 0.54, 0.32]	2.067667E-04
C3	[0.82, 0.94, 0.13, 0.86, 0.13, 0.61, 0.29, 0.90, 1.29, 0.35, 0.36, 0.05, 0.21, 0.92]	2.043703E-04

Dari tabel 4.3 dapat diketahui bahwa C1 memiliki nilai fitness 6.81097E-05, C2 memiliki nilai 2.86621E-05, dan C3 memiliki nilai fitness 2.60891E-05. Ketika nilai fitness offspring lebih baik dari parent, maka nilai σ dinaikkan dengan rumusan $\sigma' = \sigma \times 1,1$. Dan jika nilai fitness offspring lebih buruk dari parent, maka nilai σ diturunkan dengan rumusan $\sigma' = \sigma \times 0,9$.

4.3.4 Perhitungan Penalti Gizi

Gen ke-	1	2	3	4	5	6	7	8	9	10	11	12	13	14
P1	0.79	0.37	0.18	0.49	0.81	0.19	0.07	0.66	0.48	0.54	0.30	0.70	0.79	0.58

Perhitungan pertama, menghitung berat sebenarnya dengan menggunakan persamaan (4.1) yaitu mencari berat bahan makanan dengan rumus :

$$\text{berat} = \text{gen} \times 350 \text{ g} \dots\dots\dots 4- 1$$

Berat makanan sebenarnya untuk makan pagi:

Gen 1: $0.79 \times 350 = 276.8 \text{ g}$

Gen 2: $0,37 \times 350 = 130.1 \text{ g}$

Berat makanan sebenarnya untuk snack 1:

Gen 3: $0.18 \times 350 = 66.4 \text{ g}$

Perhitungan dihitung seterusnya sampai gen ke 11.

Lalu, menghitung berat komponen penyusun energi, karbohidrat, lemak, dan protein untuk setiap makanya dengan menggunakan persamaan (4.2), (4.3),(4.4),(4.5) dengan rumus :

$$\begin{aligned} eg &= \frac{\text{berat}}{100} \times \text{nilaiGiziEgi} \dots\dots\dots 4- 2 \\ kb &= \frac{\text{berat}}{100} \times \text{nilaiGiziKbi} \dots\dots\dots 4- 3 \\ lm &= \frac{\text{berat}}{100} \times \text{nilaiGiziLmi} \dots\dots\dots 4- 4 \\ pt &= \frac{\text{berat}}{100} \times \text{nilaiGiziPti} \dots\dots\dots 4- 5 \end{aligned}$$

Bahan makan pagi :

Untuk 276.8 g roti putih mengandung:

$$\text{Energi} = \frac{276.8}{100} \times 248 = 686.6244 \text{ Kal}$$

$$\text{Karbohidrat} = \frac{276.8}{100} \times 50 = 138.4323 \text{ g}$$

$$\text{Lemak} = \frac{276.8}{100} \times 1.2 = 3.3224 \text{ g}$$

$$\text{Protein} = \frac{276.8}{100} \times 8 = 22.1492 \text{ g}$$

Untuk susu skim 130.1 g mengandung :

$$\text{Energi} = \frac{130.1}{100} \times 36 = 46.8705 \text{ Kal}$$

$$\text{Karbohidrat} = \frac{130.1}{100} \times 5.1 = 6.64 \text{ g}$$

$$\text{Lemak} = \frac{130.1}{100} \times 0.1 = 0.13 \text{ g}$$

$$\text{Protein} = \frac{130.1}{100} \times 3.5 = 4.55 \text{ g}$$

Setelah menghitung menghitung berat komponen penyusun energi, karbohidrat, lemak, dan protein untuk setiap makanannya, menghitung total dari komponen yang sudah dihitung tadi.

$$\text{Total Energi} = 686.6244 + 46.8705 = 733.4949$$

$$\text{Total Karbohidrat} = 138.4323 + 6.6400 = 145.0723$$

$$\text{Total Lemak} = 3.3224 + 0.1302 = 3.4526$$

$$\text{Total Protein} = 22.1492 + 4.5569 = 26.7060$$

Setelah itu, menghitung nilai penalti per jenis menu makan untuk perhitungan penalti gizi (4.6) dengan rumus :

$$\text{penaltiMenui} = |(egi - (kebEnergi * BJM))| + |(kbi - (kebKarbohidrat * BJM))| + |(lmi - (kebLemak * BJM))| + |(pti - (kebProtein * BJM))| \dots\dots\dots 4- 10$$

Berdasarkan data yang dimasukkan pasien tadi, di dapatkan hasil kebutuhan energi, karbohidrat, lemak dan protein yaitu :

Keb. Energi = 2273.75 kkal
 Karbohidrat = 369.484375 gr
 Lemak = 50.52777778 gr
 Protein = 85.265625 gr

Selisi Energi = $|733.4949 - (2273.75 \times 0.25)| = 165.0574$
 Selisi Karbohidrat = $|145.0723 - (369.48 \times 0.25)| = 52.7012$
 Selisi Lemak = $|3.4526 - (50.52 \times 0.25)| = 9.1794$
 Selisi Protein = $|26.7060 - (85.26 \times 0.25)| = 5.3896$

Setelah diketahui nilai penalti per jenis menu makan, maka dilakukan perhitungan nilai total penalti dengan menggunakan persamaan (4.7).

$penaltiMenu = 165.0574 + 52.7012 + 9.1794 + 5.3896 = 232.3277$

Selanjutnya dilanjutkan dengan cara yang sama sampai gen ke 11.

Maka, di dapatkan penaltiMenu dari masing-masing bahan untuk parent 1 yaitu :

No	Bahan	Nilai Penalti Bahan
1	Makan Pagi	232.3277
2	Snack 1	277.7699
3	Makan siang	647.2528
4	Snack 2	70.6971
5	Makan Sore	1062.004388

Sehingga, nilai penaltinya adalah total dari nilai penalti menu yang didapatkan :

$penalti = 232.3277 + 277.7699 + 647.2528 + 70.6971 + 1062.004388 = 2290.0518$

Dengan cara yang sama, nilai penalti gizi individu yang lain adalah seperti pada Tabel 4.4:

Tabel 4. 4 Nilai Penalti Gizi

No	Individu	Niali Penalti Gizi
1	Parent 1	2290.0518
2	Parent 2	1196.9252
3	Parent 3	1268.8006
4	Child 1	3102.7226
5	Child 2	2596.3336
6	Child 3	2601.8701

4.3.5 Penalti Harga

Untuk perhitungan penalti harga dilakukan menggunakan persamaan (4.8) yaitu :

$$totHarga = \sum_{i=0}^n \frac{berat}{100} \times harga_i \dots \dots \dots 4-8$$

Total harga bahan makanan untuk makan pagi:

$$\text{Untuk } 276.8 \text{ g roti} = \frac{276.8}{100} \times 3300 = 9136.534121$$

$$\text{Untuk } 130.1 \text{ g telur ayam} = \frac{130.1}{100} \times 2000 = 2603.918943$$

Setelah itu dicari total harga.

$$totHarga = 9136.53 + 2603.91 + 3322.21 + 1389.52 + 2861.88 + 333.95 + 701.57 + 4215.68 + 2203.26 + 1890.64 + 1584.08 + 23327.94 + 1660.69 + 3072.76 = 58304.69$$

Lalu jika total harga lebih kecil dari anggaran (45.000), maka nilai penalti harganya adalah 0.

Dengan cara yang sama, nilai penalti harga individu yang lain adalah seperti pada Tabel 4.5 (lihat Lampiran B Perhitungan Manual):

Tabel 4. 5 Nilai Penalti Harga

No	Individu	Penalti
1	Parent 1	13304.6910
2	Parent 2	18027.6698
3	Parent 3	0
4	Child 1	20125.5924
5	Child 2	2239.034
6	Child 3	2290.2078

4.3.6 Fitness

Untuk perhitungan fitness dengan menggunakan persamaan (2.6) dimana seperti dibawah ini :

$$f = \frac{1}{penaltiGizi + penaltiHarga + 1} \dots \dots \dots 2-6$$

Maka, dengan menggunakan persamaan 2-6 didapatkan :

$$Fitness = \frac{1}{2290.0518 + 13304.6910 + 1} = 0.000064120$$

Dengan cara yang sama, nilai fitness kromosom-kromosom yang lain adalah seperti pada Tabel 4.6 :

Tabel 4. 6 Nilai Fitness

No	Individu	Fitness
1	Parent 1	0.135426714
2	Parent 2	0.072631478
3	Parent 3	0.041425619
4	Child 1	6.810969E-05
5	Child 2	2.608914E-05
6	Child 3	2.866207E-05

4.3.7 Seleksi

Setelah dilakukan tahap reproduksi, pada tahapan seleksi semua individu parent dan offspring digabung untuk nantinya dipilih individu baru sejumlah populasi awal yang akan dijadikan parent pada generasi berikutnya.

Tabel 4. 7 Individu Gabungan

Individu	Kromosom	Fitness
P1	[0.79, 0.37, 0.18, 0.49, 0.81, 0.19, 0.07, 0.62, 0.48, 0.54, 0.30, 0.70, 0.79, 0.58]	0.000064120
P2	[0.73, 0.79, 0.33, 0.95, 0.19, 0.24, 0.15, 0.61, 0.14, 0.05, 0.65, 0.75, 0.55, 0.40]	0.000052014
P3	[0.63, 0.94, 0.17, 0.94, 0.16, 0.17, 0.08, 0.24, 0.85, 0.27, 0.80, 0.03, 0.96, 0.80]	0.000787525
C1	[0.89, 0.61, 0.09, 0.18, 0.21, 0.06, 1.2126, 0.72, 0.51, 0.64, 0.07, 0.55, 0.89, 1.12]	4.304905E-05
C2	[0.18, 0.07, 0.28, 0.42, 0.71, 0.12, 0.86, 0.61, 0.37, 0.25, 0.85, 0.40, 0.54, 0.32]	2.067667E-04
C3	[0.82, 0.94, 0.13, 0.86, 0.13, 0.61, 0.29, 0.90, 1.29, 0.35, 0.36, 0.05, 0.21, 0.92]	2.043703E-04

Kemudian sistem akan melakukan sorting fitness secara descending (dari terbesar ke terkecil)

Tabel 4. 8 Individu gabungan berdasarkan nilai fitness terbesar

Individu	Kromosom	Fitness
P3	[0.63, 0.94, 0.17, 0.94, 0.16, 0.17, 0.08, 0.24, 0.85, 0.27, 0.80, 0.03, 0.96, 0.80]	0.000787525

P1	[0.79, 0.37, 0.18, 0.49, 0.81, 0.19, 0.07, 0.62, 0.48, 0.54, 0.30, 0.70, 0.79, 0.58]	0.000064120
P2	[0.73, 0.79, 0.33, 0.95, 0.19, 0.24, 0.15, 0.61, 0.14, 0.05, 0.65, 0.75, 0.55, 0.40]	0.000052014
C2	[0.18, 0.07, 0.28, 0.42, 0.71, 0.12, 0.86, 0.61, 0.37, 0.25, 0.85, 0.40, 0.54, 0.32]	2.067667E-04
C3	[0.82, 0.94, 0.13, 0.86, 0.13, 0.61, 0.29, 0.90, 1.29, 0.35, 0.36, 0.05, 0.21, 0.92]	2.043703E-04
C1	[0.89, 0.61, 0.09, 0.18, 0.21, 0.06, 1.2126, 0.72, 0.51, 0.64, 0.07, 0.55, 0.89, 1.12]	4.304905E-05

Memilih individu sesuai dengan jumlah populasi yang telah ditentukan di awal untuk menjadi parent di generasi berikutnya.

Tabel 4. 9 Individu yang lolos ke generasi berikutnya

Individu	Kromosom	Fitness
P3	[0.63,0.94,0.17,0.94,0.16,0.17,0.08,0.24,0.85,0.27,0.80, 0.03, 0.96, 0.80]	0.000787525
P1	[0.79 ,0.37, 0.18, 0.49, 0.81, 0.19, 0.07, 0.62, 0.48, 0.54, 0.30, 0.70, 0.79, 0.58]	0.000064120
P2	[0.73, 0.79, 0.33, 0.95, 0.19, 0.24, 0.15, 0.61, 0.14, 0.05, 0.65, 0.75, 0.55, 0.40]	0.000052014

Terbentuk suatu populasi baru

Tabel 4. 10 Populasi Baru

Individu	Kromosom	Fitness
P1	[0.63,0.94,0.17,0.94,0.16,0.17,0.08,0.24,0.85,0.27,0.80, 0.03, 0.96, 0.80]	0.000787525
P2	[0.79 ,0.37, 0.18, 0.49, 0.81, 0.19, 0.07, 0.62, 0.48, 0.54, 0.30, 0.70, 0.79, 0.58]	0.000064120
P3	[0.73, 0.79, 0.33, 0.95, 0.19, 0.24, 0.15, 0.61, 0.14, 0.05, 0.65, 0.75, 0.55, 0.40]	0.000052014

4.2 Perancangan User Interface

Dalam perancangan user interface pada optimasi biaya dan asupan gizi pasien diet khusus dengan menggunakan algoritma evolution strategies ini terdiri 3 halaman utama, yaitu Beranda, data dan proses ES, bahan makanan.

4.2.1 Rancangan Halaman Beranda

Tampilan menu utama berfungsi untuk input data gizi, makanan yang dikonsumsi, serta parameter algoritma evolution strategies. Rancangan user interface tampilan menu utama dapat dilihat pada Gambar 4.10 berikut:

Gambar 4. 9 Gambar Halaman Utama

Keterangan :

1. Input data pasien berupa jenis penyakit
2. Input data pasien berupa jenis kelamin
3. Input data pasien berupa usia
4. Input data pasien berupa berat badan
5. Input data pasien berupa tinggi badan
6. Input data pasien berupa jenis aktivitas (ringan, sedang, berat)
7. Tombol menghitung kebutuhan gizi pasien
8. Hasil perhitungan angka metabolisme basal pasien
9. Hasil perhitungan kebutuhan energi pasien (kal)
10. Hasil perhitungan kebutuhan karbohidrat pasien (gr)
11. Hasil perhitungan kebutuhan lemak pasien (gr)
12. Hasil perhitungan kebutuhan protein pasien (gr)
13. Input parameter algoritma ES untuk ukuran populasi

14. Input parameter algoritma ES untuk ukuran *offspring*
15. Input parameter algoritma ES untuk ukuran generasi
16. Input data bahan makanan berupa anggaran bahan makanan dalam satu hari (Rp)
17. Input jenis bahan makanan (makanan pagi, snack1, makan siang, snack 2, makan sore)
18. Memilih bahan makanan yang dikonsumsi
19. Daftar bahan makanan berdasarkan yang dipilih pasien
20. Tombol untuk mengambil bahan makan
21. Tombol untuk menghapus bahan makanan
22. Tombol untuk mengatur ulang bahan makan

4.2.2 Rancangan Halaman Data dan Proses Algoritma Evolution Strategies

Tampilan data dan proses algoritma *evolution strategies* berisi data beserta hasil proses algoritma *evolution strategies*. Rancangan user interface tampilan data dan proses algoritma *evolution strategies* dapat dilihat pada Gambar 4.11 berikut:



Gambar 4. 10 Halaman Data dan Proses ES

Keterangan :

1. Hasil seleksi Algoritma ES pada generasi terakhir yang berisi nama kromosom, susunan kromosom, nilai fitness, harga, nilai energi, nilai protein, nilai lemak, dan nilai karbohidrat

2. Hasil kromosom terbaik dengan detail nama kromosom, susunan kromosom, nilai fitness, harga, nilai energi, nilai protein, nilai lemak, dan nilai karbohidrat

4.3 Perancangan Uji Coba dan Evaluasi

Untuk mengevaluasi program diperlukan uji coba untuk mendapatkan nilai parameter yang optimal. Uji coba tersebut antara lain:

1. Uji coba untuk menentukan ukuran populasi dan offspring yang optimal.
2. Uji coba untuk menentukan banyaknya generasi yang optimal.

4.3.1 Rancangan Uji Coba Ukuran Populasi (μ) dan Offspring (λ)

Pada pengujian ini dilakukan *input* pada *inputan* ukuran populasi (μ) dan ukuran offspring (λ). Pada penelitian ini, dilakukan pengujian pada ukuran populasi sebesar 20, 40, 60, 80, 100, 150, 200 dan 250. Sedangkan untuk ukuran offspring sebesar 1 μ , 2 μ , 3 μ , 4 μ , 5 μ , 6 μ , 7 μ , 8 μ , 9 μ , dan 10 μ . Pengujian ini bertujuan untuk mengetahui pengaruh dari input ukuran populasi (μ) dan input ukuran offspring (λ). Setelah didapatkan nilai *fitness* dari masing-masing kombinasi μ dan λ , akan terlihat pula rata-rata nilai *fitness* terbaik. Sehingga kemudian dapat dilakukan analisis terhadap pengaruh ukuran populasi (μ) dan ukuran offspring (λ) tersebut. Rancangan uji coba ukuran populasi dan offspring dapat dilihat pada Tabel 4.11 dan Tabel 4.12

Tabel 4. 11 Rancangan Uji Coba Banyaknya Populasi

Banyak populasi	generasi , λ					Rata-rata nilai <i>fitness</i>
	Percobaan ke-					
	1	2	3	4	5	
20						
40						
60						
80						
100						
150						
200						
250						

Tabel 4. 12 Tabel Rancangan Uji Coba Ukuran Offspring

4.3.2 Rancangan Uji Coba Banyak Generasi

Untuk ukuran generasi yang digunakan adalah 50, 100, 150, 200, 250 dan 300. Untuk pengujian dari setiap ukuran generasi dilakukan sebanyak 5 kali uji

yang kemudian diambil rata-ratanya. Berikut adalah rancangan uji banyaknya generasi yang dapat ditunjukkan pada Tabel 4.13

Banyak Offspring	generasi, populasi					Rata-rata nilai <i>fitness</i>
	Percobaan ke-					
	1	2	3	4	5	
1μ						
2μ						
3μ						
4μ						
5μ						
6μ						
7μ						
8μ						
9μ						
10μ						
11μ						

Tabel 4. 13 Rancangan Uji Coba Ukuran Generasi

Banyak Generasi	μ , λ					Rata- rata nilai <i>fitness</i>
	Percobaan ke-					
	1	2	3	4	5	
50						
100						
150						
200						
250						
300						


```

4 public Makanan [] dataMakanan;
5 public Makanan(int idPenyakit) throws IOException{
6     setAllMakanan(idPenyakit);
7 }
8 public Makanan (String nam,double eg,double karb,double lmk,double, double hrg ){
9     nama=nam;
10    energi=eg;
11    karbo=karb;
12    lemak= lmk;
13    protein= prot;
14    harga=hrg;
15 }
16 public void setAllMakanan(int idPenyakit)throws IOException{
17     File inputWorkbook = new File(inputFile);
18     Workbook w;
19     try {
20         w = Workbook.getWorkbook(inputWorkbook);
21         Sheet sheet = w.getSheet(idPenyakit);
22         dataMakanan = new Makanan [sheet.getRows()];
23         for (int i = 1; i < sheet.getRows(); i++) {
24             int id = i;
25             String nam = sheet.getCell(0,id).getContents();
26             double eg = Double.parseDouble(sheet.getCell(1,id).getContents());
27             double karb = Double.parseDouble(sheet.getCell(2,id).getContents());
28             double lmk = Double.parseDouble(sheet.getCell(3,id).getContents());
29             double prot = Double.parseDouble(sheet.getCell(4,id).getContents());
30             double hrg = Double.parseDouble(sheet.getCell(16,id).getContents());
31             dataMakanan[i] = new Makanan(nam,eg,karb,lmk,prot,hrg);
32         }
33     } catch (BiffException e) {
34         e.printStackTrace();
35     }
36 }
37 public Makanan getMakanan(int id){
38     return dataMakanan[id];
39 }

```

Gambar 5. 1 Source Code Mengambil Data Makanan

Penjelasan *source code* dari Gambar 5.1 adalah sebagai berikut :

1. Baris 2-3 merupakan proses mendeklarasikan variable yang ada di data makanan, dimana di dalam data makanan ada nama, energi, karbo, lemak, protein dan harga.
2. Baris 4-5 merupakan proses menunjukkan path atau lokasi pengambilan data data dokumen excel.
3. Baris 7-12 merupakan konstruktor untuk memberikan nilai awal saat objek makanan dibuat.

4. Baris 18- 28 merupakan proses pengambilan data dari dokumen excel.

5.2.2 Implementasi Inisialisasi Populasi Awal

Pada Gambar 5.2.1 berikut menunjukkan potongan source code program pada proses pembentukan populasi awal. Proses pembentukan populasi awal dilakukan dengan memberi nilai pada gen dari setiap individu kromosom. Jumlah individu tiap kromosom ditentukan oleh jumlah inputan bahan makan dari user. Proses pembentukan populasi awal dapat dilihat pada Source Code 5.2 berikut:

1	public class Kromosom {
2	private double isi_kromosom[];
3	private double sigma[];
4	public Kromosom(int jml_mknan) {
5	isi_kromosom = new double[jml_mknan];
6	sigma = new double[isi_kromosom.length];
7	for (int i = 0; i < jml_mknan; i++) {
8	isi_kromosom[i] = Math.random();
9	sigma[i] = Math.random();
10	}
11	}

Gambar 5. 2 Source Code Inisialisasi Populasi Awal

Penjelasan *source code* dari Gambar 5.2 adalah sebagai berikut :

1. Baris 1-3 merupakan proses pembuatan kelas bernama kromosom.
2. Baris 4 merupakan pembuatan objek baru bernama isi kromosom bertipe data array double dimana jumlah individu kromosom berdasarkan jumlah inputan bahan makanan.
3. Baris 6 merupakan pembuatan objeek bernama sigma dimana panjang sigma sesuai panjang isi kromosom.
4. Baris 8-9 merupakan proses pengisian nilai isi kromosom dan sigma dimana disini bilangan random antara 0-1.

5.2.3 Implementasi Mutasi

Berikut potongan *source code* program proses mutasi pada *evolution strategies* yang dapat ditunjukkan pada Gambar 5.3

1	public Kromosom mutasi() {
2	double[] r1 = new double[isi_kromosom.length];
3	double[] r2 = new double[isi_kromosom.length];
4	double[] eN = new double[isi_kromosom.length];
5	double[] isi = new double[isi_kromosom.length];

```

6      Kromosom child = new Kromosom(isi_kromosom.length);
7      for (int i = 0; i < isi_kromosom.length; i++) {
8          r1[i] = Math.random();
9          r2[i] = Math.random();
10         eN[i] = Math.sqrt((-2) * Math.log(r1[i])) * Math.sin(2 * Math.PI * r2[i]);
11         isi[i] = Math.abs(isi_kromosom[i] + sigma[i] * eN[i]);
12     }
13     child.setIsi_Kromosom(isi);
14     child.setSigma(this.sigma);
15
16
17     return child;
18 }
19 public void setMutasi() throws IOException {
20     int indexMut = 0;
21     for (int i = 0; i < jumlah_populasi; i++) {
22         boolean findBetter = false;
23         for (int j = 0; j < jumlah_offspring; j++) {
24             arr_mutasi[indexMut] = populasi[i].mutasi();
25             if (arr_mutasi[indexMut].getFitness(pasien) >=
26                 populasi[i].getFitness(pasien)) {
27
28                 findBetter = true;
29                 arr_mutasi[indexMut].updateSigma(1.1);
30             } else {
31                 arr_mutasi[indexMut].updateSigma(0.9);
32             }
33             indexMut++;
34         }
35         if (findBetter == true) {
36             populasi[i].updateSigma(1.1);
37         } else {
38             populasi[i].updateSigma(0.9);
39         }
40     }
41 }
42

```

Gambar 5. 3 Source Code Proses Mutasi

Penjelasan *source code* dari Gambar 5.3 adalah sebagai berikut :

1. Baris 2-5 merupakan proses mendeklarasikan variable r1,r2,eN, isi dimana bertipe array double dengan panjang array isi kromosom.
2. Baris 8-9 merupakan proses isi dari r1 dan r2 adalah nilai acak antara 0-1.
3. Baris 10 merupakan rumus dari perhitungan N.
4. Baris 11 merupakan rumus mutasi dimana disini bernama isi.

5. Baris 19-42 merupakan proses update sigma, dimana jika parent memiliki child dengan fitness lebih besar dari parent minimal satu, maka sigma dikalikan 1.1 untuk proses generasi berikutnya. Tetapi jika parent tidak memiliki child dengan fitness lebih besar dari parent maka sigma parent diturunkan dengan dikali 0.9. begitu juga dengan child jika fitness child lebih besar dari parent maka sigma child dikalikan 1.1 tetapi jika tidak maka diturunkan dengan dikali 0.9.

5.2.4 Perhitungan Fitness

Proses perhitungan nilai *fitness* dapat ditunjukkan pada potongan *source code* pada Gambar 5.4 berikut:

```

1 public double getFitness(Pasien pasien) {
2     double[] berat = new double[isi_kromosom.length];
3     double[] kalori = new double[isi_kromosom.length];
4     double[] protein = new double[isi_kromosom.length];
5     double[] karbo = new double[isi_kromosom.length];
6     double[] lemak = new double[isi_kromosom.length];
7     double[] harga = new double[isi_kromosom.length];
8     String[] penyakit = new String[isi_kromosom.length];
9     double[] eg = new double[isi_kromosom.length];
10    double[] prot = new double[isi_kromosom.length];
11    double[] kb = new double[isi_kromosom.length];
12    double[] lmk = new double[isi_kromosom.length];
13    double[] hg = new double[isi_kromosom.length];
14    for (int i = 0; i < isi_kromosom.length; i++) {
15        berat[i] = (350 * isi_kromosom[i]);
16        Makanan a = pasien.daftarMakanan.getMakanan(pasien.makanan[i]);
17        kalori[i] = a.energi;
18        protein[i] = a.protein;
19        lemak[i] = a.lemak;
20        karbo[i] = a.karbo;
21        harga[i] = a.harga;
22        eg[i] = (berat[i] / 100) * kalori[i];
23        kb[i] = (berat[i] / 100) * karbo[i];
24        lmk[i] = (berat[i] / 100) * lemak[i];
25        prot[i] = (berat[i] / 100) * protein[i];
26        hg[i] = (berat[i] / 100) * harga[i];
27    }
28
29    double[][] sum = new double [pasien.panjangMenu.length][4];
30    double[][] selisihMenu = new double [pasien.panjangMenu.length][4];
31    double [] bobot = {0.25,0.1,0.3,0.1,0.25};
32    int count=0;
33    for (int i=0; i < pasien.panjangMenu.length; i++) {

```



```

34      sum[i][0]=0; //kalori
35      sum[i][1]=0; //karbo
36      sum[i][2]=0; //lemak
37      sum[i][3]=0; //protein
38      for (int j=0 ; j < pasien.panjangMenu[i]; j++) {
39          sum[i][0]=sum[i][0]+eg[count];
40          sum[i][1]=sum[i][1]+kb[count];
41          sum[i][2]=sum[i][2]+lmk[count];
42          sum[i][3]=sum[i][3]+prot[count];
43          count++;
44      }
45      selisihMenu[i][0]=Math.abs(sum[i][0]-(pasien.getKalori()*bobot[i]));
46      selisihMenu [i][1]=Math.abs(sum[i][1]-(pasien.getKarbo()*bobot[i]));
47      selisihMenu [i][2]=Math.abs(sum[i][2]-(pasien.getLemak()*bobot[i]));
48      selisihMenu [i][3]=Math.abs(sum[i][3]-(pasien.getProtein()*bobot[i]));
49      double    total    =    selisihMenu[i][0]+selisihMenu[i][1]+selisihMenu
50 [i][2]+selisihMenu[i][3];
51      }
52
53      double penalty_bahan=0;
54      for (int i = 0; i < selisihMenu.length; i++) {
55          for (int j = 0; j < selisihMenu[0].length; j++) {
56              penalty_Menu=penalty_Menu+selisihMenu [i][j];
57          }
58      }
59      double penalty_harga;
60      double sumHg = DoubleStream.of(hg).sum();
61      if ((sumHg - pasien.anggaran) < 0) {
62          penalty_harga = 0;
63      } else {
64          penalty_harga = (sumHg - pasien.anggaran);
65      }
66      double fitness = 1 / (1 + (penalty_Menu + penalty_harga));
67      return fitness;
68  }

```

Gambar 5. 4 Source Code Proses Perhitungan Fitness

Penjelasan *source code* dari Gambar 5.4 adalah sebagai berikut :

1. Baris 2-13 merupakan proses mendeklarasikan variable berat, kalori, protein, karbo, lemak, harga, penyakit, eg, prot, kb, lmk, hg dimana panjang array sesuai dengan panjang isi kromosom.
2. Baris 15 merupakan proses rumus dari berat bahan makanan. Dimana berat disini gen yang di dalam individu kromosom dikalikan 350.

3. Baris 17-21 merupakan proses pengambilan data bahan makanan ke dalam variable yang sudah dibuat sebelumnya khusus untuk makanan.
4. Baris 22-26 merupakan proses perhitungan energi, karbo, lemak, protein, harga tiap-tiap makanan dengan rumus berat yang sudah didapatkan dari sebelumnya dibagi 100 karena berat makanan per 100 gram lalu dikali kandungan dan harga setiap makanan.
5. Baris 29-30 merupakan proses mendeklarasikan variable sum dan selisihMenu bertipe data double array dimana panjang array berdasarkan inputan user yaitu pilihan makanan dan array yang lain berisikan kandungan tiap-tiap makanan yaitu 4 (energi, karbohidrat, lemak, protein).
6. Baris 31 merupakan bobot setiap bahan makanan, dimana untuk makan pagi bobot nya 0.25, snack 1 0.1, makan siang 0.3, sncak 2 0.1, dan makan sore 0.25.
7. Baris 38-43 merupakan proses perhtiungan total gizi tiap-tiap kandungan makanan.
8. Baris 45-48 merupakan proses perhitungan selisih menu makanan tiap-tiap kandungan gizi makanan dengan menggunakan bobot setiap bahan tadi.
9. Baris 49-50 merupakan proses perhitungan jumlah selisih menu.
10. Baris 53-58 merupakan proses perhitungan penalty menu dimana selisish-selisih Menu yang didapatkan per tiap Menu dijumlahkan.
11. Baris 59-65 merupakan proses mencari penalty harga dengan menjumlahkan total harga per tiap makanan yang sudah dikalikan dengan berat tadi lalu jika jumlah harga lebih kecil dari anggaran pasien maka nilai penalty harga 0, tetapi jika jumlah harga lebih besar makan penalty harga ialah jumlah harga dikurang anggaran.
12. Baris 66-68 merupakan proses perhitungan fitness dimana menggunakan perhitungan penalty gizi dan penalty harga yang sudah didaptkan sebelumnya.

5.2.5 Implementasi Seleksi dengan Elitism

Proses seleksi *elitism* pada *evolution strategies* dapat ditunjukkan pada potongan *source code* program pada Gambar 5.5 berikut.

```

1 public void setSeleksi() {
2     for (int i = 0; i < arraySeleksi1.length; i++) {
3         if (i < jumlah_populasi) {
4             arraySeleksi1[i] = populasi[i];
5         } else {
6             arraySeleksi1[i] = arr_mutasi[i - jumlah_populasi];
7         }
8     }
9     //sorting
10    for (int i = 0; i < arraySeleksi1.length; i++) {
11        int pointer = i;
12        double max = arraySeleksi1[i].getFitness(pasien);
13        for (int j = i; j < arraySeleksi1.length; j++) {
14            if (max < arraySeleksi1[j].getFitness(pasien)) {
15                max = arraySeleksi1[j].getFitness(pasien);
16                pointer = j;
17            }
18        }
19        Kromosom temp = arraySeleksi1[pointer];
20        arraySeleksi1[pointer] = arraySeleksi1[i];
21        arraySeleksi1[i] = temp;
22    }
23    //elitism
24    for (int i = 0; i < jumlah_populasi; i++) {
25        populasi[i] = arraySeleksi1[i];
26    }
27 }
28

```

Gambar 5. 5 Source Code Proses Seleksi

Penjelasan *source code* dari Gambar 5.5 adalah sebagai berikut :

1. Baris 1-8 merupakan proses penggabungan antara individu-individu array populasi dan array mutasi(child)
2. Baris 10-9-28 merupakan proses sorting dengan menggunakan array penggabungan tadi, dimana dipilih fitness yang terbesar antaran individu-individu lalu menggantikan posisi menjadi parent setelah diseleksi.

BAB VI PENGUJIAN DAN ANALISIS

Bab ini akan menjelaskan mengenai pengujian pada algoritma *Evolution Strategies* untuk optimasi biaya dan asupan gizi pasien diet khusus. Dalam pengujian ini akan dilakukan empat macam jenis pengujian yaitu pengujian ukuran populasi (*miu*), pengujian ukuran *offspring* (*lambda*), pengujian banyaknya generasi serta pengujian parameter terbaik. Pada pengujian pertama, bertujuan untuk mengetahui pengaruh ukuran populasi (*miu*), pada pengujian kedua bertujuan untuk mengetahui pengaruh ukuran *offspring* (*lambda*). Pada pengujian ketiga, bertujuan untuk mengetahui pengaruh *inputan* banyaknya generasi terhadap nilai *fitness* dimana yang sebelumnya juga telah diinputkan nilai *miu* dan *lambda* pada proses pengujian sebelumnya. Sedangkan pada pengujian keempat, bertujuan untuk mengetahui pengaruh dari parameter-parameter uji terbaik dari masing-masing solusi optimal yang telah dilakukan pada proses pengujian sebelumnya.

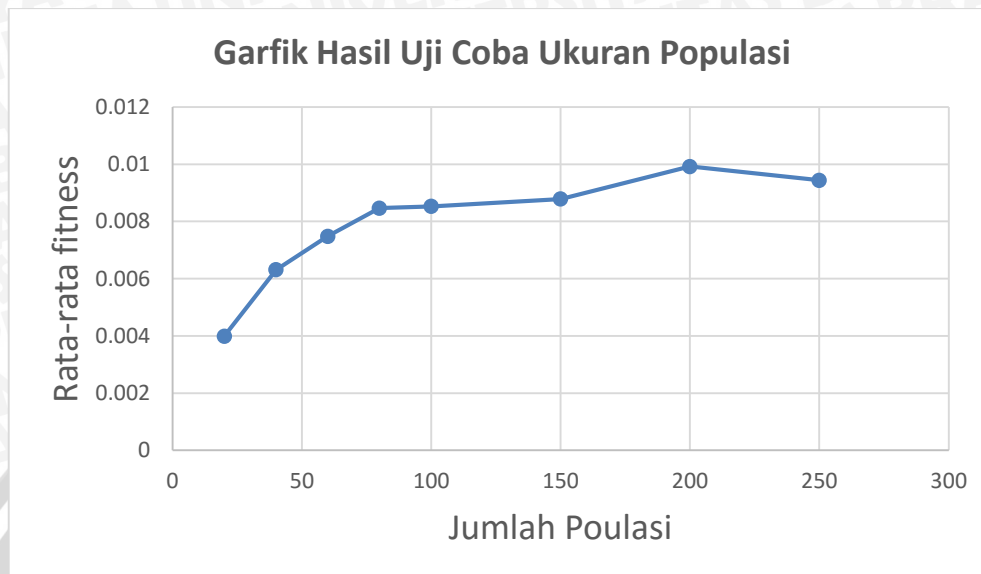
6.1 Hasil dan Analisis Uji Coba Ukuran Populasi (*miu*) dan *Offspring* (*lambda*)

Pada pengujian ini dilakukan *input* pada *inputan* ukuran populasi (*miu*) dan ukuran *offspring* (*lambda*). Pada penelitian ini, dilakukan pengujian pada ukuran variasi populasi sebesar 20, 40, 60, 80, 100, 150, 200 dan 250. Sedangkan untuk ukuran *offspring* sebesar 1 μ , 2 μ , 3 μ , 4 μ , 5 μ , 6 μ , 7 μ , 8 μ , 9 μ , 10 μ dan 11 μ . Pengujian ini bertujuan untuk mengetahui pengaruh dari input ukuran populasi (*miu*) dan input ukuran *offspring* (*lambda*). Setelah didapatkan nilai *fitness* dari masing-masing *miu* dan *lambda*, akan terlihat pula rata-rata nilai *fitness* terbaik. Sehingga kemudian dapat dilakukan analisis terhadap pengaruh ukuran populasi (*miu*) dan ukuran *offspring* (*lambda*) tersebut. Nilai *fitness* dari variasi *inputan* ukuran populasi (*miu*) dan ukuran *offspring* (*lambda*) dapat ditunjukkan pada Tabel 6.1.

Tabel 6. 1 Hasil Coba Ukuran Populasi

Banyak populasi	Generasi=50 , $\lambda = 5\mu$					Rata-rata nilai <i>fitness</i>
	Percobaan ke-					
	1	2	3	4	5	
20	0.00299	0.00299	0.00474	0.002953	0.00628	0.003988
40	0.00718	0.00548	0.00593	0.006738	0.00625	0.006314
60	0.00824	0.00816	0.00792	0.007329	0.00577	0.007484
80	0.01077	0.00867	0.0093	0.005873	0.00773	0.008469
100	0.01064	0.00793	0.00732	0.006679	0.01007	0.008528
150	0.00657	0.01112	0.00882	0.009844	0.00759	0.008788
200	0.010082	0.010661	0.008659	0.0114532	0.008767	0.009924
250	0.010139	0.009994	0.007969	0.009742	0.009362	0.009441

Grafik pengaruh ukuran populasi (μ) dapat ditunjukkan pada Gambar 6.1.



Gambar 6. 1 Grafik Pengaruh Ukuran Populasi (μ)

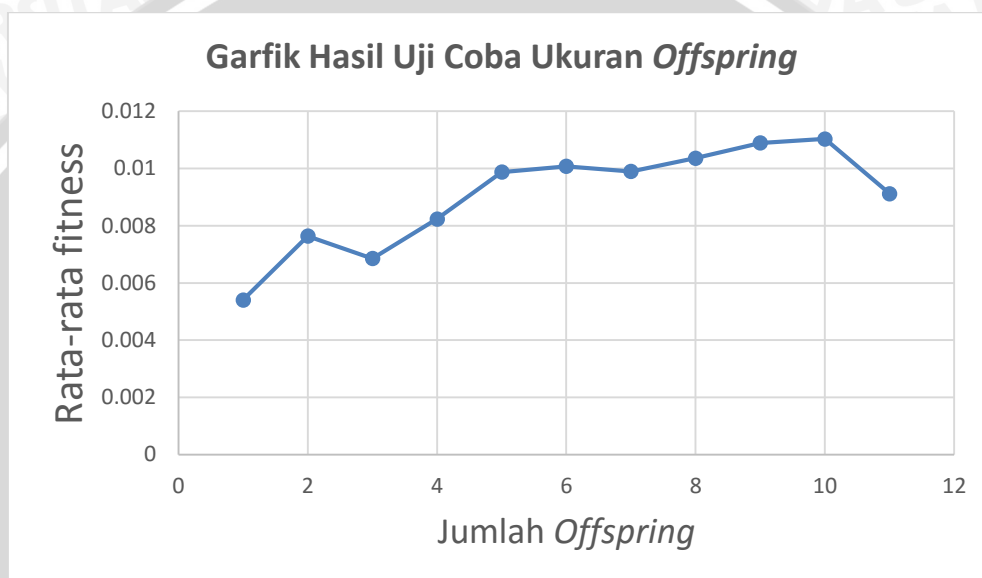
Dari Gambar 6.1 dapat dilihat pada ukuran populasi 20 hingga 200 nilai rata-rata fitness selalu mengalami kenaikan. Kemudian pada ukuran populasi 250 nilai rata-rata fitness mengalami penurunan. Pada umumnya, ukuran populasi berpengaruh terhadap nilai rata-rata fitness yang didapatkan. Ukuran populasi mempengaruhi pembangkitan awal individu dan juga jumlah individu baru yang dihasilkan. Tentunya hal ini akan berpengaruh terhadap variasi fitness yang dihasilkan oleh individu-individu baru. Sehingga peluang untuk mendapatkan nilai fitness yang tinggi semakin besar. Namun, dengan melihat grafik pada Gambar 6.1, ukuran populasi yang besar tidak menjamin akan menghasilkan nilai fitness yang besar. Hal tersebut dikarenakan pembangkitan awal individu yang dilakukan secara random dalam proses algoritma evolution strategies. Dengan adanya proses mutasi dalam ukuran populasi yang besar, memungkinkan jumlah individu baru yang dihasilkan juga semakin besar dan beragam. Selain itu, semakin besar ukuran populasi juga berpengaruh pada waktu pemrosesan algoritma evolution strategies yang semakin lama.

Tabel 6. 2 Hasil Coba Ukuran Offspring

Banyak Offspring	Generasi=50 , populasi=200					Rata-rata nilai <i>fitness</i>
	Percobaan ke-					
	1	2	3	4	5	
1μ	0.00648	0.0049	0.00509	0.005131	0.00543	0.005406
2μ	0.00814	0.00874	0.00579	0.006505	0.00901	0.007636
3μ	0.00792	0.00669	0.00676	0.005439	0.00747	0.006855
4μ	0.00749	0.0077	0.01024	0.007543	0.00821	0.008235
5μ	0.01077	0.00765	0.01027	0.010246	0.01044	0.009877
6μ	0.00813	0.00987	0.0118	0.01051	0.01008	0.010077

7μ	0.007882	0.009689	0.010016	0.0108545	0.011039	0.009896
8μ	0.011082	0.009367	0.009619	0.009557	0.012184	0.010362
9μ	0.010696	0.013401	0.010518	0.0108183	0.009034	0.010893
10μ	0.010087	0.011976	0.010772	0.0097459	0.012583	0.011033
11μ	0.011968	0.007927	0.009995	0.0100679	0.005627	0.009117

Grafik pengaruh ukuran *offspring* (*lambda*) dapat ditunjukkan pada Gambar 6.2.



Gambar 6. 2 Grafik Pengaruh Ukuran *Offspring* (*lambda*)

. Dari Gambar 6.2 dapat dilihat bahwa nilai terbesar *fitness* adalah 0.011033 terjadi ketika ukuran *offspring* sebanyak 10 μ . Dan nilai rata-rata *fitness* terkecil pada *fitness* 0.005406 pada ukuran *offspring* 1 μ . Dari grafik dilihat dengan jumlah *offspring* yang terus bertambah maka rata-rata *fitness* cenderung naik. Namun tidak semua mengalami kenaikan. Hal ini dikarenakan inisialisasi kromosom pada *Evolution Strategies* dibangkitkan secara random

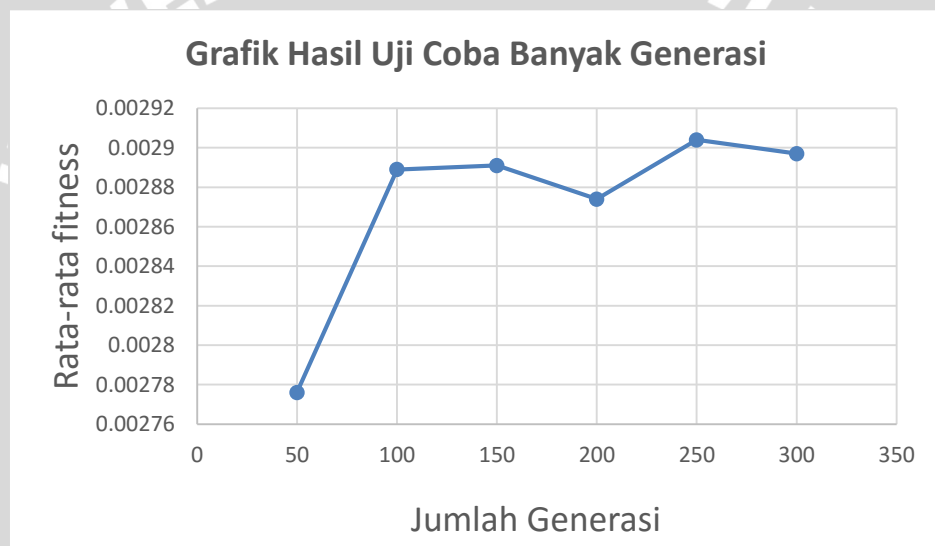
6.2 Hasil dan Analisis Uji Coba Banyak Generasi

Pada pengujian ini dilakukan *input* pada *inputan* ukuran generasi. untuk ukuran generasi yang digunakan adalah 50, 100, 150, 200, 250 dan 300. Pengujian ini bertujuan untuk mengetahui pengaruh dari input ukuran populasi generasi. Setelah didapatkan nilai *fitness* dari masing-masing generasi, akan terlihat pula rata-rata nilai *fitness* terbaik. Sehingga kemudian dapat dilakukan analisis terhadap pengaruh generasi tersebut. Untuk pengujian dari setiap ukuran generasi dilakukan sebanyak 5 kali uji yang kemudian diambil rata-ratanya. Nilai *fitness* dari *inputan* ukuran generasi dapat ditunjukkan pada Tabel 6.3.

Tabel 6. 3 Hasil Coba Banyaknya Generasi

Banyak Generasi	$\mu = 200, \lambda = 7\mu$					Rata-rata nilai <i>fitness</i>
	Percobaan ke-					
	1	2	3	4	5	
50	0.00277	0.00279	0.00281	0.002753	0.00276	0.002776
100	0.00289	0.00285	0.0029	0.002907	0.0029	0.002889
150	0.00287	0.00292	0.00287	0.002882	0.00291	0.002891
200	0.00285	0.00287	0.00289	0.00287	0.00288	0.002874
250	0.00287	0.00292	0.00292	0.002918	0.0029	0.002904
300	0.00293	0.00289	0.00287	0.002903	0.00289	0.002897

Untuk grafik pengaruh banyaknya generasi dapat ditunjukkan pada Gambar 6.3.



Gambar 6. 3 Grafik Pengaruh Ukuran generasi

Dari Gambar 6.3 dapat dilihat bahwa rata-rata nilai *fitness* tertinggi adalah 0,002904 terjadi ketika ukuran generasinya 250 dan rata-rata nilai *fitness* terendah adalah 0,002776 terjadi ketika ukuran generasinya 50. Pada umumnya, banyak jumlah generasi berpengaruh terhadap nilai rata-rata *fitness* yang didapatkan. Namun, dengan melihat grafik pada Gambar 6.3, jumlah generasi yang banyak tidak menjamin akan menghasilkan nilai *fitness* yang besar dan jumlah generasi yang kecil tidak menjamin akan menghasilkan nilai *fitness* yang kecil pula. Hal tersebut dikarenakan pembangkitan awal individu (starting point) yang dilakukan secara random dalam proses algoritma evolution strategies. Pada setiap satu generasi, akan dilakukan beberapa kali proses mutasi, dimana pada proses ini individu baru akan dihasilkan. Semakin sering proses mutasi dilakukan maka individu baru yang terbentuk akan semakin banyak dan variasi nilai *fitness* akan semakin beragam. Sehingga peluang untuk mendapatkan nilai *fitness* yang tinggi semakin besar. Semakin banyak generasi maka semakin besar pula waktu eksekusi

yang dibutuhkan. Selain itu, semakin banyak generasi juga berpengaruh pada waktu pemrosesan algoritma evolution strategies yang semakin lama.

6.3 Hasil dan Analisa Pengujian Sistem

Pengujian sistem dilakukan dengan membandingkan hasil konsultasi menu gizi yang didapatkan secara manual oleh ahli gizi dengan pengujian pada kasus yang sama dari sistem. Dari rumah sakit tersebut didapatkan 4 hasil konsultasi pasien dengan penyakit diabetes melitus, jantung, dan hati. Batas toleransi nilai gizi yang ditetapkan oleh ahli gizi untuk bahan makanan yang direkomendasikan adalah $\pm 10\%$ dari kebutuhan gizi pasien diet khusus. Pengujian dilakukan dengan menggunakan data dari 4 pasien yang berbeda. Parameter algoritma evolution strategies yang digunakan adalah hasil terbaik dari pengujian parameter algoritma evolution strategies.

6.3.1 Kasus 1

Pasien berjenis kelamin perempuan berusia 49 tahun, berat badan 52 kg, tinggi badan 151 cm, dan menderita penyakit diabetes melitus. Kategori aktifitas pasien adalah sedang. Dengan anggaran biaya Rp 15.000 per hari.

Parameter algoritma evolution strategies yang digunakan adalah sebagai berikut: Banyak generasi : 250

Ukuran populasi : 200

Banyak Offspring : 10

Didapatkan kromosom terbaik dengan kombinasi berat bahan makanan yang telah mengalami proses algoritma evolution strategies sebagai berikut:

Tabel 6. 4 Menu Kasus 1

Nama Bahan	Berat Bahan Manual	Berat Bahan (Program)	Harga Bahan Manual (Rp)	Harga Bahan Program(Rp)
Makan Pagi				
Nasi Putih	200 g	25.86 g	2000	258.6
Wortel	30 g	147.99 g	210	1035.93
Buncis	20 g	30.34 g	120	182.04
Tahu	30 g	23.37 g	180	140.22
Ikan Kakap	40 g	4.75 g	1100	130.62
Minyak Kelapa	5 g	16.07 g	65	160.7
Snack 1				
Teh	100 g	0.84 g	1000	8
Gula Pasir	50 g	66.29 g	525	696.04

Makan siang				
Nasi Putih	200 g	123.79 g	2000	1237.9
Wortel	50 g	4.18 g	350	29.26
Jagung	50 g	252.01 g	200	1008
Tahu	30 g	91.61 g	180	549.66
Ikan Kakap	40 g	39.45 g	1100	1084.87
Minyak Kelapa	5 g	11.27 g	65	112.7
Snack 2				
Tomat	30 g	306.09 g	105	1071
Gula Pasir	30 g	44.28 g	210	464.94
Makan sore				
Nasi Putih	200 g	100.77 g	2000	1007
Kangkung	75 g	340.71 g	225	1022
Minyak Kelapa	5 g	13.69 g	65	136.9
Tahu	30 g	35.90 g	180	215.4
Ayam	40 g	0.30 g	1110	8.34

Dengan melakukan pengujian , didapatkan hasil perhitungan kecukupan gizi dari Menu kasus 1 dari sistem adalah sebagai berikut:

Tabel 6. 5 Hasil Pemenuhan Gizi dan jumlah biaya dari untuk Menu Kasus 1

	Energi	Karbohidrat	Lemak	Protein
Kebutuhan Pasien	2424.2 kkal	393.72 g	53.87 g	90.91 g
Hasil sistem	2424.2 kkal	436.72 g	53.92 g	79.85 g
%Gizi Tercukupi	100	110	100	90.89
%Selisih nilai gizi	0%	10%	0%	9.11%
Total biaya Manual	Rp 12.992			
Total biaya Program	Rp 10.557			

Berdasarkan Tabel 6.5 , diperoleh selisish nilai energi yang dibutuhkan pasien tercukupi 0%, selisih nilai gizi kebutuhan karbohidrat pasien tercukupi 10%, selisih nilai gizi kebutuhan lemak pasien tercukupi 0% %, dan selisih nilai gizi kebutuhan protein pasien tercukupi 9.11%. Total biaya yang dihasilkan manual sebesar Rp 12.992 dan total biaya yang dihasilkan system Rp 10.557. Dari hasil pengujian tersebut dapat disimpulkan bahwa nilai gizi dari komposisi bahan

makanan yang direkomendasikan oleh sistem masih dalam batas toleransi yang ditetapkan oleh ahli gizi yaitu $\pm 10\%$ dari kebutuhan gizi pasien diet khusus. Selain itu, hasil biaya yang dihasilkan sistem lebih kecil dari hasil biaya yang dihasilkan secara manual.

6.3.2 Kasus 2

Pasien berjenis kelamin perempuan berusia 47 tahun, berat badan 68,5 kg, tinggi badan 157,8 cm, dan menderita penyakit diabetes melitus. Kategori aktifitas pasien adalah sedang. Dengan anggaran biaya Rp 17.000 per hari.

Parameter algoritma evolution strategies yang digunakan adalah sebagai berikut:

Banyak generasi : 250

Ukuran populasi : 200

Banyak Offspring : 10

Didapatkan kromosom terbaik dengan kombinasi berat bahan makanan yang telah mengalami proses algoritma evolution strategies sebagai berikut:

Tabel 6. 6 Menu Kasus 2

Nama Bahan	Berat Bahan Manual	Berat Bahan (Program)	Harga Bahan Manual (Rp)	Harga Bahan Program(Rp)
Makan Pagi				
Nasi Putih	120 g	110.19 g	1200	1101.99
daging sapi	40 g	3.35 g	3800	318.89
Bayam	100 g	58.94 g	800	471.59
Oyong	25 g	64.30g	125	321.51
Tempe	25 g	79.84g	200	638.78
Minyak Kelapa	5 g	13.35 g	65	173.65
Snack 1				
Pisang hijau	150 g	207.58 g	1950	2698.56
Manga	200 g	36.90 g	1333	369.08
Makan siang				
Nasi Putih	120 g	156.65 g	1200	1566.5
Telur	60 g	37.79 g	1080	680.29
Minyak	5 g	9.74 g	65	126.64
Tahu	50 g	104.27 g	300	625.679
Labu siam	100 g	55.88 g	300	167.65

Tauge	25 g	15.75 g	375	236.3
tomat	25	44.86 g	87	157.03
Snack 2				
Pepaya	300 g	266.14g	1650	1463
Makan sore				
Nasi Putih	120 g	120.52 g	1200	1205.22
Ikan mujair	50 g	42.11 g	1250	1052.95
Minyak	5 g	12.35 g	65	160.64
Wortel	50 g	48.22 g	350	337.55
Buncis	50 g	32.16 g	300	192.99
tomat	20 g	59.52 g	70	208.34
Jamur kuping	30 g	9.26 g	195	60.22
Kembang tahu	20 g	8.14 g	40	40.71
Daun bawang	10 g	46.30 g	50	231.50
seledri	10 g	19.57 g	150	293.64

Dengan melakukan pengujian , didapatkan hasil perhitungan kecukupan gizi dari Menu kasus 2 dari sistem adalah sebagai berikut:

Tabel 6. 7 Hasil Pemenuhan Gizi dari Sistem untuk Menu Kasus 2

	Energi	Karbohidrat	Lemak	Protein
Kebutuhan Pasien	2703.6 kkal	493.65 g	60.08 g	101.39 g
Hasil sistem	2693.7 kkal	478.81 g	54.27 g	92.67 g
%Gizi Tercukupi	99.63	108.98	90.32	91.39
%Selisih nilai gizi	0.37%	8.98%	9.68%	8.61%
Total biaya Manual	Rp 16550.5			
Total biaya Program	Rp 16450.9			

Berdasarkan Tabel 6.7, diperoleh selisih nilai energi yang dibutuhkan pasien tercukupi 0.37%, selisih nilai gizi kebutuhan karbohidrat pasien tercukupi 8.98%, selisih nilai gizi kebutuhan lemak pasien tercukupi 9.68 %, dan selisih nilai gizi kebutuhan protein pasien tercukupi 8.61%. Total biaya yang dihasilkan manual sebesar Rp 16550 dan total biaya yang dihasilkan sistem Rp16450. Dari hasil pengujian tersebut dapat disimpulkan bahwa nilai gizi dari komposisi bahan makanan yang direkomendasikan oleh sistem masih dalam batas toleransi yang

ditetapkan oleh ahli gizi yaitu $\pm 10\%$ dari kebutuhan gizi pasien diet khusus. Selain itu, hasil biaya yang dihasilkan sistem lebih kecil dari hasil biaya yang dihasilkan secara manual.

6.3.3 Kasus 3

Pasien berjenis kelamin laki-laki berusia 47 tahun, berat badan 68,5 kg, tinggi badan 160,5 cm, dan menderita penyakit jantung . Kategori aktifitas pasien adalah sedang. . Dengan anggaran biaya Rp 18.000 per hari.

Parameter algoritma evolution strategies yang digunakan adalah sebagai berikut:

Banyak generasi : 250

Ukuran populasi : 200

Banyak Offspring : 10

Didapatkan kromosom terbaik dengan kombinasi berat bahan makanan yang telah mengalami proses algoritma evolution strategies sebagai berikut:

Tabel 6. 8 Menu Kasus 3

Nama Bahan	Berat Bahan Manual	Berat Bahan (Program)	Harga Bahan Manual (Rp)	Harga Bahan Program(Rp)
Makan Pagi				
Nasi Putih	150 g	94.84 g	1500	948.49
Ayam	40 g	52.82 g	1112	1468.66
Buncis	100 g	107.84 g	600	647.07
Minyak	5 g	3.75 g	65	37.53
Papaya	190 g	205.57 g	1045	1130.67
Snack 1				
Melon	190 g	143.99 g	950	719.95
Gula pasir	20 g	29.73 g	210	627.21
Makan siang				
Nasi Putih	200 g	133.04 g	2000	1330.42
Tahu	110 g	115.10 g	660	690.63
Kacang panjang	100 g	113.14 g	450	509.13
Minyak	10 g	14.01 g	130	182.21
Jagung muda	110 g	199.18 g	440	796.74
Snack 2				

Jambu biji	100 g	38.72 g	1200	464.64
Gula pasir	20 g	27.97 g	210	713.7
Makan sore				
Nasi Putih	150 g	126.48 g	1500	1264.81
Daging sapi	35 g	31.50 g	3325	2993.27
Kol merah	100 g	144.36 g	400	577.47
Papaya	110 g	3.22 g	605	17.74
Minyak	5 g	7.16 g	65	93.19
Santan	40 g	13.77 g	1720	592.17

Dengan melakukan pengujian , didapatkan hasil perhitungan kecukupan gizi dari Menu kasus 2 dari sistem adalah sebagai berikut:

Tabel 6. 9 Hasil Pemenuhan Gizi dari Sistem untuk Menu Kasus 3

	Energi	Karbohidrat	Lemak	Protein
Kebutuhan Pasien	2663.9 kkal	432.8 g	59.2 g	99.9 g
Hasil sistem	2663.9 kkal	473.8 g	56.42 g	93.63 g
%Gizi Tercukupi	100	91.34	104.9	106.73
%Selisih nilai gizi	0%	8.66%	4.9%	6.73%
Total biaya Manual	Rp 18187			
Total biaya Program	Rp 15805			

Berdasarkan Tabel 6.9 , diperoleh selisih nilai energi yang dibutuhkan pasien tercukupi 0%, selisih nilai gizi kebutuhan karbohidrat pasien tercukupi 8.66%, selisih nilai gizi kebutuhan lemak 4.9%, dan selisih nilai gizi kebutuhan protein pasien 6.73%. Total biaya yang dihasilkan manual sebesar Rp 18188 dan total biaya yang dihasilkan sistem Rp15805. Dari hasil pengujian tersebut dapat disimpulkan bahwa nilai gizi dari komposisi bahan makanan yang direkomendasikan oleh sistem masih dalam batas toleransi yang ditetapkan oleh ahli gizi yaitu $\pm 10\%$ dari kebutuhan gizi pasien diet khusus. Selain itu, hasil biaya yang dihasilkan sistem lebih kecil dari hasil biaya yang dihasilkan secara manual.

Dari hasil uji coba sistem terhadap 3 studi kasus dengan data pasien diet khusus yang berbeda, dapat dilihat bahwa komposisi bahan makanan untuk pasien diet khusus yang dihasilkan oleh sistem masih dalam batas toleransi yang ditetapkan oleh ahli gizi. Selain itu anggaran biaya dari 3 studi kasus, anggaran masih di bawah anggaran yang dihasilkan oleh perhitungan yang dilakukan secara manual.

BAB VII PENUTUP

7.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis yang telah dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Cara mengimplementasikan optimasi biaya dan asupan gizi untuk pasien diet khusus dengan menggunakan algoritma evolution strategies yaitu pertama kita tentukan panjang dan pengkodean kromosom yang digunakan untuk penyelesaian masalah optimasi asupan gizi untuk pasien diet khusus. Panjang kromosom sesuai jumlah bahan makanan yang dimasukkan pengguna. Kromosom berisi informasi berupa kombinasi berat bahan makanan untuk memenuhi kebutuhan gizi. Semakin panjang kromosom (semakin banyak jumlah bahan makanan) belum tentu menghasilkan solusi yang memenuhi kebutuhan gizi, namun dengan pemilihan bahan makanan yang sesuai dapat menghasilkan solusi yang memenuhi kebutuhan gizi. Setelah itu melakukan mutasi untuk menghasilkan *offspring*. Setelah mutasi, maka didapatkan kromosom dari parent dan *offspring*, dilanjutkan dengan menghitung fitness dari setiap individu parent dan *offspring*. Setelah mendapatkan nilai fitness dilakukan seleksi elitism dengan melibatkan hasil fitness parent dan *offspring*. Setelah itu terpilih fitness yang terbesar untuk masuk ke generasi berikutnya. Proses akan berhenti sesuai dengan jumlah generasi yang diinputkan dan fitness terbesar akan terpilih menjadi solusi yang paling terbaik.
2. Penyelesaian masalah optimasi asupan gizi untuk pasien diet khusus menggunakan algoritma evolution strategies dipengaruhi oleh parameter algoritma evolution strategies. Hasil pengujian parameter pada penelitian ini untuk ukuran populasi terbaik 200, ukuran *offspring* terbaik 10 μ , dan ukuran generasi terbaik sebanyak 250 generasi. Hasil uji coba banyak generasi menunjukkan bahwa semakin banyak generasi belum tentu akan diikuti dengan bertambahnya nilai fitness. Begitu juga pada ukuran populasi, semakin besar ukuran populasi belum tentu akan diikuti dengan bertambahnya nilai fitness. Hal ini dikarenakan pembangkitan awal individu (starting point) yang dilakukan secara random dalam proses algoritma evolution strategies.
3. Secara umum, sistem dapat menghasilkan nilai gizi didalam batas toleransi terhadap kebutuhan gizi dengan lebih baik dan biaya yang optimal untuk permasalahan optimasi biaya dan asupan gizi untuk pasien diet khusus berdasarkan pengujian sistem yang dilakukan dengan 3 kasus.

7.2 Saran

Saran yang dapat diberikan untuk pengembangan lebih lanjut adalah sebagai berikut:

1. Ditambahkan fitur untuk menambah atau ingin mengedit data bahan makanan di database.
2. Ditambahkan untuk pasien penyakit komplikasi lain.



DAFTAR PUSTAKA

- Almatsier, Sunita. 2004a. *Penuntun Diet*. Jakarta: Gramedia Pustaka Utama.
- Almatsier, Sunita. 2004b. *Prinsip Dasar Ilmu Gizi*. Jakarta: Gramedia Pustaka Utama.
- Anonim, 2015. *Pantangan Makanan Penderita Penyakit Hati*. <http://penyakithati.org/pantangan-makanan-penderita-penyakit-hati/>. Diakses 18 Maret 2016
- Aryanto. 2015. *Diet Sehat Untuk Penderita Penyakit Hati*. <http://mengobatihepatitisbkronis.com/2015/03/24/diet-sehat-untuk-penderita-penyakit-hati/>. Diakses 26 Januari 2016
- Beyer, H. G, Schwefel, H. P. 2002. *Evolution Strategies: A Comprehensive Introduction*. Journal Natural Computing 1, 2002, 3-52
- Brownlee, J. 2011. *Clever Algorithms: Nature-Inspired Programming Recipes*. EBook. Retrieved from : <http://www.lulu.com/shop/jason-brownlee/cleveralgorithms-nature-inspired-programmingreocipes/ebook/product-0200704.html>
- Eiben, A.E & Smith J.E. 2004. *Introduction to Evolutionary Computing*. <http://www.cs.vu.nl/~gusz/ecbook/ecbook.html>. Diakses 26 Januari 2016
- Hartono, Andry., *Terapi Gizi dan Diet Rumah Sakit*. Edisi Kedua. Jakarta : Buku Kedokteran EGC
- Harun, IA, Mahmudy, WF & Yudistira, N 2014. *Implementasi evolution strategies untuk penyelesaian vehicle routing problem with time windows pada distribusi minuman soda XYZ*. DORO: Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya, vol. 4, no. 1.
- Jones. G. 1998. *Genetic and Evolutionary Algorithms*. <http://www.wiley.com/legacy/wileychi/ecc/samples/sample10.pdf> Diakses 26 Januari 2015
- Karyadi, D. & Muhilal, 1988. *Kecukupan Gizi Yang Dianjurkan*. Pertama ed. Jakarta: PT. Gramedia Jakarta.
- Mahmudy, WF, Marian, RM & Luong, LHS 2013, *Real coded genetic algorithms for solving flexible job-shop scheduling problem – Part I: modeling*, Advanced Materials Research, vol. 701, pp. 359-363.
- Mahmudy, Wayan Firdaus. 2013. *Algoritma Evolusi*. Program Teknologi Informasi dan Ilmu Komputer. Universitas Brawijaya. Malang
- Nurvenus, Karid, Ratnawati, DE & Marji. (2015). *Penerapan Algoritma genetika Untuk Optimasi Asupan Gizi Pasien Diet Khusus dengan Biaya Minimal*, DORO : Repositori Jurnal Mahasiswa PTIIK UB, vol. 6 no. 30

Pramono 2010. *Diet pada penyakit hati*. <https://gizisehat.com/2010/07/18/diet-pada-penyakit-hati/.html> Diakses 26 Januari 2016

Peraturan Pemerintah No.41 Tahun 2014 tentang Kesehatan Republik Indonesia. Jakarta: Menteri Hukum Dan Hak Asasi Manusia Republik Indonesia.

Riksa, ilma. 2011. *Basal metabolic Rate (BMR)*. http://biologilma.co.id/2011/01/basal-metabolic-rate-bmr_09.html. Diakses 8 Maret 2016

Suci, W. W., 2015. Optimasi Biaya Pemenuhan Gizi dan Nutrisi pada Manusia Lanjut Usia Menggunakan Algoritma Genetika, S1: Universitas Brawijaya.

Wahyuningsih

Susanto, T. & Dewanti, T., 2004. *Ilmu Pangan dan Gizi*. Pertama ed. Yogyakarta: Akademika.

Suyanto. 2011, *Artificial Intelligence*, Bandung : Informatika.

Tyas, Rizkhy Ayuning. 2013, "*Implementasi Algoritma Genetika untuk Optimasi 0/1 Multi-Dimensional Knapsack Problem dalam Penentuan Bahan Makanan Sehat*", Program Teknologi Informasi dan Ilmu Komputer (PTIIK) Universitas Brawijaya.





LAMPIRAN A DAFTAR BAHAN MAKANAN

Nama makanan	Per 100 gr				Harga	Penyakit
	energi	karbo	lemak	protein		
agar agar	0	0	0.2	0	5000	DM,SH
apel	58	14.9	0.4	0.3	1800	DM,HT,SH
ayam	302	0	25	18.2	2780	DM,JT,HT,SH
bayam	36	6.5	0.5	3.5	800	DM,JT,HT,SH
bayam merah	51	10	0.5	4.6	2000	JT,HT,SH
beras	360	78.9	0.7	6.8	1000	JT,SH
beras giling	360	78.9	0.7	6.8	1000	DM,JT,HT,SH
beras ketan hitam	356	78	0.7	7	1100	DM,JT,HT,SH
beras ketan putih	362	79.4	0.7	6.7	1200	DM,JT,HT,SH
beras merah tumbuk	359	77.6	0.9	7.5	1400	DM,JT,HT,SH
bihun	360	82.1	0.1	4.7	1700	DM,JT,SH
biskuit	458	75.1	14.4	6.9	3700	JT,HT,SH
blewah melon	77	18.3	0.15	0.45	300	JT,HT,SH
brokoli	34	6.64	0.37	2.82	1500	DM,SH
buncis	35	7.7	0.2	2.4	600	DM,JT,HT,SH
cabe merah	31	7.3	0.3	1	7200	JT,SH
daging sapi	207	0	14	18	9500	DM,JT,HT,SH
daun bawang	29	5.2	0.7	1.8	500	DM,SH
daun bawang prei	45	10.3	0.3	2.2	500	DM,SH
daun pakis	35	6.4	0.3	4	500	DM,SH
genjer	33	7.7	0.2	1.7	500	DM,SH
gula pasir	364	94	0	0	1050	DM,JT,SH
ikan asin kering	193	0	1.5	42	2200	DM,JT,HT,SH
ikan bandeng	129	0	4.8	20	1850	DM,SH
ikan kakap	92	0	0.7	20	2750	DM,SH
ikan kembung	103	0	1	22	4500	DM,JT,HT,SH
ikan lele	93	0	2.2	18.2	1600	DM,JT,HT,SH
ikan mas	86	0	2	16	2200	DM,JT,HT,SH
ikan mujair	89	0	1.25	23.37	2500	DM,JT,HT,SH
ikan pindang	156	0	4	28	3000	DM,SH
ikan teri	74	4.1	1.4	10.3	2100	DM,SH
jagung muda dg tongkol	33	7.4	0.1	2.2	400	DM,SH
jambu biji	49	12.2	0.3	0.9	1200	DM,JT,SH
jamur kuping kering	128	64.6	0.9	16	650	DM,SH
jamur kuping segar	15	0.9	0.6	3.8	650	DM,SH
jantung pisang segar	31	7.1	0.3	1.2	400	DM,SH
jeruk bali	48	12.4	0.2	0.6	1500	JT,HT,SH
jeruk manis	45	11.2	0.2	0.9	1300	DM,JT,HT,SH

jeruk manis/sari jeruk	44	11	0.2	0.8	1300	JT,HT,SH
jeruk nipis	37	12.3	0.1	0.8	900	JT,HT,SH
kacang arab	330	60.2	1.4	23.8	6000	DM,JT,HT,SH
kacang ijo	345	62.9	1.2	22.2	900	DM,JT,HT,SH
kacang kedelai	381	24.9	16.7	40.4	900	DM,JT,HT,SH
kacang merah	336	59.5	1.7	23.1	1300	DM,JT,HT,SH
kacang panjang	44	7.8	0.3	2.7	450	DM,JT,HT,SH
kacang tanah	525	17.4	42.7	27.9	1400	DM,SH
kangkung	29	5.4	0.3	3	300	DM,JT,HT,SH
kapri muda	42	9	0.2	3.3	1000	DM,SH
kecap	46	9	1.3	5.7	1700	DM,SH
kembang kol	25	4.9	0.2	2.4	1000	DM,SH
kembang tahu	380	23.3	13.8	48.9	500	DM,SH
kentang	83	19.1	0.1	2	750	DM,JT,HT,SH
kentang hitam	142	33.7	0.4	0.9	550	DM,JT,HT,SH
ketela pohon	154	36.8	0.3	1	200	DM,SH
ketela pohon kuning	157	37.9	0.3	0.8	250	DM,SH
ketimun	12	2.7	1.7	0.7	1000	DM,JT,SH
kol merah/putih	24	5.3	0.2	1.4	400	DM,JT,SH
kolang kaling	191	39.72	1.5	4.48	1000	DM,SH
kucai	45	10.3	0.3	2.2	1000	DM,SH
kurma	296	71.5	0	2.5	3000	DM,SH
labu siam	26	6.7	0.1	0.6	300	DM,JT,HT,SH
labu waluh	29	6.6	0.3	1.1	1000	DM,SH
lychee	66	14.8	0.53	0.83	2500	DM,SH
madu	294	79.5	0	0.3	10800	DM,SH
mangga	52	12.3	0	0.7	1000	DM,SH
melon	34	8.16	0.19	0.84	500	DM,JT,HT,SH
mi basah	86	14	3.3	0.6	1400	DM,JT,HT,SH
mi kering	337	50	11.8	7.9	1400	DM,JT,HT,SH
minyak kelapa	870	0	98	1	1000	DM,JT,SH
minyak kelapa sawit	902	0	100	0	1300	DM,HT,SH
oncom	187	22.6	6	13	1000	DM,SH
oyong	18	4.1	0.2	0.8	500	DM,JT,SH
pare	29	6.6	0.3	1.1	400	DM,SH
pepaya	46	12.2	0	0.5	550	DM,JT,HT,SH
pepaya muda	26	4.9	0.1	2.1	250	DM,SH
pisang ambon	99	25.8	0.2	1.2	1100	DM,JT,HT,SH
pisang hijau	121	28.9	0.1	1	1300	DM,SH
pisang kepok	109	26.3	0	0.8	1100	DM,JT,HT,SH
pisang raja	120	31.8	0.2	1.2	1300	DM,JT,HT,SH
pisang raja susu	118	31.1	0.2	1.2	2000	DM,JT,HT,SH
rambutan	69	18.1	0.1	0.9	1500	DM,SH

rebung	27	5.2	0.3	2.6	800	DM,SH
roti putih	248	50	1.2	8	3300	DM,JT,HT,SH
sagu manis	287	2.4	3.9	0.5	750	DM,SH
sagu tepung sagu	209	51.6	0.2	0.3	800	DM,JT,HT,SH
salak	77	20.9	0	0.4	1600	DM,SH
santan	324	5.6	34.3	4.2	4300	DM,JT,SH
sawi	22	4	0.3	2.3	1000	DM,SH
sawo	92	22.4	1.1	0.5	1000	DM,JT,HT,SH
selada air	17	3	0.3	1.7	1000	JT,SH
seledri	20	4.6	0.1	1	1500	DM,SH
semangka	28	6.9	0.2	0.5	500	DM,JT,HT,SH
sirsak	65	16.3	0.3	1	1500	DM,SH
susu skim	36	5.1	0.1	3.5	2000	DM,SH
tahu	68	1.6	4.6	7.8	600	DM,JT,HT,SH
tauge kacang hijau	23	4.1	0.2	2.9	1500	DM,JT,HT,SH
tauge kacang kedele	67	6.4	2.6	9	1600	JT,HT,SH
teh	132	67.8	0.7	19.5	1000	DM,JT,SH
telur ayam	162	0.7	11.5	12.8	1800	DM,JT,HT,SH
telur ayam kampung	174	1.2	14	10.8	3000	JT,HT,SH
telur ayam ras	154	0.7	10.8	12.4	3500	JT,HT,SH
telur bebek	189	0.8	14.3	13.1	2000	JT,HT,SH
telur bebek. asin	195	1.4	13.6	13.6	2000	JT,HT,SH
tempe kedelai murni	149	12.7	4	18.3	800	DM,JT,HT,SH
tempe lamtoro	142	20.4	2.5	11	600	DM,SH
tepung beras	364	80	0.5	7	1000	JT,HT,SH
tepung terigu	365	77.3	1.3	8.9	810	JT,HT,SH
terong	24	5.5	0.2	1.1	400	DM,SH
tomat	20	4.2	0.3	1	350	DM,JT,HT,SH
ubi jalar kuning	114	26.7	0.5	0.8	250	DM,SH
ubi jalar merah	151	35.4	0.3	1.6	250	DM,SH
ubi jalar putih	123	27.9	0.7	1.8	250	DM,SH
ubi kuning. mentah	119	25.1	0.4	0.5	250	DM,SH
ubi manis	83	18.8	0.2	1.5	250	DM,SH
ubi putih. mentah	88	20.6	0.4	0.4	250	DM,SH
ubi tinta	108	25.6	0.4	0.5	250	DM,SH
wortel	42	9.3	0.3	1.2	700	DM,JT,HT,SH

LAMPIRAN B MANUALISASI

P1		Gen	Berat(gr)	Harga	Energi(kalori)	Karbo	Lemak	Protein	Penalti
Makan Pagi	roti putih	0.791041915	276.8646703	9136.534121	686.6244	138.4323	3.3224	22.1492	
	susu skim	0.37198842	130.1959472	2603.918943	46.8705	6.6400	0.1302	4.5569	
Total Gizi					733.4949	145.0723	3.4526	26.7060	
Selisih Bahan					165.0574	52.7012	9.1794	5.3896	232.3277
Snack 1	agar	0.189840735	66.44425711	3322.212856	0.0000	0.0000	0.1329	0.0000	
Total Gizi					0.0000	0.0000	0.1329	0.0000	
Selisih Bahan					227.3750	36.9484	4.9199	8.5266	277.7699
Makan Siang	bayam	0.496260254	173.6910888	1389.52871	62.5288	11.2899	0.8685	6.0792	
	beras giling	0.817682009	286.1887031	2861.887031	1030.2793	225.8029	2.0033	19.4608	
	daun bawang	0.190829027	66.79015928	333.9507964	19.3691	3.4731	0.4675	1.2022	
	ayam	0.07210449	25.2365714	701.5766849	76.2144	0.0000	6.3091	4.5931	
Total Gizi					1188.3917	240.5659	9.6485	31.3353	647.2528
Selisih Bahan					506.2667	129.7206	5.5099	5.7556	
Snack 2	apel	0.669155892	234.204562	4215.682117	135.8386	34.8965	0.9368	0.7026	
	jeruk bali	0.484233466	169.4817131	2203.26227	76.2668	18.9820	0.3390	1.5253	
Total Gizi					212.1054	53.8784	1.2758	2.2279	

Selisih Bahan					15.2696	16.9300	3.7770	34.7205	70.6971
Makan Sore	beras giling	0.540184177	189.0644619	1890.644619	680.6321	149.1719	1.3235	12.8564	
	brokoli	0.301731283	105.6059489	1584.089233	35.9060	7.0122	0.3907	2.9781	
	daging sapi	0.701592271	245.5572948	23327.94301	508.3036	0.0000	34.3780	44.2003	
	tahu	0.790807882	276.7827587	1660.696552	188.2123	4.4285	12.7320	21.5891	
	seledri	0.585288384	204.8509345	3072.764018	40.9702	9.4231	0.2049	2.0485	
Total Gizi					1454.0241	170.0358	49.0291	83.6723	1062.004388
Selisih Bahan					885.5866485	77.6646688	36.3971279	62.3559425	
				58304.69096					2290.0518

P2		Gen	Berat(gr)	Harga	Energi(kalori)	Karbo	Lemak	Protein	Penalti
Makan Pagi	roti putih	0.737737968	258.2082889	8520.873533	640.3566	129.1041	3.0985	20.6567	
	susu skim	0.798739351	279.5587727	5591.175454	100.6412	14.2575	0.2796	9.7846	
Total Gizi					740.9977	143.3616	3.3781	30.4412	241.9295
Selisih Bahan					172.5602	50.9905	9.2539	9.1248	
Snack 1	agar agar	0.39831053	139.4086855	6970.434273	0.0000	0.0000	0.1329	0.0000	
Total Gizi					0.0000	0.0000	0.1329	0.0000	
Selisih Bahan					227.3750	36.9484	4.9199	8.5266	277.7699
Makan Siang	bayam	0.956826997	334.8894489	2679.115591	120.5602	21.7678	1.6744	11.7211	
	beras giling	0.195134028	68.29690977	682.9690977	245.8689	53.8863	0.4781	4.6442	
	daun bawang	0.249017691	87.15619175	435.7809587	25.2753	4.5321	0.6101	1.5688	
	ayam	0.156475131	54.76629601	1522.503029	165.3942	0.0000	13.6916	9.9675	

Total Gizi					557.0986	80.1862	16.4542	27.9016	
Selisih Bahan					125.0264	30.6591	1.2959	2.3219	159.3033
Snack 2	apel	0.611310663	213.9587321	3851.257178	124.0961	31.8799	0.8558	0.6419	
	jeruk bali	0.149949143	52.48220002	682.2686002	23.6170	5.8780	0.1050	0.4723	
Total Gizi					147.7131	37.7579	0.9608	1.1142	
Selisih Bahan					79.6619	0.8094	4.0920	7.4123	91.9757
Makan Sore	beras giling	0.055065932	19.27307637	192.7307637	69.3831	15.2065	0.1349	1.3106	
	brokoli	0.650936899	227.8279145	3417.418718	77.4615	15.1278	0.8430	6.4247	
	daging sapi	0.757201897	265.0206639	25176.96307	548.5928	0.0000	37.1029	47.7037	
	tahu	0.557945071	195.2807748	1171.684649	132.7909	3.1245	8.9829	15.2319	
	seledri	0.406189499	142.1663247	2132.494871	28.4333	6.5397	0.1422	1.4217	
Total Gizi					856.6615	39.9984	47.2058	72.0926	
Selisih Bahan					288.2240319	52.3727196	34.5739053	50.7761933	425.9468501
				63027.66978					1196.9252

P3		Gen	Berat(gr)	Harga	Energi(kalori)	Karbo	Lemak	Protein	Penalti
Makan Pagi	roti putih	0.637322966	223.0630382	7361.080261	553.1963	111.5315	2.6768	17.8450	
	susu skim	0.944867599	330.7036597	6614.073193	119.0533	16.8659	0.3307	11.5746	
Total Gizi					672.2497	128.3974	3.0075	29.4197	
Selisih Bahan					103.8122	36.0263	9.6245	8.1033	157.5662
Snack 1	agar agar	0.17044049	59.65417133	2982.708566	0.0000	0.0000	0.1193	0.0000	
Total Gizi					0.0000	0.0000	0.1193	0.0000	

Selisih Bahan					227.3750	36.9484	4.9335	8.5266	277.7835
Makan Siang	bayam	0.940871389	329.3049861	2634.439889	118.5498	21.4048	1.6465	11.5257	
	beras giling	0.747347	261.57145	2615.7145	941.6572	206.3799	1.8310	17.7869	
	daun bawang	0.177085796	61.98002869	309.9001434	17.9742	3.2230	0.4339	1.1156	
	ayam	0.008467438	2.963603453	82.38817598	8.9501	0.0000	0.7409	0.5394	
	Total Gizi				1087.1313	231.0077	4.6523	30.9675	
Selisih Bahan					405.0063	120.1623	10.5060	5.3879	541.0626
Snack 2	apel	0.242165857	84.75804992	1525.644899	49.1597	12.6289	0.3390	0.2543	
	jeruk bali	0.856205512	299.6719294	3895.735082	134.8524	33.5633	0.5993	2.6970	
Total Gizi					184.0120	46.1922	0.9384	2.9513	
Selisih Bahan					43.3630	9.2438	4.1144	5.5752	62.2964
Makan Sore	beras giling	0.270233485	94.58171991	945.8171991	340.4942	74.6250	0.6621	6.4316	
	brokoli	0.804719365	281.6517778	4224.776668	95.7616	18.7017	1.0421	7.9426	
	daging sapi	0.035584659	12.45463062	1183.189909	25.7811	0.0000	1.7436	2.2418	
	tahu	0.965910667	338.0687336	2028.412402	229.8867	5.4091	15.5512	26.3694	
	seledri	0.802726425	280.9542489	4214.313733	56.1908	12.9239	0.2810	2.8095	
Total Gizi					748.1145	111.6597	19.2799	45.7949	
Selisih Bahan					179.6769701	19.2885565	6.64800345	24.4784681	230.0919982
			40618.19462						
									1268.8006

C1		Gen	Berat(gr)	Harga	Energi(kalori)	Karbo	Lemak	Protein	Penalti
Makan Pagi	roti putih	0.892133777	312.246822	10304.14513	774.3721	156.1234	3.7470	24.9797	
	susu skim	0.617016195	215.9556682	4319.113364	77.7440	11.0137	0.2160	7.5584	

Total Gizi					852.1162	167.1372	3.9629	32.5382	
Selisi Bahan					283.6787	74.7661	8.6690	11.2218	378.3355
Snanck 1	agar agar	0.093596734	32.75885699	1637.94285	0.0000	0.0000	0.0655	0.0000	
Total Gizi					0.0000	0.0000	0.0655	0.0000	
Selisi Bahan					227.3750	36.9484	4.9873	8.5266	277.8373
Makan Siang	bayam	0.186691185	65.34191474	522.7353179	23.5231	4.2472	0.3267	2.2870	
	beras giling	0.217063215	75.97212541	759.7212541	273.4997	59.9420	0.5318	5.1661	
	daun bawang	0.067174003	23.5109011	117.5545055	6.8182	1.2226	0.1646	0.4232	
	ayam	1.212587714	424.4056998	11798.47845	1281.7052	0.0000	320.4263	0.0000	
Total Gizi					1585.5461	65.4118	321.4494	7.8763	
Selisi Bahan					903.4211	45.4335	306.2911	17.7034	1272.8491
Snack 2	apel	0.723623254	253.268139	4558.826502	146.8955	37.7370	1.0131	0.7598	
	jeruk bali	0.513276226	179.6466789	2335.406826	80.8410	20.1204	0.3593	1.6168	
Total Gizi					227.7365	57.8574	1.3724	2.3766	
Selisi Bahan					0.3615	20.9089	3.6804	6.1499	31.1008
Makan Sore	beras giling	0.644960343	225.73612	2257.3612	812.6500	178.1058	1.5802	15.3501	
	brokoli	0.079842627	27.94491945	419.1737917	9.5013	1.8555	0.1034	0.7880	
	daging sapi	0.550474487	192.6660704	18303.27669	398.8188	0.0000	26.9732	34.6799	
	tahu	0.890168603	311.5590109	1869.354065	211.8601	4.9849	14.3317	24.3016	
	seledri	1.128095704	394.8334965	5922.502447	78.9667	18.1623	0.3948	3.9483	
Total Gizi					1511.7969	203.1086	43.3833	79.0679	
Selisi Bahan					943.359397	110.737533	30.7514025	57.7515271	1142.599859
				65125.5924					3102.7226

C2		Gen	Berat(gr)	Harga	Energi(kalori)	Karbo	Lemak	Protein	Penalti
Makan Pagi	roti putih	0.188099529	65.83483514	2172.54956	163.2704	32.9174	0.7900	5.2668	
	susu skim	0.077065873	26.97305539	539.4611078	9.7103	1.3756	0.0270	0.9441	
Total Gizi					172.9807	34.2930	0.8170	6.2108	
Selisih Bahan					395.4568	58.0781	11.8150	15.1056	480.4554
Snack 1	agar agar	0.280110737	98.03875794	4901.937897	0.0000	0.0000	0.1961	0.0000	
Total Gizi					0.0000	0.0000	0.1961	0.0000	
Selisih Bahan					227.3750	36.9484	4.8567	8.5266	277.7067
Makan Siang	bayam	0.422943117	148.0300909	1184.240727	53.2908	9.6220	0.7402	5.1811	
	beras giling	0.717443642	251.1052748	2511.052748	903.9790	198.1221	1.7577	17.0752	
	daun bawang	0.122293419	42.80269657	214.0134829	12.4128	2.2257	0.2996	0.7704	
	ayam	0.859997577	300.999152	8367.776427	909.0174	0.0000	75.2498	54.7818	
Total Gizi					1878.7000	209.9698	78.0473	77.8085	
Selisih Bahan					1196.5750	99.1244	62.8890	52.2288	1410.8173
Snack 2	apel	0.612354981	214.3242433	3857.83638	124.3081	31.9343	0.8573	0.6430	
	jeruk bali	0.374003057	130.9010699	1701.713908	58.9055	14.6609	0.2618	1.1781	
Total Gizi					183.2135	46.5952	1.1191	1.8211	
Selisih Bahan					44.1615	9.6468	3.9337	6.7055	64.4474
Makan Sore	beras giling	0.256468098	89.76383438	897.6383438	323.1498	70.8237	0.6283	6.1039	
	brokoli	0.854032992	298.9115471	4483.673207	101.6299	19.8477	1.1060	8.4293	
	daging sapi	0.407173678	142.5107871	13538.52478	294.9973	0.0000	19.9515	25.6519	

	tahu	0.542567755	189.8987143	1139.392286	129.1311	3.0384	8.7353	14.8121	
	seledri	0.329375924	115.2815733	1729.223599	23.0563	5.3030	0.1153	1.1528	
Total Gizi					871.9645	99.0127	30.5365	56.1501	
Selisih Bahan					303.5269995	6.64163011	17.9045077	34.8336972	362.9068346
			47239.03445						2596.3336

C3		Gen	Berat(gr)	Harga	Energi(kalori)	Karbo	Lemak	Protein	Penalti
Makan Pagi	roti putih	0.825375698	288.8814941	9533.089307	716.4261	144.4407	3.4666	23.1105	
	susu skim	0.949148148	332.2018517	6644.037033	119.5927	16.9423	0.3322	11.6271	
Total Gizi					836.0188	161.3830	3.7988	34.7376	
Selisih Bahan					267.5813	69.0119	8.8332	13.4212	358.8476
Snack 1	agar agar	0.139482535	48.81888723	2440.944362	0.0000	0.0000	0.0976	0.0000	
Total Gizi					0.0000	0.0000	0.0976	0.0000	
Selisih Bahan					227.3750	36.9484	4.9551	8.5266	277.8051
Makan Siang	bayam	0.860957734	301.335207	2410.681656	108.4807	19.5868	1.5067	10.5467	
	beras giling	0.136999175	47.94971121	479.4971121	172.6190	37.8323	0.3356	3.2606	
	daun bawang	0.619611662	216.8640818	1084.320409	62.8906	11.2769	1.5180	3.9036	
	ayam	0.296310298	103.7086042	2883.099197	313.2000	0.0000	25.9272	18.8750	
Total Gizi					657.1902	68.6960	29.2875	36.5858	
Selisih Bahan					24.9348	42.1493	14.1292	11.0061	92.2194
Snack 2	apel	0.901652882	315.5785088	5680.413158	183.0355	47.0212	1.2623	0.9467	
	jeruk bali	1.290823493	451.7882227	5873.246895	203.3047	657.8037	0.4066	5.9202	

Total Gizi					386.3402	704.8249	1.6689	6.8670	
Selisih Bahan					158.9652	667.8764	3.3839	1.6596	831.8851
Makan Sore	beras giling	0.351876561	123.1567964	1231.567964	443.3645	971.7071	3.1036	66.0761	
	brokoli	0.369215205	129.2253218	1938.379827	43.9366	8.5806	0.4781	3.6442	
	daging sapi	0.053741786	18.80962518	1786.914392	38.9359	0.0000	2.6333	3.3857	
	tahu	0.214614519	75.11508176	450.6904905	51.0783	1.2018	3.4553	5.8590	
	seledri	0.924443038	323.5550632	4853.325949	64.7110	14.8835	0.3236	3.2356	
Total Gizi					642.0263	996.3731	9.9939	82.2005	
Selisih Bahan					73.58876881	904.001965	2.63806314	60.8840918	1041.112889
				47290.20775					
									2601.8701

	Total Harga	Penalti Harga	Penalti Gizi	Fitness
P1	58304.6910	13304.6910	2290.0518	0.000064120
P2	63027.6698	18027.6698	1196.9252	0.000052014
P3	40618.1946	0	1268.8006	0.000787525
C1	65125.5924	20125.5924	3102.7226	4.304905E-05
C2	47239.0344	2239.034	2596.3336	2.067667E-04
C3	47290.20775	2290.2078	2601.8701	2.043703E-04

SELEKSI (*elitism*)

	Total Harga	Penalti Harga	Penalti Gizi	Fitness
P3	40618.1946	0	1268.8006	0.000787525
P1	58304.6910	13304.6910	2290.0518	0.000064120
P2	63027.6698	18027.6698	1196.9252	0.000052014
C2	47239.0344	2239.034	2596.3336	2.067667E-04
C3	47290.20775	2290.2078	2601.8701	2.043703E-04
C1	65125.5924	20125.5924	3102.7226	4.304905E-05

} Lolos ke Generasi Berikutnya

