

## BAB 4 PERANCANGAN SISTEM

### 4.1 Deskripsi Umum Sistem

Optimasi gizi pada bahan makanan balita merupakan sistem yang dibangun untuk mempermudah pengguna dalam menyajikan komposisi bahan makanan yang seimbang untuk balita. Sistem ini dibangun dengan *inputan* dari pengguna berupa berat badan, dan umur. Hasil akhir dari sistem berupa rekomendasi bahan makanan untuk balita yang tersedia pada waktu pagi, siang dan malam dalam jumlah hari sesuai *inputan* pengguna.

### 4.2 Deskripsi Data

Data yang digunakan untuk penelitian ini yaitu data perkembangan balita pada bulan Desember 2016 yang diperoleh dari Posyandu Gladiol 34 Desa Wringinagung Kab Jember. Data selanjutnya yaitu daftar bahan makanan yang mengandung karbohidrat, protein, lemak, sayur dan buah disertai anjuran porsi sehari dalam ukuran rumah tangga. Daftar bahan makanan, berat bahan makanan dan harga bahan makanan dijelaskan pada lampiran D. Harga bahan makanan berdasarkan harga pasar di kota Jember pada bulan Maret 2017 . Selanjutnya yaitu data anjuran porsi yang digunakan untuk menentukan porsi makan setiap individu sesuai dengan sumber bahan makanan. 1 porsi makan sama dengan berat makanan yang ada pada lampiran D data makanan. Tabel 4.1. merupakan daftar anjuran porsi makanan berdasarkan kelompok umur.

**Tabel 4.1 Daftar Anjuran Porsi Makan**

Tipe	Umur	KH	PH	PN	Lemak	Buah	Sayur
1	1 – 3,5 tahun	3 porsi	1 porsi	1 porsi	3 porsi	3 porsi	1,5 porsi
2	>3,5 – 5 tahun	4 porsi	2 porsi	2 porsi	4 porsi	3 porsi	2 porsi

Sumber: Pedoman Gizi Seimbang, Kementerian Kesehatan RI (2014)

#### Keterangan :

KH : Karbohidrat

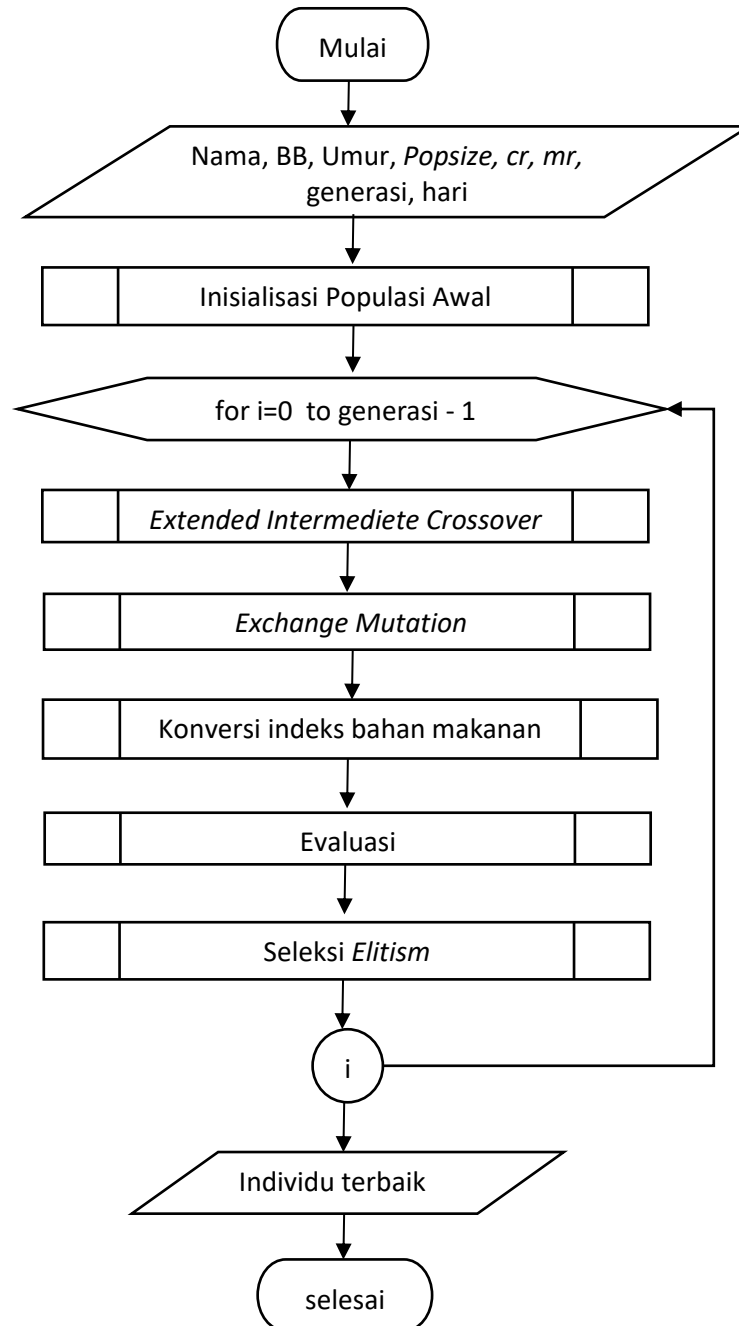
PH : Protein Hewani

PN : Protein Nabati

Anjuran porsi tersebut digunakan untuk sekali waktu makan pagi, siang, dan malam.

### 4.3 Siklus Penyelesaian menggunakan Algoritme Genetika

Pada Sub bab ini membahas tentang proses penyelesaian optimasi gizi pada bahan makanan balita menggunakan algoritme genetika. Tahapan penyelesaian menggunakan algoritme genetika ditunjukkan pada Gambar 4.1



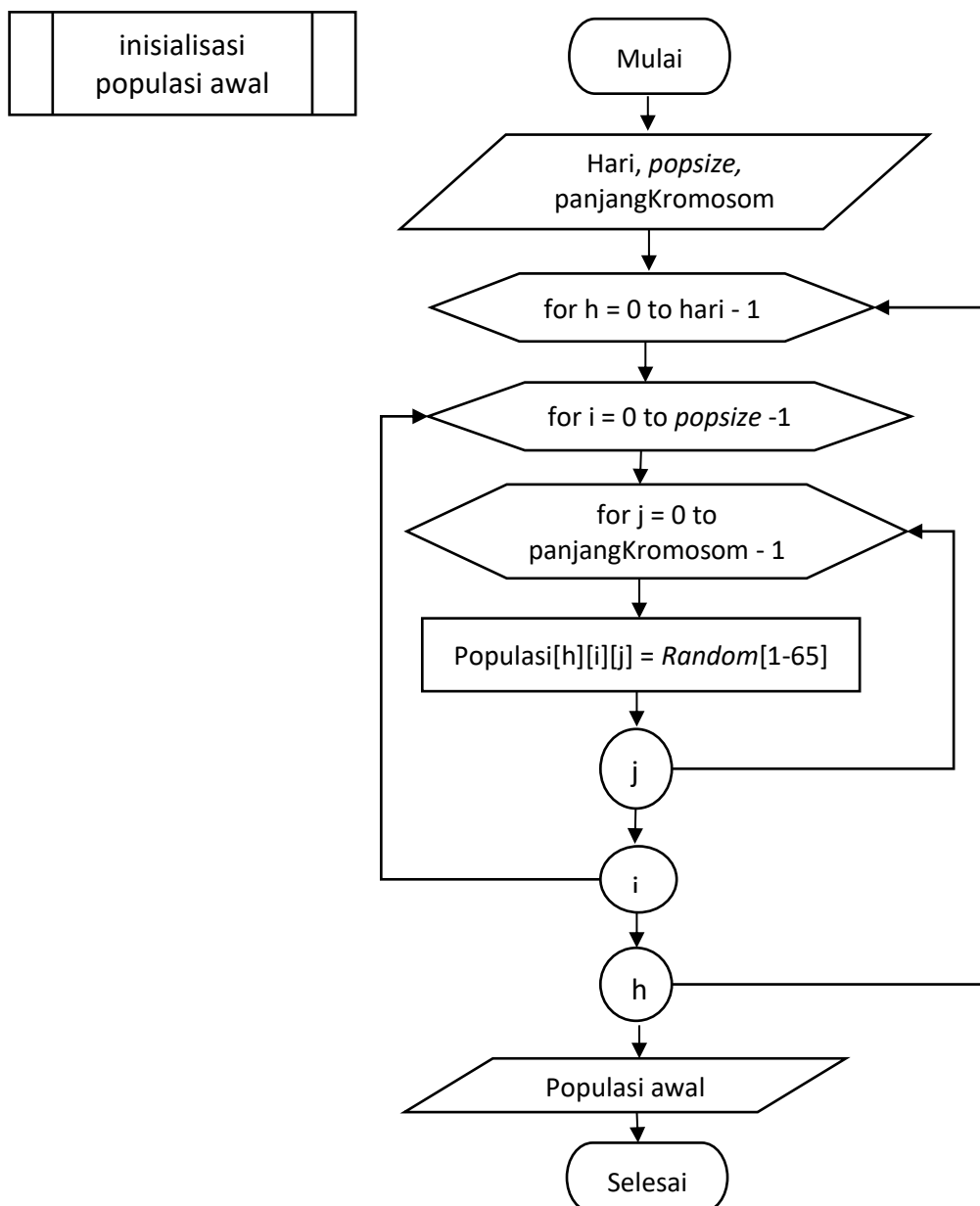
**Gambar 4.1 Diagram Alir Proses Algoritme Genetika**

Tahapan-tahapan penyelesaian optimasi pada bahan makanan balita menggunakan algoritme genetika berdasarkan Gambar 4.1 sebagai berikut:

1. *Input* berupa parameter-parameter yang diperlukan untuk optimasi gizi balita. Parameter tersebut antara lain nama balita, berat badan(BB), umur balita, jumlah populasi (*Popsi*), *crossover rate* (*cr*), *mutation rate* (*mr*), dan jumlah generasi.
2. Proses Inisialisasi populasi awal sesuai jumlah populasi (*popsi*) yang sudah ditentukan.
3. Proses Reproduksi yang pertama yaitu *crossover*, menggunakan metode *extended intermediate crossover*.

4. Metode mutasi yang digunakan yaitu *exchange mutation*. Proses mutasi dilakukan dengan mengambil salah satu *parent* secara acak dan menukar nilai gen yang dipilih secara acak.
5. Konversi gen ke indeks makanan untuk menentukan jenis bahan makanan pada setiap indeks gen.
6. Proses evaluasi dilakukan untuk mengetahui nilai *fitness* pada masing-masing individu.
7. Proses seleksi dilakukan dengan mengambil beberapa individu yang memiliki nilai *fitness* terbaik (terbesar) sebanyak jumlah populasi (*popsi*).
8. Proses berulang sampai memenuhi jumlah generasi
9. Hasil akhir berupa individu terbaik.

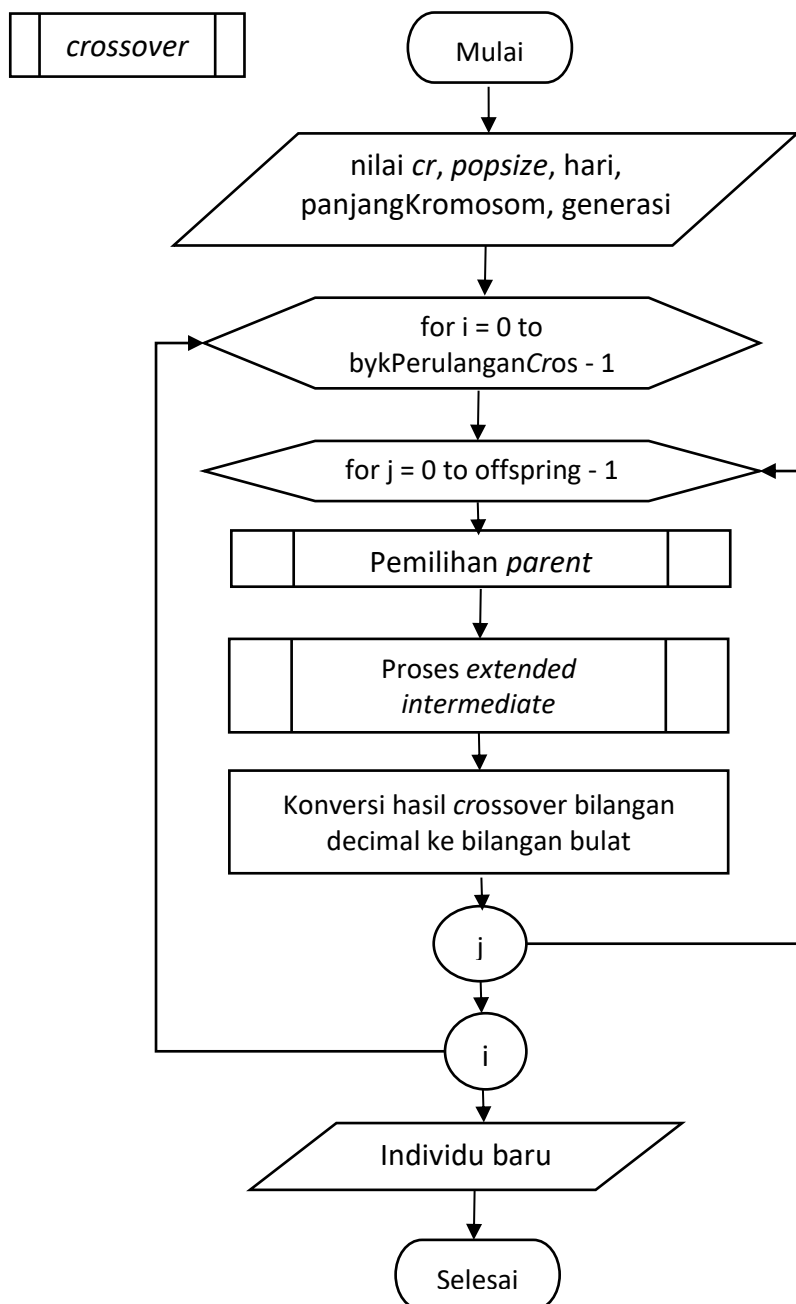
#### 4.3.1. Inisialisasi Populasi Awal



Gambar 4.2 Diagram Alir Inisialisasi Populasi Awal

Berdasarkan Gambar 4.2 proses inialisasi populasi awal digunakann untuk membentuk populasi awal. Parameter yang diperlukan untuk proses inialisasi populasi awal yaitu hari, jumlah populasi dan panjang kromosom. Dalam satu hari panjang kromosom sudah ditentukan yaitu 18 gen, sedangkan jumlah hari dan jumlah populasi (*popsize*) yang ditentukan sesuai dengan masukan pengguna. Representasi kromosom yang digunakan yaitu permutasi. Nilai indeks gen didapatkan secara acak dengan batasan nilai 1 sampai dengan 65. Proses pembentukan kromosom sesuai parameter dan hasil akhir berupa populasi awal.

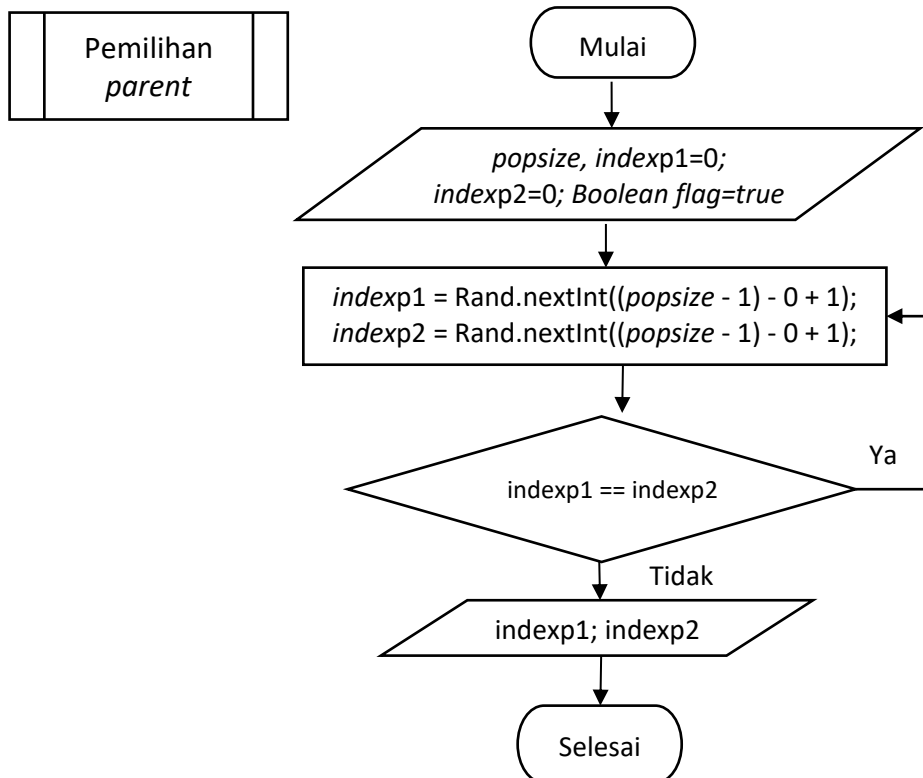
### 4.3.2 Crossover



Gambar 4.3 Diagram Alir Proses Crossover

Langkah pertama untuk melakukan reproduksi yaitu mencari nilai *crossover*. Gambar 4.3 diagram alir proses *crossover* menggunakan metode *crossover extended intermediate*, langkah pertama menentukan banyakperulangan dalam satu kali proses *crossover* yaitu 2 *child*, sehingga jika jumlah *offspring* bernilai 3 (ganjil) maka banyak perulangan *crossover* sebanyak 2 kali dengan jumlah *child* 3. Jumlah *offspring* didapatkan dari hasil perkalian *cr* dikalikan *popsiz*e. Langkah selanjutnya menentukan 2 *parent* secara acak, kemudian proses *extended intermediate* dengan menentukan nilai *a* secara acak dan menghitung nilai *child* sesuai Persamaan *extended intermediate crossover*, hasil akhir masing-masing gen bernilai bilangan decimal sehingga harus dikonversi dalam bentuk bilangan bulat dengan cara hasil *crossover* dikurangi nilai batas atas jumlah data pada bahan makanan masing-masing kandungan gizi. Kemudian hasil konversi disimpan dalam *array* dan menghasilkan individu baru. Terdapat 2 subbab proses *crossover* pada Gambar 4.3 diagram alir proses *crossover*. Subbab tersebut antara lain pemilihan *parent*, dan proses *extended intermediate*. Berikut penjelasan kedua subbab:

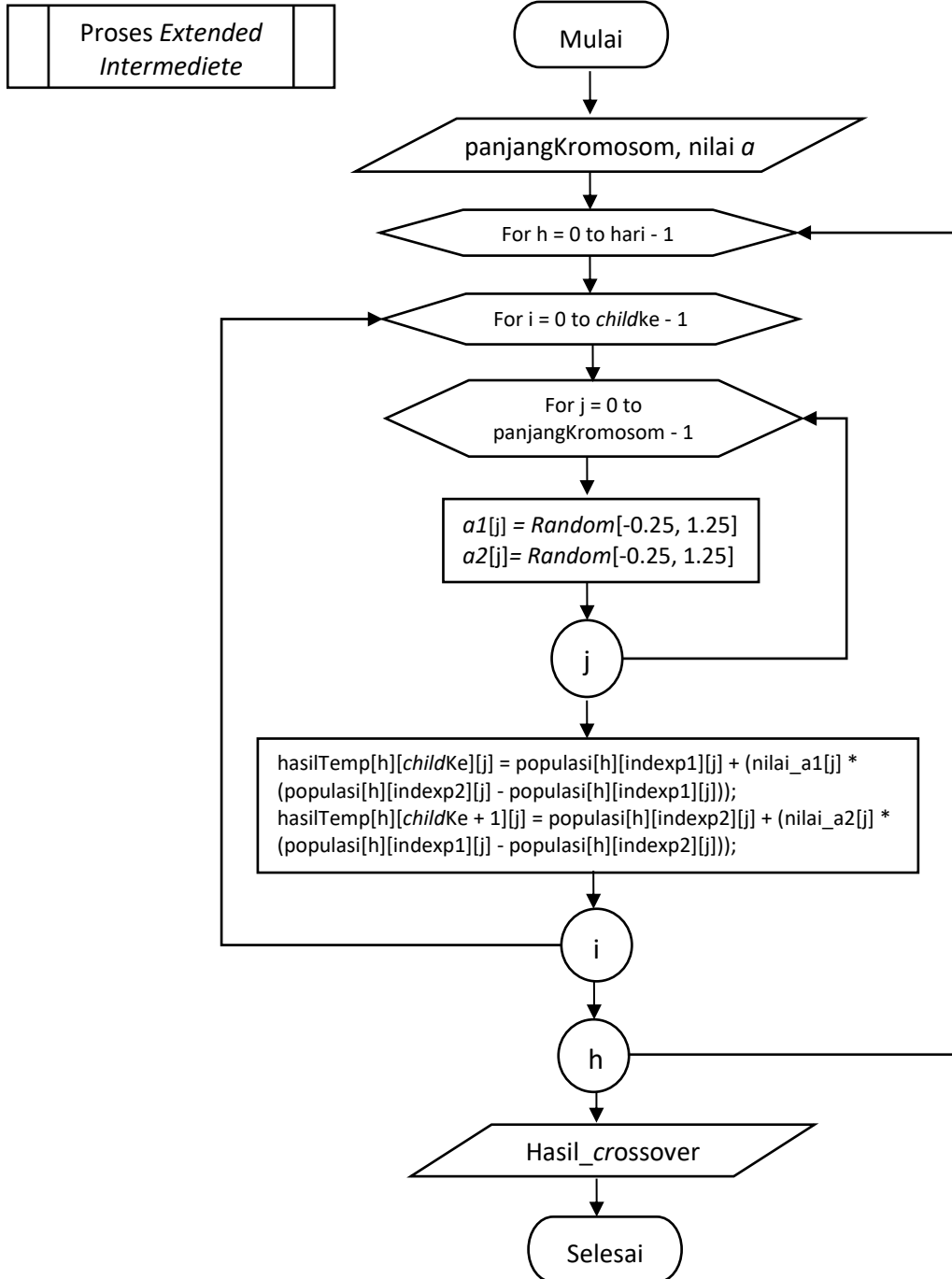
#### 4.3.2.1 Pemilihan Parent



**Gambar 4.4 Diagram Alir Pemilihan Parent**

Pada Gambar 4.4, dilakukan pemilihan 2 *parent* secara acak. *Parent* pada indeks1 dipilih secara acak kemudian *parent* pada indeks2 dipilih secara acak dengan batas *random* sesuai jumlah *popsiz*e yang sudah ditentukan. Hasil akhir berupa *parent* 1 dan *parent* 2 yang akan digunakan untuk proses *extended intermediate* pada masing-masing gen.

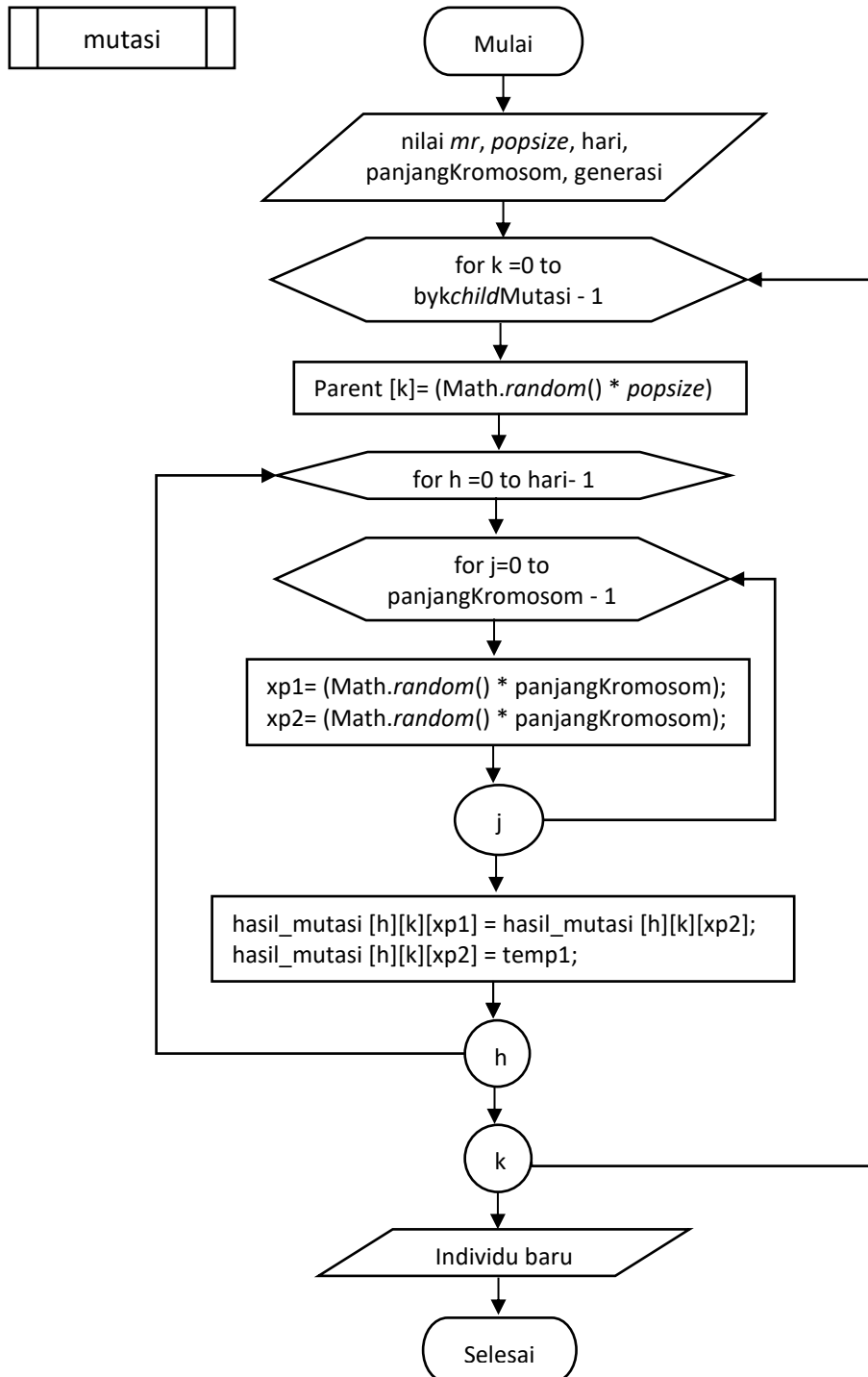
### 4.3.2.2 Proses *Extended Intermediete*



**Gambar 4.5 Diagram Alir Proses *Extended Intermediete***

Gambar 4.5 menjelaskan diagram alir proses *extended intermediete* yang digunakan untuk menghitung nilai *crossover*. Langkah pertama yaitu memilih nilai *a* secara *random* dengan *range* nilai yang sudah ditetapkan yaitu batas *minimal* (-0,25) dan batas *maksimal* (1,25). Nilai *a* dipilih secara *random* pada masing-masing gen. Langkah selanjutnya menghitung nilai *child* sesuai dengan rumus Persamaan yang sudah ditentukan. Hasil *crossover* disimpan dalam *array hasil\_crossover*.

### 4.3.3 Mutasi

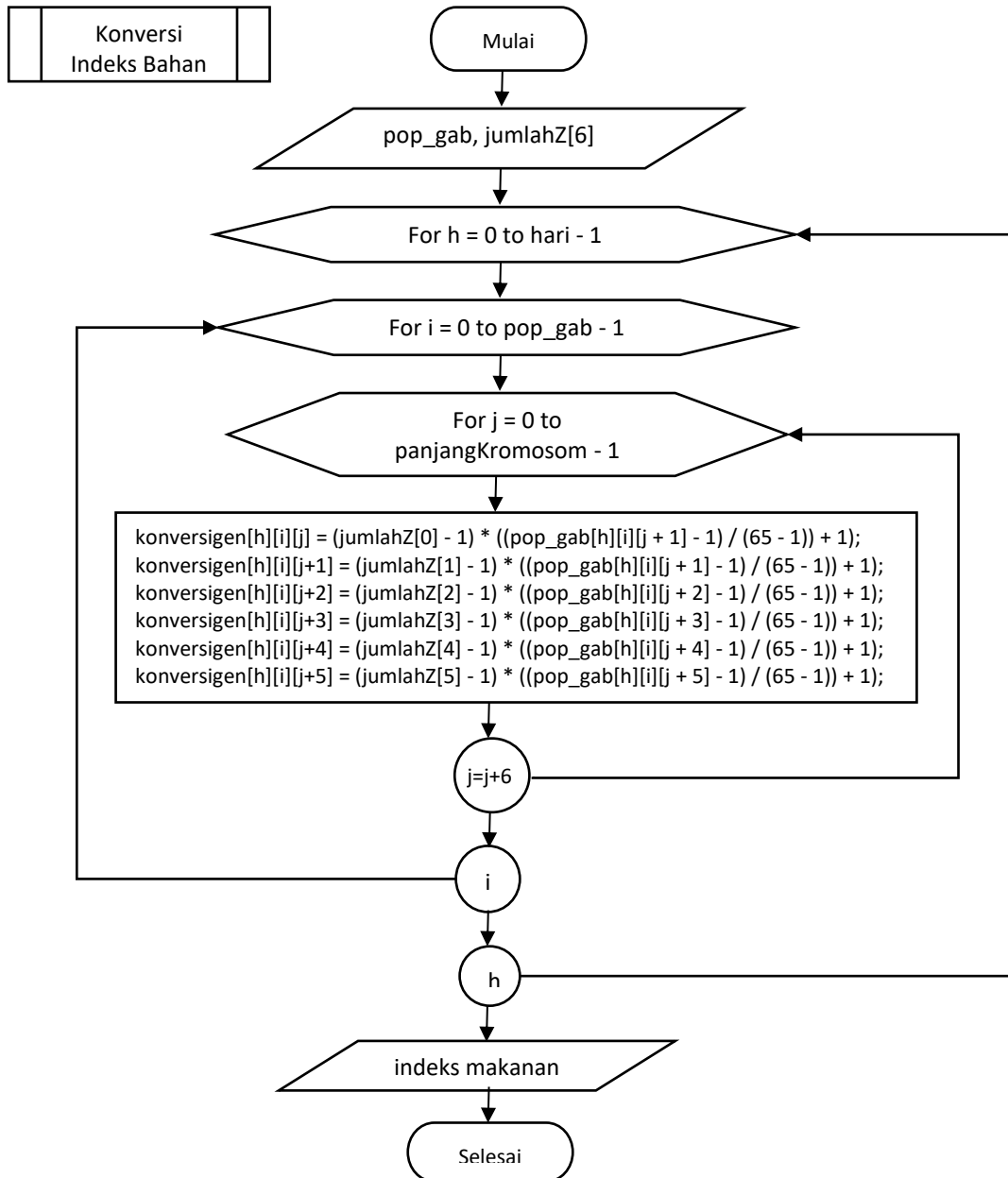


**Gambar 4.6 Diagram Alir Proses Mutasi**

Mutasi digunakan untuk menghasilkan *child* (reproduksi). Gambar 4.6 menjelaskan proses mutasi menggunakan metode *exchange mutataion*. Langkah pertama yaitu menghitung *bykchildMutasi* (*offspring*) yang dihasilkan dari nilai *mr* dikalikan *popsize*. Selanjutnya menentukan *parent* secara *random*, dengan batasan nilai *random* sejumlah banyaknya *popsize*. Langkah berikutnya yaitu

menentukan 2 titik gen terplih secara *random* dengan cara *math.random* dikalikan *panjangKromosom*. Hasil dari pemilihan tersebut disimpan dalam variabel *xp1* dan *xp2*, kemudian proses *exchange mutation* dilakukan dengan cara menukar hasil *xp1* dengan *xp2*, selanjutnya hasil pertukaran gen disimpan dalam *array hasil\_mutasi*. Proses berulang sampai memenuhi jumlah *offspring* dan sebanyak hari yang telah ditentukan. Hasil akhir berupa individu baru dari hasil mutasi.

#### 4.3.4 Konversi Indeks Bahan Makanan



**Gambar 4.7 Diagram Alir Konversi Indeks Gen Ke Indeks Bahan Makanan**

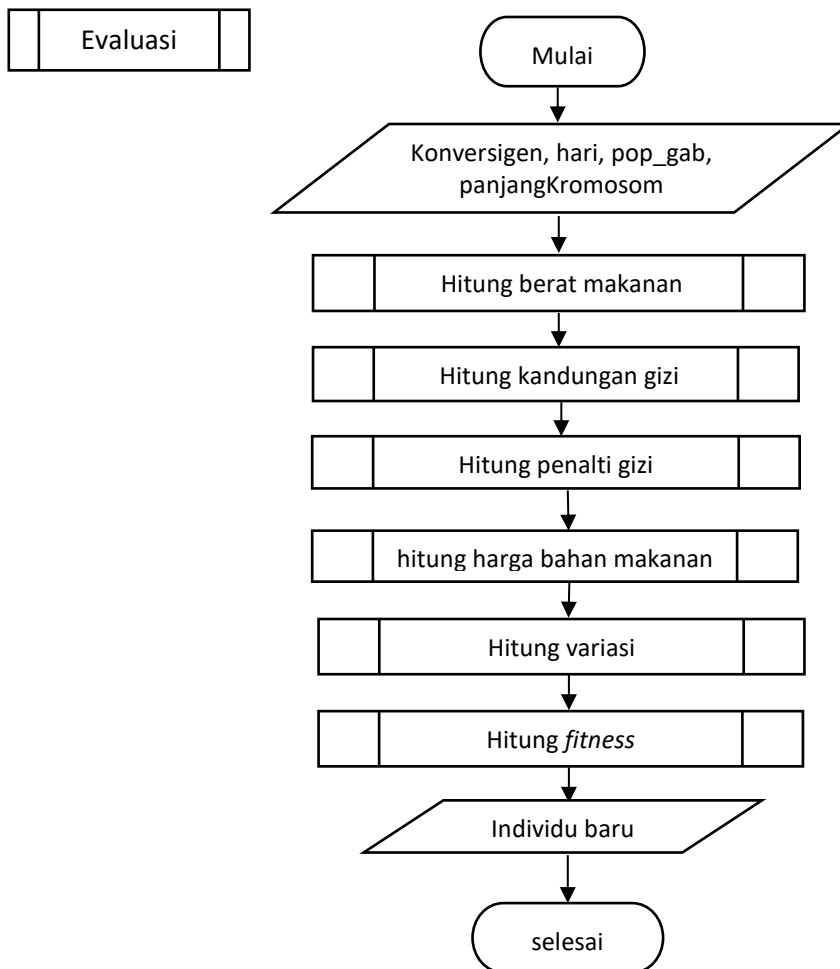
Gambar 4.7 menjelaskan mengenai konversi indeks bahan makanan. Konversi indeks bahan makanan digunakan untuk mengetahui indeks bahan makanan, sehingga indeks gen kromosom lebih mudah dikenali sesuai nama



bahan makanan. Pada penelitian ini, panjang kromosom sudah ditentukan yaitu 18 gen dibagi 3 waktu yaitu pagi, siang dan malam. Pada satu waktu misalnya, waktu pagi terdapat 6 gen kromosom yang mewakili kandungan gizi pada makanan. Urutan indeks gen yaitu karbohidrat, protein hewani, protein nabati, lemak, buah dan sayuran. Langkah pertama sesuai Gambar 4.7 yaitu memasukkan indeks gen yang disimpan dalam *array pop\_gab* yang berisi seluruh individu (*parent* dan *child*). Kemudian Proses konversi gen ke indeks bahan makanan berdasarkan Persamaan 2.11, jumlah data bahan makanan pada setiap kandungan disimpan dalam *array jumlahZ*, jumlah data pada *array* ke 0 mewakili jumlah data bahan makanan pada karbohidrat begitu seterusnya. Proses tersebut terus berulang hingga jumlah *pop\_gab* terpenuhi pada masing-masing hari. Hasil akhir berupa indeks bahan makanan yang disimpan dalam *array* konversigen.

### 4.3.5 Evaluasi

Tahapan evaluasi digunakan untuk menghitung nilai kebugaran setiap individu (*fitness*). Dalam penelitian ini terdapat beberapa langkah untuk melakukan evaluasi pada setiap individu. Langkah-langkah tersebut dijelaskan pada Gambar 4.8 merupakan diagram alir untuk menentukan nilai *fitness*.



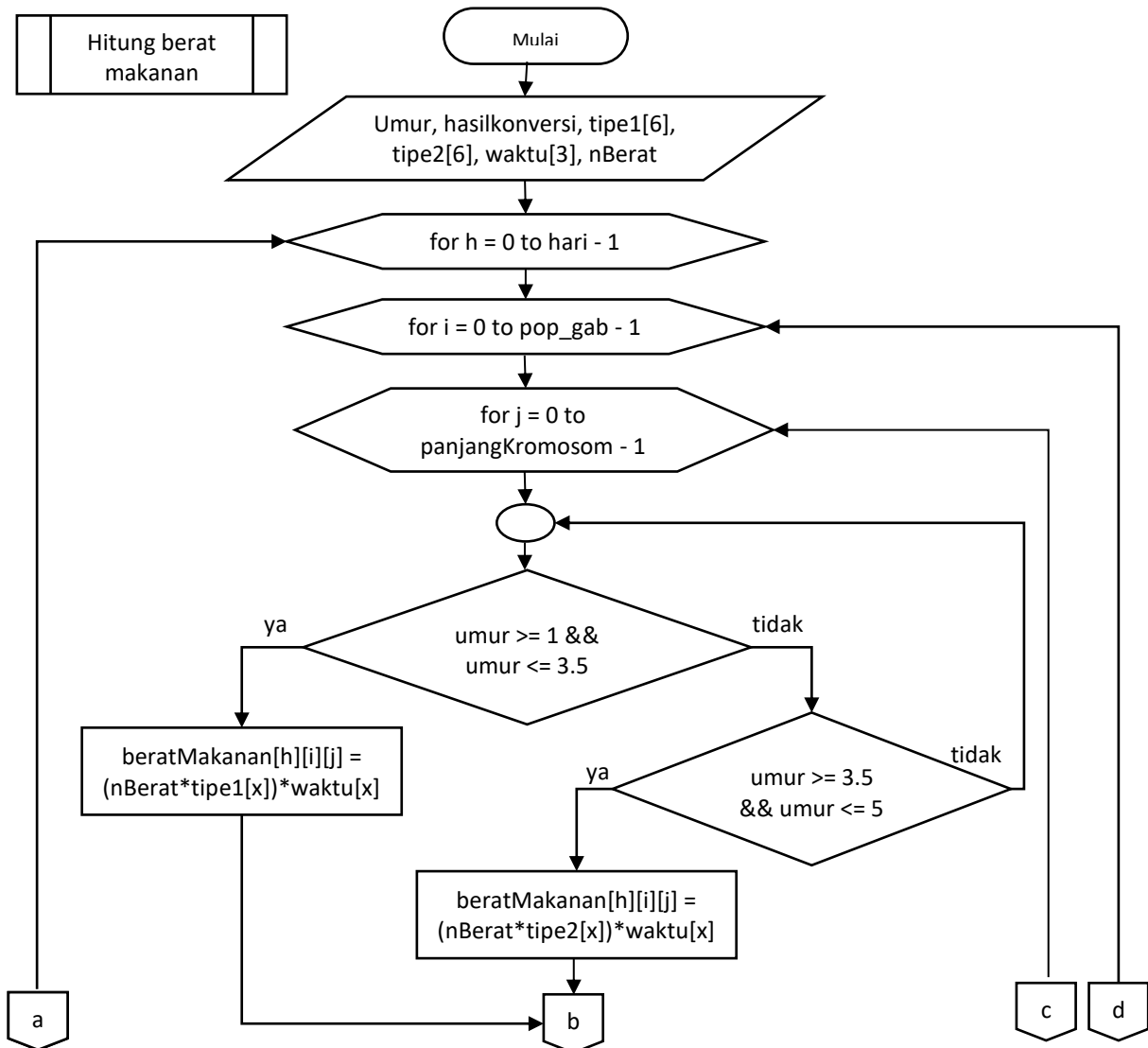
**Gambar 4.8 Diagram Alir Proses Evaluasi**

Tahapan evaluasi untuk optimasi gizi pada balita menggunakan algoritme genetika adalah sebagai berikut:

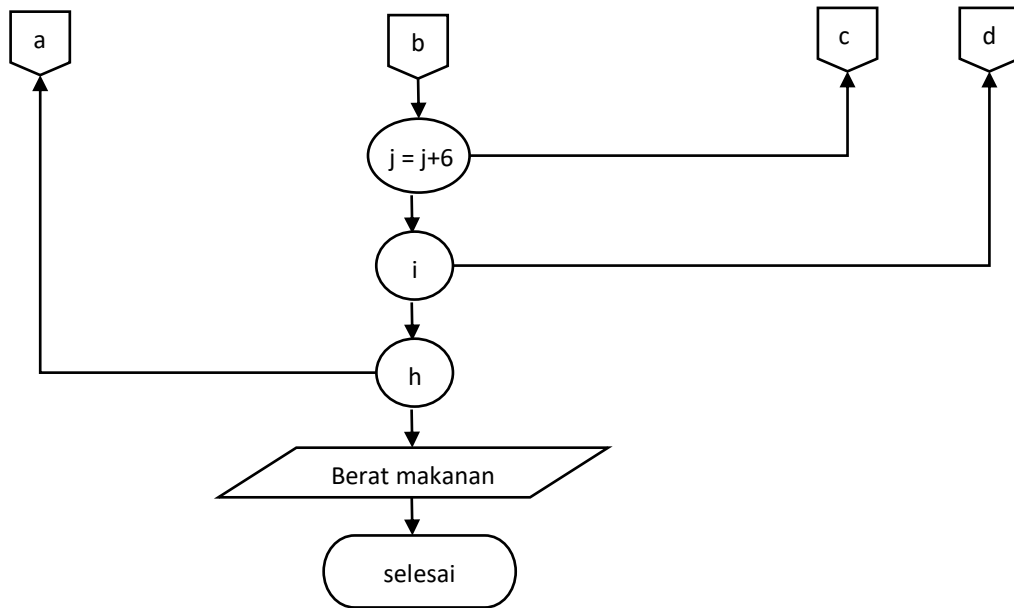
1. *Input* berupa hasil konversigen, hari, jumlah populasi gabungan dan panjangKromosom.
2. Proses menghitung berat bahan makanan dari setiap indeks bahan makanan.
3. Proses menghitung kandungan gizi pada setiap bahan makanan.
4. Proses perhitungan penalti gizi pada setiap individu.
5. Proses menghitung harga bahan makanan pada setiap balita.
6. Proses konversi nama BM (Bahan Makanan).
7. Proses menghitung variasi bahan makanan.
8. Proses menghitung nilai *fitness* setiap individu.

Berdasarkan tahapan-tahapan penyelesaian evaluasi, berikut ini proses dari setiap tahapan untuk menghitung nilai kebugaran (*fitness*) setiap individu:

### 1. Menghitung berat makanan



Gambar 4.9 Diagram Alir Proses Perhitungan Berat Makanan



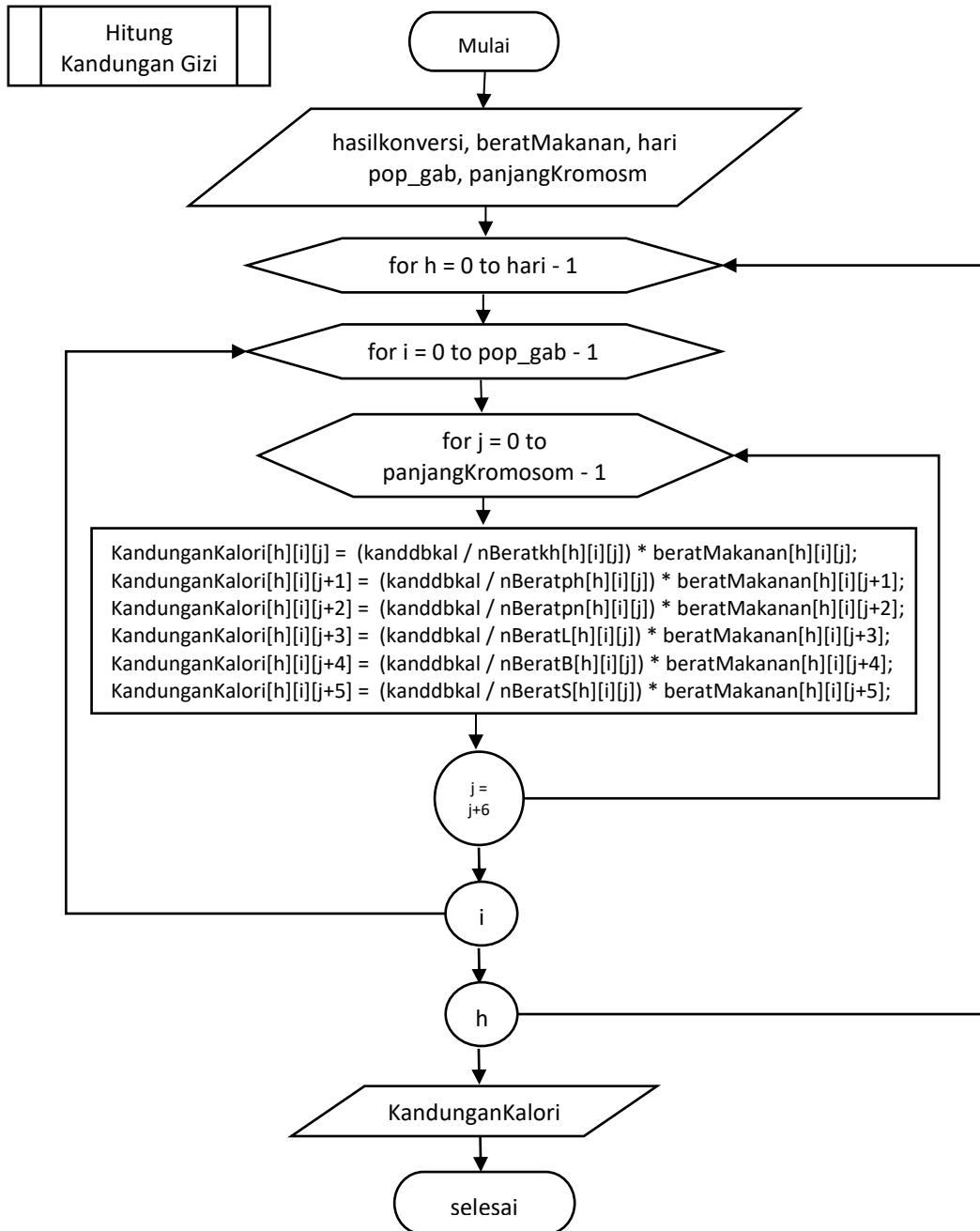
**Gambar 4.10 Diagram Alir Proses Perhitungan Berat Makanan (lanjutan)**

Berdasarkan Gambar 4.9 dan Gambar 4.10 dapat dijelaskan tahapan menghitung berat bahan makanan. Langkah pertama yaitu memasukkan parameter berat bahan makanan berupa umur balita, *hasilkonversi*, tipe porsi bahan makanan, waktu dan berat bahan makanan. Langkah kedua yaitu menghitung masing-masing berat bahan makanan sesuai dengan 2 kondisi yang sudah ditentukan, kondisi tersebut adalah jika umur balita berkisar antara 1 sampai kurang dari 3,5 tahun maka Persamaan yang digunakan sesuai dengan diagram alir pada Gambar 4.9 yaitu  $beratMakanan[h][i][j] = (nBerat * tipe1[x]) * waktu[x]$ . Sedangkan jika umur balita lebih dari 3,5 sampai 5 tahun, maka Persamaan yang digunakan yaitu  $beratMakanan[h][i][j] = (nBerat * tipe2[x]) * waktu[x]$ . Perbedaan pada 2 kondisi tersebut terletak pada tipe porsi bahan makanan yaitu tipe1 dan tipe2 sesuai Tabel 4.1. daftar anjuran porsi makan. Jumlah waktu yang digunakan yaitu pagi, siang dan malam dengan jumlah persentase masing-masing waktu sesuai dengan Persamaan 2.12. Nilai *nBerat* diambil dari data bahan makanan sesuai dengan indeks bahan makanan yang disimpan dalam database. Langkah tersebut berulang sampai kondisi hari, *pop\_gab* (individu), dan *panjangKromosom* terpenuhi. Hasil akhir dari proses tersebut, berupa berat bahan makanan yang dianjurkan pada balita.

## 2. Menghitung Kandungan Gizi

Terdapat 4 kandungan zat gizi yang dihitung dalam tahap ini, antara lain kandungan kalori, kandungan karbohidrat, kandungan protein dan kandungan lemak. Masing-masing bahan makanan dihitung sesuai dengan keempat kandungan zat gizi tersebut. Setiap bahan makanan memiliki kandungan zat gizi yang berbeda-beda, misalnya bahan makanan yang bersumber dari karbohidrat yaitu bihun dengan berat sebesar 50 gram dan memiliki kandungan kalori sebanyak 175, kandungan protein sebanyak 4, serta kandungan lemak sebesar 0

dan kandungan karbohidrat sebanyak 40. Gambar 4.11 merupakan diagram alir menghitung kandungan gizi zat Kalori.

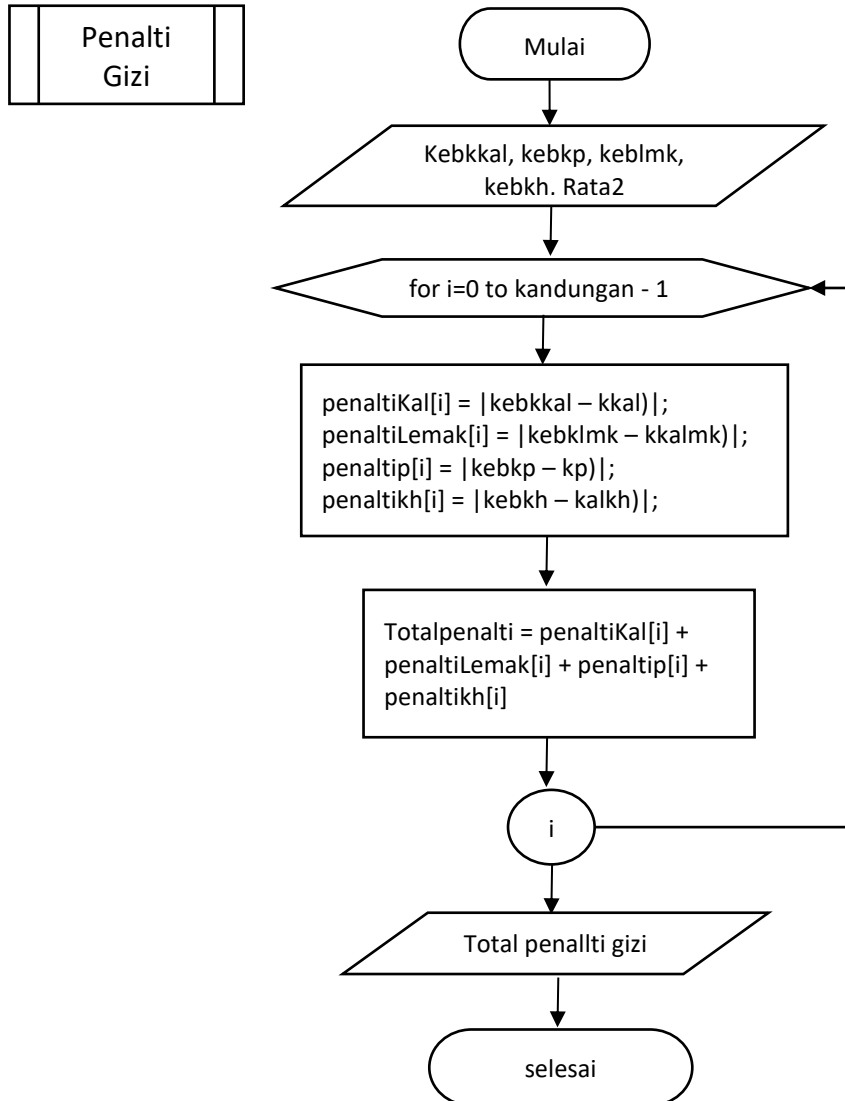


**Gambar 4.11 Diagram Alir Proses Perhitungan Kandungan Gizi**

Gambar 4.11 merupakan salah satu dari keempat kandungan zat gizi yang dihitung. Langkah pertama yaitu *inputkan* nilai berat makanan yang telah dihitung pada subbab 4.3.4, kemudian menghitung sesuai dengan persamaan yang telah ditentukan, nilai  $nBerat[x]$  diambil dari database sesuai sumber bahan makanan yaitu  $nBeratk$  yang diambil dari sumber makanan dari karbohidrat, dan begitu seterusnya. Nilai  $kanddbkal$  diambil dari database masing-masing sumber bahan makanan yang memiliki zat gizi kalori. Diagram alir dari perhitungan keempat

kandungan zat gizi tersebut sama, hanya terdapat perbedaan pada kandungan zat gizi yang dihitung yaitu  $kandbkalori$ ,  $kanddbkh$ ,  $kanddbpro$  dan  $kanddblmk$ . Hasil akhir diperoleh berupa kandungan zat gizi (kalori, karbohidrat, protein dan lemak) pada masing-masing bahan makanan.

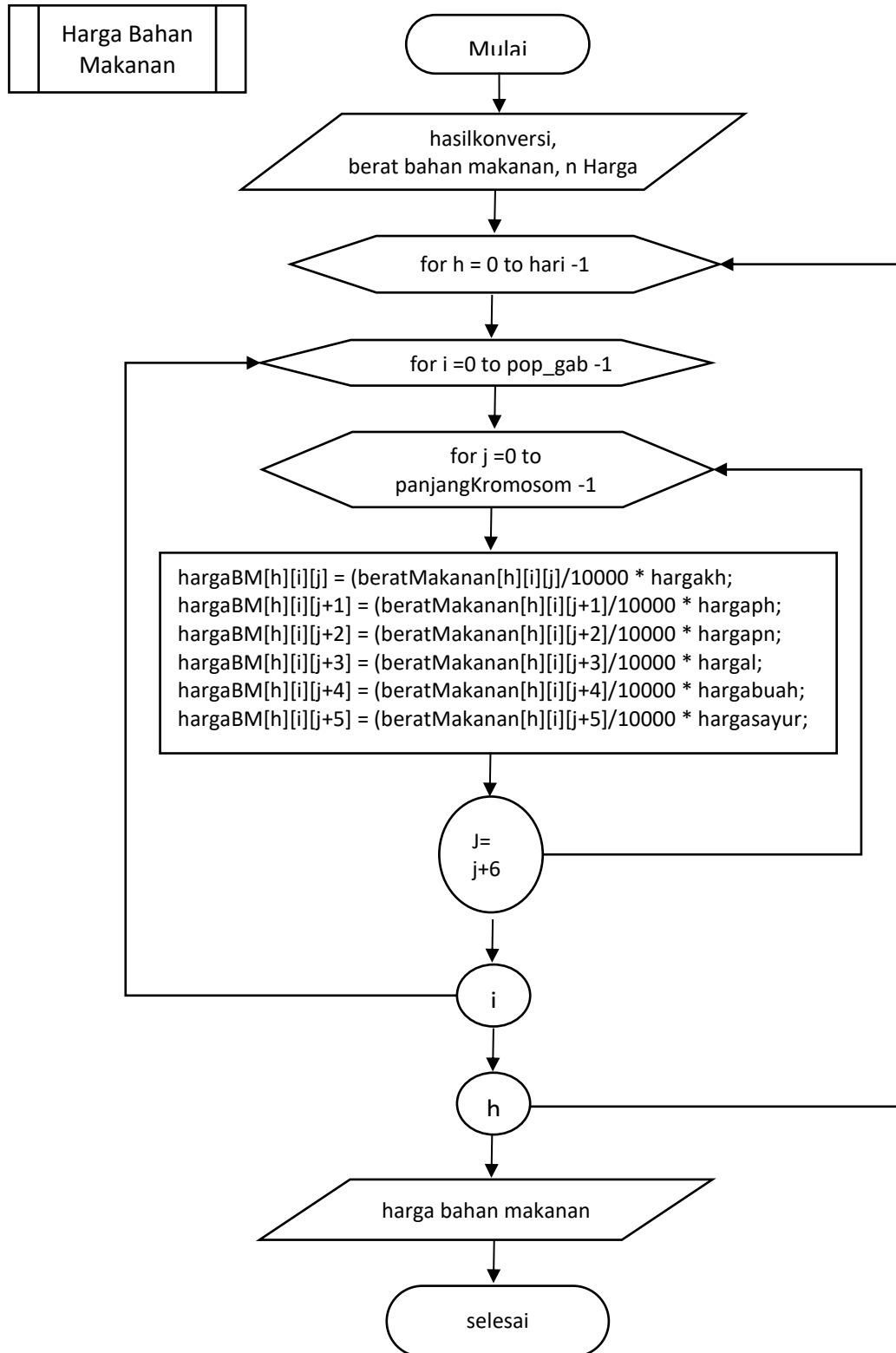
### 3. Menghitung Penalti Gizi



**Gambar 4.12 Diagram alir Proses Pehitungan Penalti Gizi**

Berdasarkan Gambar 4.12 mengenai diagram alir, pada proses perhitungan penalti dapat diketahui parameter *input* untuk menghitung penalti gizi pada balita yaitu  $kebh$ ,  $kebkal$ ,  $kebkp$ , dan  $keblmk$ . Penalti gizi dihitung sesuai dengan Persamaan 2.14 yaitu *kebutuhan gizi - rata2 kandungan gizi*. Setelah mendapatkan nilai penalti pada keempat zat gizi karbohidrat, lemak, protein dan kalori. Langkah selanjutnya yaitu menghitung nilai total penalti pada masing-masing zat gizi. Langkah tersebut terus berulang sampai jumlah mendapat nilai total penalti pada masing-masing individu. Hasil akhir berupa nilai total penalti yang akan digunakan untuk menghitung nilai *fitness*.

#### 4. Menghitung Harga Bahan Makanan

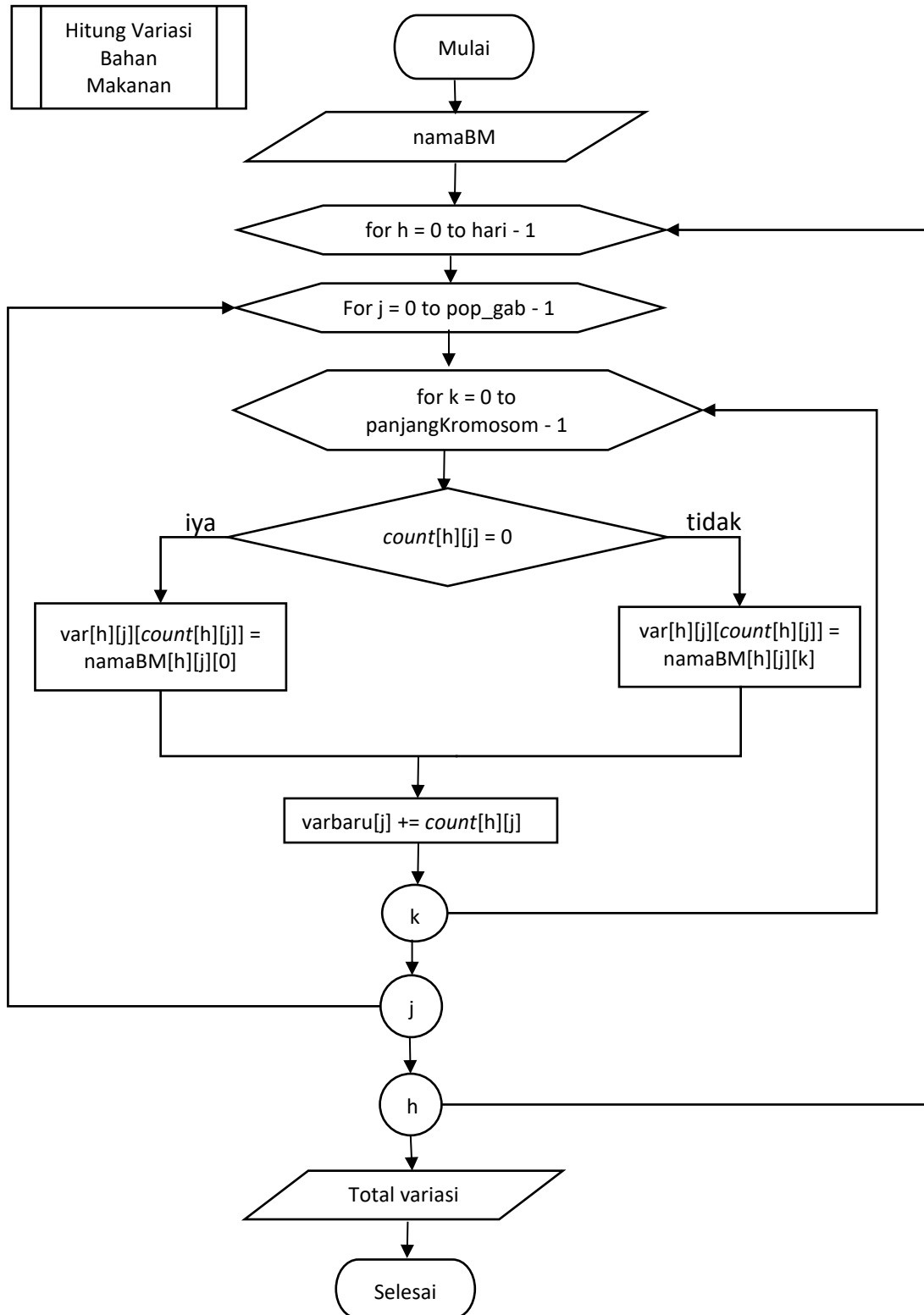


**Gambar 4.13 Diagram Alir Proses Perhitungan Harga Bahan Makanan**

Gambar 4.13 merupakan diagram alir proses perhitungan harga bahan makanan. Proses perhitungan harga makanan dimulai dari parameter *input*

antarlain *hasilkonversi*, *nHarga*, dan *beratMakanan*. *nHarga* diambil dari database masing-masing harga bahan makanan sesuai dengan indeks. Hasil akhir berupa harga bahan makanan dari setiap jenis bahan makanan.

### 5. Menghitung Variasi Bahan Makanan

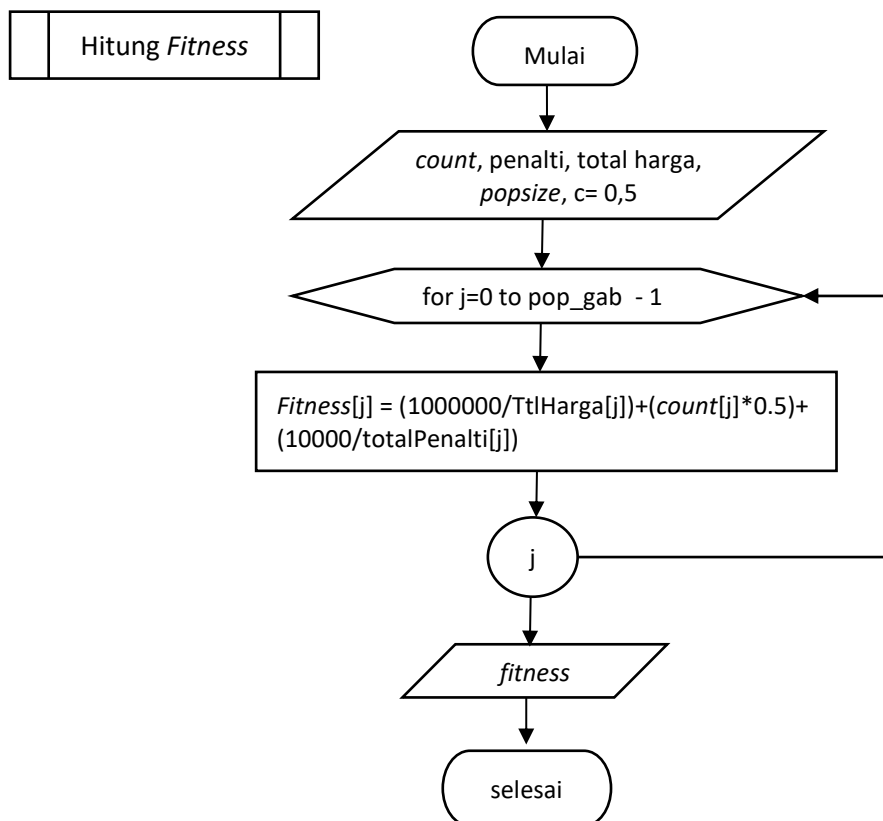


Gambar 4.14 Diagram Alir Proses Perhitungan Variasi

Gambar 4.14 merupakan proses perhitungan variasi bahan makanan yang digunakan untuk mengetahui jumlah variasi bahan makanan dalam setiap individu. Untuk menghitung variasi bahan makanan menggunakan pengecekan setiap nama bahan makanan, jika nama bahan makanan pada satu individu selama satu hari memiliki nama yang sama maka nilai variasi tersebut 0. Tetapi jika nama bahan makanan pada satu individu selama satu hari memiliki nama yang berbeda maka nilai variasi bahan makanan yaitu 1. Hasil akhir berupa total variasi bahan makanan setiap individu selama jumlah hari yang ditentukan.

## 6. Menghitung *Fitness*

Proses menghitung *fitness* dilakukan untuk mengetahui nilai kebugaran masing-masing individu. Langkah pertama yang dilakukan untuk menghitung nilai *fitness* yaitu, memasukan parameter *fitness* berupa total penalti, nilai total variasi (*count*), dan nilai total harga bahan makanan. Selanjutnya menghitung nilai *fitness* sesuai dengan Persamaan 2.10. Hasil akhir berupa nilai *fitness* pada masing-masing individu. Gambar 4.15 menjelaskan proses perhitungan nilai *fitness*.

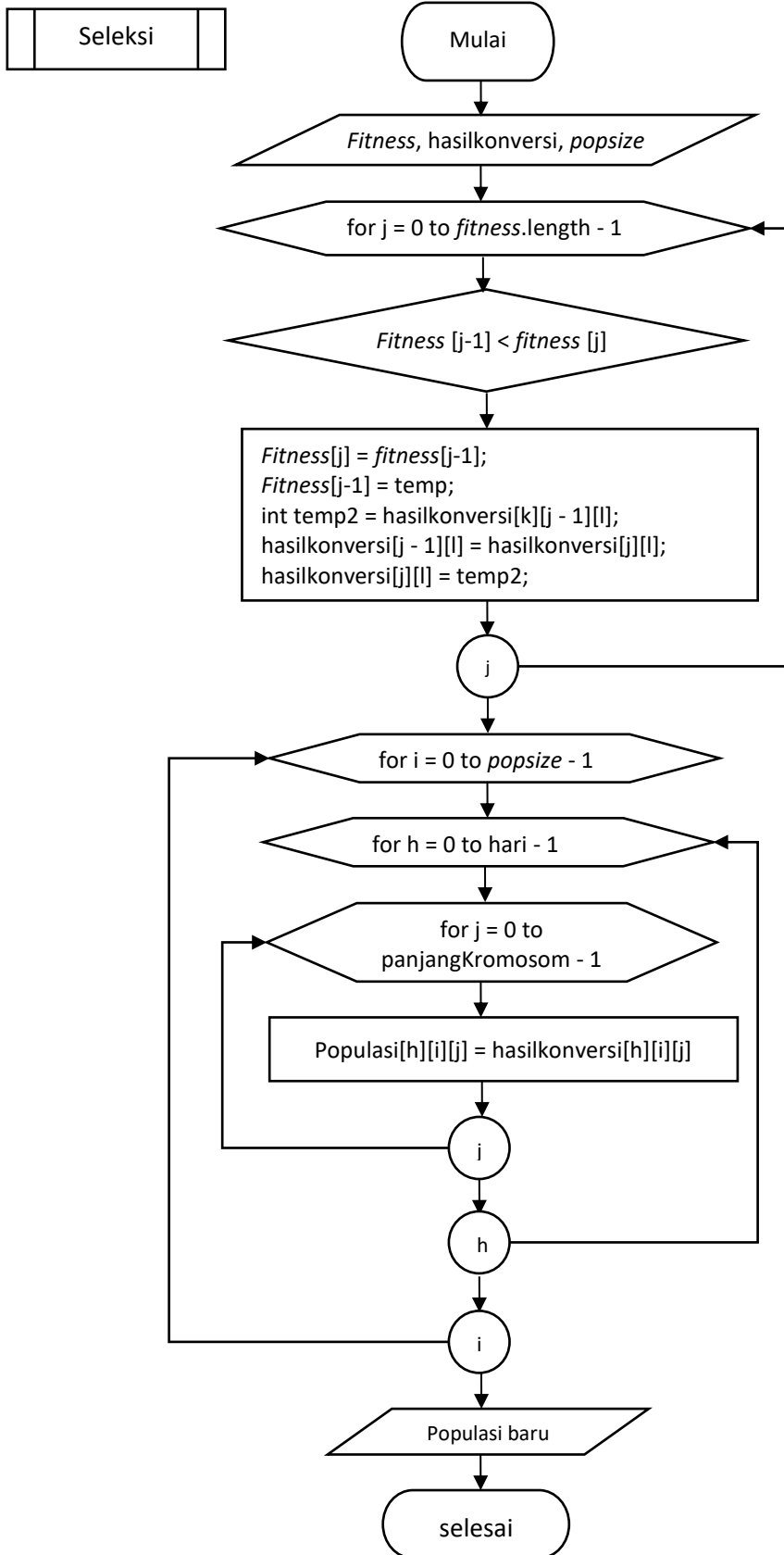


Gambar 4.15 Diagram Alir Proses Perhitungan *Fitness*

### 4.3.6 Seleksi

Gambar 4.16 merupakan proses seleksi menggunakan *elitism selection*.





**Gambar 4.16 Diagram Alir Proses Seleksi**

Seleksi digunakan untuk memilih individu terbaik dengan jumlah *fitness* terbesar. Proses seleksi diawali dari *inputan* parameter seleksi antarlain *fitness*, *hasilkonversi* dan *popsi*. Langkah pertama yaitu mengurutkan nilai *fitness* dari yang terbesar ke nilai *fitness* yang terkecil. Hasil dari pengurutan nilai *fitness* disimpan dalam variabel *hasilkonversi*. Selanjutnya diambil nilai *fitness* terbesar sebanyak jumlah *popsi*, yang akan digunakan pada generasi selanjutnya. Hasil seleksi disimpan dalam variabel *populasi*. Hasil akhir proses ini menghasilkan populasi baru, nilai *fitness* tertinggi dalam tiap generasi akan diambil menjadi individu terbaik.

#### 4.4 Contoh Permasalahan

Terdapat data balita sebagai berikut:

**Tabel 4.2 Data Balita**

No	Nama	Umur	BB
1	Balita 17	3,3 <sup>th</sup>	8,5 <sup>kg</sup>

Berdasarkan Tabel 4.2 diatas, maka didapatkan kebutuhan gizi balita sebagai berikut:

1. Menghitung berat badan ideal (BBI) menggunakan Persamaan 2.1  
 $BBI = (3,3 \times 2) + 8 = 14,6 \text{ kg}$
2. Menghitung kebutuhan Energi (KKAL) menggunakan Persamaan 2.2  
 $\text{Kebutuhan Energi} = 100 \text{ kal/kg} \times 14,6 \text{ kg} = 1460 \text{ kal/hari}$
3. Menghitung kebutuhan Protein (KP) menggunakan Persamaan 2.3  
 $KP = (10\% \times 1460) \div 4 = 36,5 \text{ gram}$
4. Menghitung kebutuhan Lemak (KL) menggunakan Persamaan 2.4  
 $KL = (20\% \times 1460) \div 9 = 32,44 \text{ gram}$
5. Menghitung kebutuhan Karbohidrat (KH) menggunakan Persamaan 2.5  
 $KH = (70\% \times 1460) \div 4 = 255,5 \text{ gram}$

Hasil kebutuhan gizi tiap balita disertai tipe anjuran porsi pada Tabel 4.3.

**Tabel 4.3 Hasil Kebutuhan Gizi Balita**

Nama	KKAL	KP	KL	KH	Tipe Porsi
Balita 17	1460 kal/hari	36,5 Gram	32,44 Gram	255,5 Gram	1

Berdasarkan Tabel 4.3 dapat ditentukan tipe anjuran porsi makan untuk setiap balita sesuai kelompok umur. Pada contoh permasalahan diatas terdapat 4 kandungan gizi yang digunakan yaitu Karbohidrat (KH), protein hewani (PH), lemak dan buah. Tabel 4.4 menjelaskan anjuran porsi makan yang harus dipenuhi untuk setiap balita dalam sekali waktu makan pagi, siang dan malam sesuai tipe porsi.

**Tabel 4.4 Anjuran Porsi Makan pada Balita**

Nama	Tipe Porsi	KH	PH	Lemak	Buah
Balita 17	1	3 porsi	1 porsi	3 porsi	3 porsi

## 4.5 Penyelesaian menggunakan Algoritme Genetika

### 4.5.1 Parameter Algoritme Genetika

Terdapat beberapa parameter algoritme genetika yang diperlukan untuk proses optimasi. Parameter-parameter tersebut antarlain *popsize* yang digunakan untuk menentukan jumlah populasi dalam satu generasi. Parameter yang kedua yaitu nilai *crossover rate* (*cr*) digunakan untuk menentukan jumlah *offspring* dari hasil reproduksi *crossover* sedangkan nilai *mutation rate* (*mr*) digunakan untuk menentukan jumlah *offsring* dari hasil reproduksi mutasi. Parameter selanjutnya yaitu jumlah generasi digunakan untuk menentukan banyaknya jumlah perulangan dalam satu kali proses. Contoh pada perhitungan manual menggunakan nilai parameter sebagai berikut:

*Popsize* : 3  
*cr* : 0,7  
*mr* : 0,3  
 Jumlah generasi : 1  
 Angka Permutasi : [1-65]

### 4.5.2 Representasi Kromosom

Terdapat 169 data makanan yang terdiri dari karbohidrat, protein hewani dan nabati, lemak, sayuran dan buah-buahan. Representasi kromosom yang digunakan yaitu *integer*. Angka *random* yang digunakan yaitu 1 sampai 65 disesuaikan dengan angka permutasi yang optimal pada penelitian (Eliantara, F., Cholissodin, I., & Indriati, 2016), angka tersebut merupakan batas angka yang ditentukan secara acak untuk mempermudah pembentukan gen pada setiap jenis bahan makanan . Setiap jenis bahan makanan memiliki jumlah makanan berbeda, sehingga diperlukan *range* permutasi 1-65 untuk menentukan batasan nilai yang harus diisi setiap gen. Jika *range* nilai setiap gen disesuaikan dengan jumlah setiap bahan makanan maka, pada *iterasi* selanjutnya akan menghasilkan nilai gen yang melebihi dari *range* yang terdapat pada setiap jenis bahan makanan. Contoh pada perhitungan manual menggunakan panjang kromosom 18 gen dalam sehari, terdiri dari 6 jenis bahan makanan (karbohidrat, protein hewani, protein nabati, lemak, buah dan sayur), tersusun dalam 3 waktu yaitu pagi, siang dan malam. Proses representasi kromosom pada Tabel 4.5.

**Tabel 4.5 Representasi Kromosom**

	Hari 1											
	Pagi						Siang					
	KH	PH	PN	L	B	S	KH	PH	PN	L	B	S
P1	61	53	54	39	39	61	30	49	41	47	53	17
P2	8	59	53	18	44	22	41	40	51	3	43	17

	Malam					
	KH	PH	PN	L	B	S
P1	52	54	32	20	62	29
P2	18	59	46	15	46	52

**Keterangan :**

- KH : Karbohidrat
- PH : Protein Hewani
- PN : Protein Nabati
- L : Lemak
- B : Buah
- S : Sayur

**4.5.3 Inisialisasi Populasi Awal**

Inisialisasi populasi awal digunakan untuk pembentukan kromosom pada setiap *parent*. Jumlah populasi awal sesuai dengan jumlah populasi yang sudah ditentukan, dan nilai pada setiap gen diambil secara acak sesuai angka permutasi yaitu 1 sampai 65. Berdasarkan contoh permasalahan yang dijelaskan sebelumnya, maka dapat ditentukan jumlah populasi yaitu 3, panjang kromosom berdasarkan jumlah kandungan gizi dalam satu waktu yaitu 6 kandungan gizi dengan 3 waktu yaitu pagi, siang, dan malam sehingga memiliki panjang kromosom 18 gen dalam satu hari. Hasil proses inisialisasi populasi awal ditunjukkan pada Tabel 4.6.

**Tabel 4.6 Hasil Proses Inisialisasi Populasi Awal**

**\*Hari ke- 1**

P : 1																	
60	9	46	24	40	29	64	40	1	53	51	57	52	47	10	13	56	20
P : 2																	
57	42	31	62	25	59	7	47	56	2	26	15	34	32	34	37	15	5
P : 3																	
46	45	42	8	63	9	25	47	12	45	40	55	6	48	4	64	49	9

**\*Hari ke-2**

P : 1																	
36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
P : 2																	
34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34
P : 3																	
53	53	53	53	53	53	53	53	53	53	53	53	53	53	53	53	53	53

#### 4.5.4 Crossover

Langkah pertama untuk melakukan reproduksi yaitu mencari nilai *crossover*. Teknik *Crossover* yang digunakan yaitu *extended intermediate crossover*. Langkah pertama yang dilakukan yaitu menentukan nilai *cr* untuk mendapatkan jumlah *offspring*. Berdasarkan permasalahan sebelumnya, ditentukan nilai *cr* 0,7. Maka nilai *offspring*:

$$\text{Offspring} = 3 \times 0,7 = 2,1$$

Hasil *offspring* di atas, dibulatkan kebawah menjadi 2. Maka proses *crossover* menggunakan *extended intermediate* akan menghasilkan 2 *offspring* (*child*). Setelah ditentukan jumlah *child*, langkah selanjutnya yaitu menentukan 2 *parent* secara acak. Kemudian menentukan nilai *a* yang dipilih secara acak pada masing-masing *child*, kemudian dihitung sesuai Persamaan 2.6 untuk *child* 1 dan Persamaan 2.7 untuk *child* 2. Contoh *parent* terpilih hari pertama yaitu *parent* 1 dan *parent* 2. Sedangkan hari kedua *parent* yang terpilih yaitu *parent* 3 dan *parent* 1. Berikut ini hasil proses *crossover* pada Tabel 4.7.

**Tabel 4.7 Proses Crossover**

Hari ke-1														
<b>a1</b>	0,63	0,4	- 0,04	0,42	0,53	- 0,16	1,2	- 0,02	0,21	.... .	.... .	0,34	- 0,03	0,66
<b>a2</b>	0	0,01	- 0,05	0,03	0,61	0,14	0,26	0,13	1,2	.... .	.... .	0,97	0,57	1,06
<b>C1</b>	54,8 2	30,6 0	41,8 4	14,7 2	50,8 1	5,80	71,8 0	47,1 4	9,69	.... .	.... .	46,6 6	48,7 9	16,2 6
<b>C2</b>	60	9,36	46,2	23,5 2	54,0 3	26,2	53,8 6	40,9 1	14,2	.... .	.... .	62,4 7	52,0 1	8,34
Hari ke-2														
<b>a1</b>	0,25	0,86	0,6	0,89	0,27	0,85	1,09	0,71	- 0,16	.... .	.... .	- 0,05	0,8	0,18
<b>a2</b>	0,73	0,63	0,72	0,58	0,41	0,05	0,6	0,09	0,68	.... .	.... .	0,56	0,69	- 0,02
<b>C1</b>	40,2 5	13,9 8	17,4 0	57,2 8	22,0 0	26,9 0	13,4 8	28,3 0	51,0 8	.... .	.... .	15,9 5	17,8 0	38,3 6
<b>C2</b>	40,5 9	17,4 1	11,3 2	32,8 4	22	28,3	32,8	34,3	44,8 4	.... .	.... .	16,4 4	36,9 1	40,0 4

**Keterangan:**

- Warna Biru : Waktu Makan Pagi
- Warna Putih : Waktu Makan Siang
- Warna Kuning : Waktu Makan Malam

Proses *crossover* tersebut dalam satu kali *crossover* hanya menghasilkan 2 *child*, sehingga dalam satu hari hanya ada *child*1 dan *child*2 yang dihitung sesuai Persamaan 2.6 dan Persamaan 2.7. Nilai *a*1 digunakan untuk nilai interval pada *child*1 sedangkan nilai *a*2 digunakan untuk nilai interval pada *child*2. Hasil dari *Crossover* tersebut perlu dinormalisasi ke dalam bilangan *real*, untuk menghindari

indeks yang melebihi bahan makanan. Ketika hasil *crossover* ada yang melebihi jumlah bahan makanan maka, hasil *crossover* tersebut dikurangi dengan jumlah data bahan makanan tersebut. Tabel 4.8 hasil normalisasi hasil *crossover*.

**Tabel 4.8 Normalisasi Hasil Crossover**

**#Hari 1**

<b>C1</b>	55	31	42	15	51	6	72	47	10	...	...	47	49	16
<b>Int</b>	26	31	29	7	14	6	43	10	10	...	...	39	12	16
<b>C2</b>	60	9	46	24	54	26	54	41	14	...	...	62	52	8
<b>Int</b>	31	9	33	16	17	26	25	4	1	...	...	54	15	8

**#Hari2**

<b>C1</b>	40	14	17	57	22	27	13	28	51	...	...	16	18	38
<b>Int</b>	11	14	4	49	22	27	13	28	38	...	...	8	18	38
<b>C2</b>	41	17	11	33	22	28	33	34	45	...	...	16	37	40
<b>Int</b>	12	17	11	25	22	28	4	34	32	...	...	8	37	40

**4.5.5 Mutasi**

Langkah reproduksi selanjutnya yaitu mutasi. Jenis mutasi yang digunakan yaitu *reciprocal exchange mutation* dengan cara memilih *parent* secara acak dan memilih 2 titik gen secara acak, kemudian nilai gen tersebut ditukar. Langkah pertama yaitu menentukan jumlah *offspring* yang akan dihasilkan. Berdasarkan contoh permasalahan, nilai *mr* yaitu 0,5 maka *offspring* yang dihasilkan sebagai berikut:

$$Offspring = 3 \times 0,3 = 0,9$$

Nilai tersebut dibulatkan keatas menjadi 1 sehingga jumlah *child* yang dihasilkan sejumlah 1. Langkah selanjutnya yaitu memilih 1 *parent*. *Parent* terpilih pada hari pertama yaitu *parent 1* dan pada hari kedua yaitu *parent 1*. Proses mutasi ditunjukkan pada Tabel 4.9.

**Tabel 4.9 Proses Mutasi**

<b>Hari ke-1</b>																		
<b>xp1 = 2 &amp; xp2 = 13</b>																		
<b>P1</b>	60	9	46	24	40	29	64	40	1	53	51	57	52	47	10	13	56	20
<b>C3</b>	60	9	47	24	40	29	64	40	1	53	51	57	52	46	10	13	56	20
<b>Hari ke-2</b>																		
<b>xp1 = 10 &amp; xp2 = 8</b>																		
<b>P1</b>	36	20	6	11	22	15	44	7	49	33	19	5	55	29	60	16	49	38
<b>C4</b>	36	20	6	11	22	15	44	7	19	33	49	5	55	29	60	16	49	38

**4.5.6 Konversi Gen ke Indeks Makanan**

Berdasarkan hasil reproduksi *crossover*, ada beberapa gen yang melebihi jumlah indeks bahan makanan sehingga indeks gen tidak dapat dikenali jenis bahan makanan tersebut. Perhitungan konversi gen ke indeks bahan makanan sesuai Persamaan 2.11. Contoh perhitungan pada *parent 1* untuk hari pertama:

$$Gen 1 = \left( (29 - 1) * \frac{60 - 1}{65 - 1} \right) + 1 = 27$$

$$Gen 2 = \left( (37 - 1) * \frac{9 - 1}{65 - 1} \right) + 1 = 6$$

$$Gen 3 = \left( (13 - 1) * \frac{46 - 1}{65 - 1} \right) + 1 = 9$$

Hasil konversi gen ke indeks bahan makanan disertai nama bahan makanan untuk *parent 1* ditunjukkan pada Tabel 4.10.

**Tabel 4.10 Konversi Gen menjadi Indeks Bahan Makanan**

Hari ke-	Waktu makan	Jenis bahan makanan	P1	Indeks bahan makanan	Nama Bahan Makanan
1	Pagi	KH	60	27	Tepung singkong
		PH	9	6	Bakso
		PN	46	9	Keju
		Lemak	24	4	Minyak Kelapa Sawit
		Buah	40	23	Kedondong
		Sayur	29	20	Taoge Kacang Hijau
	Siang	KH	64	29	Jagung Segar
		PH	40	23	Hati Ayam
		PN	1	1	Tahu
		Lemak	53	7	Mayonaise
		Buah	51	29	Kedondong
		Sayur	57	40	Daun Katuk
	Malam	KH	52	23	Roti putih
		PH	47	27	Putih telur ayam
		PN	10	3	Keju
		Lemak	13	2	Santan Kelapa
		Buah	56	32	Jambu Air
		Sayur	20	14	Brokoli
2	Pagi	KH	36	16	Roti putih
		PH	20	12	Bakso
		PN	6	2	Tempe
		Lemak	11	2	Keju Krim
		Buah	22	13	Gula
		Sayur	15	11	Daun Lobak
	Siang	KH	44	20	Roti putih
		PH	7	4	Ikan Gabus
		PN	49	10	Petai Segar
		Lemak	33	5	Keju Krim

		Buah	19	11	Semangka
		Sayur	5	4	Kacang Kapri
	Malam	KH	55	25	Singkong
		PH	29	17	Bakso
		PN	60	12	Tempe
		Lemak	16	3	Minyak Kelapa
		Buah	49	28	Pepaya
		Sayur	38	26	Nangka Muda

#### 4.5.7 Evaluasi

Digunakan untuk menghitung kebugaran (*fitness*) setiap kromosom. Langkah-langkah menghitung *fitness* sebagai berikut:

##### 1. Menghitung Berat Makanan

Langkah pertama yaitu menghitung berat setiap bahan makanan. Perhitungan berat makanan digunakan untuk mengetahui jumlah kebutuhan berat makanan setiap balita sesuai dengan tipe porsi yang sudah ditentukan pada Tabel 4.4 berisi daftar anjuran porsi. Daftar bahan makanan diambil pada Tabel 4.10. Perhitungan berat makanan menggunakan Persamaan 2.12. Hasil perhitungan berat bahan makanan pada Tabel 4.11. Contoh perhitungan pada *p1* pada waktu pagi hari:

$$\text{Tepung Singkong} = (50 * 3) * 25\% = 37,5 \text{ gr}$$

$$\text{Bakso} = (170 * 1) * 25\% = 2,5 \text{ gr}$$

$$\text{Keju} = (15 * 1) * 25\% = 10 \text{ gr}$$

**Tabel 4.11 Berat Bahan Makanan**

Hari ke-	Waktu makan	Jenis bahan makanan	Indeks bahan makanan	Nama Bahan Makanan	Berat Makanan (gram)
1	Pagi	KH	27	Tepung singkong	37,5
		PH	6	Bakso	2,5
		PN	9	Keju	10
		Lemak	4	Minyak Kelapa Sawit	3,75
		Buah	23	Kedondong	35,625
		Sayur	20	Taoge Kacang Hijau	75
	Siang	KH	29	Jagung Segar	36
		PH	23	Hati Ayam	14
		PN	1	Tahu	8
		Lemak	7	Mayonaise	6
Buah	29	Kedondong	24		



	Malam	Sayur	40	Daun Katuk	120
		KH	23	Roti putih	52,5
		PH	27	Putih telur ayam	10,5
		PN	3	Keju	7
		Lemak	2	Santan Kelapa	5,25
		Buah	32	Jambu Air	28,875
2	Pagi	Sayur	14	Brokoli	105
		KH	16	Roti putih	52,5
		PH	12	Bakso	7,5
		PN	2	Tempe	6,25
		Lemak	2	Keju Krim	3,75
		Buah	13	Gula	41,25
	Siang	Sayur	11	Daun Lobak	75
		KH	20	Roti putih	120
		PH	4	Ikan Gabus	16
		PN	10	Petai Segar	22
		Lemak	5	Keju Krim	48
		Buah	11	Semangka	63
	Malam	Sayur	4	Kacang Kapri	120
		KH	25	Singkong	52,5
		PH	17	Bakso	31,5
		PN	12	Tempe	17,5
		Lemak	3	Minyak Kelapa	5,25
		Buah	28	Pepaya	21
Sayur	26	Nangka Muda	105		

## 2. Menghitung Kandungan Gizi

Langkah selanjutnya yaitu menghitung kandungan gizi setiap makanan berdasarkan kandungan kalori, karbohidrat, protein dan lemak. Perhitungan kandungan gizi dilakukan untuk mengetahui jumlah kandungan gizi pada setiap bahan makanan yang dikonsumsi balita. Kandungan gizi pada makanan dapat diketahui menggunakan Persamaan 2.13. Contoh perhitungan pada *p1* untuk nama bahan tepung singkong:

$$\text{*Kandungan Kalori (KKAL) : Tepung singkong} = \left(\frac{37,5}{50}\right) \times 175 = 131,25 \text{ gram}$$

$$\text{*Kandungan Karbohidrat (KH) : Tepung singkong} = \left(\frac{37,5}{50}\right) \times 40 = 30 \text{ gram}$$

$$\text{*Kandungan Protein (KP) : Tepung singkong} = \left(\frac{37,5}{50}\right) \times 4 = 3 \text{ gram}$$

\*Kandungan Lemak (KL) : Tepung singkong =  $\left(\frac{37,5}{50}\right) \times 0 = 0 \text{ gram}$

Tabel 4.12 menunjukkan Hasil perhitungan kandungan gizi pada setiap makanan.

**Tabel 4.12 Kandungan Gizi pada Makanan**

Hari ke-	Waktu makan	Jenis bahan makanan	P1	Berat Makanan	KKAL	KH	KP	Lemak
1	Pagi	KH	27	37,5	131,25	30	3	0
		PH	6	2,5	12,5	0	1,75	0,5
		PN	9	10	18,75	1,75	1,25	0,75
		Lemak	4	3,75	37,5	0	0	3,75
		Buah	23	35,625	18,75	4,5	0	0
		Sayur	20	75	0	0	0	0
	Siang	KH	29	36	210	48	4,8	0
		PH	23	14	20	0	2,8	0,8
		PN	1	8	30	2,8	2	1,2
		Lemak	7	6	60	0	0	6
		Buah	29	24	30	7,2	0	0
		Sayur	40	120	0	0	0	0
	Malam	KH	23	52,5	183,75	42	4,2	0
		PH	27	10,5	26,25	0	2,45	1,75
		PN	3	7	26,25	2,45	1,75	1,05
		Lemak	2	5,25	52,5	0	0	5,25
		Buah	32	28,875	26,25	6,3	0	0
		Sayur	14	105	26,25	5,25	1,05	0
2	Pagi	KH	16	52,5	131,25	30	3	0
		PH	12	7,5	12,5	0	1,75	0,5
		PN	2	6,25	18,75	1,75	1,25	0,75
		Lemak	2	3,75	37,5	0	0	3,75
		Buah	13	41,25	18,75	4,5	0	0
		Sayur	11	75	37,5	7,5	2,25	0
	Siang	KH	20	120	210	48	4,8	0
		PH	4	16	20	0	2,8	0,8
		PN	10	22	30	2,8	2	1,2
		Lemak	5	48	60	0	0	6
		Buah	11	63	30	7,2	0	0
		Sayur	4	120	30	6	1,2	0

	Malam	KH	25	52,5	183,75	42	4,2	0
		PH	17	31,5	17,5	0	2,45	0,7
		PN	12	17,5	26,25	2,45	1,75	1,05
		Lemak	3	5,25	52,5	0	0	5,25
		Buah	28	21	26,25	6,3	0	0
		Sayur	26	105	0	0	0	0
<b>JUMLAH</b>					<b>1852,5</b>	<b>308,75</b>	<b>52,5</b>	<b>41,05</b>
<b>Kandungan/hari</b>					<b>926,25</b>	<b>154,375</b>	<b>26,25</b>	<b>20,525</b>

**Keterangan:**

KH : Karbohidrat

PH : Protein Hewani

PN : Protein Nabati

L : Lemak

B : Buah

S : Sayuran

Hasil diatas menjelaskan setiap bahan makanan mengandung kandungan gizi Karbohidrat, Protein Nabati, Protein Hewani, Lemak, Buah dan Sayur yang berbeda-beda. Semua kandungan gizi pada makanan dijumlah untuk menghasilkan jumlah kandungan dari gizi tersebut dan dihitung nilai kandungan/hari.

### 3. Menghitung Penalti Gizi

Langkah selanjutnya yaitu menghitung penalti gizi untuk mengetahui jumlah kebutuhan gizi yang tidak terpenuhi dari masing-masing kandungan gizi pada kalori, karbohidrat, protein dan lemak. Kebutuhan gizi masing-masing balita berdasarkan Tabel 4.3 hasil kebutuhan gizi balita dan hasil perhitungan kandungan gizi pada makanan pada Tabel 4.13.

**Tabel 4.13 Hasil Perhitungan Kandungan Gizi**

kandungan Gizi	KKAL	KH	KP	KL
<b>JUMLAH</b>	<b>1852,5</b>	<b>308,75</b>	<b>52,5</b>	<b>41,05</b>
<b>Kandungan/hari</b>	<b>926,25</b>	<b>154,375</b>	<b>26,25</b>	<b>20,525</b>

Menghitung penalti gizi menggunakan Persamaan 2.14. contoh perhitungan penalti:

$$\text{penalti Kalori} = 1460 - 926,25 = 533,75 \text{ gram}$$

$$\text{penalti KH} = 1100 - 154,375 = 101,125 \text{ gram}$$

Hasil perhitungan penalti pada *parent 1* selama 2 hari ditunjukkan pada Tabel 4.14.

**Tabel 4.14 Hasil Perhitungan Penalti Gizi**

Parent	KKAL	KH	KP	KL	Total Penalti
<b>P1</b>	533,75	101,125	10,25	11,9194	657,04

**Keterangan:**

- KKAL : Kandungan Kalori  
 KH : Kandungan Karbohidrat  
 KP : Kandungan Protein  
 KL : Kandungan Lemak

**4. Hitung Harga Bahan Makanan**

Perhitungan harga bahan makanan menggunakan Persamaan 2.15. Contoh perhitungan harga bahan makanan pada *p1*:

$$\text{Tepung Singkong} = \left(\frac{37,5}{1000}\right) \times 21000 = \text{Rp. } 281,25$$

$$\text{Bakso} = \left(\frac{2,5}{1000}\right) \times 65000 = \text{Rp. } 450$$

$$\text{Keju} = \left(\frac{10}{1000}\right) \times 15000 = \text{Rp. } 350$$

Tabel 4.15 merupakan hasil perhitungan harga bahan makanan berdasarkan jumlah berat bahan makanan dan harga setiap bahan makanan.

**Tabel 4.15 Hasil Perhitungan Harga Bahan Makanan**

Hari ke-	Waktu makan	Jenis bahan makanan	P1	Nama BM	Berat Makanan	Harga
1	Pagi	KH	27	Tepung singkong	37,5	281,25
		PH	6	Bakso	2,5	450
		PN	9	Keju	10	350
		Lemak	4	Minyak Kelapa Sawit	3,75	187,5
		Buah	23	Kedondong	35,625	641,25
		Sayur	20	Taoge Kacang Hijau	75	225
	Siang	KH	29	Jagung Segar	36	900
		PH	23	Hati Ayam	14	784
		PN	1	Tahu	8	160
		Lemak	7	Mayonaise	6	162
		Buah	29	Kedondong	24	672
		Sayur	40	Daun Katuk	120	2160
	Malam	KH	23	Roti putih	52,5	420
		PH	27	Putih telur ayam	10,5	241,5
		PN	3	Keju	7	175
		Lemak	2	Santan Kelapa	5,25	540,75
		Buah	32	Jambu Air	28,875	346,5
		Sayur	14	Brokoli	105	1365
2	Pagi	KH	16	Roti putih	52,5	2362,5

		<b>PH</b>	12	Bakso	7,5	225
		<b>PN</b>	2	Tempe	6,25	75
		<b>Lemak</b>	2	Keju Krim	3,75	386,25
		<b>Buah</b>	13	Gula	41,25	618,75
		<b>Sayur</b>	11	Daun Lobak	75	1125
	<b>Siang</b>	<b>KH</b>	20	Roti putih	120	5760
		<b>PH</b>	4	Ikan Gabus	16	384
		<b>PN</b>	10	Petai Segar	22	242
		<b>Lemak</b>	5	Keju Krim	48	1200
		<b>Buah</b>	11	Semangka	63	945
		<b>Sayur</b>	4	Kacang Kapri	120	2400
	<b>Malam</b>	<b>KH</b>	25	Singkong	52,5	735
		<b>PH</b>	17	Bakso	31,5	1102,5
		<b>PN</b>	12	Tempe	17,5	210
		<b>Lemak</b>	3	Minyak Kelapa	5,25	420
		<b>Buah</b>	28	Pepaya	21	420
		<b>Sayur</b>	26	Nangka Muda	105	2940
<b>TOTAL HARGA</b>						<b>31612,75</b>

## 5. Menghitung Variasi Bahan Makanan

Variasi bahan makanan digunakan untuk mengetahui jumlah keberagaman variasi jenis bahan makan dalam satu waktu. Nilai variasi didapat berdasarkan keragaman dari setiap jenis bahan makanan. Jika nama makanan sama maka variasi bernilai 0 begitu sebaliknya jika nama bahan makanan dalam satu jenis yang sama memiliki nama yang berbeda maka nilai variasi bernilai 1. Proses perhitungan bahan makanan dilakukan dengan cara menghitung setiap variasi bahan makanan pada masing-masing kandungan gizi. Tabel 4.16 menunjukkan hasil perhitungan variasi bahan makanan.

**Tabel 4.16 Hasil Perhitungan Variasi Bahan Makanan**

Hari ke-	Waktu makan	Jenis bahan makanan	Nama BM	Variasi
1	Pagi	<b>KH</b>	Tepung Terigu	1
		<b>PH</b>	Ikan Gabus	1
		<b>PN</b>	Oncom	1
		<b>Lemak</b>	Mentega	1
		<b>Buah</b>	Nangka	1
		<b>Sayur</b>	Gambas	1
	Siang	<b>KH</b>	Kerupuk Udang/Ikan	1
		<b>PH</b>	Udang Segar	1

		<b>PN</b>	Kacang Hijau	1	
		<b>Lemak</b>	Minyak Kelapa	1	
		<b>Buah</b>	Pisang Raja	1	
		<b>Sayur</b>	Slada Air	1	
	<b>Malam</b>	<b>KH</b>	Tepung beras	1	
		<b>PH</b>	Hati Ayam	1	
		<b>PN</b>	Kacang Merah	1	
		<b>Lemak</b>	Minyak Jagung	1	
		<b>Buah</b>	Sawo	1	
		<b>Sayur</b>	Daun Koro	1	
	<b>2</b>	<b>Pagi</b>	<b>KH</b>	Roti warna coklat	1
			<b>PH</b>	Ikan Mujaer	1
<b>PN</b>			Kacang Kedelai	1	
<b>Lemak</b>			Minyak Jagung	1	
<b>Buah</b>			Jeruk Manis	1	
<b>Sayur</b>			Daun Katuk	1	
<b>Siang</b>		<b>KH</b>	Tape beras ketan	1	
		<b>PH</b>	Daging Ayam tanpa kulit	1	
		<b>PN</b>	Petai Segar	1	
		<b>Lemak</b>	Santan Kelapa	1	
		<b>Buah</b>	Jeruk Bali	1	
		<b>Sayur</b>	Brokoli	1	
<b>Malam</b>		<b>KH</b>	Tepung Sagu	1	
		<b>PH</b>	Kerang	1	
		<b>PN</b>	Tempe	1	
		<b>Lemak</b>	Minyak Kacang Kedelai	1	
		<b>Buah</b>	Pisang Mas	1	
		<b>Sayur</b>	Kangkung	1	
<b>Total Variasi</b>				<b>36</b>	

## 6. Menghitung Nilai *Fitness*

Menghitung nilai *fitness* digunakan untuk mengetahui nilai kebugaran masing-masing individu. Menghitung nilai *fitness* berdasarkan Persamaan 2.12. Hasil perhitungan *fitness* pada Tabel 4.17.

**Tabel 4.17 Hasil Perhitungan Nilai *Fitness***

<b>Individu</b>	<b>Harga</b>	<b>Variasi</b>	<b>Penalti</b>	<b><i>Fitness</i></b>
<b>P1</b>	31613	36	657,04	<b>64,8525</b>
<b>P2</b>	50343	30	554,74	52,8899
<b>P3</b>	38324	33	641,34	58,1853
<b>P4</b>	43549	34	614,92	56,2247
<b>P5</b>	34398	35	643,34	<b>62,1155</b>
<b>P6</b>	31842	35	657,04	<b>64,125</b>

#### **4.5.8 Seleksi**

Seleksi digunakan untuk menentukan individu terbaik yang lolos pada generasi berikutnya. *Elitism selection* dilakukan dengan cara menggabungkan seluruh nilai *fitness* dari masing-masing *parent* dan *child*, kemudian mengurutkan nilai *fitness* dari yang terbesar sampai terkecil. Selanjutnya nilai *fitness* terbesar dipilih dan lolos pada generasi berikutnya. Berdasarkan nilai *fitness* pada Tabel 4.17 maka diambil 3 individu baru sesuai dengan jumlah populasi. Tabel 4.18 merupakan hasil seleksi.

**Tabel 4.18 Populasi Baru**

<b>Generasi (1)</b>	<b>Generasi (2)</b>	<b><i>Fitness</i></b>
P1	P1	64,8525
P6	P2	64,125
P5	P3	62,1155

#### **4.6 Perancangan Antarmuka**

Perancangan antarmuka digunakan untuk mempermudah pengguna dalam menggunakan sistem. Halaman awal sistem merupakan tampilan awal ketika pengguna masuk ke sistem. Pada halaman ini akan menampilkan form balita dan parameter algoritme genetika yang nantinya nilai tersebut merupakan inputan dari pengguna. Selanjutnya form sebelah kanan akan menampilkan hasil rekomendasi daftar bahan makanan. Hasil akhir sistem akan disimpan dalam bentuk file txt. Tampilan sistem optimasi gizi pada makanan balita menggunakan algoritme genetika dapat dilihat pada Gambar 4.17.

**SISTEM OPTIMASI GIZI PADA BAHAN MAKANAN BALITA  
MENGUNAKAN ALGORITME GENETIKA**

<p style="text-align: center;"><b>Data Balita</b></p> <p>Nama <input style="width: 80%;" type="text"/></p> <p>BB <input style="width: 80%;" type="text"/></p> <p>Umur <input style="width: 80%;" type="text"/></p> <p style="text-align: center;"><b>Parameter Algoritma Genetika</b></p> <p><i>Cr</i> <input style="width: 80%;" type="text"/></p> <p><i>Mr</i> <input style="width: 80%;" type="text"/></p> <p><i>Popsize</i> <input style="width: 80%;" type="text"/></p> <p>Generasi <input style="width: 80%;" type="text"/></p> <p>Hari <input style="width: 80%;" type="text"/></p>	<p>Hasil Rekomendasi</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------

Proses
Simpan

**Gambar 4.17 Tampilan Awal Sistem**

## 4.7 Perancangan Pengujian

Perancangan uji coba digunakan untuk merancang evaluasi sistem. Dalam penelitian ini menggunakan beberapa pengujian antara lain:

### 4.7.1 Pengujian Jumlah Populasi

Uji coba ini digunakan untuk mengetahui jumlah populasi yang optimal untuk menghasilkan nilai *fitness* yang terbaik. Pengujian jumlah populasi dilakukan sebanyak 10 kali dengan kelipatan jumlah populasi yaitu 10. Tabel 4.19 menunjukkan rancangan uji coba jumlah populasi.

**Tabel 4.19 Uji Coba Jumlah Populasi**

Populasi	Nilai <i>Fitness</i>										Rata-rata <i>Fitness</i>
	1	2	3	4	5	6	7	8	9	10	
10											
20											
30											
40											
50											



60											
70											
80											
90											
100											

#### 4.7.2 Pengujian Kombinasi Nilai *Crossover Rate* (*cr*) dan Nilai Mutasi Rate (*mr*)

Pengujian ini digunakan untuk mengetahui kombinasi nilai *cr* dan *mr* yang optimal untuk menghasilkan nilai *fitness* terbaik. Pengujian ini dilakukan sebanyak 10 kali percobaan. Tabel 4.20 menunjukkan uji coba kombinasi nilai *crossover* dan mutasi .

**Tabel 4.20 Uji Coba Kombinasi Nilai *Cr* dan *Mr***

		Nilai <i>Fitness</i>										Rata-rata <i>Fitness</i>
<i>Cr</i>	<i>Mr</i>	1	2	3	4	5	6	7	8	9	10	
0	1											
0,1	0,9											
0,2	0,8											
0,3	0,7											
0,4	0,6											
0,5	0,5											
0,6	0,4											
0,7	0,3											
0,8	0,2											
0,9	0,1											
1	0											

#### 4.7.3 Pengujian Jumlah Generasi

Pengujian banyak generasi digunakan untuk mengetahui banyaknya generasi yang optimal untuk menghasilkan nilai *fitness* terbaik. Percobaan jumlah generasi dilakukan sebanyak 10 kali dengan kelipatan 10. Tabel 4.21 menunjukkan tabel uji coba jumlah generasi.

**Tabel 4.21 Uji Coba Banyak Generasi**

<b>Generasi</b>	<b>Nilai <i>Fitness</i></b>										<b>Rata-rata <i>Fitness</i></b>
	1	2	3	4	5	6	7	8	9	10	
10											
20											
30											
40											
50											
60											
70											
80											
90											
100											