OPTIMASI KOMPOSISI MAKANAN BAGI PENDERITA OBESITAS PADA ANAK MENGGUNAKAN ALGORITMA GENETIKA

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Febry Agristya Indaswari
NIM: 125150200111047

PROGRAM STUDI INFORMATIKA/ILMU KOMPUTER
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

PENGESAHAN

OPTIMASI KOMPOSISI MAKANAN BAGI PENDERITA OBESITAS PADA ANAK MENGGUNAKAN ALGORITMA GENETIKA

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Febry Agristya Indaswari
NIM: 125150200111047

Skripsi ini telah diuji dan dinyatakan lulus pada
7 April 2016
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D
NIP. 19720919 199702 1 001

Mengetahui
Ketua Program Studi Informatika/Ilmu Komputer

Drs. Marji, M.T
NIP. 19670801 199203 1 001

PERNYATAAN ORISINALITAS

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

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

Malang, 7 April 2016

Febry Agristya Indaswari

NIM: 125150200111047

KATA PENGANTAR

Puji dan syukur Penulis panjatkan kehadirat Allah SWT, karena hanya dengan rahmat dan bimbingannya Penulis dapat menyelesaikan tugas akhir dengan judul "Optimasi Komposisi Makanan Bagi Penderita Obesitas Pada Anak Menggunakan Algoritma Genetika" dengan baik. Tanpa rahmat dan bimbingan dari Tuhan Yang Maha Esa, maka niscaya Penulis tidak akan dapat menyelesaikan tugas akhir ini dengan baik dan tepat waktu.

Penulisan dan penyusunan laporan skripsi ini dapat terlaksana dengan baik karena adanya bantuan secara langsung maupun tidak langsung dari pihak tertentu diantaranya:

1. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D selaku dosen pembimbing I yang telah banyak memberikan ilmu dan saran untuk laporan skripsi.
2. Bapak Drs. Marji, M.T selaku Ketua Program Studi Informatika / Ilmu Komputer Fakultas Ilmu Komputer Universitas Brawijaya.
3. Bapak Imam Cholissodin, S.Si, M.Kom selaku dosen pembimbing akademik yang telah banyak memberikan saran dan motivasi selama masa perkuliahan.
4. Kedua orang tua saya Agus Sugiharto dan Ristyaningrum yang telah memberi motivasi, kasih sayang serta dukungan moril dan materil. Kedua adik saya (Dewi Rafardha Dharmai Harti dan Nanda Risma Oktavia Rini) dan seluruh keluarga besar yang telah memberikan semangat dan doa dari awal sampai akhir pengerjaan skripsi ini.
5. Teman-teman seperjuangan skripsi Dwi Ratnawati, Qoirul Kotimah, Ziya El Arief, Rezqi Andini, Danu Akbar, Yuyon Alif, Suherni, Dito Prayogi, Ibnu Aqli, M. Ilham Akbar, Nancy Rahmadhani, Yuni Widyaningtyas, Nurrizka Ajeng, Irsyad Fauzan, dan Aldi Yudha yang saling menyemangati dan membantu satu sama lain.

Semoga jasa dan amal baik mendapatkan balasan dari Allah SWT. Dengan segala kerendahan hati, penulis menyadari sepenuhnya bahwa skripsi ini masih jauh dari sempurna karena keterbatasan materi dan pengetahuan yang dimiliki penulis. Saran dan kritikan yang bersifat membangun dapat disampaikan melalui email penulis. Semoga skripsi ini dapat bermanfaat dan berguna bagi pembaca terutama mahasiswa Fakultas Ilmu Komputer Universitas Brawijaya.

Malang, 7 April 2016

Febry Agristya Indaswari
febri.agristya1602@gmail.com

ABSTRAK

Obesitas merupakan kelebihan berat badan akibat penimbunan zat lemak di dalam tubuh. Obesitas tidak hanya dialami oleh orang dewasa tetapi juga dialami oleh anak-anak. Obesitas pada anak beresiko menjadi obesitas lanjutan pada saat dewasa dan menyebabkan timbulnya penyakit. Pengaturan diet anak dapat dilakukan dengan memberikan komposisi makanan yang baik bagi anak. Dalam pelaksanaannya, orang tua terkadang masih sulit untuk mengatur pola makan anak dikarenakan kurangnya pengetahuan dalam hal tersebut. Mengatasi kasus tersebut dibutuhkan algoritma genetika. Pada penelitian ini menggunakan 133 nama makanan dengan kandungan gizi (karbohidrat, lemak, dan protein). Dalam proses algoritmanya menggunakan representasi bilangan integer dengan panjang kromosom 15 dimana angka dalam kromosom merepresentasikan nomor makanan, metode *single point crossover*, *reciprocal exchange mutation*, dan seleksi *elitism*. Dari hasil pengujian yang dilakukan diperoleh parameter optimal sebesar 160 populasi dengan nilai rata-rata *fitness* sebesar 0.07744299, 100 generasi dengan nilai rata-rata *fitness* sebesar 0.07744299 dan kombinasi cr = 0.6 dan mr = 0.4. dengan nilai rata-rata *fitness* sebesar 0.0782747. Hasil yang diperoleh pada penelitian ini berupa komposisi makanan untuk pagi hari, siang hari, dan malam hari dengan biaya minimum.

Kata kunci: Algoritma Genetika, Obesitas Anak, Komposisi Makanan

repository.ub.ac.id

ABSTRACT

Obesity is excess weight due to accumulation of fat substances in the body. Obesity is not experienced by adults but also to the children. Childhood obesity able to risk to becoming obese in adulthood and further cause many diseases. To organize diet for children, it can be done by giving the good composition of food. But in the daily life, parent has a little knowledge to organize it. Overcoming this problem, it can be solved by genetic algorithm. In this experiment is used 133 data of food which contain nutrients like carbohydrates, fats, and proteins. In algorithm process is used representate with integer value and has 15 gens length of chromosome which each digit of the chromosome is a number of food. For the reproduction process, it used single point crossover and reciprocal exchanged mutation. For the selection, it used elitism selection. From the test results are to be obtained optimal parameter of 160 populations with average fitness = 0.07744299, 100 generations with average fitness = 0.07744299, and combination of cr = 0.6 and mr = 0.4 with average fitness = 0.0782747. Final result of this experiments is a combination of food for breakfast, lunch, and dinner with minimum cost.

Keywords: Genetic Algorithm, Childhood Obesity, Food Composition

UNIVERSITAS BRAWIJAYA

DAFTAR ISI

PENGESAHAN i
PERNYATAAN ORISINALITAS ii
KATA PENGANTAR iii
ABSTRAK iv
ABSTRACT v
DAFTAR ISI vi
DAFTAR TABEL ix
DAFTAR GAMBAR x
DAFTAR LAMPIRAN xi
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 2
1.3 Batasan Masalah 2
1.4 Tujuan Penelitian 3
1.5 Manfaat 3
1.6 Sistematika Pembahasan 3
BAB II TINJAUAN PUSTAKA 5
2.1 Kajian Pustaka 5
2.2 Kebutuhan Energi 7
2.2.1 Karbohidrat 7
2.2.2 Lemak 7
2.2.3 Protein 7
2.3 Obesitas 8
2.3.1 Obesitas Pada Anak 8
2.3.2 Perhitungan Status Gizi 10
2.4 Algoritma Genetika 11
2.4.1 Struktur Algoritma Genetika 12
2.4.2 Komponen Algoritma Genetika 12
2.4.3 Kondisi Berhenti 16
BAB III METODOLOGI PENELITIAN 17

3.1 Tahapan Penelitian 17
3.1.1 Studi Literatur 17
3.1.2 Pengumpulan Data 18
3.1.3 Analisis dan Perancangan 18
3.1.4 Implementasi 18
3.1.5 Pengujian dan Analisis 18
3.1.6 Kesimpulan 18
3.2 Teknik Pengumpulan Data 19
3.3 Algoritma yang Digunakan 19
3.4 Lingkungan Uji Coba 19
3.5 Pengujian Algoritma 19
BAB IV PERANCANGAN 21
4.1 Formulasi Permasalahan 21
4.2 Siklus Algoritma Genetika 23
4.2.1 Representasi Kromosom 24
4.2.2 Inisialisasi Populasi Awal 29
4.2.3 Reproduksi 30
4.2.4 Seleksi 34
4.3 Perancangan Database 35
4.4 Perancangan User Interface 38
4.4.1 Data Makanan 38
4.4.2 Beranda 39
4.4.3 Proses Algen 40
4.3.4 Detail Kromosom Terbaik 41
4.3.5 Hasil Optimasi 41
BAB V IMPLEMENTASI 43
5.1 Struktur Database 43
5.2 Struktur Class 43
5.3 Potongan Source Code 44
5.3.1 Mengambil Data Makanan dari Database 44
5.3.2 Pembangkitan Populasi Awal 46

5.3.3 Perhitungan Penalti dan Nilai Fitness 47
5.3.4 Proses Crossover 48
5.3.5 Proses Mutasi 50
5.3.6 Proses Seleksi 52
5.3.7 Proses Pemilihan Individu dengan Nilai Fitness Terbaik 53
BAB VI PENGUJIAN DAN ANALISA 54
6.1 Hasil dan Analisa Pengujian Ukuran Populasi 54
6.2 Hasil dan Analisa Pengujian Ukuran Generasi 56
6.3 Hasil dan Analisa Pengujian Terhadap Kombinasi Cr dan Mr 58
6.4 Analisa Hasil 60
BAB VII PENUTUP 63
7.1 Kesimpulan 63
7.2 Saran 63
DAFTAR PUSTAKA 65

DAFTAR TABEL

Tabel 2. 1 Kajian Pustaka 5
Tabel 2. 2 Tingkatan Aktivitas Fisik 9
Tabel 2. 3 Tabel Aktifitas Fisik Anak.....11
Tabel 2. 4 Inisialisasi Kromosom dengan Biner.....13
Tabel 2. 5 Himpunan populasi parent dan offspring 15
Tabel 2. 6 Hasil perankingan himpunan populasi parent dan offspring.....16
Tabel 2. 7 Populasi baru (parent) pada generasi selanjutnya 16

Tabel 4. 1 Kebutuhan Nutrisi Anak Usia 7-9 tahun.....21
Tabel 4. 2 Representasi Kromosom 25
Tabel 4. 3 Detail Kromosom 1 dan Perhitungan Fitness.....28
Tabel 4. 4 Individu Baru Hasil Seleksi.....34
Tabel 4. 5 Atribut Tabel makanan_pokok.....35
Tabel 4. 6 Atribut Tabel sumber_nabati 36
Tabel 4. 7 Atribut Tabel sayuran 36
Tabel 4. 8 Atribut Tabel sumber_hewani.....37
Tabel 4. 9 Atribut Tabel pelengkap 37

Tabel 6. 1 Hasil Uji Coba Populasi 54
Tabel 6. 2 Hasil Uji Coba Generasi 56
Tabel 6. 3 Hasil Uji Coba Kombinasi Cr dan Mr.....59
Tabel 6. 4 Hasil Kebutuhan dan Keterpenuhan Kandungan Gizi 61

DAFTAR GAMBAR

Gambar 2. 1 Pencarian Solusi dalam Algoritma Genetika 12

Gambar 3. 1 Diagram Blok Metode 17

Gambar 4. 1 Flowchart Algoritma Genetika 23
Gambar 4. 2 Flowchart Perhitungan Nilai *Fitness*.....26
Gambar 4. 3 FLowchart Perhitungan Penalti.....27
Gambar 4. 4 Flowchart Pembangkitan Populasi Awal.....29
Gambar 4. 5 *Flowchart* Proses Crossover 31
Gambar 4. 6 Flowchart Single Point Crossover.....31
Gambar 4. 7 Contoh Proses Crossover 32
Gambar 4. 8 Contoh Proses Mutasi 32
Gambar 4. 9 Flowchart Reciprocal Exchange Mutation 33
Gambar 4. 10 Flowchart Proses Mutasi.....33
Gambar 4. 11 *Flowchart* Seleksi Elitism 34
Gambar 4. 12 Perancangan Database.....35
Gambar 4. 13 Perancangan Halaman Data Makanan.....38
Gambar 4. 14 Perancangan Beranda 39
Gambar 4. 15 Perancangan Proses Algoritma Genetika.....40
Gambar 4. 16 Perancangan Individu Terbaik.....41
Gambar 4. 17 Perancangan Halaman Hasil Optimasi 42

Gambar 5. 1 Struktur Database 43
Gambar 5. 2 *Class Diagram*.....44

Gambar 6. 1 Uji Coba Populasi.....56
Gambar 6. 2 Uji Coba Generasi 58
Gambar 6. 3 Uji Coba Cr dan Mr 60
Gambar 6. 4 Hasil Makanan Terpilih dari Kromosom Terbaik.....61

DAFTAR LAMPIRAN

Lampiran 1.1 Data Anak 67
Lampiran 1.2 Prosentase Kebutuhan Energi Manusia 67
Lampiran 1.3 Data Makanan Pokok 68
Lampiran 1.4 Data Sumber Nabati 69
Lampiran 1.5 Data Sayuran 71
Lampiran 1.6 Data Sumber Hewani 72
Lampiran 1.7 Data Makanan Pelengkap 74

BAB I PENDAHULUAN

1.1 Latar Belakang

Pola makan dengan mengkonsumsi makanan bergizi merupakan hal penting yang perlu diperhatikan. Sumber daya manusia yang baik dihasilkan dari tubuh yang sehat dengan mengkonsumsi makanan yang bergizi. Makanan yang bergizi dikonsumsi untuk mendapatkan asupan gizi yang dibutuhkan dalam melakukan segala aktivitas. Namun, seiring dengan perkembangan industri dan teknologi, pola makan di dunia mengalami perubahan. Masyarakat saat ini lebih tertarik dengan makanan yang diolah secara industri dalam bentuk makanan cepat saji maupun makanan kaleng (Almatsier, 2006). Perubahan pola makan yang terjadi tidak hanya mempengaruhi pola makan pada orang dewasa, namun perubahan ini juga terjadi pada anak-anak. Dewasa ini anak-anak cenderung lebih menyukai makanan cepat saji yang mengakibatkan kegemukan atau obesitas (Almatsier, 2004).

Obesitas merupakan kelebihan berat badan akibat penimbunan zat lemak di dalam tubuh. Kelebihan berat badan yang dimaksudkan adalah IMT (Indeks Masa Tubuh) penderita yang melebihi batas standart deviasi. Seseorang dikatakan mengalami obesitas ketika IMT lebih dari 2 Standart Deviasi. Berdasarkan hasil survei yang dilakukan oleh RISKESDAS (Riset Kesehatan Dasar) Indonesia pada tahun 2010, obesitas pada anak sekolah usia 6-12 tahun mencapai prosentase sebesaar 9,2% (Kementerian Kesehatan, 2012). Survei lainnya juga dilakukan WHO pada tahun 2013 dimana obesitas pada anak usia dibawah 19 tahun mencapai prosentase sebesar 47,1% dibandingkan dengan orang dewasa sebesar 27,5% (WHO, 2015).

Obesitas pada anak beresiko menjadi obesitas lanjutan pada saat dewasa dan menyebabkan timbulnya penyakit. Penyakit yang dapat diderita antara lain kardiovaskuler, meningkatnya gula darah yang menyebabkan diabetes mellitus, hipertensi, jantung, dan penyakit lainnya (WHO, 2015). Selain itu, anak usia 6-7 tahun juga dapat menurunkan tingkat kecerdasan dikarenakan faktor malas akibat kelebihan berat badan (Sartika, 2011).

Pengaturan diet anak dapat dilakukan dengan memberikan komposisi makanan yang baik bagi anak. Dalam pelaksanaannya, orang tua terkadang masih sulit untuk mengatur pola makan anak karena kurangnya pengetahuan dalam menyajikan komposisi makanan yang baik bagi anak. Selain kurangnya pengetahuan, masalah biaya menjadi masalah dalam mengatur pola makanan bergizi bagi anak. Dari permasalahan tersebut dapat dicarikan sebuah solusi dengan cara mengembangkan sistem yang mampu memberikan rekomendasi komposisi makanan bagi anak-anak yang mengalami masalah obesitas.

Pengaturan komposisi makanan dapat disusun berdasarkan beberapa kombinasi. Kombinasi penyusunan tersebut dapat dilakukan dengan

menggunakan algoritma genetika. Algoritma genetika merupakan algoritma pencarian heuristik yang didasarkan pada mekanisme evolusi biologis. Dalam pengaplikasiannya, algoritma genetika secara khusus dapat diterapkan untuk memecahkan masalah optimasi yang kompleks. Oleh karena itu algoritma genetika baik untuk aplikasi yang memerlukan strategi pemecahan masalah (Mahmudy, 2013).

Penggunaan algoritma genetika sebagai optimasi komposisi makanan pernah dibahas pada beberapa penelitian, misalkan penelitian yang dilakukan oleh Wahid dan Mahmudy (2015) yang memberikan optimasi bahan makanan bagi penderita Kolesterol agar memperoleh menu makanan dengan komposisi yang optimal dengan harga minimum. Selain itu, penelitian terkait dengan penentuan komposisi menu makanan dilakukan oleh Uyun dan Hartati (2011) yang menentukan menu makanan menggunakan algoritma genetika bagi penderita penyakit ginjal dan saluran kemih. Penelitian ini menghasilkan kombinasi bahan pangan yang optimal dilihat dari nilai fitness yang terbaik.

Berdasarkan penelitian-penelitian yang telah dipaparkan di atas, algoritma genetika mampu diterapkan dalam penentuan optimasi komposisi makanan. Oleh karena itu, pada skripsi ini akan dilakukan penelitian terhadap optimasi komposisi makanan bagi penderita obesitas pada anak. Pada skripsi ini, algoritma genetika digunakan untuk menyelesaikan masalah komposisi makanan bagi anak-anak penderita obesitas dengan rekomendasi menu makanan seimbang dengan asupan kalori dan biaya minimal.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, maka dapat dirumuskan permasalahan yang akan dijadikan objek penelitian untuk skripsi ini, antara lain:

1. Bagaimana menerapkan Algoritma Genetika untuk optimasi komposisi makanan bagi anak-anak yang menderita obesitas?
2. Bagaimana menentukan representasi kromosom untuk optimasi komposisi makanan bagi anak-anak yang menderita obesitas?
3. Bagaimana menentukan parameter Algoritma Genetika yang tepat?

1.3 Batasan Masalah

Batasan masalah pada penelitian ini sebagai berikut:

1. Penelitian ini menggunakan Algoritma genetika tanpa membandingkan dengan metode lain.
2. Penelitian ini digunakan untuk menentukan komposisi makanan bagi penderita obesitas pada anak dengan harga minimal.
3. Data makanan yang digunakan sebanyak 133 data makanan dari Tabel Komposisi Makanan Indonesia serta makalah hasil penelitian oleh Pusat Penelitian dan Pengembangan Gizi dan Makanan Departemen Kesehatan RI

serta Program Nutrien Survey Gizi Universitas Brawijaya dari penelitian Rianawati tahun 2015.
4. Penelitian ini hanya memilih jenis makanan dengan kandungan berat makanan atau bobot makanan yang telah ditentukan.
5. Target usia adalah 4-9 tahun dan tidak memiliki penyakit bawaan.
6. Kandungan Gizi yang digunakan hanya meliputi karbohidrat, lemak, dan protein.
7. Tidak dilakukan pengecekan kesamaan menu dalam sehari untuk setiap individu.

1.4 Tujuan Penelitian

Tujuan dari penelitian skripsi ini antara lain:

1. Untuk mengetahui bagaimana cara menerapkan Algoritma Genetika agar dapat membantu merekomendasikan pengaturan komposisi makanan bagi anak-anak yang menderita obesitas dengan biaya yang minimal.
2. Untuk mengetahui bagaimana cara menentukan representasi kromosom untuk mengoptimasi komposisi makanan bagi anak-anak yang menderita obesitas.

1.5 Manfaat

Manfaat dari penelitian skripsi ini, sebagai berikut:

a. Bagi Ahli Gizi
   Dapat membantu merekomendasikan komposisi makanan bagi anak-anak mengalami obesitas dan terdapat perhitungan untuk mengetahui kebutuhan pasien tersebut.
b. Bagi Orang Tua
   Dapat membantu orang tua dalam mengatur pola makan anak sehari-hari dengan adanya rekomendasi berupa asupan kalori dan harga dari makanan tersebut.

1.6 Sistematika Pembahasan

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

BAB I PENDAHULUAN
Bab ini berisi tentang latar belakang masalahan, rumusan masalah, batasan masalah, tujuan penelitian, manfaat, serta sistematika penyusunan laporan.

BAB II TINJAUAN PUSTAKA
Bab ini berisi tentang teori-teori yang diambil dari sumber pustaka dan referensi yang terkait dengan teori-teori dalam optimasi komposisi makanan menggunakan algoritma genetika yang dapat memberikan solusi permasalahan yang diangkat dalam penelitian ini.

BAB III METODE PENELITIAN

Bab ini menguraikan tentang metode dan langkah kerja yang dilakukan dalam penulisan laporan penelitian yang terdiri dari studi literatur, perancangan sistem perangkat lunak, implementasi sistem perangkat lunak, pengujian dan analisis, serta penulisan laporan.

BAB IV PERANCANGAN

Bab ini menguraikan tentang perancangan sistem yang terdiri dari rangkaian proses kerja serta desain sistem pada permasalahan optimasi komposisi makanan bagi penderita obesitas pada anak menggunakan algoritma genetika.

BAB V IMPLEMENTASI

Bab ini menjelaskan tentang implementasi dari perangkat lunak sesuai dengan perancangan sistem yang telah dibangun dalam penelitian ini.

BAB VI PENGUJIAN DAN ANALISIS

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

BAB VII PENUTUP

Bab ini memuat kesimpulan dari keseluruhan uraian bab-bab sebelumnya, serta saran-saran dari hasil yang diperoleh, yang diharapkan dapat bermanfaat dalam pengembangan selanjutnya.

BAB II TINJAUAN PUSTAKA

2.1 Kajian Pustaka

Kajian pustaka berisi penelitian-penelitian sebelumnya yang terkait atau memiliki kesesuaian dengan penelitian penulis. Kajian pustaka bermanfaat sebagai pembanding penulis dalam melakukan penelitian ini. Perbandingan tersebut ditampilkan pada Tabel 2.1.

Tabel 2. 1 Kajian Pustaka

No.	Judul (Tahun)	Penulis	Perbandingan	
			Kajian pustaka	Skripi penulis
1.	Optimasi Komposisi Makanan Untuk Penderita Kolesterol Menggunakan Algoritma Genetika (Wahid dan Mahmudy, 2015)	Nurbaiti Wahid	Penelitian ini menggunakan algoritma genetika untuk mengoptimalisasi menu makanan bagi penderita kolesterol dengan harga minimal. Dalam penelitian ini, satu kromosom terdiri dari 9 gen yang didapatkan secara acak dan dalam proses algoritma genetikanya menggunakan single point crossover, reciprocal exchange mutation, dan seleksi *Elitism*. Hasil dari komposisi makanan digunakan untuk satu kali makan dengan biaya minimal.	Pada skripsi penulis satu kromosom terdiri dari 15 gen yang didapatkan secara acak berdasarkan nomor indeks jenis makanannya dan dalam proses algoritma genetikanya menggunakan single point crossover, reciprocal exchange mutation, dan seleksi *Elitism*. Hasil dari penelitian berupa komposisi makanan digunakan untuk tiga kali makan dengan biaya minimum.
2.	Penentuan Komposisi Bahan Pangan Untuk Diet Penyakit Ginjal dan Saluran Kemih dengan	Shofwatul Uyun dan Sri Hartati	Pada penelitian ini bertujuan untuk menentukan komposisi makanan yang berguna sebagai diet bagi penderita ginjal dan saluran kemih. Panjang	Pada penelitian ini bertujuan untuk menentukan komposisi makanan bagi penderita obesitas pada anak. Panjang kromosom terdiri dari

	Algoritma Genetika (Uyun dan Hartati, 2011)		kromosom dari penelitian ini sebesar 10 dengan gen yang berisi nomor indeks makanan dan dipilih secara acak. Hasil dari penelitian berupa rekomendasi makanan bagi penderita penyakit ginjal dan saluran kemih dalam satu hari.	15 gen yang didapatkan secara acak dari nomor indeks jenis makanannya. Hasil dari penelitian berupa komposisi makanan digunakan untuk tiga kali makan dengan biaya minimum.
3.	Rancang Bangun Aplikasi Penilaian Status Gizi dan Penentuan Menu Makanan (Susanto, 2011)	Johan Agus Susanto	Pada penelitian ini bertujuan untuk memberikan penilaian terhadap status gizi pengguna aplikasi dan merekomendasikan menu makanan yang sesuai dengan hasil penilaian status gizi. Hasil penilaian menggunakan metode perhitungan gizi Krause. Peneliti menggunakan aplikasi dalam bentuk web dimana pengguna memasukkan beberapa data dan aplikasi akan menghitung kebutuhan kalorinya lalu merekomendasikan menu makanan serta dilengkapi dengan konsultasi gizi dengan ahli gizi yang terhubung ke dalam sistem.	Pada skripsi ini penulis menggunakan algoritma genetika untuk menentukan menu makanan bagi penderita obesitas pada anak dan kombinasi makanan yang dihasilkan didapatkan dari hasil perhitungan proses algoritma genetika seperti crossover, mutasi, dan seleksi *Elitism*.

Sumber : Wahid dan Mahmudy (2015), Uyun dan Hartati (2011), dan Susanto (2011)

Berdasarkan penelitian sebelumnya yang menggunakan algoritma genetika sebagai metode optimasi komposisi makanan, maka algortima genetika mampu menjadi salah satu alternatif untuk menyelesaikan permasalahan yang kompleks terkait dengan optimasi dan penentuan komposisi makanan.

2.2 Kebutuhan Energi

Energi merupakan hukum pertama termodinamika yang dapat berubah bentuk namun tidak dapat diciptakan maupun dimusnahkan. Energi memiliki satuan bernama kilokalori (kkal) untuk menyatakan jumlah kalori tertentu. Energi pada umumnya dikeluarkan dalam bentuk panas sebagai bentuk proses perubahan energi menjadi bentuk energi lain dan dapat diukur dengan *calorimeter*. Menurut WHO (1985), kebutuhan energi pada manusia merupakan kebutuhan yang didapatkan dari konsumsi makanan yang diperlukan untuk memenuhi kebutuhan sesuai dengan ukuran dan komposisi tubuh serta untuk memenuhi pengeluaran energi yang berasal dari aktifitas fisik (Almatsier, 2006).

2.2.1 Karbohidrat

Karbohidrat terdiri atas unsur karbon (C), Hidrogen (H), dan Oksigen (O_2). Karbohidrat memegang peranan sekitar 55-67 %. Karbohidrat tidak dapat larut dalam air tetapi dapat dicerna dengan baik oleh tubuh sebagai energi cadangan (Baskora dkk, 2011). Dalam dunia gizi, karbohidrat dibagi menjadi dua jenis, yaitu kabohidrat sederhana dan karbohidrat kompleks. Karbohidrat sederhana terdiri dari monosakarida, disakarida, gula alcohol, dan oligosakarida. Sedangkan pada karbohidrat kompleks terdiri atas polisakarida dan serat. (Almatsier, 2006).

2.2.2 Lemak

Lemak merupakan senyawa heterogen dan komponen struktural. Lemak memiliki beberapa sifat yaitu larut dalam pelarut nonpolar seperti etanol, eter, kloroform, dan benzena (Almatsier, 2006). Dalam tubuh, lemak dibutuhkan sekitar 20-30% dari total kalori per hari (Baskora dkk, 2012). Lemak diklasifikasikan menjadi tiga bagian yaitu trigliserida, fosfolipid, dan sterol. Trigliserida merupakan lemak tubuh dengan prosentase tertinggi yaitu sebanyak 99% dan terdiri dari asam lemak dan gliserol (Hardinsyah dkk, 2012). namun tubuh tidak dapat memproduksi atau mensintesisnya. (Almatsier, 2006).

2.2.3 Protein

Protein merupakan bagian dari sel hidup dan terbesar setelah air. Protein terdiri dari rantai panjang asam amino yang terikat satu sama lain dalam ikatan peptide. Rantai panjang asam amino dalam protein dibagi menjadi 9 asam amino esensial dan 11 asam amino nonesensial (Almatsier, 2006). Protein memegang peranan sekitar 13-15% dalam tubuh namun tidak untuk disimpan lama. Protein akan diolah menjadi sumber energi secara tidak langsung beriringan dengan aktivitas (Baskora dkk, 2011).

2.3 Obesitas

Obesitas merupakan keadaan dimana terdapat penimbunan zat lemak di dalam tubuh. Obesitas biasanya disebut sebagai kelebihan berat badan yang tidak normal dan disebabkan karena kurangnya aktivitas fisik (Kementerian Kesehatan, 2012). Pada tubuh normal, perbandingan antara lemak tubuh dan berat badan pada wanita sekitar 12-35% sedangkan pada pria sekitar 18-23% (Wijayanti, 2013).

Obesitas dapat menjadi pemicu datangnya berbagai macam penyakit. Penyakit yang dapat disebabkan karena obesitas meliputi Kardiovaskuler, Diabetes Mellitus, Dislipidemia, Kolesterol, dan Hipertensi yang menimbulkan penyakit komplikasi lainnya (Hariadi dan Ali, 2005). Faktor penyebab obesitas menurut Hidayati dkk (2009) sebagai berikut :

1. Faktor Genetik

   Faktor genetik merupakan faktor yang berperan besar. Kemungkinan anak 80% menderita obesitas disebabkan karena orang tuanya juga menderita obesitas, 40 % obesitas bila salah satu orang tuanya obesitas, dan 14% obesitas jika kedua orang tuanya tidak menderita obesitas.
2. Faktor Lingkungan
   a. Aktifitas Fisik

      Aktifitas fisik memiliki sekitar 20-50% peranan sebagai komponen utama *Energy Expenditure* (EE) dari total keseluruhan EE. Individu dengan aktivitas rendah cenderung mengalami kenaikan berat badan (BB) sebesar 5 kg.
   b. Faktor Nutrisional

      Pemberian nutrisi sudah dimulai dari saat sejak dalam kandungan dimana jumlah lemak tubuh dan pertumbuhan bayi dipengaruhi oleh berat badan ibu.
   c. Faktor Sosial Ekonomi

      Perubahan gaya hidup mempengaruhi pemilihan jenis dan jumlah makanan yang dikonsumsi sehari-hari. Selain itu juga ketersediaan makanan cepat saji yang mudah didapatkan saat ini beresiko menimbulkan obesitas.

2.3.1 Obesitas Pada Anak

Obesitas bisa dialami oleh berbagai jenis usia dan jenis kelamin. Obesitas pada anak dikenal dengan istilah *Juvenil Obesity* (Anggraini, Suciaty 2008). Obesitas pada anak dapat dipengaruhi oleh beberapa faktor, antara lain asupan makanan kurang bergizi, aktivitas fisik, dan riwayat pemberian ASI eksklusif pada saat balita. Obesitas pada anak beresiko lebih besar sebanyak 1,8 kali menjadi obesitas pada masa dewasa dan berpotensi mengalami penyakit metabolik dan degeneratif dikemudian hari. Anak-anak yang mengalami obesitas akan memiliki resiko hipertensi lebih besar sekitar 20-30% (Hidayati dkk, 2009)

Untuk menentukan apakah anak termasuk dalam kategori obesitas, dapat diketahui dengan cara pemeriksaan laboratorik antara lain (Hidayati dkk, 2009) :

1. Anak-anak dikatakan obesitas ketika BB > 120% BB standart. Hasil tersebut didapatkan dari pengukuran berat badan (BB) yang dibandingkan dengan standart.
2. Pengukuran BB dibanding dengan TB yang dipangkat dua (dalam meter) dan menghasilkan IMT > Z score (+ 2 SD).
3. Pengukuran *skinfold thicknes* (tebal lipatan kulit). Anak dikatakan obesitas ketika tebal lipatan kulit Triceps > persentil ke 85.
4. Pengukuran lemak laboratorik dengan metode DXA yang dianggap sebagai metode paling akurat tetapi tidak praktis untuk dilapangan

Pada anak-anak terdapat beberapa tatalaksana dalam mengatasi obesitas yang diderita. Berikut adalah prinsip tatalaksana obesitas pada anak-anak (Hidayati dkk, 2009), antara lain :

1. Menetapkan penurunan berat badan (BB)

   Penurunan BB didasarkan pada usia anak yang menderita obesitas. Anak penderita obesitas usia di bawah 7 tahun tanpa ada penyakit penyerta disarankan untuk tetap mempertahankan berat badannya. Sedangkan yang menderita obesitas di atas 7 tahun tanpa penyakit penyerta, target penurunan BB sebesar 2,5 – 5 kg per bulan.
2. Pengaturan diet

   Dalam pengaturan diet, hendaknya tetap memperhatikan dan mempertahankan pertumbuhan anak-anak secara normal.
3. Pengaturan aktifitas fisik

   Pengaturan aktifitas fisik memiliki pengaruh kuat terhadap laju metabolisme dalam tubuh anak-anak. Kemampuan fisik ini diharapkan dapat mengurangi kalori dalam tubuh yang tidak seimbang. Tingkatan aktivitas fisik sehari-hari dapat dilihat pada Tabel 2.2

Tabel 2. 2 Tingkatan Aktivitas Fisik

Level Aktivitas	Jenis Kegiatan
Sedentary (Ringan)	Kegiatan yang berdiri atau duduk, jalan kaki, melukis, kegiatan laboratorium, menulis, bermain kartu, bermain musik
Low Active (Sedang)	Bersepeda, Jalan kaki 6 km/jam, bermain tenis, bermain ski, jogging
Active (Berat)	bermain sepak bola, lari, bersepeda, berenang, bermain basket, berlari
Very Active (Sangat berat)	Berlari marathon, mendayung dengan cepat, bekerja sangat keras

(Sumber: Bowman dan Rusel dalam Rianawati, 2015)

4. Mengubah pola hidup/perilaku

   Dalam hal ini diperlukan kerjasama dari orang tua sebagai komponen intervensi, caranya sebagai berikut :

a. Pengawasan terhadap berat badan (BB) dan mencatat perkembangannya
b. Mengontrol rangsangan untuk makan
c. Mengubah perilaku makan
d. Pengendalian diri dengan menghindari makanan berlemak tinggi

2.3.2 Perhitungan Status Gizi

Cara yang digunakan untuk menentukan jumlah kalori yang dibutuhkan oleh anak-anak umur 4-9 tahun adalah dengan memperhitungkan kebutuhan kalori yang dapat ditambah atau dikurangi bergantung pada jenis kelamin, umur, aktivitas, dan berat badan.

1. Perhitungan pertama yaitu menghitung Indeks Masa Tubuh (IMT) yang digunakan untuk mengetahui apakah anak menderita obesitas atau tidak. Hasil dari perhitungan IMT akan dicocokan dengan tabel Z Score WHO tahun 2005. Rumus perhitungannya sebagai berikut (Susanto, 2009):

   $$\boldsymbol{IMT = BB \div (TB\ x\ TB)} \qquad \textbf{(2-1)}$$

   Keterangan :
   BB = Berat badan (kilogram)
   TB = Tinggi badan (meter)

2. Perhitungan selanjutnya merupakan perhitungan dengan menambahkan jenis kelamin, umur, aktivitas fisik, dan berat badan. Dalam buku panduan Krause tentang nutrisi, dijelaskan persamaan yang digunakan untuk menghitung TEE (*Total Energy Expenditure*) yang dibutuhkan untuk anak usia 4-9 tahun (Krause, 2012):
   Persamaan kecukupan kalori anak usia 4-9 tahun:

   $$\boldsymbol{TEE\ (boys) = 88.5 - (61,9\ x\ U) + (PA\ x\ ((26,7\ x\ BB) + (903\ x\ TB)) + 20\ kkal} \qquad \textbf{(2-2)}$$

   $$\boldsymbol{TEE\ (girls) = 135,3 - (30,8\ x\ U) + (PA\ x\ ((10\ x\ BB) + (934\ x\ TB)) + 20\ kkal} \qquad \textbf{(2-3)}$$

   Keterangan :
   U = Umur
   PA = *Physical Activity* (Aktifitas Fisik)

   Untuk usi 4-9 tahun, *Physical Activity* atau aktifitas fisik anak yang dapat dilihat pada Tabel 2.3 di bawah ini:

Tabel 2. 3 Tabel Aktifitas Fisik Anak

	Boys	Girls
Sedentary	1	1
Low Active	1.13	1.16
Active	1.26	1.31
Very Active	1.42	1.56

(Sumber: Krause, 2012)

3. Setelah mengetahui jumlah kalori yang diperlukan, maka dilakukan perhitungan untuk mencari kebutuhan asupan karbohidrat, protein, dan lemak dalam satu hari.

Umur	Energi Protein (%)	Energi Lemak (%)	Energi Karbohidrat (%)	Total
4-6 tahun	13.1	27	59.8	100
7-9 tahun	13.3	27.2	59.5	100

(Sumber: Hardinsyah dkk, 2012)

2.4 Algoritma Genetika

Algoritma genetika merupakan salah satu bagian dari algoritma evolusi berbasis populasi (Mahmudy, 2013). Algoritma gentika diusulkan pertama kali oleh John Holland dan rekan-rekannya di Universitas Michigan. Dalam pengaplikasiannya, algoritma genetika dapat diterapkan untuk memecahkan masalah yang kompleks. Oleh karena itu algoritma genetika baik untuk aplikasi yang memerlukan strategi pemecahan masalah (Gen dalam Uyun dan Hartati, 2011).

Berdasarkan pengertian di atas, maka dapat diketahui bahwa Algoritma genetika merupakan metode pencarian berdasarkan pada proses seleksi dan proses-proses genetika pada umumnya (Carwoto, 2007). Proses genetika yang ada dalam Algoritma genetika mengadaptasi dari proses evolusi biologi seperti *crossover*, *gen*, *chromosome*, dan mutasi. Hasil dari pemecahan masalah yang dipecahkan oleh Algoritma genetika tidak selalu menghasilkan hasil yang terbaik namun mendekati terbaik (Phetegem dalam Devi dkk, 2010).

Secara umum, algoritma genetika memiliki 5 komponen dasar, meliputi (Michalewicz dalam Uyun dan Hartati, 2011):

- Algoritma genetika menjadi representatif genetik dari solusi yang berasal dari suatu permasalahan.
- Algoritma genetika merupakan suatu cara yang dapat menciptakan inisial populasi dan berasal dari beberapa solusi.

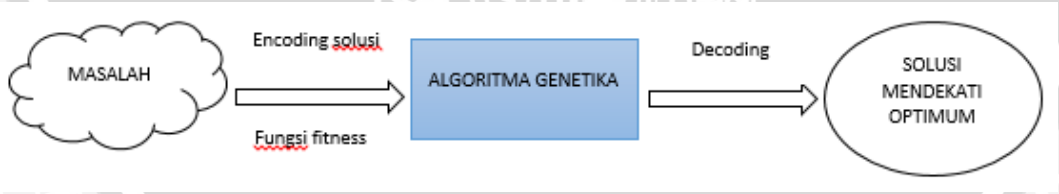
- Terdapat evaluasi fungsi solusi melalui nilai *fitness*. Nilai *fitness* dimiliki oleh masing-masing individu yang terbentuk.
- Adanya proses pembentukan anak atau reproduksi dari operator genetika yang diinisialisasikan.
- Adanya nilai untuk beberapa parameter yang digunakan dalam memproses algoritma genetika.

Terdapat beberapa kelebihan algoritma genetika sebagai metode optimasi (Mahmudy, 2013):

- Algoritma genetika memiliki daerah pencarian yang luas dan kompleks sehingga memungkinkan untuk mencari solusi
- Dapat mengurangi waktu komputasi dikarenakan individu pada populasi dapat diletakkan pada sub populasi dan diproses secara paralel pada sejumlah computer.
- Dapat menyelesaikan masalah kompleks yang memiliki banyak variabel dan dapat menghasilkan solusi mendekati optimal dikarenakan algoritma genetika mengkodekan solusi melalui kromosomnya.
- Algoritma genetika cukup fleksibel untuk diimplementasikan atau dihibridisasi dengan algoritma yang lain.

2.4.1 Struktur Algoritma Genetika

Suatu permasalahan direpresentasikan dengan bentuk kromosom dan dibangkitkan sejumlah solusi atau yang disebut dengan populasi (Kusumadewi dalam Rianawati, 2013). Individu atau kromosom yang telah dipetakan (*encoding*) menggambarkan variabel keputusan yang digunakan dalam mengambil solusi. Individu yang telah direpresentasikan dimasukkan ke dalam algoritma genetika dan diproses menggunakan proses pada algoritma genetika dan dihasilkan suatu nilai solusi (nilai *fitness*). Nilai solusi ini nantinya akan dikonversi (*decoding*) menjadi sebuah solusi yang mendekati optimum (Mahmudy, 2013).



Gambar 2. 1 Pencarian Solusi dalam Algoritma Genetika

Sumber: [Mahmudy, 2013]

2.4.2 Komponen Algoritma Genetika

Pada algoritma genetika terdapat beberapa komponen penting yang digunakan dalam menyelesaikan permasalahan:

A. Representasi Kromosom

Representasi kromosom dalam algoritma genetika dapat menggunakan bilangan real, desimal atau biner. Permasalahan yang akan menggunakan algoritma genetika sebagai pemecahannya harus dikonversikan lebih dahulu ke dalam bentuk individu yang diwakili lebih dari satu *chromosome* dengan kode tertentu (Suryanto dalam Panharesi, 2015). Berikut adalah representasi *chromosome* menggunakan beberapa bilangan yang telah disebutkan di atas :

- Representasi kromosom menggunakan bilangan real dengan interval [0,R] dimana R adalah bilangan real positif dan biasanya R = 1.
- Representasi kromosom menggunakan bilangan decimal dengan deretan bilangan bulat dalam interval [0,9].
- Representasi kromosom menggunakan bilangan biner dengan deretan nilai 0 atau 1.
- Representasi kromosom menggunakan permutasi integer dengan setiap individu berisi satu kromosom yang berisi gen-gen.

B. Inisialisasi

Inisialisasi atau yang biasa disebut juga dengan populasi awal adalah proses membangkitkan himpunan solusi baru secara acak yang terdiri dari atas sejumlah string kromosom dan ditempatkan pada penampungan yang disebut populasi (Mahmudy, 2013).

Dalam tahap ini harus ditentukan ukuran populasi (*popSize*). Ukuran populasi menyatakan banyaknya *chromosome* yang ditampung. Semakin besar ukuran populasi maka kemampuan eksplorasi semakin meningkat (Pitaloka, 2014).

Contoh:

Diketahui popSize = 6 dan representasi chromosome biner basis 2. Nilai x ditentukan antara 0 sampai 31 dan bilangan biner dengan panjang 5 sudah menjangkau nilai x (ingat 11111_2) = 31. Jadi stringLen = 5.

Maka inisialisasi kromosomnya dapat dilihat pada Tabel 2.4 sebagai berikut:

Tabel 2. 4 Inisialisasi Kromosom dengan Biner

	Kromosom	X
P1	[0 0 0 1 1]	3
P2	[0 0 1 1 0]	5
P3	[0 1 1 0 0]	12
P4	[1 1 0 0 0]	24
P5	[1 0 0 0 1]	17

C. Reproduksi

Reproduksi digunakan untuk menghasilkan keturunan. Individu – individu dalam populasi akan dipilih sebagai *parent* dan menghasilkan keturunan yang disebut *offspring*. Ada 2 proses dalam reproduksi atau yang biasa disebut dengan operator genetika, yaitu *crossover* dan mutasi. Masing – masing operator mempunyai banyak sekali metode yang sudah dikembangkan oleh para peneliti dan biasanya bersifat spesifik terhadap masalah dan representasi *chromosome* yang digunakan (Mahmudy, 2013).

1. Crossover

Crossover digunakan dalam proses reproduksi dengan perkawinan silang antara satu atau lebih kromosom induk yang terpilih. Proses *crossover* ini mampu menghasilkan lebih dari satu *offspring*. Secara umum, mekanisme dari proses *crossover* adalah sebagai berikut (Mahmudy, 2013):

1. Menentukan tingkat *crossover* atau *crossover rate*

   Pada proses ini harus ditentukan tingkat *crossover*-nya atau *crossover rate* (cr). Nilai ini yang akan menyatakan jumlah *offspring* yang dihasilkan pada proses *crossover* terhadap ukuran populasi atau *popsize*. Rumus untuk menghitung *offspring* :

   $$Offspring = cr\ x\ popsize$$

2. Memilih kromosom sebagai induk dari populasi secara *random*
3. Memilih titik perkawinan silang secara *random*
4. Melakukan penukaran antar segmen *chromosome* induk untuk menghasilkan *chromosome* anak.

2. Mutasi

Mutasi digunakan dalam proses reproduksi dengan merubah secara acak nilai gen pada *chromosome*. Pada umumnya mutasi memilih satu individu sebagai *parent* dan menghasilkan satu *offspring*. Secara umum mekanisme dari mutasi adalah sebagai berikut (Mahmudy, 2013):

1. Menentukan tingkat mutasi atau *mutation rate*

   Sama seperti *crossover*, tingkat mutasi atau *mutation rate* (mr) juga harus ditentukan untuk mengetahui jumlah *offspring* yang dihasilkan pada proses mutasi terhadap ukuran populasi atau *popsize*.

   Rumus untuk menghitung *offspring* :

   $$Offspring = mr\ x\ popsize$$

2. Memilih secara random individu untuk dijadikan *parent*
3. Memilih secara random titik yang akan dimutasi
4. Melakukan perubahan atau pertukaran pada nilai gen berdasarkan titik mutasi tersebut.

D. Evaluasi

Evaluasi merupakan proses perhitungan nilai *fitness* tiap kromosom. Nilai *fitness* merupakan tolak ukur dalam menentukan generasi selanjutnya. Biasanya, individu yang memiliki nilai *fitness* yang tinggi akan lolos menjadi generasi selanjutnya. Selain itu, nilai *fitness* ini juga digunakan untuk menentukan kualitas dari kromosom tersebut. Pada kasus untuk menghitung nilai maksimum, nilai *fitness* bisa dihitung langsung dengan rumus (Mahmudy, 2013):

$$fitness = f(x)$$

Sedangkan untuk kasus untuk menghitung nilai minimum, nilai *fitness* bisa dihitung dengan salah satu dari 2 rumus berikut ini (Mahmudy, 2013):

$$fitness = C - f(x)$$

$$fitness = \frac{1}{f(x)}$$

E. Seleksi

Seleksi merupakan himpunan populasi (*parent*) dan *offspring* yang dipertahankan untuk hidup pada generasi selanjutnya (Mahmudy, 2013). Individu yang terpilih adalah individu yang mempunyai nilai *fitness* paling baik. Seleksi *elitism* merupakan seleksi yang sering digunakan dalam algoritma genetika.

- *Elitism selection*
  Elitism selection digunakan untuk mengurutkan individu dari nilai fitness yang paling tinggi hingga yang paling rendah. Individu terbaik akan dipilih dari nomor urut 1 sampai ukuran *popSize*. Misalnya himpunan populasi (*parent*) dan *offspring* seperti Tabel 2.5:

Tabel 2. 5 Himpunan populasi parent dan offspring

	Fitness
P1	30,8312
P2	27,2321
P3	20,1211
P4	10,0123
C1	19,2312
C2	12,7231
C3	20,0927

Setelah dilakukan perankingan maka hasilnya sepeerti Tabel 2.6:

Tabel 2. 6 Hasil perankingan himpunan populasi parent dan offspring

	Fitness
P1	30,8312
P2	27,2321
P3	20,1211
C3	20,0927
C1	19,2312
C2	12,7231
P4	10,0123

Berdasarkan himpunan populasi (*parent*) diketahui ukuran populasi awal=4 maka individu yang terpilih adalah individu dengan nomor urut 1 sampai dengan 4 sehingga populasi untuk generasi selanjutnya ditunjukkan pada Tabel 2.7:

Tabel 2. 7 Populasi baru (parent) pada generasi selanjutnya

	Fitness
P1	30,8312
P2	27,2321
P3	20,1211
C3	20,0927

2.4.3 Kondisi Berhenti

Suatu iterasi dalam algoritma genetika akan berhenti jika mengalami kondisi seperti berikut ini (Mahmudy, 2013):

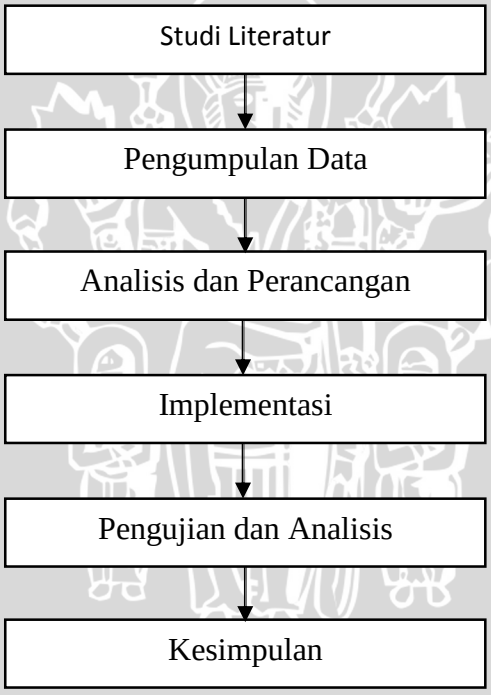
- Iterasi berhenti ketika iterasi ke – n (Yogeswaran, Ponnambalam & Tiwari dalam Mahmudy, 2013).
- Iterasi berhenti ketika setelah n generasi berurutan tidak dijumpai solusi yang lebih baik (Mahmudy, Marian & Luong dalam Mahmudy, 2013).
- Iterasi berhenti setelah satuan waktu (t) tercapai dan biasanya digunakan untuk melakukan perbandingan performa algoritma.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan tahapan penelitian yang digunakan dalam menyusun skripsi dengan judul "Optimasi Komposisi Makanan Bagi Penderita Obesitas Pada Anak Menggunakan Algoritma Genetika".

3.1 Tahapan Penelitian

Tahapan penelitian atau yang disebut dengan metodologi penelitian menjelaskan mengenai langkah – langkah yang digunakan dalam pembuatan aplikasi optimasi komposisi makanan bagi anak-anak yang menderita obesitas. Tahapan penelitian yang dilakukan dalam penelitian ini memiliki beberapa tahap yaitu studi literatur, analisis dan perancangan sistem perangkat lunak, implementasi sistem perangkat lunak, pengujian dan analisis, serta penarikan kesimpulan. Gambar 3.1 berikut ini menunjukkan langkah – langkah yang dilakukan dalam penelitian penelitian.



Gambar 3. 1 Diagram Blok Metode Penelitian

3.1.1 Studi Literatur

Pada tahap ini, peneliti mengumpulkan studi literatur terkait dengan penelitian yang dilakukan. Studi literatur ini digunakan sebagai landasan teori untuk dipelajari dan dianalisis terkait dengan permasalahan yang ada guna menunjang penelitian. Adapun teori – teori yang dibahas sebagai berikut:

1. Kebutuhan Energi

2. Obesitas, Obesitas Anak, dan Perhitungan Status Gizi
3. Algoritma Genetika

Studi literatur yang digunakan dalam penelitian ini berasal dari jurnal, skripsi, buku dan artikel.

3.1.2 Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data makanan yang digunakan pada perangkat lunak dan data anak penderita obesitas.

1. Data makanan dan kandungan gizi dalam penelitian ini merupakan data dari Tabel Komposisi Makanan Indonesia serta makalah hasil penelitian oleh Pusat Penelitian dan Pengembangan Gizi dan Makanan Departemen Kesehatan RI serta Program Nutrien Survey Gizi Universitas Brawijaya dari penelitian Rianawati tahun 2015.
2. Selain itu, penelitian ini juga menggunakan data anak penderita obesitas yang didapatkan dari Puskesmas Mulyorejo, Malang bulan Februari 2015 dan hasil wawancara di SDN Gedongan 1 Kota Mojokerto.

3.1.3 Analisis dan Perancangan

Pada tahap ini dilakukan analisis dan perancangan yang bertujuan untuk menganalisis permasalahan serta perancangan aplikasi pada permasalahan optimasi komposisi makanan menggunakan algoritma genetika yang bertujuan untuk meminimalkan biaya dan menentukan komposisi makanan yang tepat. Tahap analisis permasalahan digunakan sebagai pemecahan kasus agar dapat dipahami dan digunakan untuk menyelesaikan studi kasus optimasi komposisi makanan bagi anak penderita obesitas.

3.1.4 Implementasi

Pada tahap ini, aplikasi dibuat sesuai dengan analisis dan perancangan sistem yang dilakukan pada tahap sebelumnya. Permasalahan optimasi komposisi makanan dengan menerapkan algoritma genetika diimplementasikan agar dapat diuji dan dianalisis terhadap permasalahan yang ada.

3.1.5 Pengujian dan Analisis

Pada tahap ini, permasalahan optimasi komposisi makanan yang sudah diimplementasikan berupa aplikasi diuji apakah sistem telah mampu bekerja sesuai dengan spesifikasi kebutuhannya.

3.1.6 Kesimpulan

Pada tahap ini dilakukan penarikan kesimpulan yang menjadi akhir dari seluruh rangkaian proses pembuatan aplikasi. Penarikan kesimpulan didasarkan

pada hasil uji yang telah dilakukan. Selain itu, pada kesimpulan diharapkan dapat menjadi acuan bagi peneliti lain untuk perkembangan aplikasi berikutnya.

3.2 Teknik Pengumpulan Data

Dalam penelitian ini, data yang digunakan sebagai berikut

1. Data penderita obesitas pada anak di dapatkan dari Puskesmas Mulyorejo Kota Malang bulan Februari 2015 dan hasil wawancara di SDN Gedongan 1 Kota Mojokerto.
2. Data makanan dan kandungan gizi didapatkan dari Tabel Komposisi Makanan Indonesia serta makalah hasil penelitian oleh Pusat Penelitian dan Pengembangan Gizi dan Makanan Departemen Kesehatan RI serta Program Nutrien Survey Gizi Universitas Brawijaya dari penelitian Rianawati tahun 2015.

3.3 Algoritma yang Digunakan

Sistem yang dibangun dalam penelitian ini adalah sistem optimasi komposisi makanan bagi penderita obesitas pada anak menggunakan algoritma genetika. Algoritma genetika dianggap mampu dalam penyelesaian masalah optimasi makanan sesuai dengan penjelasan yang telah dipaparkan pada Bab 2. Dalam penyelesaian masalahnya, algoritma genetika menghasilkan nilai *fitness* yang digunakan untuk menentukan individu mana yang baik sebagai hasil optimasi komposisi makanan.

3.4 Lingkungan Uji Coba

Dalam pembuatan aplikasi, peneliti memerlukan beberapa kebutuhan baik berupa kebutuhan perangkat lunak maupun perangkat keras serta data yang diolah pada aplikasi ini. Berikut ini adalah kebutuhan sistem yang menunjang pembuatan aplikasi:

1. Kebutuhan Perangkat Keras, meliputi:
   - Processor Intel® Core i3-4030U 1,90 GHz 1,90GHz
   - RAM 2.00 GB
   - Harddisk 500 GB

2. Kebutuhan Perangkat Lunak, meliputi:
   - Operating System Windows 8.1 64 bit
   - Netbeans IDE 8.1
   - Microsoft Office Word dan Excel 2016
   - Astah Professional 7

3.5 Pengujian Algoritma

Pengujian dilakukan dengan menguji keakuratan algoritma melalui nilai fitness yang dihasilkan oleh masing-masing hasil optimasi makanan yang dihasilkan selama proses pengujian. Pengujian terdiri dari tiga skenario, yaitu:

1. Skenario Pengujian Ukuran Populasi (*popSize*)

Pada skenario pengujian ini akan dilakukan pengujian ukuran populasi terhadap nilai *fitness*. Jumlah populasi yang digunakan adalah kelipatan 20 mulai dari populasi 20 sampai populasi 200. Setiap ukuran populasi yang diuji akan dilakukan 10 kali percobaan.

2. Skenario Pengujian Banyaknya Generasi

Pada skenario pengujian ini akan dilakukan pengujian ukuran generasi terhadap perubahan nilai *fitness*. Jumlah generasi yang digunakan adalah kelipatan 25 mulai dari 25 generasi sampai 150 generasi. Setiap ukuran banyaknya generasi yang diuji akan dilakukan 10 kali percobaan.

3. Skenario Pengujian Kombinasi *Crossover Rate* (*cr*) dan *Mutation Rate* (*mr*)

Pada skenario pengujian ini akan dilakukan pengujian kombinasi *crossover rate* (*cr*) dan *mutation rate* (*mr*) terhadap perubahan nilai *fitness*. Nilai cr dan mr yang akan diujikan adalah antara 0 sampai dengan 1. Setiap kombinasi *cr* dan *mr* yang diuji akan dilakukan sebanyak 10 kali percobaan.

BAB IV PERANCANGAN

4.1 Formulasi Permasalahan

Permasalahan yang diangkat melibatkan anak-anak yang mengalami obesitas. Anak penderita obesitas dianggap mengkonsumsi makanan yang kurang sesuai dengan asupan gizinya. Porsi makanan yang dikonsumsi sehari-hari oleh anak terkadang tidak diperhitungkan oleh orang tua disebabkan karena kurang pengetahuan terhadap pola konsumsi anak sehari-hari. Selain itu proses perhitungan manual yang dilakukan untuk mendapatkan makanan yang baik bagi anak membutuhkan waktu yang relatif lama.

Perhitungan manual mengenai asupan gizi pada anak, dapat dicontohkan dari perhitungan di bawah ini. Misalkan diketahui data anak penderita obesitas:

Berat Badan = 38 kg
Tinggi Badan = 120 cm
Umur = 7 tahun
Jenis Kelamin = Perempuan
Aktivitas anak (PA) = Active

Setelah mendapat data anak yang mengalami obesitas, kita dapat menghitung total kalori yang dianjurkan bagi anak tersebut menggunakan persamaan (2-2). Contoh perhitungannya sebagai berikut:

BEE perempuan (Total Kalori) = 135.3 – (30.8 x 7) + (1.31 x ((10 x 38) + (934 x 1,2)) + 20 kkal = 1558.3 kkal

Langkah selanjutnya setelah mendapatkan total kalori harian anak penderita obesitas, maka dihitung asupan gizi meliputi kandungan karbohidrat, lemak, protein. Kebutuhan nutrisi tersebut dapat dilihat pada Tabel 4.1 sebagai berikut:

Tabel 4. 1 Kebutuhan Nutrisi Anak Usia 7-9 tahun

Karbohidrat (%)	Lemak (%)	Protein (%)
59,5 %	27,2 %	13,3 %

(Sumber: Hardinsyah dkk, 2012)

Berdasarkan Tabel 4.1 tersebut, selanjutnya dilakukan proses perhitungan kebutuhan nutrisi yang dibutuhkan oleh anak yang mengalami obesitas. Perhitungannya dapat dilakukan seperti:

Total Kalori = 1558.3 kkal
Karbohidrat = 59.5% x 1558.3 = 927.1885 kkal
Lemak = 27.2% x 1558.3 = 423.8576 kkal

Protein = 13.3% x 1558.3 = 207.2539 kkal

Dalam perhitungan di atas didapatkan energi karbohidrat, protein, lemak dalam satuan kkal. Selanjutnya satuan tersebut dikonversikan ke dalam gram dimana untuk 1 gram karbohidrat = 4 kkal, 1 gram protein = 4 kkal, dan 1 gram lemak = 9 kkal. Berikut hasil konversi hasil kandungan gizi dari kkal menuju gram:

Karbohidrat = 927.1885 kkal / 4 = 231.797 gram
Lemak = 423.8576 kkal / 9 = 47.095 gram
Protein = 207.2539 kkal / 4 = 51.813 gram

Setelah melakukan proses perhitungan total kalori dan kandungan gizi anak akan diteruskan dengan memilih makanan yang disediakan pada Lampiran 1.3-1.7. Makanan yang disediakan sebanyak 133 dengan kandungan nutrisi karbohidrat, lemak, dan protein. Makanan yang disediakan dibagi menjadi 5 kategori yaitu makanan pokok dengan bobot sebesar 180 gram, sumber nabati dengan bobot sebesar 50 gram, sumber hewani dengan bobot sebesar 80 gram, sayuran dengan bobot sebesar 200 gram, serta pelengkap dengan bobot sebesar 150 gram. Dari kategori makanan tersebut, dimisalkan makanan yang dipilih adalah sebagai berikut:

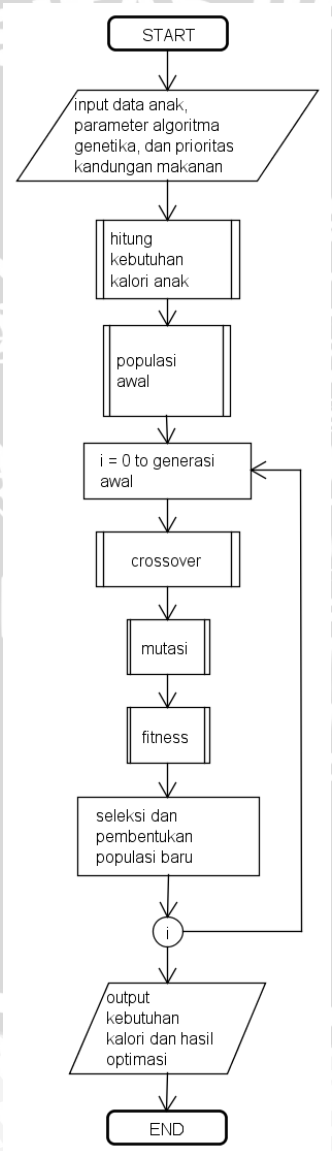
	Makanan	Karbohidrat (gram)	Protein (gram)	Lemak (gram)	Harga
Makan Pagi	Nasi	51.5	4.3	0.4	2070
	Biji Kecipir	12.7	15.4	8.4	2200
	Bawal	0	14.6	0.6	1040
	Ketimun	5.6	1.4	0.2	5600
	Alpukat	17.7	0.8	5.7	900
Makan Siang	Nasi Jagung	49	5	1.1	1080
	Kacang Tolo	10.4	3.8	0.3	300
	Bebek	0	15.2	22.7	3600
	Jengkol	15.6	5.4	0.2	6400
	Arbei	19.2	1	0.6	3750
Makan Malam	Nasi Uduk	42.3	3.8	2.7	2100
	Ampas Tahu	4.8	20.3	12	500
	Daging Sapi	0	19.9	14.4	7600
	Gambas Oyong	8.6	1.8	0.6	1500
	Duku	23	0.3	0.6	1875
	Total	260.4	113	70.5	40515

Dalam makanan yang terpilih di atas, total kandungan karbohidrat sebesar 260.4 gram, total kandungan lemak sebesar 70.5 gram, dan total kandungan protein sebesar 113 gram. Hal tersebut menunjukkan makanan yang kita pilih

masih belum memenuhi keterpenuhan dari kandungan nutrisi yang seharusnya dikonsumsi oleh anak dan biaya yang didapatkan juga sangat besar. Oleh sebab itu, dibutuhkan algoritma genetika yang mampu memberikan kombinasi makanan yang mendekati baik berdasarkan besarnya nilai *fitness* tertinggi yang dihasilkan dengan harga yang minimum.

4.2 Siklus Algoritma Genetika

Siklus penyelesaian masalah menggunakan algoritma genetika dijelaskan melalui beberapa tahapan – tahapan yang dilalui sistem dalam menyelesaikan masalah optimasi komposisi makanan dengan algoritma genetika. Penjelasan lebih lanjut dapat dilihat pada Gambar 4.1 berikut ini:



Gambar 4. 1 Flowchart Algoritma Genetika

Secara garis besar, proses algoritma genetika diilustrasikan pada Gambar 4.1 prosesnya meliputi:

1. Inisialisasi parameter
   a. Data anak penderita obesitas meliputi berat badan, tinggi badan, umur, jenis kelamin, dan aktivitas sehari-hari
   b. Parameter algoritma gentika meliputi *popsize*, generasi, *crossover rate*, dan *mutation rate* serta nilai prioritas kebutuhan gizi (nilai alpha α).
2. Menghitung kebutuhan kalori harian anak
   Pada proses ini digunakan untuk menghitung kalori, kebutuhan lemak, kebutuhan karbohidrat, dan kebutuhan protein harian
3. Membangkitkan populasi awal yang terdiri dari beberapa individu secara acak.
4. Melakukan proses *crossover* yang bertujuan untuk menghasilkan *offspring* sesuai dengan *crossover rate* dari kromosom yang ada.
5. Melakukan proses mutasi berdasarkan nilai *mutation rate* dari kromosom yang ada
6. Melakukan proses seleksi elitism untuk memilih individu baru sebanyak jumlah populasi awal dan dijadikan populasi pada generasi selanjutnya.

Permasalahan dalam menentukan komposisi makanan menggunakan algoritma genetika telah dijabarkan melalui diagram alir pada Gambar 4.1. Dari diagram alir tersebut, selanjutnya dapat ditetapkan inisialisasi awal parameter algoritma genetika sebagai berikut:

Ukuran populasi (popsize) = 5
Jumlah generasi = 1
Crossover rate (cr) = 0.6
Mutation rate (cr) = 0.2

4.2.1 Representasi Kromosom

Representasi kromosom pada penelitian ini direpresentasikan menggunakan bilangan integer. Bilangan integer berisi nomor (indeks) makanan yang akan dikonsumsi dengan bobot makanan masing-masing sebagai berikut: makanan pokok 180 gram, sumber nabati sebesar 50 gram, sumber hewani sebesar 80 gram, sayuran sebesar 200 gram, dan pelengkap sebesar 150 gram. Satu kromosom berisi 15 gen penyusun. Susunan gen tersebut dapat dikonsumsi sebanyak 3 kali makan dalam waktu satu hari.

Setiap 5 gen akan menunjukkan waktu makan bagi anak, 5 gen pertama merupakan komposisi makanan pagi hari, 5 gen kedua merupakan komposisi makanan siang hari, dan 5 gen ketiga merupakan komposisi makanan malam hari. Sebelumnya representasi kromosom bilangan integer ini pernah dilakukan untuk menerapkan optimasi komposisi makanan bagi penderita kolesterol oleh Wahid dan Mahmudy (2015). Representasi kromosom untuk Parent 1 yang digunakan dapat dilihat pada Tabel 4.2 sebagai berikut:

Tabel 4. 2 Representasi Kromosom

	Makan Pagi					Makan Siang					Makan Malam				
	P	N	H	S	PL	P	N	H	S	PL	P	N	H	S	PL
P1	0	6	3	9	2	1	9	4	6	4	3	7	8	5	5

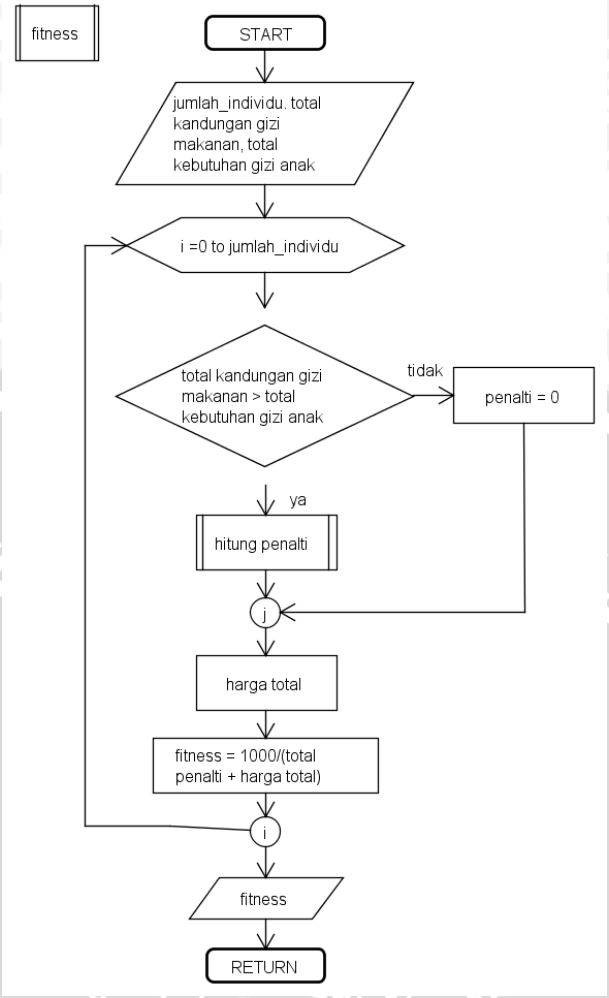
Keterangan:
P = Makanan Pokok S = Sayuran
L = Sumber Nabati PL = Pelengkap
H = Sumber Hewani

Berdasarkan representasi kromosom pada Tabel 4.2 di atas, terdapat nomor makanan yang merujuk pada nama makanan. Misalkan nilai 0 akan merujuk pada makanan pokok indeks ke-0 yaitu Nasi Putih, nilai 6 akan merujuk pada sumber nabati indeks ke-6 yaitu biji kecipir, dan seterusnya sesuai dengan penomoran nama makanan yang terdapat pada Lampiran 1.3 – 1.7.

Hasil dari tiap individu dapat dilakukan dengan cara perhitungan nilai fitness. Nilai fitness yang dihasilkan digunakan untuk menunjukkan kualitas dari masing-masing individu. Hasil perhitungan digunakan sebagai masukan dalam proses seleksi untuk mencari individu terbaik. Individu terbaik yang terpilih dapat menjadi solusi dalam penyelesaian masalah. Untuk perhitungan *fitness* dapat dihitung dengan menggunakan persamaan (4-2).

$$fitness = \frac{1000}{total\ penalti + jumlah\ harga\ makanan}$$

Dari persamaan di atas, dapat diketahui nilai konstanta (angka pembilang) sebesar 1000 diambil dari rata-rata harga semua makanan yang berkisar ribuan. Total penalti merupakan jumlah dari penalti masing-masing kandungan gizi (karbohidrat, lemak, dan protein) yang didapatkan dari perkalian prioritas kebutuhan masing-masing kandungan gizi (nilai α) dengan selisih antara total kandungan gizi dan kebutuhan gizi anak. Jumlah harga makanan merupakan penjumlahan seluruh harga satu kromosom yang terdiri dari 15 gen. Berikut adalah flowchart perhitungan untuk mendapatkan nilai *fitness*:



Gambar 4. 2 Flowchart Perhitungan Nilai *Fitness*

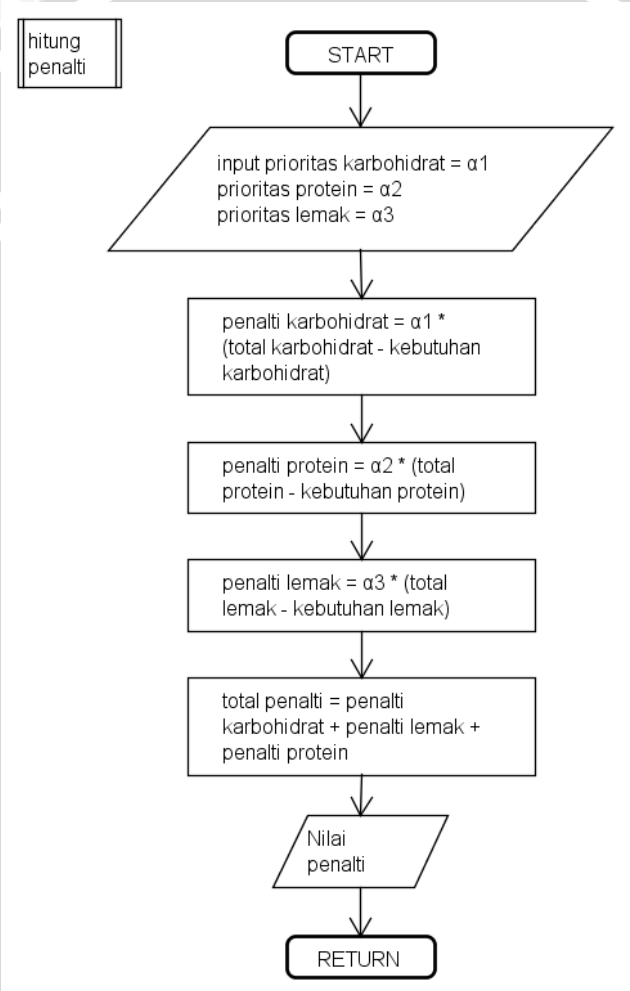
Perhitungan penalti dapat dilakukan dengan melihat flowchart perhitungan penalti pada Gambar 4.3. Penalti merupakan nilai pelanggaran yang tidak sesuai dengan aturan yang seharusnya. Dalam kasus ini, penalti terjadi apabila total kebutuhan lemak, karbohidrat, dan protein yang dikonsumsi pada makanan lebih dari total lemak, karbohidrat, dan protein yang dibutuhkan satu hari. Penalti yang didapatkan dari masing-masing kandungan lemak, karbohidrat, dan protein, kemudian dikalikan dengan nilai alpha (α) yang merupakan prioritas kebutuhan nutrisi untuk menghasilkan nilai *fitness*. Nilai alpha berfungsi untuk mendapatkan suatu individu dengan kandungan lemak yang rendah. Diharapkan bahwa individu dengan kandungan lemak yang rendah agar dapat terus terseleksi ke generasi selanjutnya.

$$Total\ penalti = (\alpha 1 * penalti\ 1) + (\alpha 2 * penalti\ 2) + (\alpha 3 * penalti\ 3\ ) \quad (4\text{-}1)$$

Keterangan :
Penalti 1 = penalti karbohidrat dengan α = 2
Penalti 2 = penalti protein dengan α = 2
Penalti 3 = penalti lemak dengan α = 10

Sesuai dengan persamaan (4-1), aturan penalti yang berlaku adalah sebagai berikut:

1. apabila total lemak, karbohidrat, dan protein pada makanan > total kebutuhan lemak, karbohidrat, dan protein yang harus dikonsumsi, maka dilakukan perhitungan selisihnya.
2. Apabila total lemak, karbohidrat, dan protein pada makanan ≤ total kebutuhan lemak yang harus dikonsumsi, maka nilai penalti yang didapatkan adalah 0.



Gambar 4. 3 Flowchart Perhitungan Penalti

Berikut adalah contoh perihitungan penalti dan nilai *fitness* pada satu individu yang telah direpresentasikan pada Tabel 4.3 sebagai berikut:

Tabel 4. 3 Detail Kromosom 1 dan Perhitungan Fitness

Waktu	No Makanan	Makanan	Karbohidrat (gram)	Protein (gram)	Lemak (gram)	Harga
Makan Pagi	0	Nasi	51.5	4.3	0.4	2070
	6	Biji Kecipir	12.7	15.4	8.4	2200
	3	Bawal	0	14.6	0.6	1040
	9	Ketimun	5.6	1.4	0.2	5600
	2	Alpukat	17.7	0.8	5.7	900
Makan Siang	1	Nasi Jagung	49	5	1.1	1080
	9	Kacang Tolo	10.4	3.8	0.3	300
	4	Bebek	0	15.2	22.7	3600
	6	Jengkol	15.6	5.4	0.2	6400
	4	Arbei	19.2	1	0.6	3750
Makan Malam	3	Nasi Uduk	42.3	3.8	2.7	2100
	7	Ampas Tahu	4.8	20.3	12	500
	8	Daging Sapi	0	19.9	14.4	7600
	5	Gambas Oyong	8.6	1.8	0.6	1500
	5	Duku	23	0.3	0.6	1875
		Total	260.4	113	70.5	40515
		Selisih	28.6029	61.1865	23.40471111	
		Total Penalti	413.6259111			
		Fitness	0.024432777			

Keterangan:

1. No. Makanan ke-n merupakan isi kromosom yang berisi nomor makanan
2. Perhitungan selisih dan penalti masing-masing kandungan gizi:
   - Selisih lemak adalah 23.40471 dikarenakan total lemak pada makanan > total kebutuhan lemak anak. Perhitungan penalti lemak = 234.0471 didapatkan dari hasil perkalian selisih lemak dengan nilai alpha lemak =10.
   - Selisih karbohidrat = 28.6029 dikarenakan total karbohidrat > kebutuhan karbohidrat anak. Nilai penalti yang dihasilkan sebesar 57.2058 yang didapatkan dari perkalian selisih karbohidart dengan nilai alpha karbohidrat = 2.
   - Selisih penalti protein = 61.1865 dikarenakan total protein > total kebutuhan protein anak. Nilai penalti yang didapatkan sebesar 122.373 yang didapatkan dari perkalian selisih protein dengan nilai alpha protein = 2.

3. Nilai total penalti didapatkan dari prioritas persamaan (4-1)

$$Penalti = (\alpha 1 * penalti\ 1) + (\alpha 2 * penalti\ 2) + (\alpha 3 * penalti\ )$$

$$Penalti = 57.2058 + 122.373 + \ 234.0471 = 413.6259$$

4. Perhitungan nilai fitness didapatkan dari persamaan (4-2)

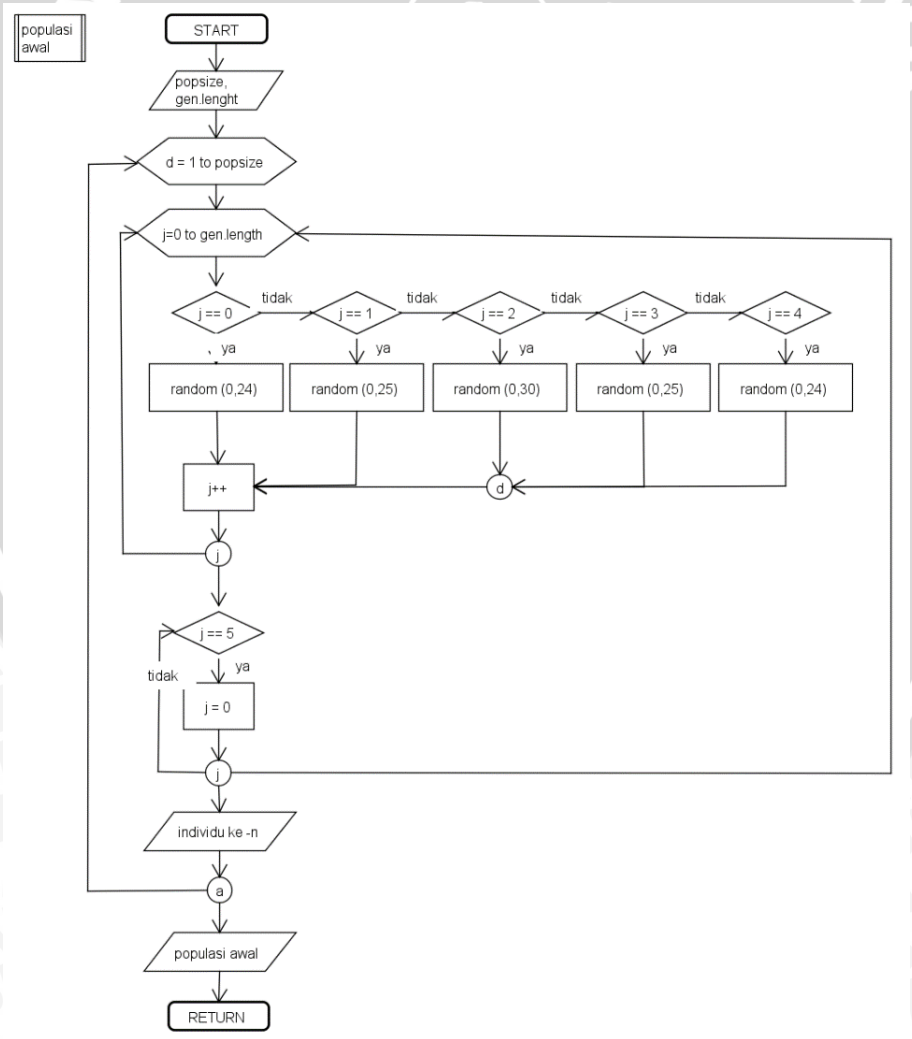
$$fitness = \frac{1000}{total\ penalti + jumlah\ harga\ makanan}$$

$$fitness = \frac{1000}{413.6259 + 40515} = 0.024432777$$

Maka nilai *fitness* yang didapatkan sebesar 0.024432777

4.2.2 Inisialisasi Populasi Awal

Proses inisialisasi populasi awal merupakan proses yang digunakan untuk membangkitkan populasi awal sebanyak ukuran populasi yang telah ditetapkan. Terdapat flowchart yang menggambarkan langkah-langkah dalam membangkitkan populasi awal:



Gambar 4. 4 Flowchart Pembangkitan Populasi Awal

Setelah melakukan proses pembangkitan populasi maka dicontohkan kita telah mendapatkan populasi awal dari setiap populasi yang terdiri dari nomor makanan yang dipilih secara random dan dapat dilihat pada Tabel 4.4 sebagai berikut:

Tabel 4. 4 Populasi Awal

Individu	P	N	H	S	Pl	P	N	H	S	Pl	P	N	H	S	Pl
P1	0	6	3	9	2	1	9	4	6	4	3	7	8	5	5
P2	3	19	2	14	3	4	25	10	8	10	7	4	19	11	9
P3	2	5	6	2	6	6	12	9	4	7	4	6	11	8	10
P4	5	0	5	19	7	9	4	7	3	2	10	15	12	2	3
P5	7	10	0	7	9	13	8	1	10	3	10	9	2	3	0

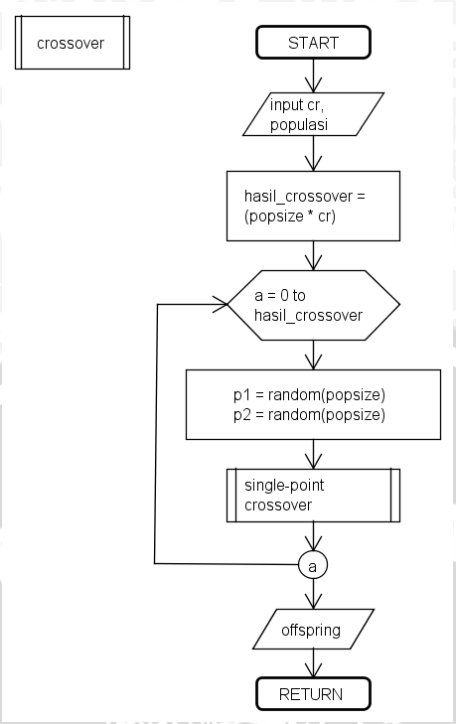
Keterangan:
Individu = urutan kromosom hasil pembangkitan

4.2.3 Reproduksi

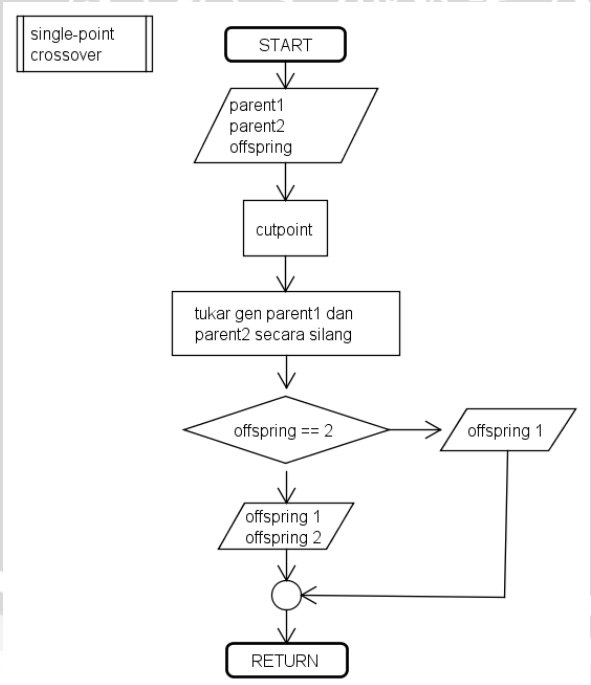
Proses reproduksi dilakukan setelah tahapan penentuan populasi untuk menghasilkan keturunan (*offspring*) dari kromosom yang terpilih untuk melakukan proses reproduksi. Hasil tersebut didapatkan melalui proses pindah silang (*crossover*) dan mutasi. Proses pindah silang (*crossover*) merupakan proses menghasilkan *offspring* dengan cara menukarkan sebagian gen dari dua induk yang dipilih secara random. Jumlah *offspring* yang akan dihasilkan diperoleh dari perkalian antara *crossover rate* dan ukuran populasi $Offspring = cr\ x\ popsize$. Mutasi merupakan proses menghasilkan *offspring* dengan cara menukarkan informasi dari induk yang sama. Jumlah *offspring* dapat dihasilkan melalui perkalian antara *mutation rate* dan ukuran populasi $Offspring = mr\ x\ popsize$.

A. Pindah Silang (Crossover)

Berdasarkan inisialisasi awal diketahui ukuran populasi sebanyak 5 dan ditetapkan *crossover rate* sebesar 0.6, maka jumlah *offspring* yang dihasilkan sebanyak 0.6 x 5 = 3 offspring. Metode yang digunakan dalam pindah silang ini adalah metode *single-point crossover*, dimana 2 induk dipilih secara random untuk menghasilkan *offspring*. Berikut adalah *flowchart* untuk menghitung banyaknya *offspring* yang dihasilkan dari proses crossover:

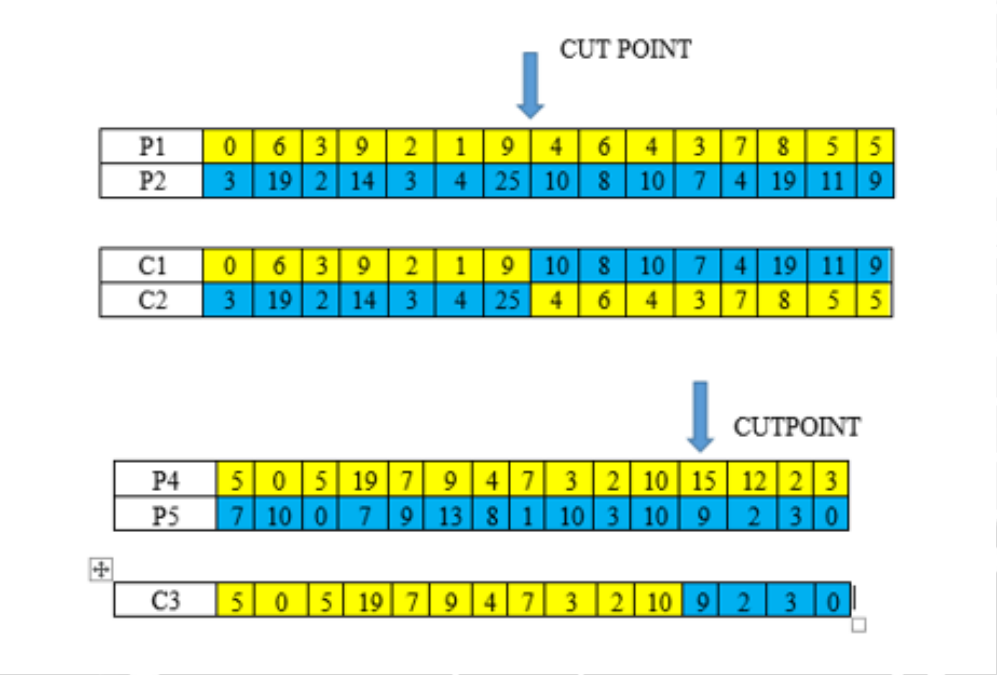


Gambar 4. 5 *Flowchart* Proses Crossover



Gambar 4. 6 Flowchart Single Point Crossover

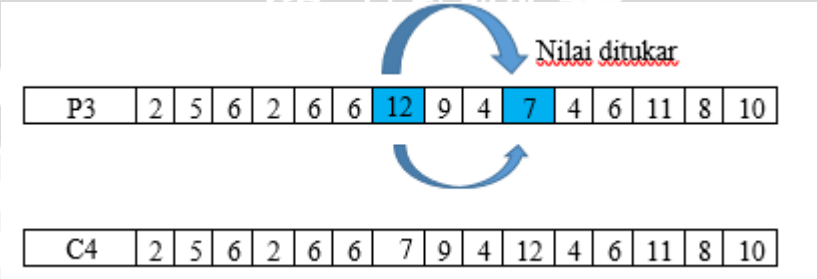
Berdasarkan *flowchart* di atas dapat dicontohkan secara manual proses pembentukan keturunan dengan *single point crossover* pada Gambar 4.7 berikut ini:



Gambar 4. 7 Contoh Proses Crossover

B. Mutasi

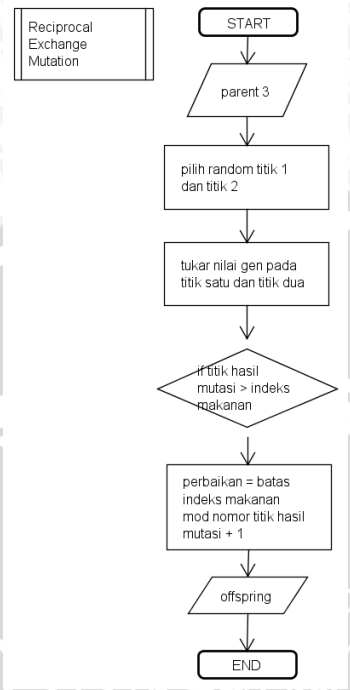
Tahapan berikutnya adalah mutasi. Sama halnya dengan *crossover*, hasil dari mutasi dihitung berdasarkan ukuran populasi dikalikan dengan *mutation rate*. Diketahui ukuran populasi sebanyak 5 dan mr sebesar 0.2, sehingga *offspring* yang dihasilkan sebanyak 5x 0.2 = 1 *offspring*. Metode mutasi yang digunakan adalah metode *Reciprocal Exchange Mutation* dua titik dimana metode ini menukar dua gen dalam satu kromosom (Mahmudy, 2013). Misalkan terpilih nomor 12 pada makanan sumber nabati dan nomor 7 pada pelengkap, maka nilai 12 dan 7 akan ditukar.



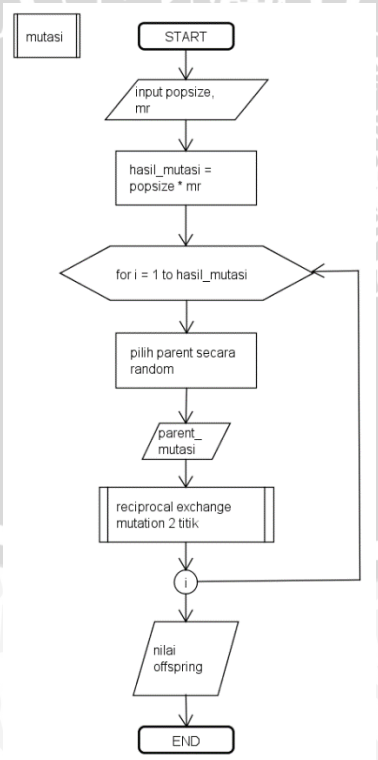
Gambar 4. 8 Contoh Proses Mutasi

Apabila dalam penukaran nilai melebihi batasan nomor indeks masing-masing jenis makanan, maka akan dilakukan proses perbaikan dengan menyesuaikan batasan dari masing-masing data jenis makanan pada database.

Berikut adalah flowchart pembentukan *offspring* dari proses Reciprocal Exchange Mutation:



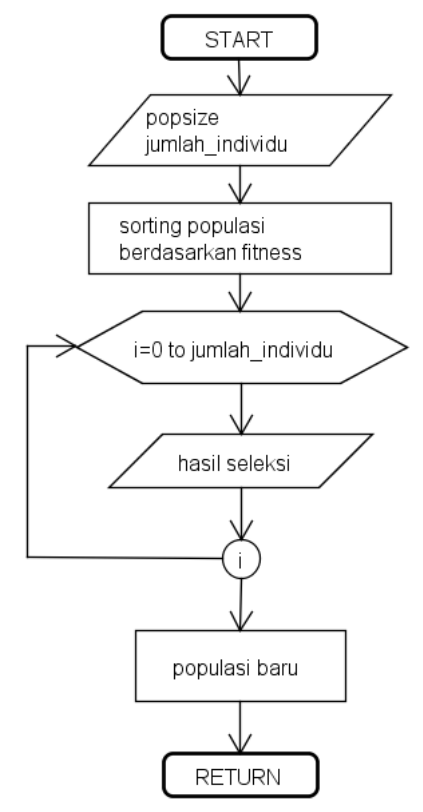
Gambar 4. 9 Flowchart Reciprocal Exchange Mutation



Gambar 4. 10 Flowchart Proses Mutasi

4.2.4 Seleksi

Seleksi merupakan proses memilih individu dengan nilai *fitness* terbaik untuk menjadi individu baru dan menjadi generasi selanjutnya. Pada penelitian ini digunakan metode seleksi *elitism*. Seleksi *elitism* dapat direpresentasikan melalui langkah-langkah dalam *flowchart* di bawah ini:



Gambar 4. 11 *Flowchart* Seleksi Elitism

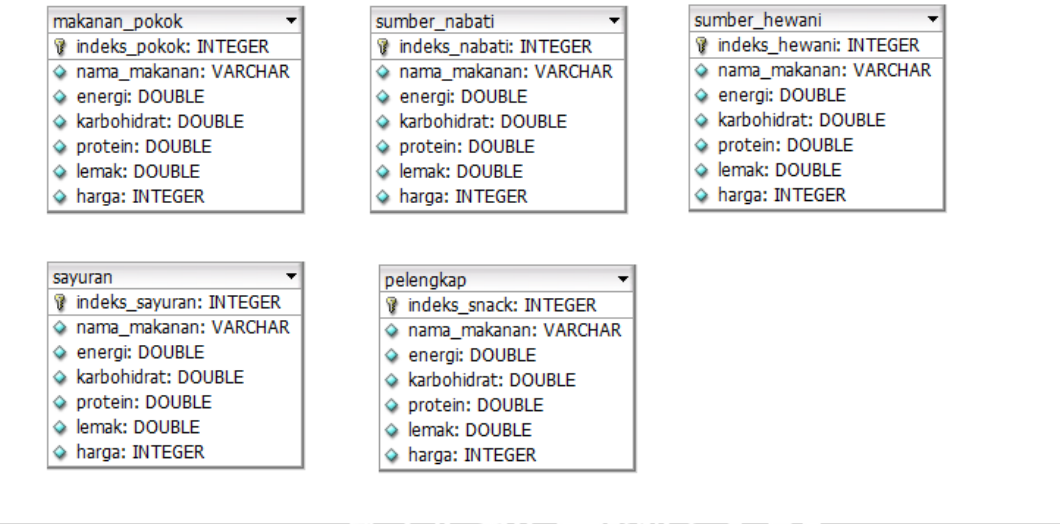
Seleksi *elitism* mengurutkan nilai *fitness* dari tinggi ke rendah dan memilih individu dengan nilai *fitness* tinggi yang berpeluang menjadi generasi berikutnya sebanyak jumlah populasi awal.

Tabel 4. 4 Individu Baru Hasil Seleksi

Individu	Asal	Total Asupan Kalori tiap Kromosom			Harga	Fitness
		Karbohidrat	Protein	Lemak		
P1	P1	260.4	113	70.5	40515	0.024432777
P2	C2	241.1	88.1	101.5	40655	0.024218807
P3	P3	393.9	157	98.5	42780	0.022816139
P4	C4	384.3	165.2	88.3	45355	0.02159882
P5	P5	319	89.5	25	47710	0.020850805

4.3 Perancangan Database

Sub bab ini menjelaskan mengenai perancangan database yang akan digunakan pada implementasi. Database terdiri dari lima tabel untuk menyimpan data makanan_pokok, lauk_pauk, sayuran, buah, dan pelengkap. Perancangan database dapat dilihat pada Gambar 4.11 berikut ini:



Gambar 4. 12 Perancangan Database

Dari rancangan pada Gambar 4.12, maka masing-masing tabel memiliki atribut sebagai berikut:

Tabel 4. 5 Atribut Tabel makanan_pokok

Nama Atribut	Tipe	Keterangan
indeks_pokok	Integer	Primary key dari tabel makanan_pokok
nama_makanan	Varchar (45)	Untuk menyimpan nama makanan
Energi	Double	Untuk menyimpan energi tiap makanan
Lemak	Double	Untuk menyimpan kandungan lemak tiap 180 gr pada tiap makanan
karbohidrat	Double	Untuk menyimpan kandungan karbohidrat tiap 180 gr pada tiap makanan

protein	Double	Untuk menyimpan kandungan protein tiap 180 gr pada tiap makanan
Harga	Integer	Untuk menyimpan harga makanan

Tabel 4. 6 Atribut Tabel sumber_nabati

Nama Atribut	Tipe	Keterangan
indeks_nabati	Integer	Primary key dari tabel sumber_nabati
nama_makanan	Varchar (45)	Untuk menyimpan nama makanan
Energi	Double	Untuk menyimpan energi tiap makanan
lemak	Double	Untuk menyimpan kandungan lemak tiap 50 gr pada tiap makanan
karbohidrat	Double	Untuk menyimpan kandungan karbohidrat tiap 50 gr pada tiap makanan
protein	Double	Untuk menyimpan kandungan protein tiap 50 gr pada tiap makanan
harga	Integer	Untuk menyimpan harga makanan

Tabel 4. 7 Atribut Tabel sayuran

Nama Atribut	Tipe	Keterangan
indeks_sayuran	Integer	Primary key dari tabel sayuran
nama_makanan	Varchar (45)	Untuk menyimpan nama makanan
Energi	Double	Untuk menyimpan energi tiap makanan
lemak	Double	Untuk menyimpan kandungan lemak tiap 200 gr pada tiap makanan
karbohidrat	Double	Untuk menyimpan kandungan karbohidrat tiap 200 gr pada tiap makanan

protein	Double	Untuk menyimpan kandungan protein tiap 200 gr pada tiap makanan
harga	Integer	Untuk menyimpan harga makanan

Tabel 4. 8 Atribut Tabel sumber_hewani

Nama Atribut	Tipe	Keterangan
indeks_hewani	Integer	Primary key dari tabel sumber_hewani
nama_makanan	Varchar (45)	Untuk menyimpan nama makanan
Energi	Double	Untuk menyimpan energi tiap makanan
lemak	Double	Untuk menyimpan kandungan lemak tiap 80 gr pada tiap makanan
karbohidrat	Double	Untuk menyimpan kandungan karbohidrat tiap 80 gr pada tiap makanan
protein	Double	Untuk menyimpan kandungan protein tiap 80 gr pada tiap makanan
harga	Integer	Untuk menyimpan harga makanan

Tabel 4. 9 Atribut Tabel pelengkap

Nama Atribut	Tipe	Keterangan
indeks_snack	Integer	Primary key dari tabel pelengkap
nama_makanan	Varchar (45)	Untuk menyimpan nama makanan
Energi	Double	Untuk menyimpan energi tiap makanan
lemak	Double	Untuk menyimpan kandungan lemak tiap 150 gr pada tiap makanan
karbohidrat	Double	Untuk menyimpan kandungan karbohidrat tiap 150 gr pada tiap makanan

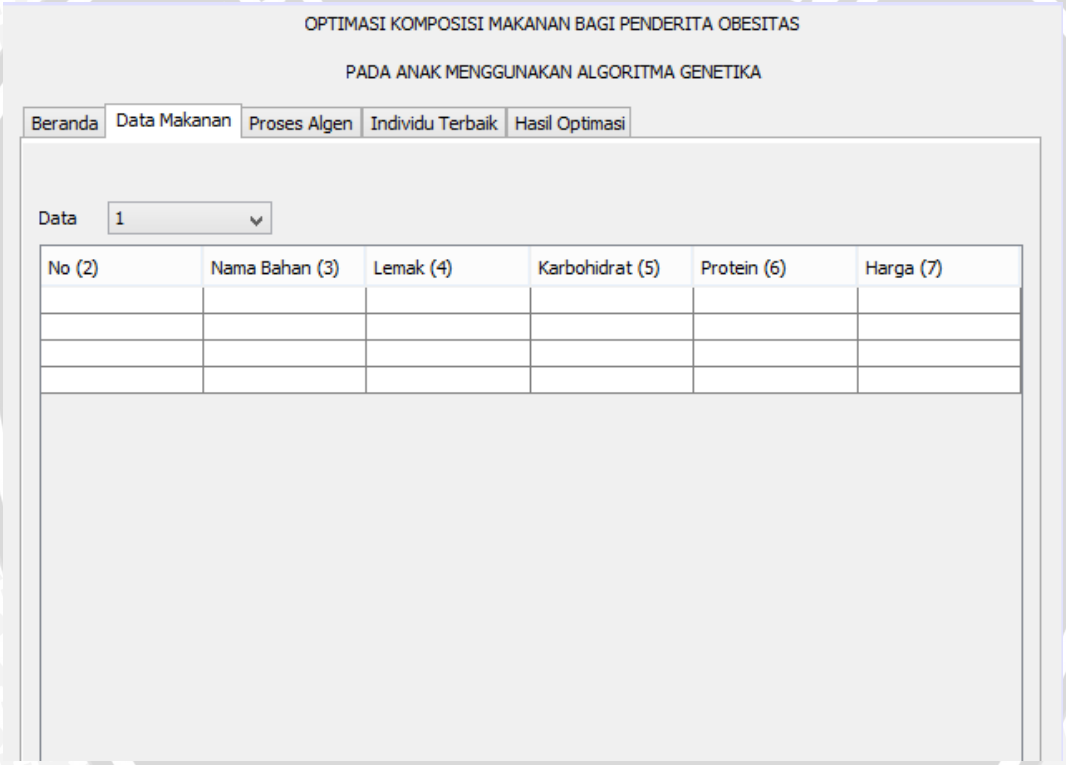
protein	Double	Untuk menyimpan kandungan protein tiap 150 gr pada tiap makanan
harga	Integer	Untuk menyimpan harga makanan

4.4 Perancangan User Interface

Perancangan *user interface* dari sistem ini sebagai berikut:

4.4.1 Data Makanan

Berisikan data makanan yang terdiri dari nama makanan, kandungan lemak, kandungan karbohidrat, kandungan protein, dan harga.



Gambar 4. 13 Perancangan Halaman Data Makanan

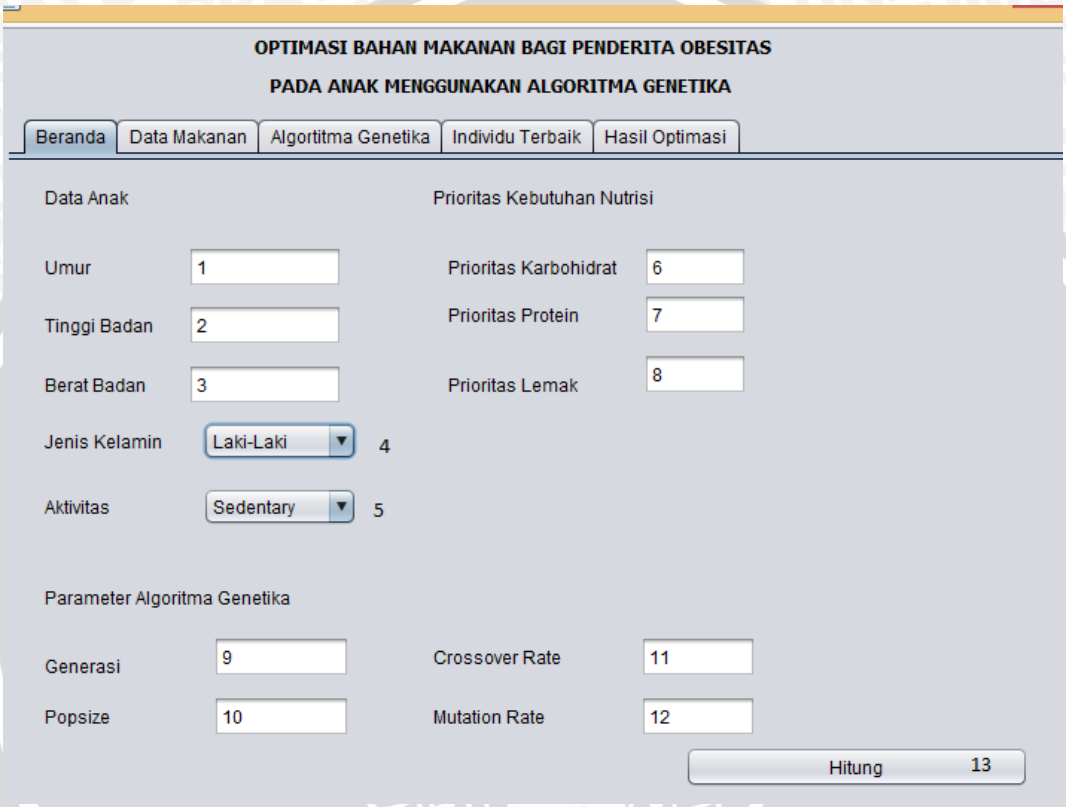
Keterangan:

1. No 1 merupakan tombol dropdown untuk memilih makanan mana yang akan ditampilkan
2. No 2 merupakan isi tabel berupa indeks makanan
3. No 3 merupakan isi tabel berupa nama makanan
4. No 4 merupakan isi tabel berupa kandungan lemak
5. No 5 merupakan isi tabel berupa kandungan karbohidrat
6. No 6 merupakan isi tabel berupa kandungan protein

7. No 7 merupakan isi tabel berupa kandungan harga

4.4.2 Beranda

Berisikan input parameter algoritma genetika dan data anak. Parameter algoritma genetika terdiri dari input *popsize*, generasi, *crossover rate*, dan *mutation rate*. Sedangkan data anak meliputi inputan tinggi badan, berat badan, jenis kelamin, jenis aktivitas, dan umur.



Gambar 4. 14 Perancangan Beranda

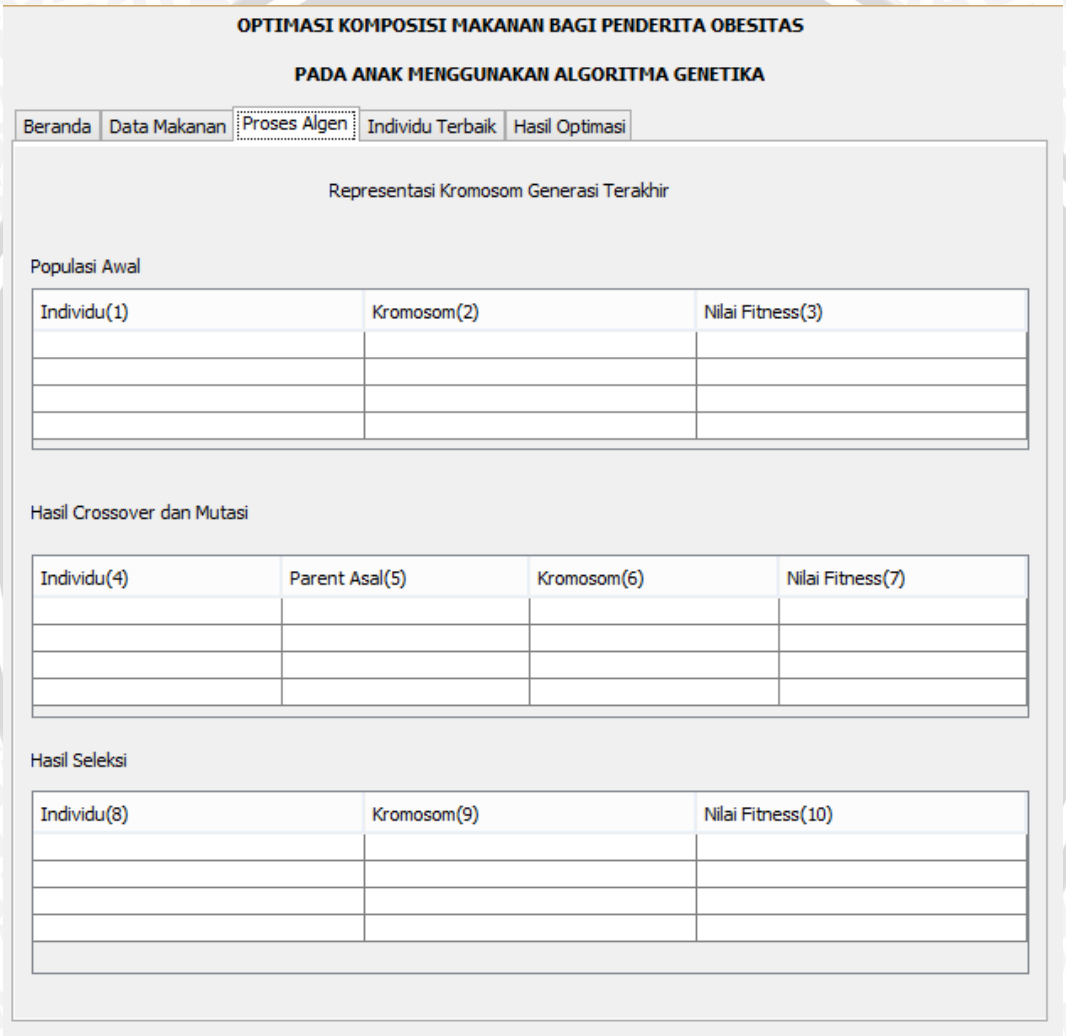
Keterangan:

1. No 1 merupakan field untuk mengisi umur anak
2. No 2 merupakan field untuk mengisi tinggi badan anak
3. No 3 merupakan field untuk mengisi berat badan anak
4. No 4 merupakan tombol dropdown untuk memilih jenis kelamin anak
5. No 5 merupakan tombol dropdown untuk memilih aktivitas anak
6. No 6 merupakan field untuk mengisi nilai alpha atau prioritas kebutuhan lemak
7. No 7 merupakan field untuk mengisi nilai alpha atau prioritas kebutuhan karbohidrat
8. No 8 merupakan field untuk mengisi nilai alpha atau prioritas kebutuhan protein
9. No 9 merupakan field untuk mengisi data jumlah generasi atau iterasi
10. No 10 merupakan field untuk mengisi jumlah individu dalam satu populasi

11. No 11 merupakan field untuk mengisi nilai crossover rate
12. No 12 merupakan field untuk mengisi nilai mutation rate
13. No 13 merupakan tombol untuk memproses sistem

4.4.3 Proses Algen

Berisikan hasil proses tahapan-tahapan algoritma genetika untuk optimasi. Dalam halaman ini akan ditampilkan populasi awal, hasil crossover, hasil mutasi, dan hasil seleksi yang telah dilakukan.



Gambar 4. 15 Perancangan Proses Algoritma Genetika

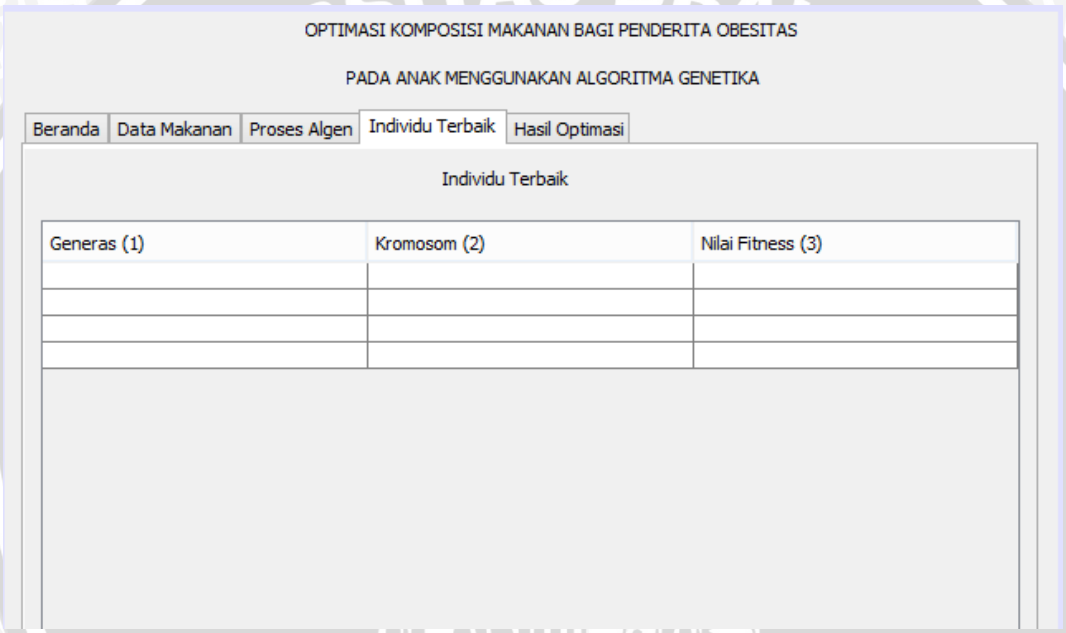
Keterangan:

1. No 1 merupakan isi tabel berupa individu untuk populasi awal
2. No 2 merupakan isi tabel berupa kromosom yang dibangkitkan pada populasi awal
3. No 3 merupakan isi tabel berupa nilai fitness yang dihasilkan

4. No 4 merupakan isi tabel berupa individu hasil reproduksi
5. No 5 merupakan isi tabel berupa parent yang melakukan reproduksi
6. No 6 merupakan isi tabel berupa kromosom yang dihasilkan oleh proses reproduksi
7. No 7 merupakan isi tabel berupa nilai fitness yang dihasilkan oleh *offspring*
8. No 8 merupakan isi tabel berupa individu yang lolos seleksi
9. No 9 merupakan isi tabel berupa kromosom yang terpilih dalam proses seleksi
10. No 10 merupakan isi tabel berupa nilai fitness yang dihasilkan

4.3.4 Detail Kromosom Terbaik

Berisikan detail kromosom terbaik yang telah terpilih.



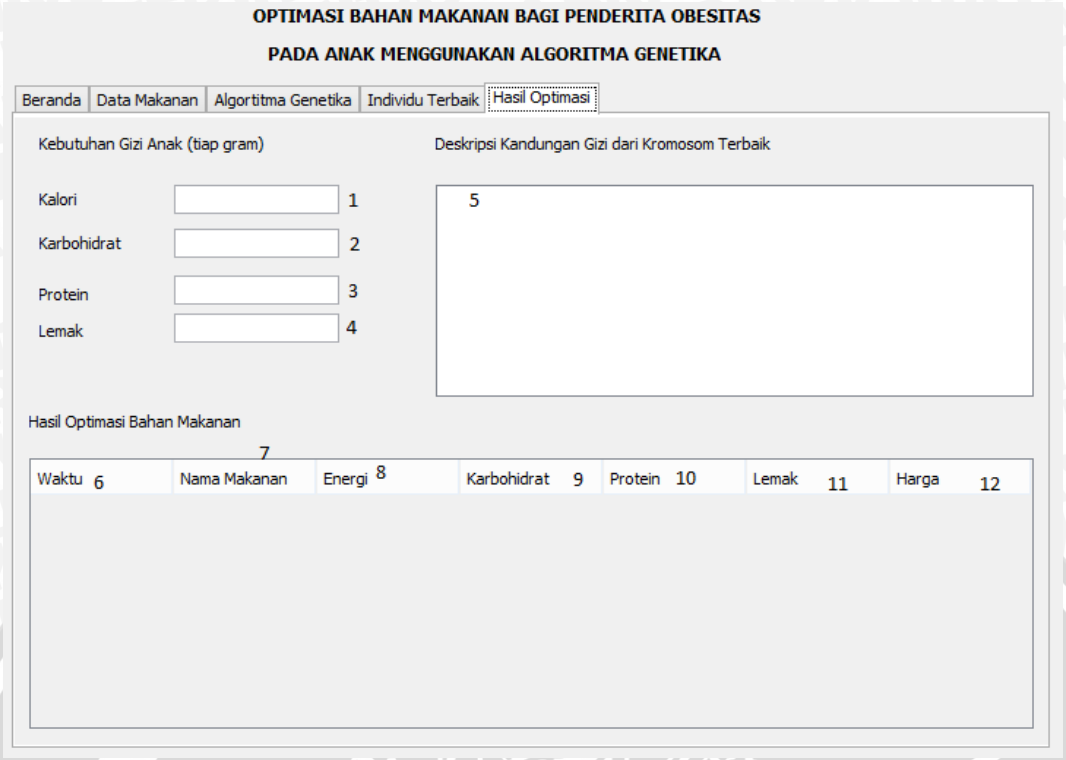
Gambar 4. 16 Perancangan Individu Terbaik

Keterangan:

1. No 1 merupakan isi tabel berupa generasi atau iterasi yang telah dilakukan dalam proses algen
2. No 2 merupakan isi tabel berupa kromosom terbaik yang dihasilkan oleh tiap generasi
3. No 3 merupakan isi tabel berupa nilai fitness yang dihasilkan

4.3.5 Hasil Optimasi

Berisikan daftar makanan yang diperlukan oleh anak penderita obesitas sesuai dengan kebutuhan nutrisi lemak, karbohidrat, dan protein.



Gambar 4. 17 Perancangan Halaman Hasil Optimasi

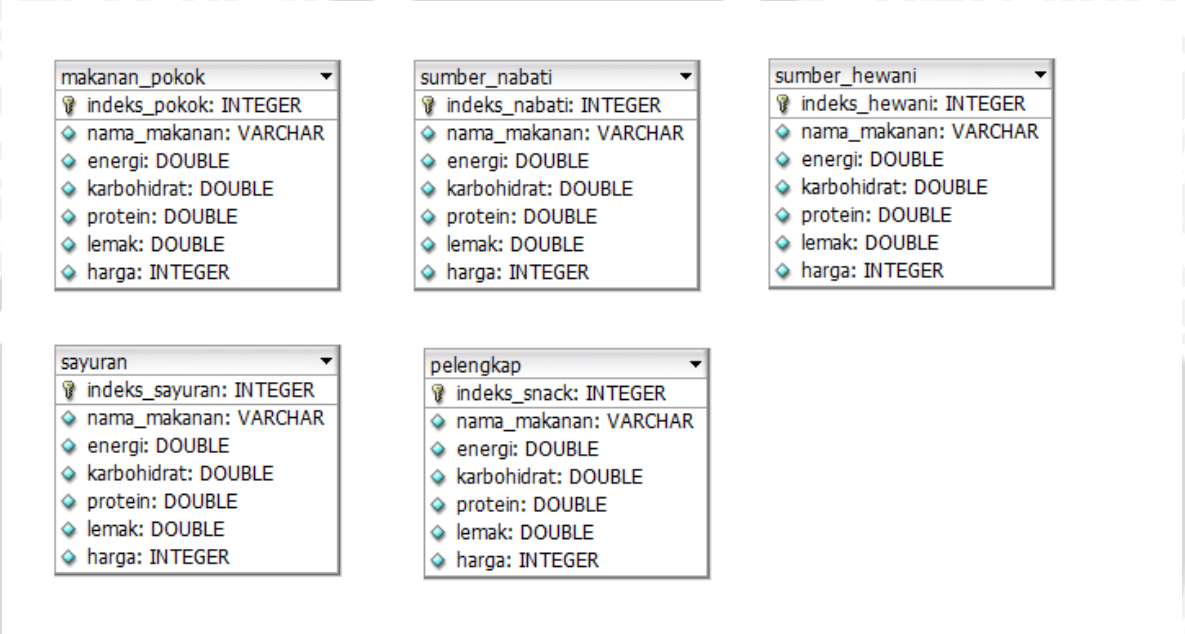
Keterangan:

1. No 1 merupakan field untuk menampilkan total kalori harian anak
2. No 2 merupakan field untuk menampilkan kebutuhan karbohidrat anak dalam satu hari
3. No 3 merupakan field untuk menampilkan kebutuhan protein anak dalam satu hari
4. No 4 merupakan field untuk menampilkan kebutuhan lemak anak dalam satu hari
5. No 5 merupakan deskripsi dari hasil komposisi makanan
6. No 6 merupakan isi tabel yang menampilkan waktu makan (pagi hari, siang hari, dan malam hari)
7. No 7 merupakan isi tabel yang menampilkan nama makanan yang terpilih untuk dikonsumsi satu hari
8. No 8 merupakan isi tabel yang menampilkan energi tiap makanan terpilih
9. No 9 merupakan isi tabel yang menampilkan kandungan karbohidrat makanan terpilih
10. No 10 merupakan isi tabel yang menampilkan kandungan protein makanan terpilih
11. No 11 merupakan isi tabel yang menampilkan kandungan lemak makanan terpilih
12. No 12 merupakan isi tabel yang menampilkan harga makanan terpilih

BAB V IMPLEMENTASI

5.1 Struktur Database

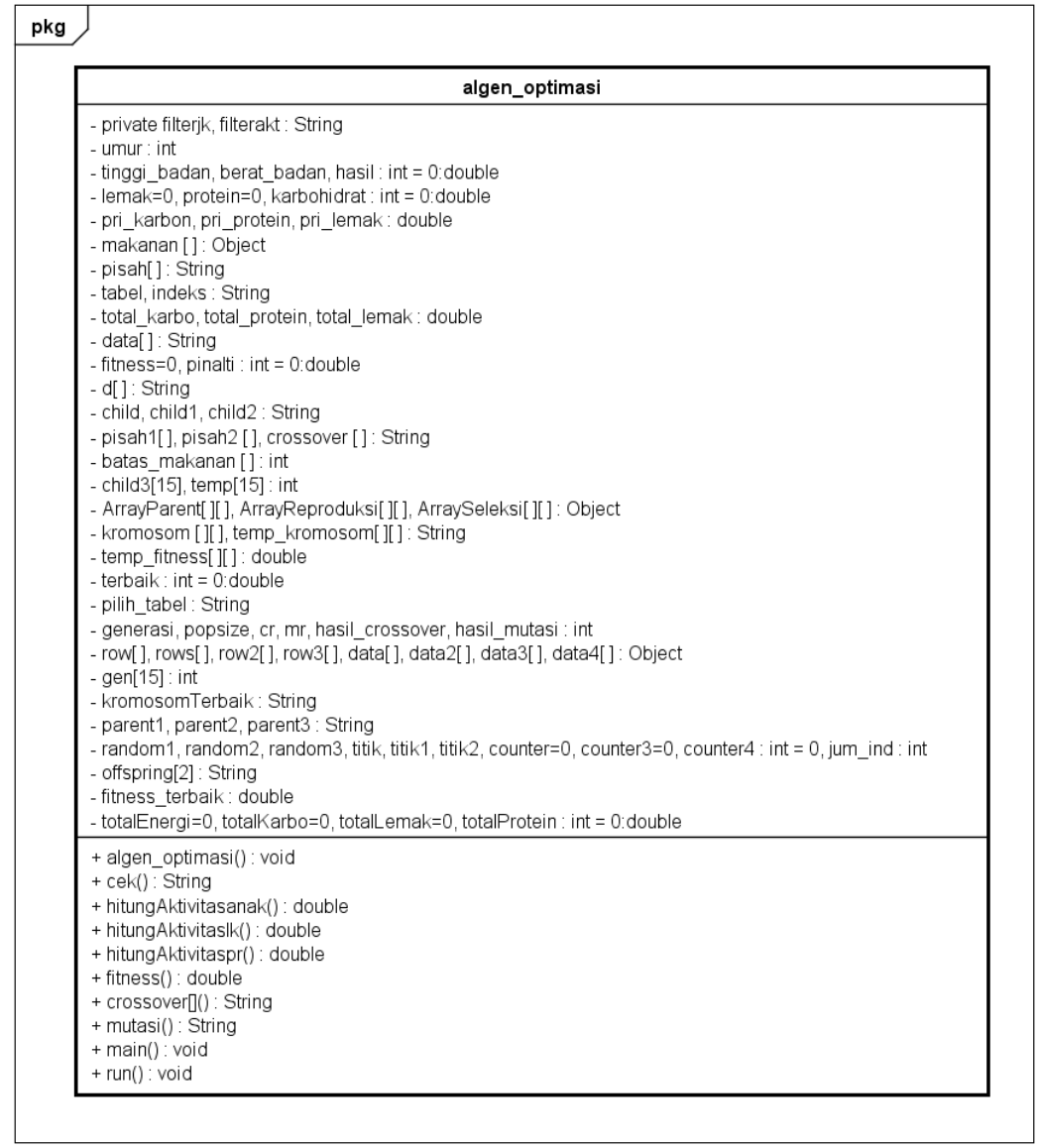
Data makanan dalam sistem disimpan dalam phpMySQL dengan total makanan sebanyak 133 dan dibagi menjadi 5 tabel yaitu tabel makanan_pokok, tabel sumber_nabati, tabel sumber_hewani, tabel sayuran, dan tabel pelengkap.



Gambar 5. 1 Struktur Database

5.2 Struktur Class

Struktur diagram *Class* dalam membangun sistem optimasi komposisi makanan bagi penderita obesitas pada anak menggunakan algoritma genetika sebagai berikut:



Gambar 5. 2 *Class Diagram*

5.3 Potongan Source Code

5.3.1 Mengambil Data Makanan dari Database

Proses mengambil data dilakukan dengan mengambil data pada database MySQL yang telah dibuat pada phpMyadmin. Database terdiri dari makanan pokok, lauk pauk, sayuran, buah, dan pelengkap disertai dengan kandungan gizi dari masing-masing makanan.

```
pilih_tabel = (String) pilih_makanan.getSelectedItem();
        tabel = null;
        indeks = null;
        Object[] row = {"Indeks", "Nama Makanan", "Energi",
"Karbohidrat", "Protein", "Lemak", "Harga"};
        model = new DefaultTabelModel(null, row);
        tabel_makanan.setModel(model);
        model.getDataVector().removeAllElements();
        model.fireTabelDataChanged();
        //memilih tabel
        if (pilih_tabel == "Makanan Pokok") {
            tabel = "makanan_pokok";
            indeks = "indeks_pokok";
        } else if (pilih_tabel == "Sumber Nabati") {
            tabel = "sumber_nabati";
            indeks = "indeks_nabati";
        } else if (pilih_tabel == "Sumber Hewani") {
            tabel = "sumber_hewani";
            indeks = "indeks_hewani";
        } else if (pilih_tabel == "Sayuran") {
            tabel = "sayuran";
            indeks = "indeks_sayuran";
        } else if (pilih_tabel == "Pelengkap") {
            tabel = "pelengkap";
            indeks = "indeks_snack";
        }
        try {
            statement = con.createStatement();
            set = statement.executeQuery("select * from " +
tabel);
            while (set.next()) {
                makanan = new Object[7];
```

```
                makanan[0] = set.getString(indeks);
                makanan[1] = set.getString("nama_makanan");
                makanan[2] = set.getString("energi");
                makanan[3] = set.getString("karbohidrat");
                makanan[4] = set.getString("protein");
                makanan[5] = set.getString("lemak");
                makanan[6] = set.getString("harga");
                model.addRow(makanan);
            }
            set.close();
        } catch (SQLException makanan) {
            System.err.println(makanan);
        }
```

Source Code 5. 1 Pengambilan Data Makanan

5.3.2 Pembangkitan Populasi Awal

Populasi awal dibangkitkan dengan merepresentasikan kromosom berdasarkan jumlah individu (*popsize*) yang telah diinputkan oleh user. Data dari hasil pembangkitan populasi awal didapatkan dari indeks masing-masing tabel pada database.

```
kromosom = new String[generasi + 2][popsize + hasil_crossover +
hasil_mutasi];
Object data[] = new Object[3];
        int gen[] = new int[15]; //panjang kromosom
        int j = 0;
        for (int ulang = 0; ulang < popsize; ulang++) {
            kromosom[0][ulang] = "";
            for (int i = 0; i < gen.length; i++) {
                if (j == 0 || j == 4) {
                    gen[i] = (int) (Math.random() * 24);
                } else if (j == 1 || j == 3) {
                    gen[i] = (int) (Math.random() * 25);
                } else {
```

```
                gen[i] = (int) (Math.random() * 30);
            }
            j++;
            if (j == 5 || j == 10) {
                j = 0;
            }
        }
        for (int i = 0; i < gen.length; i++) {
            kromosom[0][ulang] += gen[i];
            kromosom[0][ulang] += " ";
        }
        data[0] = "P" + ulang;
        data[1] = kromosom[0][ulang];
        data[2] = fitness(kromosom[0][ulang], false, 1);
    }
```

Source Code 5. 2 Pembangkitan Populasi Awal

5.3.3 Perhitungan Penalti dan Nilai Fitness

Perhitungan penalti digunakan untuk menghitung selisih apabila total kandungan pada makanan yang terpilih melebihi total kebutuhan kandungan nutrisi anak. Perhitungan nilai *fitness* digunakan untuk mengetahui nilai dari masing-masing kromosom yang dibangkitkan dengan tujuan memperoleh nilai terbaik sebagai acuan untuk pembangkitan populasi baru maupun mendapatkan komposisi makanan yang terbaik.

```
//menghitung selisih lemak
if (total_lemak < lemak) {
    selisih_lem = pri_lemak * 0;
} else if (total_lemak > lemak) {
    selisih_lem = pri_lemak * (total_lemak - lemak);
    }
//menghitung selisih karbohidrat
if (total_karbo < karbo) {
```

```
        selisih_kar = pri_karbon * 0;
} else if (total_karbo > karbo) {
        selisih_kar = pri_karbon * (total_karbo - karbo);
}
//menghitung selisih protein
if (total_protein < protein) {
        selisih_pro = pri_protein * 0;
} else if (total_protein > protein) {
        selisih_pro = pri_protein * (total_protein - protein);
}
pinalti = (selisih_kar + selisih_pro + selisih_lem);
//menghitung penalti
fitness = 1000 / (pinalti + total_harga); //menghitung nilai
fitness
```

Source Code 5. 3 Perhitungan Penalti dan Fitness

5.3.4 Proses Crossover

Proses perhitungan *crossover* menggunakan metode *single-cut-point* yaitu pemilihan dua induk secara acak dari populasi yang nantinya menghasilkan suatu *offspring* (anak) dengan cara menentukan titik potong dimana diantara titik potong tersebut dilakukan tukar silang terhadap gen.

```
//proses single point crossover
public String[] crossover(int cutpoint, String p1, String p2) {
        pisah1 = p1.split(" ");
        pisah2 = p2.split(" ");
        child1 = "";
        child2 = "";
        crossover = new String[2];

        for (int i = 0; i < cutpoint; i++) {
            child1 += pisah1[i];
            child1 += " ";
```

```
            child2 += pisah2[i];
            child2 += " ";
        }
        for (int i = cutpoint; i < pisah1.length; i++) {
            child1 += pisah2[i]; //tukar
            child1 += " ";
            child2 += pisah1[i]; //tukar
            child2 += " ";
        }
        crossover[0] = child1;
        crossover[1] = child2;
        return crossover;
    }
//proses crossover (memilih 2 parent random dan titik cutpoint
random)
for (int ulang = popsize; ulang < hasil_crossover + popsize;
ulang++) {
        random1 = (int) (Math.random() * popsize);//random
parent 1
        random2 = (int) (Math.random() * popsize);//random
parent 2
        parent1 = kromosom[counter][random1];
        parent2 = kromosom[counter][random2];
        titik = (int) (double) (Math.random() * 14); //random
titik
        if (titik == 0) {
            titik = 14;
        }
        offspring = crossover(titik, parent1, parent2);
           for (int k = 0; k < offspring.length; k++) {
             if (jum_ind < (popsize * cr) + popsize) {
                kromosom[counter][jum_ind] = offspring[k];
                data2[0] = "P" + jum_ind;
```

```
                    data2[1] = "P" + random1 + " dengan P" +
random2;
                    data2[2] = offspring[k];
                    data2[3] = fitness(kromosom[counter][jum_ind],
false, 1);
                }
            }}
```

Source Code 5. 4 Proses Crossover

5.3.5 Proses Mutasi

Proses mutasi menggunakan *reciprocal exchange mutation* dimana proses ini memilih satu induk secara acak dari populasi, kemudian memilih dua titik secara acak dalam satu kromosom dan kedua titik tersebut ditukar posisinya.

```
//proses reciprocal exchange mutation
public String mutasi(int point1, int point2, String p) {
   mutasi = "";
   int[] batas_makanan = {24, 25, 30, 25, 24, 24, 25, 30, 25,
24, 24, 25, 30, 25, 24};
   pisah = p.split(" ");
   child3 = new int[15];
   temp = new int[15];
   for (int i = 0; i < pisah.length; i++) {
      if (i < child3.length) {
         child3[i] = Integer.parseInt(pisah[i]);
         temp[i] = Integer.parseInt(pisah[i]);
      }
   }
        child3[point1] = temp[point2];//tukar titik
        child3[point2] = temp[point1];//tukar titik

    //perbaikan gen mutasi apabila ada nilai yang melebihi
batasan jumlah masing-masing makanan dalam tabel database
       for (int i = 0; i < pisah.length; i++) {
```

```
            if (child3[i] > batas_makanan[i]) {
                child3[i] = (child3[i] % batas_makanan[i]) + 1;
            }
        }
        for (int i = 0; i < pisah.length; i++) {
            if (i < child3.length) {
                mutasi += child3[i];
                mutasi += " ";
            }
        }
        return mutasi;
    }
//proses mutase untuk memilih parent secara acak dan titik tukar
secara acak
    String temp1;
    double temp;
    String parent3;
    int random3;
    int titik1, titik2;
    for (int ulang = jum_ind; ulang < jum_ind + hasil_mutasi;
ulang++) {
        random3 = (int) (Math.random() * popsize);//random
parent
        parent3 = kromosom[counter][random3];
        titik1 = (int) (Math.random() * 15);//random titik satu
        do {
            titik2 = (int) (Math.random() * 15);//random titik
dua
        } while (titik2 == titik1);
            kromosom[counter][ulang] = mutasi(titik1, titik2,
parent3);
            data2[0] = "P" + ulang;
            data2[1] = "P" + random3;
```

```
            data2[2] = kromosom[counter][ulang];
            data2[3] = fitness(kromosom[counter][ulang], false,
1);
   }
```

Source Code 5. 5 Proses Mutasi

5.3.6 Proses Seleksi

Proses seleksi menggunakan seleksi *elitism* dimana semua induk dan anak dikumpulkan dan diseleksi nilainya dari yang paling tinggi hingga paling rendah.

```
//proses seleksi
for (int ulang = 0; ulang < jum_ind + hasil_mutasi; ulang++) {
   temp_kromosom[counter][ulang] = kromosom[counter][ulang];
   temp_fitness[counter][ulang] =
fitness(kromosom[counter][ulang], false, 1);
}
//proses sorting nilai fitness dari besar ke kecil
for (int i = 1; i <= jum_ind + hasil_mutasi; i++) {
   for (int k = 0; k < jum_ind + hasil_mutasi - i; k++) {
     if (temp_fitness[counter][k] < temp_fitness[counter][k +
1]) {
         temp = temp_fitness[counter][k];
         temp1 = temp_kromosom[counter][k];
         temp_fitness[counter][k] = temp_fitness[counter][k +
1];
         temp_kromosom[counter][k] = temp_kromosom[counter][k +
1];
         temp_fitness[counter][k + 1] = temp;
         temp_kromosom[counter][k + 1] = temp1;
       }
    }
}
```

Source Code 5. 6 Proses Seleksi

5.3.7 Proses Pemilihan Individu dengan Nilai Fitness Terbaik

Proses ini digunakan untuk memilih individu terbaik dari tiap generasi yang memiliki nilai fitness paling tinggi. Hasil dari pemilihan ini digunakan untuk menampilkan kombinasi makanan yang dapat dikonsumsi oleh anak.

```
double fitness_terbaik;
if (ulang == 0) {
   data4[0] = "Generasi ke-" + counter;
   data4[1] = kromosom[counter + 1][ulang];
   data4[2] = fitness_terbaik;
   if (fitness_terbaik > terbaik) {
      terbaik = fitness_terbaik;
      kromosomTerbaik = data3[1].toString();
   }
```

Source Code 5. 7 Pengambilan Individu Terbaik tiap Generasi

BAB VI PENGUJIAN DAN ANALISA

Pada bab ini akan membahas mengenai pengujian dan analisis terhadap sistem yang telah diimplementasikan. Proses pengujian dilakukan dengan 3 pengujian, yaitu pengujian ukuran populasi, pengujian ukuran generasi, dan pengujian terhadap *crossover rate* dan *mutation rate*.

6.1 Hasil dan Analisa Pengujian Ukuran Populasi

Pengujian ukuran populasi digunakan untuk menentukan populasi yang terbaik. Pengujian ini menggunakan data 133 makanan dengan prioritas karbohidrat = 2, prioritas lemak = 10, dan prioritas protein = 2. Data anak yang digunakan adalah umur 7 tahun, tinggi 120 cm, berat badan 38 kg, jenis kelamin perempuan, dan status aktivitas adalah *active*. Parameter yang digunakan dalam pengujian adalah sebagai berikut:

a. Ukuran Populasi = 20-200
b. Generasi = 100
c. Cr = 0.6
d. Mr = 0.2

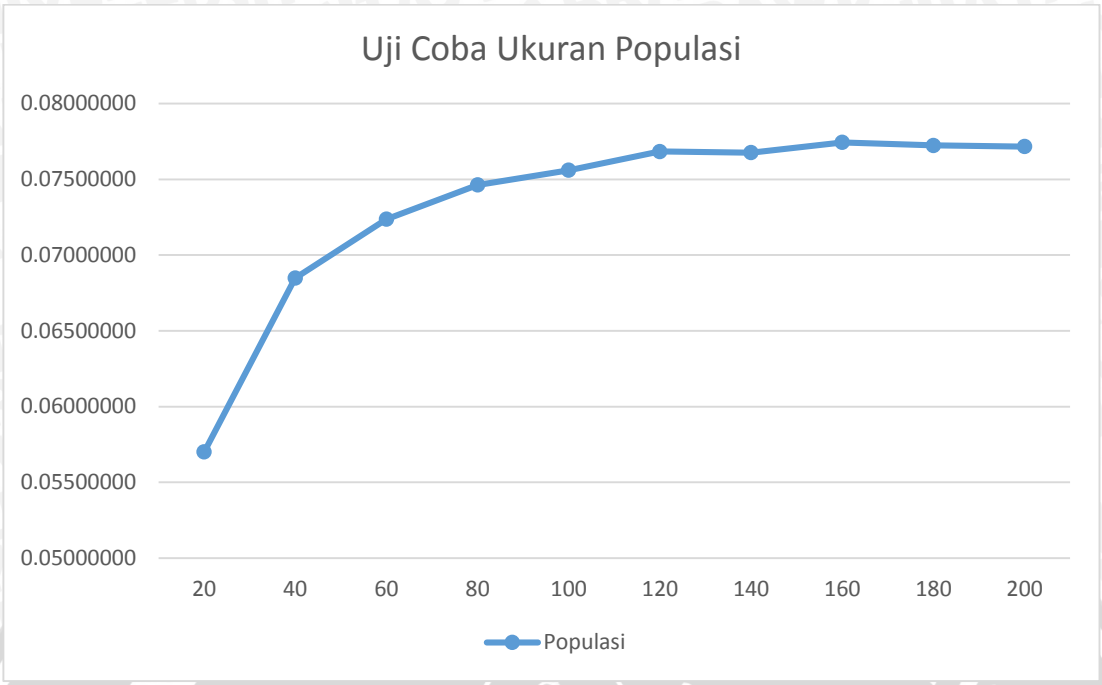
Dari pengujian sebanyak 10 kali dengan menggunakan parameter di atas, diperoleh hasil pengujian yang dapat dilihat pada Tabel 6.1 berikut ini:

Tabel 6. 1 Hasil Uji Coba Populasi

Nilai popsize	Nilai Fitness										Rata – rata *fitness*
	Pengujian ke -										
	1	2	3	4	5	6	7	8	9	10	
20	0.0491388	0.0545176	0.0495483	0.0593665	0.0555244	0.0680452	0.0581878	0.0582988	0.0560683	0.0613474	0.0570043
40	0.0658340	0.0730962	0.0709905	0.0623099	0.0717288	0.0653313	0.0712068	0.0676773	0.0701734	0.0664752	0.0684823
60	0.0735905	0.0736154	0.0681821	0.0791404	0.0741289	0.0716795	0.0752242	0.0751625	0.0624781	0.0704831	0.0723685
80	0.0722225	0.0754956	0.0760572	0.0731256	0.0666599	0.0791404	0.0782999	0.0697005	0.0764514	0.0791404	0.0746293

100	0.0760572	0.0727442	0.0791404	0.0756395	0.0749356	0.0697778	0.0781213	0.0791404	0.0781213	0.0723401	0.0756018
120	0.0791404	0.0758771	0.0791404	0.0758771	0.0791404	0.0738350	0.0791404	0.0758771	0.0769361	0.0734419	0.0768406
140	0.0782500	0.0791404	0.0716959	0.0758771	0.0791404	0.0758771	0.0779441	0.0756453	0.0758771	0.0781213	0.0767569
160	0.0791404	0.0726491	0.0791404	0.0758771	0.0742781	0.0778986	0.0791404	0.0791404	0.0791404	0.0780250	0.077443
180	0.0791404	0.0758771	0.0791404	0.0742781	0.0791404	0.0761510	0.0780250	0.0788771	0.0758771	0.0758771	0.0772384
200	0.0758771	0.0774725	0.0756395	0.0781213	0.0781213	0.0742781	0.0791404	0.0791404	0.0779411	0.0758771	0.0771609

Grafik hasil uji Pada Gambar 6.1 menyatakan bahwa, semakin besar ukuran populasi maka rata-rata *fitness* yang dihasilkan cenderung makin besar. Dapat dilihat pada populasi 20 memiliki rata-rata *fitness* terkecil yaitu sebesar 0.05700431 dan rata-rata nilai *fitness* tertinggi didapatkan pada populasi sebesar 160 dengan rata-rata nilai *fitness* sebesar 0.07744299. Namun semakin besar ukuran populasi juga belum tentu menghasilkan nilai *fitness* yang lebih baik, dapat dilihat pada ukuran populasi 160 ke 200, terjadi penurunan pada nilai *fitness* yang dihasilkan. Pada ukuran populasi 120 ke 200 juga memperlihatkan bila rata-rata nilai *fitness* yang dihasilkan tidak terlalu terjadi banyak perubahan, perubahan yang tidak terlalu besar disebabkan karena *offspring* yang dihasilkan mirip dengan induknya (Mahmudy dalam Rianawati, 2015). Pola perubahan nilai *fitness* juga terjadi pada penelitian Wahid dan Mahmudy (2015) yang menerapkan algoritma genetika pada optimasi komposisi makanan bagi penderita kolesterol. Dari hasil uji coba ukuran populasi didapatkan ukuran populasi terbaik sebesar 160 individu.



Gambar 6. 1 Uji Coba Populasi

6.2 Hasil dan Analisa Pengujian Ukuran Generasi

Pengujian ukuran generasi digunakan untuk menentukan generasi yang terbaik. Pengujian ini menggunakan data 133 makanan dengan prioritas karbohidrat = 2, prioritas lemak = 10, dan prioritas protein = 2. Data anak yang digunakan adalah umur 7 tahun, tinggi 120 cm, berat badan 38 kg, jenis kelamin perempuan, dan status aktivitas adalah *active*. Parameter yang digunakan dalam pengujian adalah sebagai berikut:

a. Ukuran Populasi = 160
b. Generasi = 25-150
c. Cr = 0.6
d. Mr = 0.2

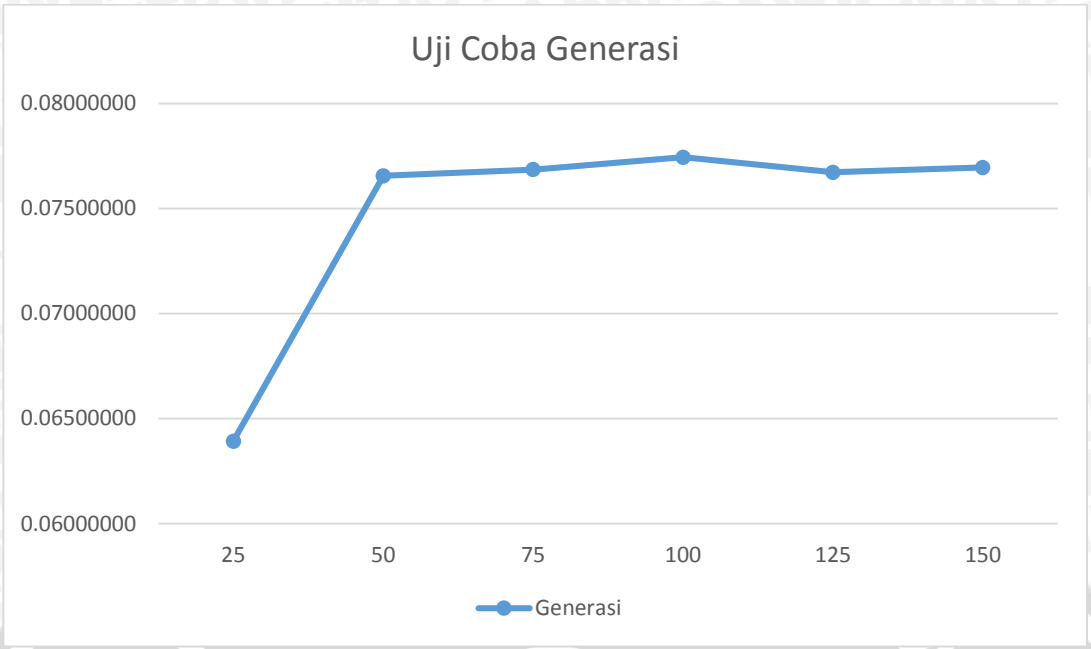
Dari pengujian sebanyak 10 kali dengan menggunakan parameter di atas, diperoleh hasil pengujian yang dapat dilihat pada Tabel 6.2 berikut ini:

Tabel 6. 2 Hasil Uji Coba Generasi

Nilai popsize	Nilai Fitness										Rata – rata *fitness*
	Pengujian ke -										
	1	2	3	4	5	6	7	8	9	10	

25	0.062 1801	0.061 7779	0.060 7352	0.065 5050	0.060 9968	0.064 2459	0.068 7181	0.062 9985	0.066 2278	0.065 8571	0.0639 2424
50	0.074 1939	0.079 1404	0.077 0297	0.079 1404	0.075 2004	0.075 2095	0.077 9411	0.077 9411	0.075 6006	0.074 1928	0.0765 5899
75	0.075 6395	0.075 7576	0.079 1404	0.076 2462	0.076 7731	0.079 1404	0.075 8771	0.079 1404	0.079 1404	0.071 7052	0.0768 5603
100	0.079 1404	0.072 6491	0.079 1404	0.075 8771	0.074 2781	0.077 8986	0.079 1404	0.079 1404	0.079 1404	0.078 025	0.0774 4299
125	0.075 8771	0.079 1404	0.079 1404	0.075 3682	0.077 5928	0.075 6395	0.075 6394	0.075 6395	0.077 4725	0.075 7576	0.0767 2674
150	0.076 6624	0.079 1404	0.079 1404	0.076 6624	0.075 8771	0.074 2781	0.074 2781	0.077 8986	0.079 1404	0.076 4912	0.0769 5691

Grafik hasil uji pada Gambar 6.2 menyatakan bahwa generasi sebesar 100 memiliki nilai rata-rata *fitness* terbaik yaitu sebesar 0.07744299. Pada generasi ke 100 sampai generasi 150 sudah menunjukkan kestabilan nilai rata-rata *fitness* dikarenakan nilai rata-rata *fitness* tidak menunjukkan perubahan yang begitu besar. Pola perubahan nilai *fitness* juga terjadi pada penelitian Wahid dan Mahmudy (2015) yang menerapkan algoritma genetika pada optimasi komposisi makanan bagi penderita kolesterol. Hasil pengujian ukuran generasi menunjukkan semakin besar generasi maka semakin besar nilai komputasinya dan belum tentu juga menghasilkan nilai yang lebih optimal (Mahmud dalam Rianawati, 2015).



Gambar 6. 2 Uji Coba Generasi

6.3 Hasil dan Analisa Pengujian Terhadap Kombinasi Cr dan Mr

Pengujian ukuran generasi digunakan untuk menentukan kombinasi nilai cr dan mr yang terbaik. Pengujian ini menggunakan data 133 makanan dengan prioritas karbohidrat = 2, prioritas lemak = 10, dan prioritas protein = 2. Data anak yang digunakan adalah umur 7 tahun, tinggi 120 cm, berat badan 38 kg, jenis kelamin perempuan, dan status aktivitas adalah *active*. Parameter yang digunakan dalam pengujian adalah sebagai berikut:

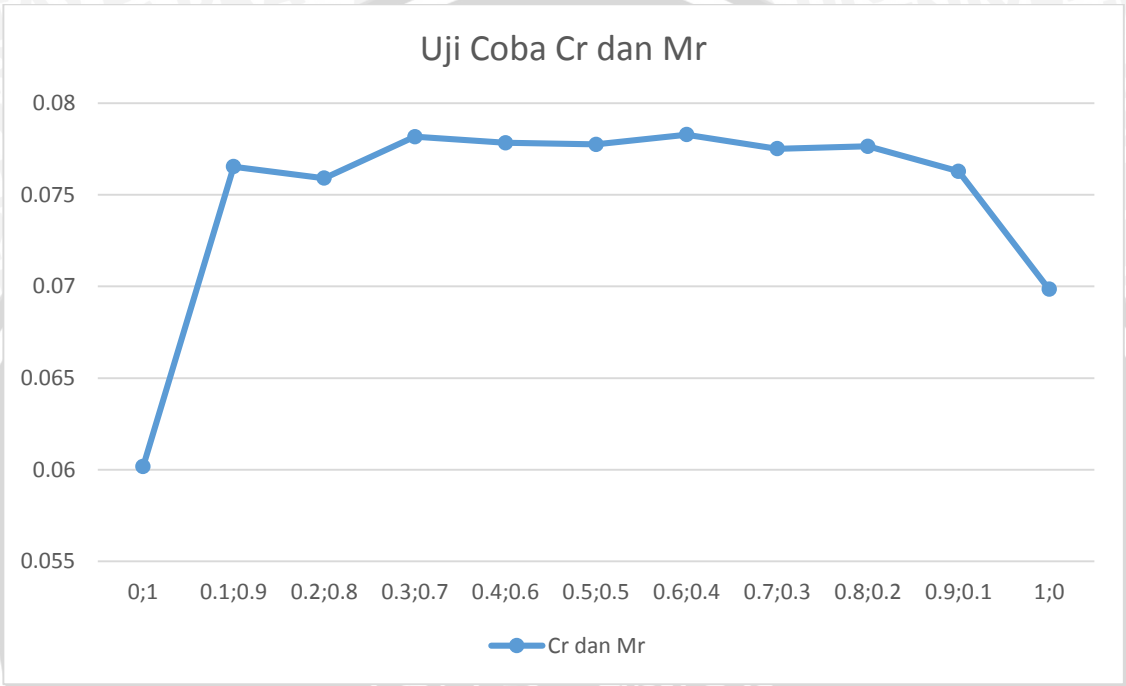
a. Ukuran Populasi = 160
b. Generasi = 100

Tabel 6. 3 Hasil Uji Coba Kombinasi Cr dan Mr

Kombi-nasi		Nilai Fitness										Rata – rata *fitness*
		Pengujian ke -										
cr	mr	1	2	3	4	5	6	7	8	9	10	
0	1	0.0599809	0.0603722	0.0633224	0.0622979	0.0619594	0.0603635	0.0547252	0.0601709	0.0610717	0.0574375	0.0601702
0.1	0.9	0.0749356	0.0791404	0.0758771	0.0742781	0.0791404	0.0746659	0.0791404	0.0758771	0.0791404	0.0730999	0.0765295
0.2	0.8	0.0696612	0.0791404	0.0742781	0.0736154	0.0758771	0.0758771	0.0791404	0.0731208	0.0791404	0.0791404	0.0758991
0.3	0.7	0.0791404	0.0791404	0.0758771	0.0758771	0.0791404	0.0791404	0.0758771	0.0791404	0.0791404	0.0791404	0.0781614
0.4	0.6	0.0791404	0.0791404	0.0758771	0.0758771	0.0791404	0.0758771	0.0791404	0.0791404	0.0758771	0.0791404	0.0778351
0.5	0.5	0.0758771	0.0791404	0.0780250	0.0738350	0.0781213	0.0791404	0.0791404	0.0791404	0.0791404	0.0758771	0.0777438
0.6	0.4	0.0771236	0.0791404	0.0780250	0.0791404	0.0791404	0.0791404	0.0791404	0.0736154	0.0791404	0.0791404	0.0782747
0.7	0.3	0.0734403	0.0791404	0.0791404	0.0791404	0.0791404	0.0791404	0.0758771	0.0791404	0.0718199	0.0791404	0.0775120
0.8	0.2	0.0791404	0.0791404	0.0791404	0.0780250	0.0756452	0.0791404	0.0761510	0.0791404	0.0757576	0.0750999	0.0776381
0.9	0.1	0.0791404	0.0791404	0.0766624	0.0767849	0.0754523	0.0781213	0.0758771	0.0722596	0.0764994	0.0728544	0.0762792
1	0	0.0690207	0.0679606	0.0711992	0.0709345	0.0759832	0.0692530	0.0702612	0.0705388	0.0690517	0.0641909	0.0698394

Grafik uji Pada Gambar 6.3 menyatakan bahwa nilai rata-rata *fitness* yang diperoleh bervariasi dikarenakan tidak adanya suatu nilai tetap yang digunakan untuk menentukan nilai cr maupun mr dalam memperoleh nilai kombinasi yang tepat (Mahmudy dalam Rianawati, 2015). Nilai cr = 0.6 dan mr = 0.4 merupakan nilai kombinasi yang terbaik dari hasil uji coba kombinasi dengan nilai rata-rata

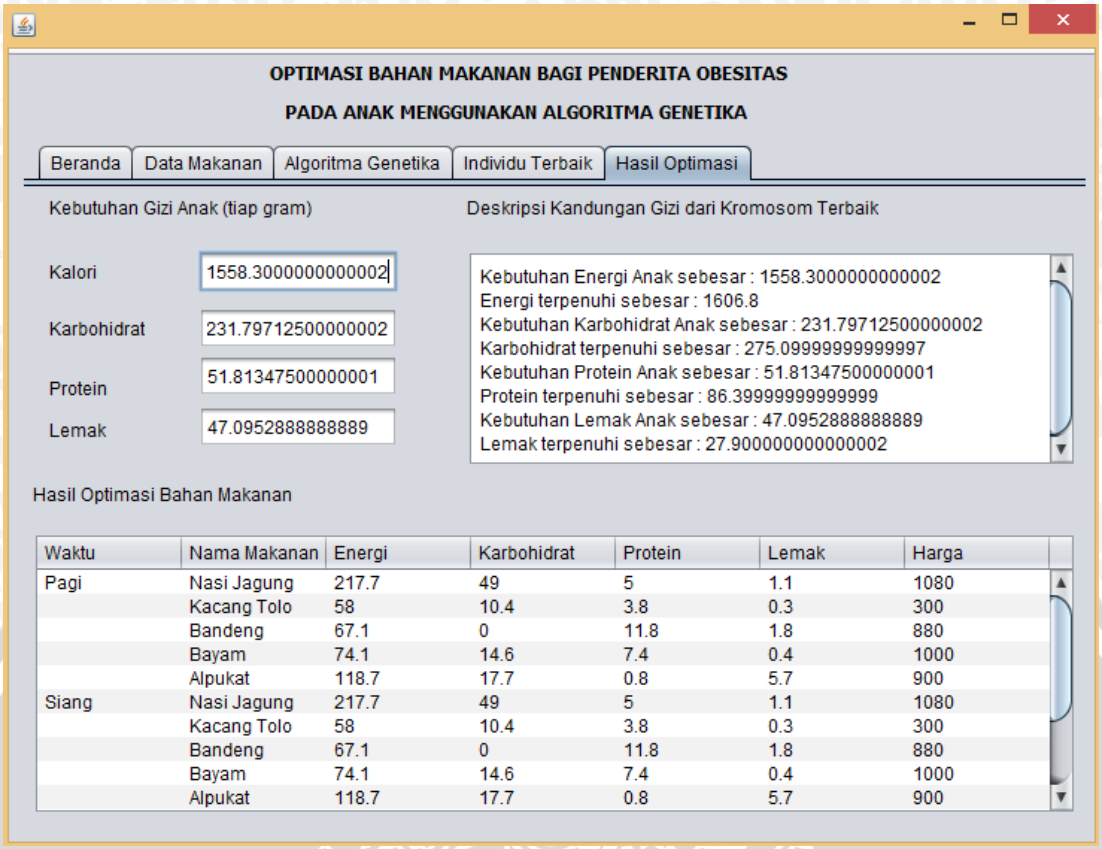
fitness sebesar 0.0782747. Menurut Mahmudy dalam Rianawati (2015), nilai cr yang terlalu rendah dan nilai mr terlalu tinggi mengakibatkan kemampuan algoritma genetika menurun dikarenakan algoritma genetika kurang efektif untuk belajar dari generasi sebelumnya. Selain itu, nilai mr yang terlalu rendah dan nilai cr yang terlalu tinggi dapat mengakibatkan konvergensi atau perubahan nilai *fitness* yang tidak terlalu besar.



Gambar 6. 3 Uji Coba Cr dan Mr

6.4 Analisa Hasil

Berdasarkan hasil pengujian yang dilakukan pada 133 data makanan dan diuji coba pada anak penderita obesitas dengan umur 7 tahun, berat badan 38 kg, tinggi badan 120 cm, jenis kelamin perempuan, dan aktivitas *active*, diperoleh ukuran populasi terbaik sebesar 160, generasi terbaik sebesar 100, dan nilai kombinasi cr dan mr terbaik sebesar 0.6 dan 0.4. Dari hasil pengujian tersebut, didapatkan hasil seperti Gambar 6.4 di bawah ini:



Gambar 6. 4 Hasil Makanan Terpilih dari Kromosom Terbaik

Pada Gambar 6.4 di atas, komposisi makanan terpilih untuk makan pagi, siang, dan malam adalah makanan pokok berupa nasi Jagung 180 gram, sumber nabati berupa kacang tolo 50 gram, sumber hewani berupa bandeng 80 gram, sayuran berupa bayam 200 gram, dan pelengkap berupa alpukat 150 gram. Makanan yang terpilih tersebut memiliki total keterpenuhan energi seperti pada Tabel 6.4:

Tabel 6. 4 Hasil Kebutuhan dan Keterpenuhan Kandungan Gizi

Kandungan Gizi	Kebutuhan Kalori Anak	Keterpenuhan Kalori Anak dari Makanan Terpilih	Penalti (Selisih Keterpenuhan Kalori dan Kebutuhan Kalori)	Komentar
Karbohidrat	231.797125 gram	275.0999 gram	43.302875	Nilai keterpenuhan karbohidrat anak melebihi kebutuhan

				karbohidrat anak
Protein	51.813475 gram	86.3999 gram	34.586525	Nilai keterpenuhan protein anak melebihi kebutuhan protein anak
Lemak	47.09528889 gram	27.9 gram	0	Nilai keterpenuhan lemak anak kurang dari kebutuhan lemak anak
Total Penali			155.7788	
Fitness			0.0791404	

Berdasarkan Tabel 6.2 tersebut diketahui bahwa masih ada nilai penalti yang terjadi pada karbohidrat dan protein dimana keterpenuhan dari makanan terpilih melebihi kebutuhan anak. Meskipun demikian, makanan terpilih memiliki nilai *fitness* yang dihasilkan baik yaitu sebesar 0.0791404 dengan total harga makanan sebesar 12480. Penalti yang terjadi pada komposisi makanan terpilih dikarenakan makanan yang terpilih tiap jenisnya memiliki bobot makanan yang sama. Misalkan pada pagi hari anak mengkonsumsi makanan pokok sebesar 180 gram dan untuk siang hari serta malam hari tetap mengkonsumsi makanan pokok sebesar 180 gram, hal inilah yang menunjukkan bahwa dalam satu hari anak mengkonsumsi makanan yang sama dan tidak ada variasi berat konsumsi dalam satu hari.

BAB VII PENUTUP

7.1 Kesimpulan

Pada penelitian skripsi berjudul "Optimasi Komposisi Makanan Bagi Penderita Obesitas Pada Anak Menggunakan Algoritma Genetika" didapatkan kesimpulan sebagai berikut:

1. Algoritma Genetika yang diterapkan dalam penelitian ini menggunakan representasi kromosom bilangan integer. Proses reproduksi yang dilakukan pada penelitian ini menggunakan metode *single point crossover*, *reciprocal exchange mutation*, dan diseleksi menggunakan seleksi *elitism*.
2. Representasi kromosom bilangan integer memiliki panjang kromosom sebesar 15 yang merepresentasikan nomor makanan.
3. Pada proses pengujian populasi, generasi, serta nilai cr dan mr ternyata ukuran dari populasi dan generasi mempengaruhi nilai *fitness* yang dihasilkan. Jika ukuran populasi dan generasi sedikit maka pencarian Algortima Genetika semakin sempit. Jika ukuran populasi dan generasi semakin besar dapat menyebabkan konvergensi dan waktu komputasi yang lama. Pada pengujian populasi didapatkan hasil optimal sebesar 160 populasi dengan nilai rata-rata *fitness* sebesar 0.07744299. Pada pengujian generasi didapatkan hasil optimal generasi sebesar 100 dengan nilai rata-rata *fitness* sebesar 0.07744299. Pada pengujian kombinasi cr dan mr didapatkan hasil optimal sebesar cr = 0.6 dan mr = 0.4 dengan nilai rata-rata *fitness* sebesar 0.0782747.
4. Hasil optimasi komposisi makanan bagi penderita obesitas pada anak berupa komposisi makanan pagi, siang, dan malam yaitu nasi jagung, kacang tolo, bandeng, bayam, dan alpukat dengan keterpenuhan karbohidrat sebesar 275.0999 gram, keterpenuhan protein sebesar 86.3999 gram, dan keterpenuhan lemak sebesar 27.9 gram berdasarkan hasil seleksi dari kromosom terbaik.

7.2 Saran

Saran yang dapat digunakan untuk penelitian lebih lanjut sebagai berikut:

1. Penelitian ini dapat dikembangkan dengan penambahan kandungan gizi lain selain karbohidrat, lemak, dan protein.
2. Penelitian ini disarankan dapat menggunakan metode lain dalam proses reproduksi dan seleksinya agar dapat memperoleh hasil yang lebih bervariasi dalam pembentukan *offspring* dan seleksinya.

3. Penelitian ini dapat dikembangkan lagi selanjutnya dengan penambahan makanan dan bobot makanan yang lebih bervariasi dalam tiap jenis makanannya.
4. Apabila penelitian ini dikembangkan lebih lanjut diharapkan adanya variasi menu yang berbeda- beda minimal satu hari.

DAFTAR PUSTAKA

Almatsier, Sunita. 2004. *Penuntun Diet (edisi baru)*. Ed. 20. Jakarta : Gramedia Pustaka Utama

Almatsier, Sunita. 2006. *Prinsip Dasar Ilmu Gizi*. Ed. 6. Jakarta : Gramedia Pustaka Utama

Angraini, Suciaty. 2008. *Faktor Risiko Obesitas Pada Anak Taman Kanak-Kanak di Kota Bogor*. Program Studi Gizi Masyarakat dan Sumberdaya Keluarga, Fakultas Pertanian, IPB.

Baskora, R. Sutardji. Woro, Oktia. 2011. *Sistem Informasi Perencanaan Pola Hidup Sehat melalui Keseimbangan Aktivitas dan Asupan Makanan*, Jurnal Media Ilmu Keolahragaan Indonesia, vol. I, ed. 2.

Carwoto. 2007. *Implementasi Algoritma Genetika untuk Optimasi Penempatan Kapasitor Shunt pada Penyulang Distribusi Tenaga Listrik*. Jurnal Teknologi Informasi DINAMIK, vol. XII, no.2.

Devi, OC. Mahmudy, WF. Setiawan, BD. 2015. *Penerapan Algoritma Genetika untuk Penjadwalan Asisten Praktikum*, DORO : Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya, vol.5, no.11.

Hardinsyah. Riyadi, Hadi. Napitupulu, Victor. 2012. *Kecukupan Energi, Protein, Lemak, dan Karbohidrat*. Departemen Gizi Masyarakat, FEMA, IPB.

Hariadi, Ali AR. 2005. *Hubungan Obesitas dengan Beberapa Faktor Risiko Penyakit Jantung Koroner di Laboratorium Klinik Prodia Makassar Tahun 2005*.

Hendarto, Aryono. Nasar, Sri S. 2002. *Aspek Praktis Nutrisi Parenteral pada Anak*. Jurnal Sari Pediatri, vol.3, no.4.

Hidayati, SN. Irawan, R. Hidayat, B. 2009. *Obesitas Pada Anak*. Surabaya: Divisi Nutrisi dan Penyakit Metabolik, Ilmu Kesehatan Anak, Fakultas Kedokteran Unair.

Kementerian Kesehatan RI. 2012. *Pedoman Pencegahan dan Penanggulangan Kegemukan dan Obesitas pada Anak Sekolah*.

Krause's Food and Nutrition Theraphy. 2004. 11th ed. International. USA : Saunders Elsevier, pp. 31

Mahmudy, Wayan Firdhaus. 2013. *Algoritma Evolusi*. Program Teknologi Informasi dan Ilmu Komputer. Universitas Brawijaya.

Panharesi, YG. Mahmudy, WF. 2015. *Optimasi Distribusi Barang dengan Algoritma Genetika*. DORO: Repository Jurnal Mahasiswa PTIIK Univeristas Brawijaya.

Pitaloka, DA. Mahmudy, WF. Sutrisno. 2014. *Penerapan Hybrid Algoritma Gentika untuk Permasalahan Vehicle Routing Problem with Time Windows (VRPTW).* DORO: Repository Jurnal Mahasiswa PTIIK Univeristas Brawijaya, vol.4, no.11.

Rianawati, A. Mahmudy, WF. 2015. *Implementasi Algoritma Genetika Untuk Optimasi Komposisi Makanan Bagi Penderita Diabetes Mellitus*, DORO: Repository Jurnal Mahasiswa PTIIK Univeristas Brawijaya, vol.5, no.14.

Sartika, RAD. 2011. *Faktor Risiko Obesitas pada Anak 5-15 Tahun di Indonesia*, MAKARA Kesehatan, vol.15, no.1, pp. 37-43.

Susanto, JA. 2011. *Penilaian Status Gizi dan Penentuan Menu Makanan*, Jurusan Sistem Informasi Sekolah Tinggi Manajemen Komputer dan Teknik Komputer, Surabaya.

Utomo, GT. Junaidi, Said. Rahayu, Setya. 2012. *Latihan Senam Aerobik untuk Menurunkan Berat Badan, Lemak, dan Kolesterol*. Journal of Sport Sciences and Fitness.

Uyun, Shofwatul. Hartati, Sri. 2011. *Penentuan Komposisi Bahan Pangan untuk Diet Penyakit Ginjal dan Saluran Kemih dengan Algoritma Genetika*, Seminar Nasional Aplikasi Teknologi Informasi, Yogyakarta, 17-18 Juni.

Wahid, N. Mahmudy, WF. 2015. *Optimasi Komposisi Makanan untuk Penderita Kolesterol Menggunakan Algoritma Genetika*. DORO: Repository Jurnal Mahasiswa PTIIK Univeristas Brawijaya, vol.5, no.15.

Wijayanti, DN. 2013. *Analisis Faktor Penyebab Obesitas dan Cara Mengatasi Obesitas Pada Remaja Putri.* Jurusan Ilmu Keolahragaan. Fakultas Ilmu Keolahragaan. Universitas Negeri Semarang. Semarang.

WHO. 2007. Tabel BMI Z-Score. WHO Reference.

WHO. 2015. Childhood Obesity: *Interim Reportof Commision on Ending Childhood Obesity*. Geneva: WHO Technical Report Series

LAMPIRAN

Lampiran 1.1 Data Anak

No.	Nama	Umur (tahun)	Jenis Kelamin	BB (kg)	TB (cm)	Aktivitas
1	Clara	7	P	38	120	Active
2	Tata	7	L	23	100	Active
3	Farros	9	L	42	130	Sedentary
4	Tasya	8	P	38	125	Sedentary
5	Mutiara	9	P	40	125	Sedentary
6	Angeline	9	P	45	131	Low Active
7	Maufal	4	L	25	107	Low Active
8	Dwi Aji	4	L	22	102	Low Active
9	Akbar	5	L	25	107	Low Active
10	Abi	4	L	25	106	Low Active
11	Putri	4	P	25	103	Low Active

Lampiran 1.2 Prosentase Kebutuhan Energi Manusia

Umur	Energi Protein	Energi Lemak	Energi Karbohidrat	Total
0-5 bulan	9.4	36.2	54.4	100
6-11 bulan	11.2	29	59.8	100
1-3 tahun	13	27.9	59.3	100
4-6 tahun	13.1	27	59.8	100
7-9 tahun	13.3	27.2	59.5	100
Laki Laki				
10-12 tahun	13	26.8	60.2	100
13-15 tahun	13.3	26.4	60.4	100
16-18 tahun	13.3	26.1	60.6	100
19-29 tahun	13.3	25.4	61.3	100
30-49 tahun	13.2	25	61.8	100
50-64 tahun	13.2	24.8	62.1	100
65-79 tahun	13.1	24.6	62.3	100
80+ tahun	13.2	24.4	62.4	100
Perempuan				
10-12 tahun	13.1	27	59.9	100
13-15 tahun	13.4	27.4	59.2	100

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

16-18 tahun	13.6	27.4	59.1	100
19-29 tahun	13.8	26.8	59.4	100
30-49 tahun	13.7	26.9	59.4	100
50-64 tahun	13.5	25.8	60.6	100
65-79 tahun	13.3	25.8	60.9	100
80+ tahun	13.2	25.3	61.5	100

Diolah dari Data Riskesdas (2010). Sumber: Hardinsyah dkk (2012)

Lampiran 1.3 Data Makanan Pokok

	Nama Bahan	Kandungan per 180 gram				Harga (Rupiah)
		Energi	Karbohidrat(gr)	Protein (gr)	Lemak (gr)	
0	Nasi	234	51.5	4.3	0.4	2070
1	Nasi Jagung	217.7	49	5	1.1	1080
2	Nasi Tim	210.8	46.3	4	0.4	2070
3	Nasi Uduk	212.5	42.3	3.8	2.7	2100
4	Bubur Ketan Hitam	273.6	54.2	2	6.1	2700
5	Bihun	685.8	164.3	0.5	0.2	2160
6	Biskuit	912.5	140.4	18.7	31.5	5130
7	Kentang	167.4	38.9	3.6	0.2	2430
8	Makaroni	635.4	127.4	21.6	3.2	2700
9	Maizena	685.8	163.3	0.5	0.2	6930
10	Tepung Beras	131.2	28.8	2.3	0.2	1800
11	Bubur	131.2	28.8	2.3	0.2	2000

12	Roti Sisir	329.5	64.8	9	3.6	4320
13	Ketan Hitam	649.6	143.1	12.1	1.1	2520
14	Beras Merah	644.5	135.4	13.3	4.7	2700
15	Bubur Pabrikan	669.4	116.6	28.1	10.8	3000
16	Ketan Putih	649.6	143.1	12.1	1.1	2500
17	Ubi Jalar Ungu	201.8	47.3	4.3	0.2	1530
18	Ubi Jalar Kuning	183.7	43.7	3.8	0.2	1530
19	Ubi Jalar Merah	185.4	43.7	4.1	0.2	1530
20	Ubi Jalar Putih	201.8	47.3	4.3	0.2	1530
21	Ubi Goreng	333	73.4	2.2	4.7	1500
22	Roti	493	93.4	15.8	5.4	4140
23	Singkonng	235.8	57.4	2	0.5	1530
24	Talas	201.8	47.3	4.3	0.2	5760

Lampiran 1.4 Data Sumber Nabati

	Nama Bahan	Kandungan per 50 gram				Harga (Rupiah)
		Energi	Karbohidrat(gr)	Protein (gr)	Lemak (gr)	
0	Kacang Hijau	58	10.4	3.8	0.3	775
1	Kecap	30	2.8	5.3	0.1	1750
2	Tahu	38	0.9	4.1	2.5	1000

3	Tempe Kedelai	74.5	6.3	9.1	2	1000
4	Kacang Merah	167.5	30.1	11.5	0.6	700
5	Kacang Tanah	207	5.9	9.4	18	500
6	Biji Kecipir	174.5	12.7	15.4	8.4	2200
7	Ampas Tahu	190	4.8	20.3	12	500
8	Kacang Panjang	17.4	4	0.9	0.2	500
9	Kacang Tolo	58	10.4	3.8	0.3	300
10	Kacang Kapri	41.9	7.8	2.7	0.1	875
11	Kacang Kedelai	207.5	15.1	18.3	10	650
12	Kemiri	138	25.9	11.2	22.2	1150
13	Kenari	328.5	6.5	7.5	33	650
14	Koro	184.5	31.4	11.8	2	825
15	Kuaci	257.5	6.9	15.3	21.05	1425
16	Lamtoro	66	11.8	4.6	0.3	1050
17	Wijen	286.4	24.9	8.9	24.9	750
18	Emping	166	32.3	7.7	1	900
19	Jamur Kuping	13.5	2.5	1.1	0.3	2250
20	Minyak Kelapa	431	0	0	50	3100
21	Minyak Kelapa Sawit	431	0	0	50	1100
22	Santan	177	7.6	1.6	16.8	1500

23	Minyak Wijen	442	0	0	50	600
24	Kedelai Hitam	207.5	15.1	18.3	10	750
25	Margarin	318	0	0	36	1250

Lampiran 1.5 Data Sayuran

	Nama Bahan	Kandungan per 200 gram				Harga (Rupiah)
		Energi	Karbohidrat(gr)	Protein (gr)	Lemak (gr)	
0	Bayam	74.1	14.6	7.4	0.4	1000
1	Bayam Merah	74.1	14.6	7.4	0.4	4300
2	Buncis	69.8	15.8	3.8	0.6	1200
3	Daun Pepaya	120	22.4	10.6	1.8	5000
4	Daun Singkong	74.1	14.6	7.4	0.4	5000
5	Gambas Oyong	40.2	8.6	1.8	0.6	1500
6	Jengkol	84.1	15.6	5.4	0.2	6400
7	Kecipir	69.8	15.8	3.8	0.6	10500
8	Kemangi	42.1	10	2.4	1.4	3000
9	Ketimun	25.8	5.6	1.4	0.2	5600
10	Kol Kembang	50.2	10.8	2.8	0.6	10500
11	Kol Merah	44	9	2	0.8	2900
12	Labu Air	40.2	8.6	1.8	0.6	1900

13	Labu Siam	40.2	8.6	1.8	0.6	2700
14	Lobak	25.8	5.6	1.4	0.2	2200
15	Melinjo	229.9	57.2	4.6	0.8	1700
16	Pare	40.2	8.6	1.8	0.6	2600
17	Rebung	25.8	4.2	2	0.4	4500
18	Sawi	30.1	4.2	4.6	0.4	1500
19	Selada	25.8	4.2	2	0.4	5000
20	Seledri	25.8	4.2	2	0.4	4200
21	Terong	55.9	13.2	1.6	0.4	1900
22	Wortel	51.6	9.6	2	0.4	2000
23	Kangkung	30.1	4.2	4.6	0.4	2500
24	Kapri Muda	69.8	15.8	3.8	0.6	3000
25	Jantung Pisang	42.1	10	2.4	1.4	6600

Lampiran 1.6 Data Sumber Hewani

	Nama Bahan	Kandungan per 80 gram				Harga (Rupiah)
		Energi	Karbohidrat(gr)	Protein (gr)	Lemak (gr)	
0	Ayam	227.9	0	21.5	15.1	2600
1	Babat	66.3	0	11.6	1.9	2080
2	Bandeng	67.1	0	11.8	1.8	880

3	Bawal	67.1	0	14.6	0.6	1040
4	Bebek	269.6	0	15.2	22.7	3600
5	Belut Air Tawar	89.7	0	17.1	1.8	2000
6	Cumi-Cumi	117.6	4	20	1.8	2720
7	Daging Kambing	215.1	0	19.9	14.4	7200
8	Daging Sapi	215.1	0	19.9	14.4	7600
9	Ikan Gabus Kering	196	1.6	42.5	1.6	8960
10	Ikan Asing Kering	244.7	0	53.1	2	6800
11	Ikan Mas	104	0	14.6	4.6	2000
12	Ikan Mujair	67.1	0	14.6	0.6	1680
13	Ikan Tongkol	88.7	0	19.2	0.8	3820
14	Ikan Kembung	89.7	0	17.1	1.8	2680
15	Kepiting	69.6	1	13.2	1	28520
16	Kerang	82.4	3.5	11.4	2.2	2200
17	Lele	67.1	0	11.8	1.8	1680
18	Belut	89.7	0	17.1	1.8	3000
19	Sidat	151.2	0	15.2	9.6	17200
20	Susu Kambing	55.3	3.6	2.9	3.3	4320
21	Susu Sapi	52.8	3.8	2.6	3.1	6000
22	Telur Ayam	124.1	0.9	10.1	8.5	1200

23	Telur Bebek	148	1.2	10.2	11	1360
24	Telur Puyuh	168	1.2	10.2	11	1440
25	Tenggiri	89.7	0	17.1	1.8	1200
26	Ikan Asin Teri	268.1	0	46.9	7.5	6800
27	Udang	63.3	0	13.4	0.7	6800
28	Ikan Kakap	67.1	0	14.6	0.6	1760
29	Hati Sapi	128.9	2.7	19.5	3.9	3200
30	Otak	127.9	0	8.9	10	3600

Lampiran 1.7 Data Makanan Pelengkap

	Nama Bahan	Kandungan per 150 gram				Harga (Rupiah)
		Energi	Karbohidrat(gr)	Protein (gr)	Lemak (gr)	
0	Agar-agar	0	0	0	0	3525
1	Alpukat	118.7	17.7	0.8	5.7	900
2	Anggur	79.6	20.7	0.2	0.2	12750
3	Apel	88.6	23	0.3	0.6	1950
4	Arbei	78.2	19.2	1	0.6	3750
5	Duku	88.6	23	0.3	0.6	1875
6	Jahe	98.9	23	3.1	0.5	1125
7	Jeruk Bali	43.4	14	1.7	0.5	3750

8	Jeruk Manis	70.6	17.7	1.3	0.2	3300
9	Jeruk Nipis	37	14	1.7	0.5	2250
10	Kesemek	97.5	25.5	0.8	0.5	1275
11	Madu	456	123.6	0.5	0	11250
12	Mangga Gadong	97.5	25.5	0.8	0.5	6975
13	Mangga Harumanis	97.5	25.5	0.8	0.5	6150
14	Mangga Muda	97.5	25.5	0.8	0.5	1500
15	Manggis	88.6	23	0.3	0.6	4425
16	Pisang Ambon	138	35.1	1.5	0.8	1125
17	Pisang Raja	138	315.1	1.5	0.8	2175
18	Rambutan	73.5	18.6	0.6	0.6	2250
19	Salak	123	31.9	0.3	0.6	3150
20	Sawo	124.4	32.1	0.5	0.9	2400
21	Semangka	48	10.8	0.9	0.6	1050
22	Sirsak	105	26.7	1	0.5	2925
23	Srikaya	118.7	30.3	1.8	0.3	3300
24	Yogurt	57	6.3	6.5	0.2	11250