



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

LANDASAN TEORI

2.1 Gizi

Ada tiga hal utama yang harus dilihat saat melakukan pengaturan gizi, yakni: jadwal makan, jumlah makanan, dan jenis makanan.

Jadwal makan umumnya dibagi menjadi lima, yakni makan pagi (sarapan), makan ringan antara pagi dan siang, makan siang, makan ringan antara siang dan malam, serta makan malam.

Pada prinsipnya semua makanan memiliki nilai positif dan negatif masing-masing. Seperti makanan *fast food* memiliki kalori yang begitu besar, sehingga untuk pengguna yang gemuk disarankan untuk mengurangi memakan makanan *fast food*. Oleh karena itu, masyarakat harus memperhatikan jenis makanan apa yang baik untuk dirinya karena bukan hanya dari makanannya, tetapi juga dari diri sendiri yang memiliki alergi terhadap jenis makanan tertentu (Almatsier, 2004).

Jumlah makanan berkaitan erat dengan jumlah kalori yang dibutuhkan tiap orang per hari. Jadi pengguna harus mengetahui jumlah gizi yang dibutuhkan oleh dirinya, sehingga bisa mengira-ngira makanan apa yang harus dimakan. Bisa dilihat jumlah gizi suatu makanan pada bungkus makanan seperti pada tabel berikut.

Tabel 2.1 Tabel Nilai Gizi pada Bungkus Makanan (Almatsier, 2006)

INFORMASI NILAI GIZI		
Takaran Saji	25 g	
JUMLAH PER SAJIAN		
Energi total	70 kkal	
Energi dari lemak	10 kkal	
%AKG*		
Lemak Total	1 g	2 %
Protein	7 g	11 %
Kabohidrat Total	8 g	3 %
Natrium	0 mg	0 %
Vitamin C		4 %
Vitamin E		6 %
Kalsium		6 %
* Persen AKG berdasarkan kebutuhan energy 2000kkal, kebutuhan energy anda mungkin lebih tinggi atau lebih rendah.		

Dalam melakukan perhitungan jumlah kalori yang diperlukan, nilai BMR (*Basal Metabolic Rate*) menjadi salah satu hal terpenting. Ada beberapa cara dalam melakukan perhitungan BMR, yakni: metode Copper, metode Krause, metode Harris Benedict, metode MIFFIN, dan rumus Dubois. Dalam dunia kesehatan, metode Harris Benedict paling sering digunakan dalam melakukan perhitungan BMR, sehingga penulis menggunakan metode Harris Benedict dalam penelitian ini (Almatsier, 2006).

Berikut beberapa langkah menghitung jumlah energi yang dibutuhkan serta kebutuhan zat gizi dalam satu hari (Arisman, 2009).

1. Mendapatkan data-data mengenai umur, jenis kelamin, berat badan, tinggi badan serta jenis pekerjaan atau aktivitas dari pasien.

2. Menghitung berat badan ideal dengan rumus :

$$BB \text{ ideal (kg)} = (TB - 100) * 90\% \quad \dots \text{Rumus 2.1}$$

BB = berat badan (kg), TB = tinggi badan (cm)

3. Selanjutnya, menghitung indeks massa tubuh (IMT) dengan rumus :

$$IMT \text{ (kg/m}^2\text{)} = BB / TB^2 \quad \dots \text{Rumus 2.2}$$

BB = berat badan (kg), TB = tinggi badan (m)

- Melihat pada tabel IMT apakah pasien kelebihan berat badan atau kekurangan.

Tabel 2.2 Tabel IMT (Indeks Massa Tubuh)

Underweight	< 18.5
Normal	18.5 – 24.9
Overweight	25 – 29.9
Obese	> 29.9

- Lalu menghitung BMR (*Basal Metabolic Rate*) menggunakan metode Harris Benedict. Rumus 2.3 digunakan untuk wanita dan Rumus 2.4 untuk pria.

$$\text{BMR (Kal/hari)} = 655 + (9.6 * \text{BB}) + (1.85 * \text{TB}) - (4.7 * \text{UM}) \quad \dots \text{Rumus 2.3}$$

$$\text{BMR (Kal/hari)} = 66.5 + (13.7 * \text{BB}) + (5 * \text{TB}) - (6.8 * \text{UM}) \quad \dots \text{Rumus 2.4}$$

BB = berat badan (kg), TB = tinggi badan (cm), UM = umur (tahun)

- Mengubah BMR dari Kal/hari menjadi Kal/jam.
- Menghitung jumlah kalori yang dibutuhkan berdasarkan aktivitas yang dilakukan.

Tabel 2.3 Pengeluaran Energi Berbagai Jenis Kegiatan Jasmani (Arisman, 2009)

Kegiatan Jasmani	Ganda BMR	
	PRIA	WANITA
Tidur	1.0	1.0
Kegiatan dasar (mandi, makan, berpakaian)	1.4	1.4
Pekerjaan ringan (pekerja kantor, ahli hukum, dokter, guru, ibu rumah tangga)	1.7	1.7
Pekerjaan sedang (pekerja industri ringan, mahasiswa, pelayan toko)	2.7	2.2
Pekerjaan berat (buruh kasar, pekerja pabrik baja, penari, atlet, pekerja bangunan)	3.8	2.8
Berdiri dengan tenang	1.4	1.5
Berjalan dengan kecepatan normal	3.2	3.4
Melakukan tugas-tugas rumah tangga	1.8 – 3.7	1.8 – 3.7
Rekreasi ringan (golf, billiard, berlayar)	2.2 – 4.4	2.1 – 4.2
Rekreasi sedang (berenang, tenis)	4.4 – 6.6	4.2 – 6.3
Rekreasi berat (sepak bola, jogging, mendayung)	6.6 ⁺	6.3

Tabel 2.4 Contoh Perhitungan Jumlah Kebutuhan Kalori per Hari (Arisman, 2009)

	Lama (jam)	Perhitungan	Total
Tidur	8	Lama * dari table di atas * BMR/jam	Lama * Perhitungan
Aktivitas dalam pekerjaan	6		(kalori)
Aktivitas luar pekerjaan :			
Olah raga	$\frac{3}{4}$		
Rumah Tangga	4		
Berdiri	2		
Berjalan	1		
Duduk	2		
Lain-lain	$\frac{1}{4}$		
Jumlah	24		

8. Jika ternyata IMT pasien termasuk *underweight*, *overweight* atau *obese* maka jumlah kalori per hari perlu ditambahkan faktor koreksi. Jika kelebihan, maka dapat dikurangi 10 – 20% dari kebutuhan kalori. Untuk kasus *overweight* dapat dikurangi 10%, sedangkan untuk kasus *obese* dapat dikurangi 20%. Jika kekurangan, maka dapat ditambahkan 10 – 20%. Untuk kasus *underweight* dapat mengambil nilai tengah, sehingga 15%.
9. Selanjutnya menghitung kebutuhan zat gizi.

$$\text{Protein (g/hari)} = 1\text{g/kg BB (berat badan)} \quad \dots \text{Rumus 2.5}$$

P/E ratio = protein per energi ratio

$$\text{P/E ratio} = (\text{Protein} \times 4) / \text{jumlah kalori per hari} \quad \dots \text{Rumus 2.6}$$

L/E ratio = lemak per energi ratio

$$\text{L/E ratio} = 20\% \text{ sampai } 30\% \text{ sehingga di ambil } 25\% \quad \dots \text{Rumus 2.7}$$

$$\text{Lemak (g/hari)} = \text{L/E ratio} * \text{jumlah kalori per hari} / 9 \quad \dots \text{Rumus 2.8}$$

K/E ratio = karbohidrat per energi ratio

$$\text{K/E ratio} = 100\% - (\text{P/E ratio} + \text{L/E ratio}) \quad \dots \text{Rumus 2.9}$$

$$\text{Karbohidrat (g/hari)} = \text{K/E ratio} * \text{jumlah kalori per hari} / 4 \quad \dots \text{Rumus 2.10}$$

10. Setelah mendapatkan jumlah protein, lemak dan karbohidrat dalam gram per hari, langkah selanjutnya mencari bahan makanan pada daftar analisis bahan makanan (Nio, 1992) yang dapat memenuhi kebutuhan zat gizi tersebut.

2.2 Algoritma Genetika

Algoritma genetika pada dasarnya adalah program komputer yang mencari suatu solusi menggunakan proses evolusi. Menurut Suyanto (2011), pada awalnya algoritma ini hanya digunakan untuk mencari parameter-parameter yang optimal. Akan tetapi, dalam perkembangannya algoritma genetika dapat digunakan untuk berbagai masalah seperti *learning*, peramalan, pemrograman otomatis dan sebagainya. Menurut Sri Kusumadewi (2003), individu yang kuat akan memiliki daya tahan yang tinggi dan kemampuan reproduksi yang lebih tinggi dibandingkan dengan yang kurang kuat. George F. Luger (2009) juga mengatakan bahwa algoritma genetika adalah algoritma yang menirukan teori Charles Darwin mengenai evolusi jerapah, yang mana jerapah yang berleher pendek (solusi buruk) akan mati dan digantikan dengan jerapah berleher panjang (solusi baik) karena jerapah berleher panjang memiliki kemampuan bertahan lebih baik dibanding jerapah berleher pendek.

Secara garis besar algoritma ini berjalan dengan membangkitkan populasi awal. Populasi yang dimaksud adalah sejumlah solusi atas suatu permasalahan. Pada setiap populasi memiliki sejumlah individu yang disebut kromosom (satu solusi). Proses populasi awal membentuk populasi baru disebut sebagai generasi. Untuk membentuk generasi baru, setiap kromosom akan memiliki nilai evaluasi yang disebut nilai *fitness*. Nilai *fitness* akan menunjukkan kualitas suatu kromosom. Generasi selanjutnya dikenal dengan istilah anak (*offspring*) dan generasi pembentuk anak dikenal dengan istilah induk (*parent*). Dalam pembentukan anak dapat digunakan operator penyilangan

(*crossover*) dan mutasi. Anak-anak terbaiklah yang akan terpilih menjadi *parent* baru untuk membentuk anak selanjutnya dalam beberapa generasi. Setelah melalui proses ini, generasi akhir akan memiliki populasi (sekumpulan solusi) terbaik yang dapat digunakan dalam menyelesaikan suatu masalah (Kusumadewi, 2003).

Terdapat enam komponen utama dalam algoritma genetika, yakni: teknik penyandian, prosedur inisialisasi, seleksi, operator genetika, elitisme dan penentuan parameter.

2.2.1 Teknik Penyandian

Suatu permasalahan harus dikonversi terlebih dahulu ke dalam bentuk individu atau kromosom dengan kode tertentu. Pengkodean dilakukan pada sekumpulan gen pada kromosom. Pada umumnya sebuah gen mewakili satu variabel. Algoritma genetika secara umum merepresentasikan gen sebagai bilangan real dengan interval $[0,R]$, dimana R adalah bilangan real positif, bilangan desimal dengan interval $[0,9]$, dan bilangan biner, sehingga gen berupa deretan nilai 0 atau 1 (Suyanto, 2007).

Penyandian yang digunakan dalam penelitian ini adalah penyandian dengan merepresentasikan gen sebagai bilangan real dengan interval $[0,R]$, dimana R merupakan bilangan real positif. Dalam satu kromosom akan terdapat 17 gen, dimana 14 gen pertama merepresentasikan 14 kategori bahan makanan, 1 merepresentasikan nilai *fitness* untuk satu kromosom, 1 merepresentasikan nilai *fitness relative*, dan 1 merepresentasikan *fitness kumulative*.

2.2.2 Prosedur Inisialisasi

Prosedur inisialisasi adalah suatu proses pembentukan populasi awal. Ukuran populasi yang dibentuk tergantung pada masalah yang akan dipecahkan dan jenis

operator genetika yang diimplementasikan. Inisialisasi kromosom yang terdapat dalam populasi awal dilakukan secara acak, namun harus tetap memperhatikan domain solusi dan kendala permasalahan (Kusumadewi, 2003).

Dalam penelitian ini, 14 gen pertama akan dimasukkan nilai acak dari 0.5 sampai 5.0, yang mana 0.5 adalah jumlah minimal suatu kategori bahan makanan digunakan dan 5.0 adalah jumlah maksimal suatu kategori bahan makanan digunakan. Pada setiap pengisian gen akan dilakukan perhitungan jumlah kebutuhan kalori, karbohidrat, lemak dan protein yang jika sudah melebihi kebutuhan akan dilakukan pembagian terhadap bilangan acak dengan nilai 2 selama tiga kali atau sudah tidak melebihi kebutuhan. Pengisian gen lainnya dilakukan dengan perhitungan sesuai 14 gen sebelumnya.

2.2.3 Fungsi Seleksi

Seleksi adalah proses menentukan individu mana saja yang akan dipilih untuk menjadi induk bagi generasi selanjutnya. Semakin baik induk yang dipilih, maka akan menghasilkan keturunan yang semakin baik pula. Individu yang baik akan dilihat dari nilai *fitness* individu tersebut. Semakin tinggi nilai *fitness* akan menunjukkan bahwa individu tersebut semakin baik. Ada banyak cara dalam seleksi induk, antara lain: *rank based fitness assignment*, *roulette wheel selection*, *stochastic universal sampling*, *local selection*, *truncation selection*, dan *tournament selection* (Kusumadewi, 2003).

Metode yang digunakan dalam penelitian ini adalah *roulette wheel selection*. Metode ini adalah metode yang menirukan permainan *roulette-wheel* dimana masing-masing individu menempati lingkaran pada *roulette* secara proporsional sesuai dengan nilai *fitness*-nya (Suyanto, 2007).

2.2.4 Operator Genetika

Operator genetika dapat dibagi menjadi dua, yakni operator penyilangan (*crossover*) dan mutasi. Penyilangan atau pindah silang merupakan operator yang melibatkan dua induk untuk menghasilkan keturunan yang baru. Pindah silang dilakukan dengan cara melakukan pertukaran gen dari dua induk secara acak. Prinsip dari pindah silang ini adalah melakukan operasi pada gen-gen yang bersesuaian dari dua induk yang menghasilkan individu baru. Proses pindah silang dilakukan pada setiap individu dengan probabilitas pindah silang yang ditentukan. Macam-macam penyilangan yang dapat digunakan dalam algoritma genetika antara lain: rekombinasi diskrit, rekombinasi menengah, rekombinasi garis yang diperluas, rekombinasi satu titik, rekombinasi banyak titik, rekombinasi seragam, dan rekombinasi dengan permutasi.

Operator penyilangan yang digunakan dalam penelitian ini adalah rekombinasi menengah. Operator ini merupakan rekombinasi yang hanya dapat digunakan untuk variabel real. Nilai variabel anak dipilih di sekitar dan antara nilai-nilai variabel induk. Anak dihasilkan menurut aturan sebagai berikut :

$$\text{anak} = \text{induk1} + \alpha (\text{induk2} - \text{induk1}) \quad \dots \text{Rumus 2.11}$$

dengan α adalah faktor skala yang dipilih secara acak pada interval $[-d, 1+d]$, biasanya $d = 0.25$. Pada penelitian ini, anak tidak boleh melebihi 5.0 atau kurang dari 0.5, sehingga apabila hasil perhitungan melebihi 5.0 atau kurang dari 0.5, maka akan dilakukan penyesuaian untuk yang melebihi akan menjadi 5.0 dan yang kekurangan akan menjadi 0.5.

Setelah mengalami proses penyilangan, dapat dilakukan mutasi. Mutasi dilakukan dengan menambahkan nilai *random* yang sangat kecil (ukuran langkah mutasi), dengan probabilitas yang rendah. Setiap individu memiliki probabilitas untuk mengalami mutasi. Jika peluang mutasi terlalu kecil, banyak gen yang mungkin berguna, tetapi

tidak pernah dievaluasi, sebaliknya jika peluang mutasi terlalu besar, maka akan mengalami gangguan acak pada keturunan yang tidak memiliki kemiripan dengan induknya (Suyanto, 2007).

2.2.5 Elitisme

Pelaksanaan seleksi yang secara acak dapat membuat suatu individu bernilai *fitness* tertinggi dapat terbuang atau mengalami kerusakan gen karena proses penyilangan dan mutasi. Untuk menjaga agar individu terbaik tidak hilang selama evolusi, maka perlu dilakukan penyimpanan individu (Suyanto, 2007).

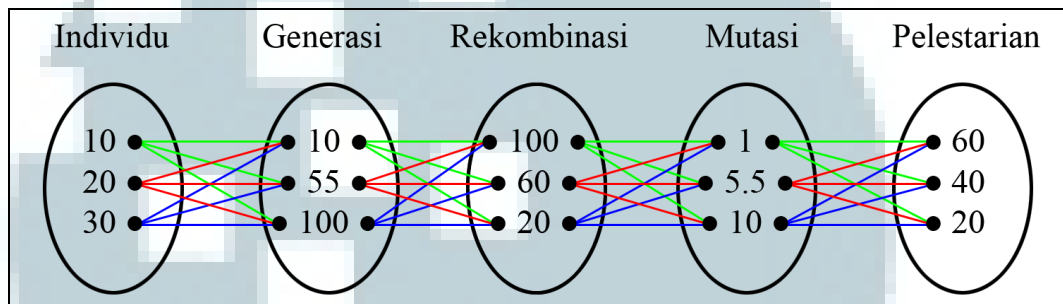
Penyimpanan individu yang dilakukan dalam penelitian ini dengan mengambil sejumlah individu terbaik dari generasi sebelumnya dan memilih secara acak individu pada generasi selanjutnya untuk digantikan dengan individu dari generasi sebelumnya.

2.2.6 Penentuan Parameter

Dalam algoritma genetika terdapat parameter kontrol, yaitu: ukuran populasi, jumlah generasi, peluang elitisme, peluang *crossover*, dan peluang mutasi. Nilai parameter ditentukan berdasarkan permasalahan yang akan dipecahkan (Kusumadewi, 2003).

Pada penelitian ini, penentuan parameter dilakukan berdasarkan data diri dari masing-masing pengguna. Setiap parameter memiliki beberapa kemungkinan seperti ukuran populasi yang terdiri dari 10, 20 dan 30. Untuk jumlah generasi terdiri dari 10, 55 dan 100. Untuk peluang elitisme terdiri dari 60%, 40% dan 20%. Untuk peluang *crossover* terdiri dari 100%, 60% dan 20%. Serta untuk peluang mutasi terdiri dari 1%, 5.5% dan 10%. Terdapat tiga anggota untuk setiap parameter, sehingga jumlah percobaan untuk setiap kasusnya terdapat 3^5 , yakni 243 percobaan.

Penulis melakukan uji coba berulang kali terhadap parameter-parameter tersebut berdasarkan data diri pengguna untuk mendapatkan parameter-parameter algoritma genetika yang tepat. Hal ini dilakukan agar hasil penelitian semakin mendekati sempurna. Pengujian dilakukan dengan melihat nilai *fitness* terbaik yang dapat dihasilkan dalam setiap proses genetika dengan parameter yang berbeda-beda.



Gambar 2.1 Jumlah Percobaan terhadap Parameter Algoritma Genetika

2.3 Android

Saat ini telepon genggam menjadi sangat populer. Pada awalnya telepon genggam hanya menjadi sarana berkomunikasi dengan kelebihan mudah dibawa kemana-mana, tetapi yang terjadi sekarang adalah telepon genggam menjadi alat yang sangat penting. Seperti yang dikatakan Mulyadi (2010) bahwa “Kebanyakan orang akan panik ketika ketinggalan telepon genggam daripada ketinggalan dompet”.

Seiring berkembangnya zaman, alat komunikasi mini ini berkembang sehingga memiliki fungsi tidak hanya telepon dan pesan singkat melainkan dapat mendengarkan musik, mengambil foto, mendengarkan radio, menonton acara TV, mengakses internet, bermain permainan, membaca dokumen, dan sebagainya (Mulyadi, 2010).

Pada saat ini banyak *platform* telepon pintar berdasarkan sistem operasi dan spesifikasi perangkat keras yang digunakan, misalnya Symbian, Blackberry OS,

Windows Phone, iOS, dan Android. Android adalah salah satu sistem operasi untuk telepon pintar yang berbasis Linux. Kehadiran robot hijau ini di masyarakat telah memiliki daya tarik tersendiri, sehingga banyak penggemar telepon pintar menggunakan Android.

Android merupakan *platform* masa depan. Android dipuji sebagai *platform* bergerak pertama yang lengkap, terbuka, dan bebas. OS ini merupakan sistem informasi yang aman dan banyak menyediakan peralatan guna membangun perangkat lunak dan memungkinkan peluang pengembangan aplikasi. *Platform* ini disediakan melalui lisensi *open source*, sehingga para pengembang dapat mengembangkan aplikasi secara bebas tanpa membayar royalti (Safaat, 2011).

2.4 Metode Penelitian

Survei adalah metode pengumpulan data dengan memberikan pertanyaan kepada beberapa orang yang berhubungan dengan penelitian. Menurut Gay dan Diehl (1992) dalam menentukan jumlah sampel sebaiknya sebanyak-banyaknya karena dengan ukuran sampel yang semakin besar, maka akan semakin merepresentasikan populasi penelitian. Akan tetapi jumlah sampel yang dibutuhkan untuk setiap penelitian berbeda berikut penjelasannya.

1. Penelitian deskriptif, sampel minimumnya adalah 10% dari populasi.
2. Penelitian korelasional, sampel minimumnya adalah 30 subjek.
3. Penelitian kausal perbandingan, sampel sebanyak 30 subjek per kelompok.
4. Penelitian eksperimental, sampel minimumnya adalah 15 subjek per kelompok.

Penelitian deskriptif adalah metode penelitian yang berusaha menggambarkan objek atau subjek yang diteliti sesuai dengan apa adanya. Penelitian korelasional adalah metode penelitian yang dilakukan untuk membandingkan persamaan atau perbedaan

dua atau lebih fakta berdasarkan suatu kerangka pemikiran. Penelitian kausal perbandingan adalah metode penelitian yang digunakan untuk mengetahui apakah antara dua variabel ada perbedaan dalam suatu aspek penelitian. Penelitian eksperimental adalah metode penelitian yang mencari sebab akibat antara beberapa faktor yang sengaja ditimbulkan oleh peneliti dengan mengeliminasi faktor-faktor yang mengganggu.

Penelitian ini adalah penelitian eksperimental yang dibagi berdasarkan jenis kelamin sehingga jumlah sampel minimumnya 15 untuk kelompok pria dan 15 untuk kelompok wanita.



UMN