PENENTUAN KOMPOSISI MENU MAKANAN UNTUK PENDERITA *DIABETES MELLITUS* MENGGUNAKAN ALGORITMA GENETIKA

SEMINAR HASIL

Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Dalam Bidang Ilmu Komputer

Oleh :
Resthy Kushardiana
0810960060-96

PROGRAM STUDI ILMU KOMPUTER
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2013

PENENTUAN KOMPOSISI MENU MAKANAN UNTUK PENDERITA *DIABETES MELLITUS* MENGGUNAKAN ALGORITMA GENETIKA

Resthy Kushardiana[1] ; Muh. Arif Rahman[2]; Achmad Ridok[3]
Program Studi Ilmu Komputer, Program Teknologi Informasi dan Ilmu Komputer
Universitas Brawijaya, Jl. Veteran No 8 Malang , 65145, Indonesia
E-mail : resthy.kushardiana@gmail.com[1] , m_arif@ub.ac.id[2] , acridokg@gmail.com[3]

ABSTRAK

Indonesia merupakan salah satu negara dengan jumlah penderita diabetes terbanyak di dunia. Sampai dengan tahun 2008, Departemen Kesehatan Republik Indonesia mencatat Indonesia sebagai peringkat ke-4 di dunia yaitu 5,6%. Pada penyakit Diabetes Mellitus, diet merupakan pengobatan paling mendasar yaitu dengan mengendalikan kadar gula darah melalui penekanan total kalori yang dimakan dan memperhatikan jumlah karbohidrat serta kombinasi dalam penyusunan menu makanannya. Penyusunan komposisi menu makanan termasuk dalam suatu permasalahan kombinatorik dan dapat diselesaikan dengan algoritma genetika. Sesuai struktur umum dari algoritma genetika, data bahan makanan dikombinasikan menjadi individu yang merupakan 1 kesatuan menu dalam sehari dan akan dievaluasi menggunakan sebuah fungsi evaluasi (fungsi fitness). Untuk mendapatkan variasi fitness individu maka akan dilakukan kombinasi nilai probabilitas crossover dan probabilitas mutasi. Dari hasil pengujian yang dilakukan, didapatkan bahwa nilai crossover yang terlalu rendah atau terlalu tinggi mengakibatkan fitness yang buruk. Sedangkan jumlah populasi dan generasi yang semakin tinggi, cenderung menghasilkan fitness yang baik. Dari beberapa hasil pengujian, dapat dihasilkan solusi lokal optimum dengan nilai fitness maksimum yaitu 1,0 sehingga sistem dikatakan dapat diimplementasikan untuk menghasilkan kombinasi menu dan berat makanan dalam sehari sesuai dengan kebutuhan kalori tubuh penderita Diabetes Mellitus.

***Kata kunci** : Algoritma Genetika, Diabetes Mellitus, menu makanan*

I. PENDAHULUAN

Masalah gizi merupakan salah satu masalah dalam dunia kesehatan. Seringkali kebutuhan gizi seseorang tidak sesuai dengan kebutuhan tubuhnya, bisa berlebih bisa kekurangan. Keadaan gizi yang tidak seimbang bisa menyebabkan gangguan metabolisme dalam tubuh baik dalam keadaan sehat maupun sakit. Pedoman Umum Gizi Seimbang (PUGS) adalah pedoman dasar tentang gizi seimbang yang disusun sebagai penuntun pada perilaku konsumsi makanan di masyarakat secara baik dan benar. PUGS menganjurkan agar 60-75% kebutuhan energi diperoleh dari karbohidrat, 10-15% dari protein dan 10-25% dari lemak [1] .

Selain untuk orang yang sehat, pemenuhan kebutuhan gizi seimbang juga diperlukan untuk seseorang dalam kondisi sakit. Indonesia merupakan salah satu negara dengan jumlah penderita diabetes terbanyak di dunia. Sampai dengan tahun 2008, Departemen Kesehatan Republik Indonesia mencatat Indonesia sebagai peringkat ke-4 di dunia yaitu 5,6% , itupun hanya jumlah data yang tercatat, belum termasuk data yang dimungkinkan tidak tercatat. *Diabetes Mellitus* dapat dikendalikan dengan cara menurunkan kadar gula darah, salah satu caranya adalah melakukan diet yang merupakan dasar utama pengobatan diabetes dengan cara menekan total kalori yang dimakan. Diet untuk penderita diabetes pun harus dikelola secermat mungkin. Prinsip terpenting adalah jumlah karbohidrat dan kombinasi pemenuhan gizi lain yang dimakan harus stabil dan harus dimakan pada saat yang sama [2] .

Konsep keilmuan pakar kesehatan dalam penyusunan menu makanan dapat dimasukkan ke dalam komputer untuk menyusun suatu menu makanan yang optimal pula. Untuk keperluan penyusunan dengan metode ini, dapat digunakan algoritma genetika. Pada algoritma genetika terdapat beberapa istilah evolusi seperti populasi, individu dan generasi. Individu merepresentasikan satu solusi. Setiap individu nantinya akan dievaluasi dengan suatu fungsi yang disebut fungsi *fitness*, dimana fungsi *fitness* tersebut dicari secara matematis untuk mencapai solusi yang optimal. Dalam algoritma genetika juga terdapat 2 jenis transformasi yaitu mutasi, yang menghasilkan individu baru dari perubahan satu individu saja dan *crossover*, yang akan menghasilkan individu baru dengan mengkombinasikan dua individu [3].

Diet yang dilakukan melalui penyusunan komposisi menu makanan termasuk dalam suatu permasalahan kombinatorik. Pada penelitian ini, kombinasi komposisi menu makanan akan dihitung dan dikombinasikan sesuai dengan jumlah kalori yang dihasilkan dari perhitungan kalori seorang penderita diabetes. Dalam pengkombinasian tersebut juga akan disediakan bahan makanan pokok beserta bahan makanan pengganti makanan pokok yang memiliki kandungan gizi yang setara. Secara garis besar, permasalahan kombinatorik ini terlihat dalam penyusunan gen dalam kromosom untuk mencari sebuah solusi yaitu suatu kesatuan menu makanan yang optimal sesuai dengan kondisi tubuh.

Sebelumnya, sudah pernah juga dilakukan penelitian tentang penyusunan menu makanan dengan algoritma genetika antara lain, pemecahan masalah kombinatorik dengan menggunakan algoritma genetika untuk menghitung komposisi bahan pangan dalam rangka penentuan diet bagi pendertita gangguan ginjal dengan menggunakan 400 data bahan pangan dan hasil peneitian tersebut juga mengatakan bahwa algoritme genetika dapat digunakan dalam pengkombinasian menu makanan [4].

II. METODOLOGI PENELITIAN

2.1 Data

Data yang digunakan pada penelitian ini adalah data bahan makanan dalam ukuran 100 gr berdasarkan Tabel Komposisi Pangan Indonesia yang diterbitkan oleh Persatuan Ahli Gizi Indonesia. Selain data bahan makanan, juga digunakan data pasien Diabetes Mellitus dari Rumah Sakit Baptis Kediri selama periode Januari-April 2012. Dari kedua data inilah akan dihasilkan sebuah menu sesuai dengan kebutuhan kalori pasien. Dari banyak penelitian ternyata indeks yang paling berpengaruh terhadap AMB (Angka Metabolisme Basal) atau kebutuhan kalori seseorang adalah berat badan menurut umur, sehingga digunakan rumus dari FAO/WHO/UNU/1985 yang menaksir AMB dari berat badan seperti pada tabel 1 berikut [5].

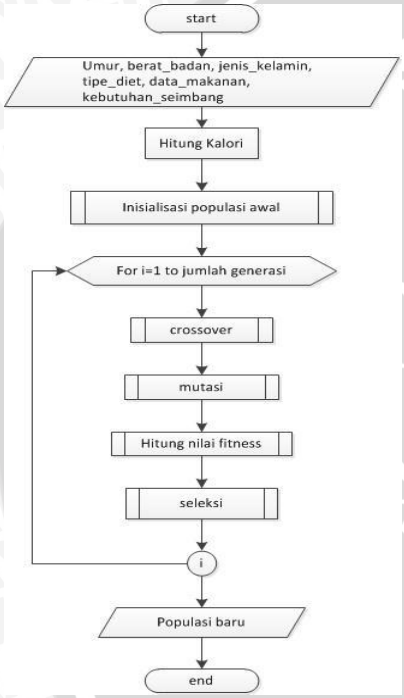
Tabel 1 Rumus Regresi Linier, FAO/WHO/UNU/1985

Kelompok Umur (tahun)	AMB (Kkal/hari)	
	Laki-laki	Perempuan
0-3	60.9 B - 54	61 B - 51
3-10	22.7 B + 495	22.5 B + 499
10-15	17.5 B + 651	12.2 B + 746
18-30	15.3 B + 679	14.7 B + 496
30-60	11.6 B + 879	8.7 B + 829
>60	13.5 B + 487	10.5 B + 596

Keterangan :
B = berat badan dalam satuan kilogram

2.2 Model Algoritma Genetika

Algoritma genetika merupakan algoritma yang dikembangkan dari proses pencarian solusi menggunakan pencarian acak, hal ini terlihat pada proses pembangkitan populasi awal yang menyatakan sekumpulan solusi yang bisa dipilih secara acak. Selanjutnya pencarian dilakukan berdasarkan proses-proses teori genetika yang memperhatikan pemikiran bagaimana memperoleh individu yang lebih baik, sehingga dalam proses evolusi diharapkan diperoleh individu yang terbaik [6]. Secara umum, blok diagram dari mekanisme kerja algoritma genetika dapat dilihat dalam gambar 1



Gambar 1 Mekanisme Kerja Algoritma Genetika

2.2.1 Inisialisasi Individu

Individu terdiri dari gen-gen yang merupakan suatu pengkodean yang digunakan untuk merepresentasikan masalah kombinasi menu makanan untuk penderita *Diabetes Mellitus*. Inisialisasi individu dapat dilihat dalam gambar 2

Makan Pagi				Makan Siang				Makan Malam				Peleng-kap		
S	N	H	Y	S	N	H	Y	S	N	H	Y	P 1	P 2	P 3

Gambar 2 Inisialisasi Individu

Dalam setiap makan pagi, siang, malam dan pelengkap terdapat komponen Karbohidrat, Lemak dan Protein. Ilustrasinya dapat dilihat pada gambar 3

S			N			H			Y			P		
K 1	L 1	R 1	K 2	L 2	R 2	K 3	L 3	R 3	K 4	L 4	R 4	K 5	L 5	R 5

Gambar 3 Ilustrasi komponen dalam setiap gen makanan

Ket :
S = Makanan Pokok H = Lauk Hewani
N = Lauk Nabati Y = Sayur
P1,P2,P3 = Pelengkap
K1 .. K5 = Jumlah karbohidrat
L1 .. L5 = Jumlah lemak
R1 .. R5 = Jumlah protein

Dalam proses pembangkitan individu tersebut, akan dilakukan pembangkitan secara acak untuk jenis dan bobot makanannya. Untuk proses pembobotan makanan sendiri akan dibangkitkan bilangan random 0 .. 1 lalu akan dihitung bobot sebenarnya . Langkah-langkah penghitungan bobot makanan sebenarnya adalah sebagai berikut :

1. Untuk masing-masing gen dalam individu, bangkitkan jenis makanan secara acak.
2. Bangkitkan bilangan acak [0 .. 1] dalam tiap gen tersebut sebagai inisialisasi bobot makanan
3. Hitung bobot sebenarnya dengan aturan rentang untuk makanan pokok adalah 0-180 gr, untuk sumber lauk nabati 0-50 gr, sumber lauk hewani 0-80 gr, sayur 0-200 gr dan makanan pelengkap 0-150 gr
4. Hitung berat masing-masing komponen penyusun (karbohidrat, lemak dan protein) makanan sesuai dengan berat sebenarnya.

2.2.2 *Crossover*

Untuk pemilihan *parent* dalam *crossover* digunakan nilai Probabilitas *Crossover* (Pc). Pertama akan dibangkitkan nilai Pc secara *random* untuk setiap individu, lalu akan diberikan nilai input Pc misal 0,8 berikutnya akan dilakukan pengecekan individu mana yang memiliki nilai Pc < 0.8 . Individu-individu itulah yang menjadi *parent* dalam proses *crossover*.

Dalam operator *crossover*, dipilih metode *simple arithmetic crossover* yaitu metode persilangan dimana keturunan (*offspring*) didapat dengan melakukan operasi pada nilai gen dari *parent* pada titik tertentu sesuai dengan persamaan 1 dan 2.

$$x'_1 = \langle x_1, \dots, x_k, y_{k+1} \cdot a + x_{k+1} \cdot (1-a), \dots, y_q \cdot a + x_q \cdot (1-a) \rangle \quad (1)$$

$$x'_2 = \langle y_1, \dots, y_k, x_{k+1} \cdot a + y_{k+1} \cdot (1-a), \dots, x_q \cdot a + y_q \cdot (1-a) \rangle \quad (2)$$

Dimana :
X = *parent* 1 ; Y= *parent* 2 ; X'1 = *child* 1; X'2 = *child* 2 ;
α = nilai *random* antara 0 .. 1

Penjelasan langkah-langkah proses *simple arithmetic crossover* yang digunakan adalah sebagai berikut:

1. Masukkan dua *parent* yang akan di-*crossover*.
2. Tentukan bilangan *random* sebagai titik potong antara 0 sampai sepanjang kromosom pada masing-masing *parent*.
3. Untuk setiap gen pada kromosom *parent*, lakukan:
   a. Jika indeks kromosom terletak sebelum titik potong, salin nilai gen *parent* untuk diberikan pada anakan (*offspring*).
   b. Jika indeks kromosom berada pada titik setelah titik potong, lakukan persamaan 1 dan 2 untuk mendapatkan nilai baru yang akan diberikan kepada anakan (*offspring*).

2.2.3Mutasi

Untuk studi kasus kali ini akan digunakan metode *random mutation* Dalam sebuah populasi semua individu bisa mengalami mutasi. Tingginya peluang mutasi akan ditentukan oleh nilai peluang mutasi (Pm) yang bernilai nol sampai dengan satu. Nilai Pm terdapat dalam tiap individu dan tiap gen yang akan dibangkitkan secara *random*. Jika nilai Pm dalam gen kurang dari nilai Pm yang ditentukan, maka akan dilakukan mutasi gen.

2.2.4 Fungsi *Fitness*

Fungsi *fitness* merupakan fungsi yang digunakan untuk mengevaluasi setiap individu yang terdapat dalam sebuah populasi. Fungsi *fitness* dalam studi kasus ini merupakan suatu fungsi maksimasi, sehingga semakin besar nilai *fitness*, maka individu akan semakin baik. Dalam studi kasus ini, nilai *fitness* tersebut juga akan dihitung dengan menggunakan penalti, dimana penalti merupakan suatu cara untuk memberikan nilai terhadap pelanggaran yang dilakukan oleh setiap individu. Fungsi *fitness* yang digunakan akan ditunjukkan dalam persamaan 3.

$$\textbf{fitness} = \frac{1}{1 + \sum_{n=1}^{3} P_n} \qquad (3)$$

Dengan :
Pn = penalti ke-n

Fungsi *fitness* akan dihitung berdasarkan komponen penyusun bahan makanan, yaitu Karbohidrat, Lemak dan Protein. Berdasarkan tipe diet untuk penderita *Diabetes Mellitus*, maka terdapat 8 jenis tipe diet beserta kebutuhan komponen seimbangnya seperti pada tabel 2 berikut [5].

Tabel 2 Kebutuhan seimbang berdasarkan tipe diet

Jenis Diet	Energi (kkal)	Protein (gr)	Lemak (gr)	Karbo-hidrat (gr)
I	1100	43	30	172
II	1300	45	35	192
III	1500	51,5	36,5	235
IV	1700	55,5	36,5	275
V	1900	60	48	299
VI	2100	62	53	319
VII	2300	73	59	369
VIII	2500	80	62	396

Dari kebutuhan seimbang di atas, dapat dibuat batasan minimum dan maksimum yang masih diperbolehkan oleh ahli gizi pihak rumah sakit seperti dalam tabel 3

Tabel 3 Pembagian jumlah kebutuhan tiap komponen karbohidrat, lemak dan protein dalam sehari sesuai tipe diet

Jenis Diet	Energi (kkal)	Kebutuhan	Protein (gr)	Lemak (gr)	Karbo-hidrat (gr)
I	1100	Pagi	9-10	6-8	34-36
		Siang	13-14	9-11	50-52
		Malam	10-12	7,5-9	42-44
		Pelengkap	10-12	7,5-9	42-44
II	1300	Pagi	11-12	8-10	37-39
		Siang	13-15	10,5-12	56-58
		Malam	10-13	8-10	46-49
		Pelengkap	10-13	8-10	46-49
III	1500	Pagi	9-11	6-8	45-49
		Siang	15-16	10-12	69-72
		Malam	12-14	9,125	57-60
		Pelengkap	12-14	9,125	57-60
IV	1700	Pagi	10-12	6-8	50-55
		Siang	15-17	9-11	80-84
		Malam	12-14	8-10	67-70
		Pelengkap	12-14	8-10	67-70
V	1900	Pagi	10-13	8-10	56-61
		Siang	17-19	13-15	80-93
		Malam	14-16	11-14	70-80
		Pelengkap	14-16	11-14	70-80
VI	2100	Pagi	11-13,5	9-11	61,5-68
		Siang	17-19	14-17	90-100
		Malam	14-16	12-15	73-85
		Pelengkap	14-16	12-15	73-85
VII	2300	Pagi	13-15	10-13	70-80
		Siang	19-22	16-18	100-110

		Malam	17-19	14-16	89-100
		Pelengkap	17-19	14-16	89-100
VIII	2500	Pagi	15,5-16,5	11-13	70-81
		Siang	22-26	18-19,5	110-121
		Malam	18-21	14-16	90-100
		Pelengkap	18-21	14-16	90-100

Nilai *fitness* didapat dari total penalti yang bergantung pada selisih tiap komponen sesuai dengan persamaan 3. Dalam kasus ini, digunakan nilai *absolute* untuk hasil perhitungan selisihnya karena baik kekurangan atau kelebihan nutrisi akan dianggap sama-sama buruk.

$$selisih\ x_i y_j = |kebutuhan\ seimbang\ x_i y_j - nilai x_i y_j| \quad (4)$$

Dengan :
x_i = nilai komponen penyusun (karbohidrat, lemak, protein), i = 0 .. 2
y_j= posisi ketika komponen tersebut diberikan (pagi, siang, malam, pelengkap), j = 0 .. 3

Tabel 4 Skor untuk pelanggaran aturan

No	Aturan	Penalti
1	Jika kekurangan atau kelebihan karbohidrat dari jumlah yang diperbolehkan pada makanan pagi, siang, malam, pelengkap	$\sum_{j=0}^{3} selisih\ x_{karbohidrat} y_j * 5$
2	Jika kekurangan atau kelebihan protein dari jumlah yang diperbolehkan pada makanan pagi, siang, malam, pelengkap	$\sum_{j=0}^{3} selisih\ x_{protein} y_j$
3	Jika kekurangan atau kelebihan lemak dari jumlah yang diperbolehkan pada makanan pagi, siang, malam, pelengkap	$\sum_{j=0}^{3} selisih\ x_{lemak} y_j$

2.2.5 Seleksi

Proses seleksi digunakan untuk memilih individu-individu yang akan digunakan dalam *iterasi* atau generasi berikutnya. Metode seleksi yang digunakan adalah *rank based fitness selection.* Dalam proses seleksi ini, digunakan semua individu awal, ditambah dengan individu baru hasil *crossover* dan mutasi dimana akan dilihat dari nilai *fitness* nya dan diurutkan berdasarkan nilai *fitness* tertinggi lalu akan diambil individu sesuai dengan jumlah individu pada populasi awal.

III. Hasil dan Pembahasan

Setelah dilakukan tahap pengujian, berikutnya akan dianalisa dan dibahas mengenai hasil pengujian tersebut. Dalam sistem ini dilakukan 3 macam pengujian , yaitu pengujian pengaruh perubahan nilai Pc dan Pm terhadap nilai *fitness* , pengaruh jumlah generasi terhadap nilai *fitness* dan pengaruh ukuran populasi terhadap nilai *fitness* .

Pengujian pertama, dilakukan terhadap pengaruh nilai Pc dan Pm terhdap nilai *fitness* . Dalam pengujian ini, dilakukan kombinasi terhadap nilai Pm dan Pc, dengan rentang nilai Pm dan Pc masing-masing antara 0.1-0.9. Jumlah populasi sebanyak 20 dan generasi sebanyak 100. Pada setiap kombinasi Pc dan Pm, dilakukan 5 kali percobaan lalu diambil nilai rata-rata *fitness*-nya. Hasil uji coba nilai Pc dan Pm terhadap nilai *fitness* dapat dilihat pada tabel 5 di lampiran.

Dari tabel 5 dapat dilihat bahwa nilai *fitness* rata-rata terbesar adalah 0,2451526. Nilai ini terdapat pada saat nilai Pc 0.9 dan nilai Pm 0.8 . Adapun grafik pengaruh probabilitas *crossover* dan probabilitas mutasi terhadap nilai *fitness* rata-rata dapat dilihat pada gambar 4 di lampiran.

Berdasarkan grafik perbandingan nilai Pc dan Pm dengan nilai *fitness* rata-rata dari gambar 4, dapat dilihat seiring perubahan nilai probabilitas *crossover* dan probabilitas mutasi maka nilai *fitness* rata-rata yang didapat *fluktuatif*. Nilai *fitness* seperti ini, dimungkinkan terjadi karena proses *random* yang dilakukan pada saat proses *crossover* dan mutasi. Untuk kombinasi nilai Pc dan Pm yang seimbang atau berdekatan, dihasilkan nilai *fitness* rata-rata puncak untuk setiap Pm, hal ini terlihat misalnya pada kombinasi Pm 0.9 dan Pc 0.8, lalu pada Pm 0.6 dan Pc 0.7, dan seterusnya.

Nilai *fitness* rata-rata yang relatif tinggi tersebut dapat disebabkan karena semakin seringnya proses *crossover* terjadi sehingga dimungkinkan diperoleh variasi individu yang semakin banyak. Semakin besar nilai probabilitas *crossover* maka semakin banyak kemungkinan individu yang disilangkan (*crossover*), tentunya akan semakin banyak individu baru yang diciptakan. Dengan demikian peluang untuk didapatkan individu yang memiliki nilai *fitness* yang tinggi akan semakin besar.

Untuk nilai Pm dan Pc yang rendah, dihasilkan nilai *fitness* rata-rata yang buruk. Hal ini disebabkan sedikitnya variasi individu yang terbentuk, sehingga nilai *fitness* yang dihasilkan cenderung seragam. Sedangkan terjadi penurunan nilai *fitness* saat nilai Pm tinggi, bisa disebabkan oleh individu yang baik ikut terkena mutasi dan setelah di

mutasi tidak menjadi solusi yang lebih baik dari keadaan sebelum di mutasi. Selain itu, dapat di lihat pada grafik di atas bahwa pada perubahan nilai peluang mutasi cenderung seragam meskipun mengalami kenaikan. Dengan metode mutasi pada jenis dan bobot makanan cukup berpengaruh pada nilai *fitness* sehingga dimungkinkan bisa menjadi lebih baik.

Pengujian berikutnya adalah pengaruh jumlah generasi terhadap nilai *fitness*. Pengujian dilakukan dengan mengubah nilai generasi dari 10, 30, 50, 80, 100 dan 150 generasi. Pada setiap 1 pengujian jumlah generasi, dilakukan 10 kali percobaan untuk kemudian diambil nilai rata-rata *fitness* nya. Hasil uji coba dapat dilihat pada tabel 6 di lampiran.

Dari tabel 6 dapat dilihat bahwa nilai *fitness* rata-rata paling besar yang dihasilkan adalah 0.4096114 pada saat jumlah generasi sebesar 100. Sedangkan nilai *fitness* rata-rata paling kecil adalah 0.0577386 pada saat jumlah generasi sebesar 10. Pergerakan nilai *fitness* rata-rata terhadap jumlah generasi ditunjukkan pada gambar 5.

Pada gambar 5, dapat dilihat bahwa perbandingan jumlah generasi dan nilai *fitness* rata-rata yang *fluktuatif*. Dalam grafik terlihat nilai *fitness* rata-rata turun pada saat generasi 80 kemudian mengalami kenaikan kembali dan turun pada saat jumlah generasi 150. Nilai yang fluktuatif ini bisa disebabkan oleh *starting point* dalam proses algoritma genetika. Pada setiap satu generasi, akan dilakukan beberapa kali proses *crossover* dan mutasi, dimana pada proses ini individu baru akan dihasilkan. Semakin sering proses *crossover* dilakukan maka individu baru yang terbentuk akan semakin banyak dan variasi *fitness* akan semakin beragam.

Jumlah generasi yang tinggi akan mengakibatkan proses evolusi semakin sering dilakukan sehingga kesempatan untuk terjadinya proses pembentukan individu-individu barupun semakin besar pula. Semakin banyak individu yang dihasilkan, memungkinkan semakin banyak variasi nilai *fitness*nya, sehingga peluang untuk mendapatkan nilai *fitness* yang tinggi akan semakin besar.

Pengujian yang terakhir adalah untuk mengetahui pengaruh ukuran populasi terhadap nilai *fitness*. Dalam pengujian ini, digunakan 5 macam ukuran populasi yaitu 10, 20, 30, 40 dan 50. Pada tiap pengujian populasi, dilakukan 10 kali percobaan lalu diambil nilai rata-rata *fitness* yang kemudian akan dibandingkan dengan setiap nilai *fitness* populasi yang lain. Hasil pengujian akan ditunjukkan dalam tabel 7.

Dari tabel 7 dapat dilihat bahwa nilai *fitness* rata-rata paling besar yang dihasilkan adalah 0.4433587 pada saat ukuran populasi sebesar 50. Sedangkan nilai *fitness* rata-rata paling kecil adalah 0.0150539 pada saat ukuran populasi sebesar 10. Pergerakan nilai *fitness* rata-rata terhadap ukuran populasi digambarkan pada gambar 6 di lampiran.

Dari gambar 6, dapat dilihat bahwa perbandingan jumlah populasi terhadap nilai *fitness* rata-rata cenderung naik, yaitu setiap pertambahan jumlah populasi akan diikuti pula bertambahnya nilai *fitness* . Dalam grafik terlihat nilai *fitness* rata-rata turun pada saat generasi 30 tapi pada akhirnya grafik akan naik kembali. Banyaknya ukuran populasi mempengaruhi jumlah individu baru yang dihasilkan. Dengan adanya proses *crossover* dan proses mutasi dalam ukuran populasi yang besar, memungkinkan jumlah individu baru yang dihasilkan juga semakin besar dan beragam. Tentunya hal ini akan berpangaruh pula terhadap variasi nilai *fitness* yang dihasilkan oleh individu – individu baru. Sehingga peluang untuk mendapatkan nilai *fitness* yang tinggi semakin besar.

IV. Kesimpulan

Berdasarkan uji coba dan hasil yang didapatkan, maka dapat disimpulkan bahwa :

1. Algoritma Genetika dapat digunakan untuk permasalahan penyusunan menu makanan untuk penderita *Diabetes Mellitus* dengan menerapkan pengkodean *floating point*, *simple arithmetic crossover*, *random mutation* dan *rank based fitness selection*.
2. Penyelesaian permasalahan penyusunan menu makanan untuk penderita *Diabetes Mellitus* penerapan algoritma genetika dipengaruhi oleh beberapa parameter algoritma genetika. Pengaruh parameter probabilitas *crossover* menunjukkan bahwa semakin besar nilai probabilitas *crossover* dan diikuti dengan nilai probabilitas mutasi yang seimbang, maka nilai *fitness* cenderung semakin tinggi (baik). Hal tersebut juga terjadi pada jumlah populasi dan jumlah generasi. Pengaruh parameter jumlah populasi dan jumlah generasi cenderung menaikkan nilai *fitness* seiring dengan bertambahnya jumlah populasi dan/atau jumlah generasi.

DAFTAR PUSTAKA

[1] Almatsier, Sunita . 2008 . **Penuntun Diet** . Jakarta : Gramedia Pustaka Utama.

[2] Marilyn, Johnson. 1998 . **Diabetes ; Terapi dan Pencegahannya** . Bandung : Indonesia Publishing House .

[3] Gen, Mitsuo and Cheng, Runwei . 2000 . **Genetic Algorithms & Engineering Optimization** . Canada : John Willey & Sons, Inc.

[4] Uyun, Shofwatun dan Hartati, Sri. 2011 . http://journal.uii.ac.id/index.php/Snati/article/viewFile/2196/2019 . **Penentuan Komposisi Bahan Pangan untuk Diet Penyakit Ginjal dan Saluran kemih dengan Algoritma Genetika** . Seminar Nasional Aplikasi Teknologi Informasi 2011 (SNATI, 2011) . Yogyakarta, 17-18 Juni 2011.

[5] Almatsier, Sunita . 2003 . **Prinsip Dasar Ilmu Gizi** . Jakarta : Gramedia Pustaka Utama.

[6] Basuki, Achmad . 2003 . **ALGORITMA GENETIKA Suatu Alternatif Penyelesaian Permasalahan Searching, Optimasi dan Machine Learning** . http://budi.blog.undip.ac.id . Diakses pada tanggal 1 April 2012 pukul 11.45 WIB

LAMPIRAN

Tabel 5 Hasil Uji Coba Probabilitas *Crossover* dan Probabilitas Mutasi terhadap Nilai *Fitness*

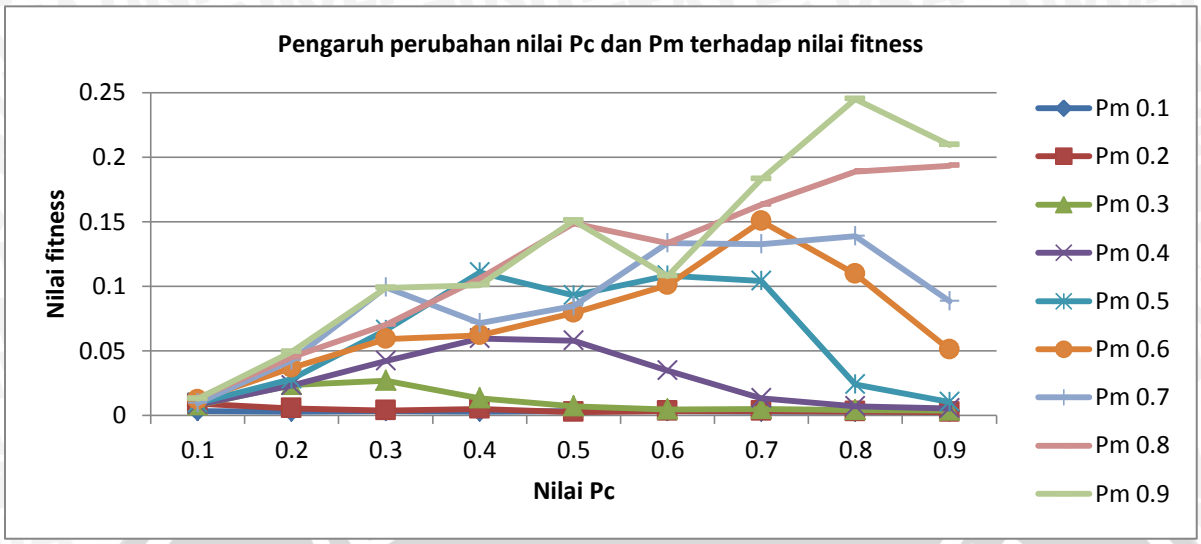
Nilai Pc	Nilai Pm								
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0.1	0.003584	0.0032216	0.0032902	0.0029114	0.003636	0.0034718	0.0032384	0.003231	0.003132
0.2	0.0091054	0.0056954	0.0038782	0.0051146	0.0026594	0.003499	0.0036688	0.0030116	0.0025738
0.3	0.0079972	0.02389328	0.0268912	0.013584	0.0073952	0.0048698	0.0051346	0.0042494	0.0037898
0.4	0.0086086	0.0232106	0.042123	0.059679	0.0579674	0.0349562	0.013495	0.0072196	0.0056784
0.5	0.0097188	0.0282494	0.0663162	0.11087	0.0929644	0.1082426	0.1042414	0.0240236	0.0105234
0.6	0.012024	0.0370048	0.0591122	0.0619686	0.0794706	0.101031	0.1503062	0.1095258	0.0510568
0.7	0.0096572	0.0425552	0.0992936	0.071781	0.0849246	0.1337482	0.1327678	0.13914254	0.0886582
0.8	0.013468	0.0450154	0.0703034	0.1063344	0.1488158	0.1334508	0.1632812	0.1888312	0.193354489
0.9	0.0131096	0.0491542	0.0990862	0.1008602	0.1509602	0.1081422	0.1831924	0.2451526	0.209472

Tabel 6 Hasil Uji Coba Pengaruh Jumlah Generasi terhadap Nilai *Fitness*

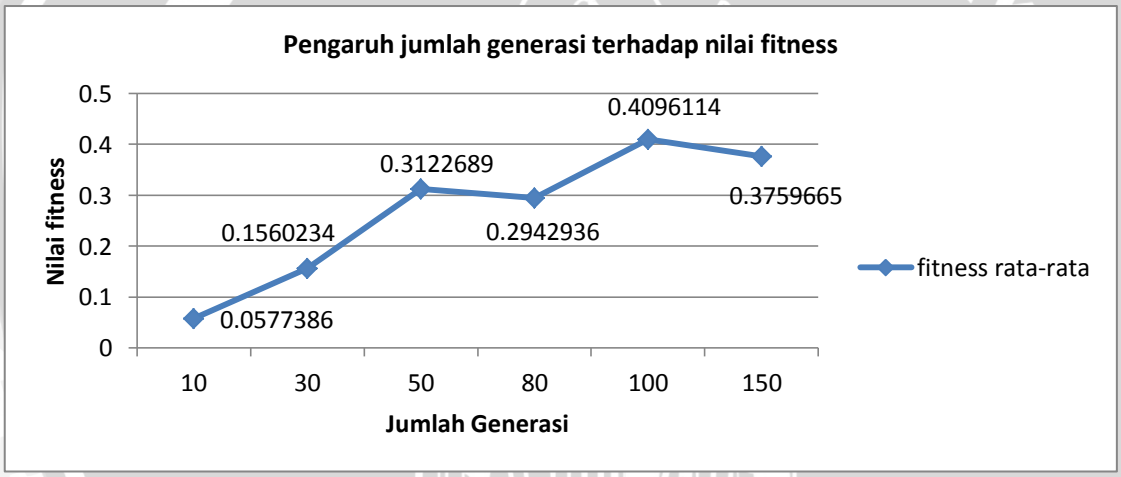
Jumlah Gene-rasi	Percobaan ke-										
	1	2	3	4	5	6	7	8	9	10	*Fitness* rata-rata
10	0.062306	0.05677	0.046251	0.046665	0.040983	0.059543	0.083564	0.053918	0.043962	0.083424	0.0577386
30	0.081519	0.109061	0.437137	0.104892	0.173316	0.155173	0.103411	0.160184	0.090759	0.144782	0.1560234
50	0.279168	0.422694	0.163739	0.176056	0.113429	0.398042	1	0.262537	0.149753	0.157271	0.3122689
80	0.342825	0.845447	0.074185	0.140411	0.178057	0.24474	0.167649	0.158119	0.167366	0.624137	0.2942936
100	0.74625	0.127689	0.179218	0.082062	0.17007	0.142731	1	1	0.153174	0.49492	0.4096114
150	1	0.135192	0.178958	0.22394	0.158922	0.615554	0.127477	0.141716	1	1	0.458175

Tabel 7 Hasil Uji Coba Pengaruh Ukuran Populasi terhadap Nilai *Fitness*

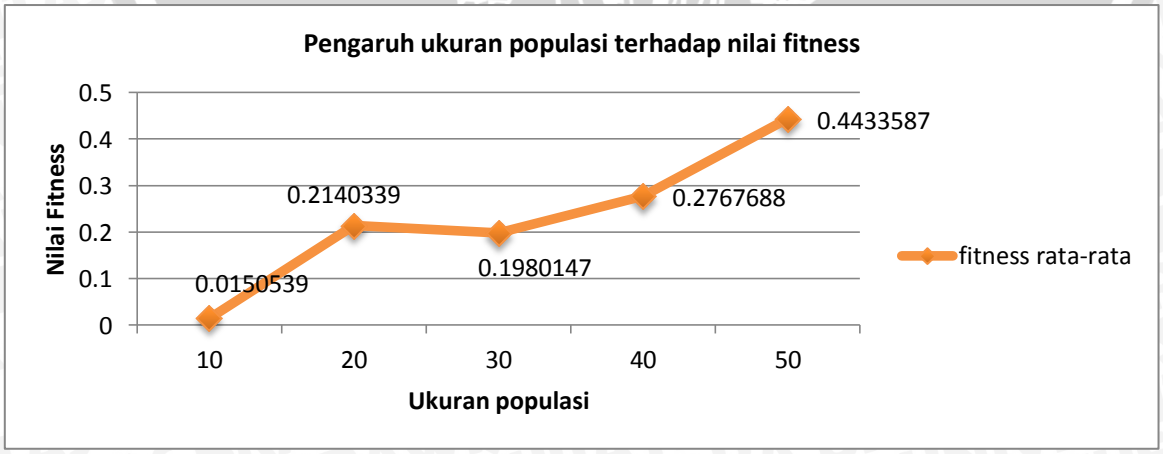
Jum Pop	Percobaan Ke -										*Fitness* Rata - rata
	1	2	3	4	5	6	7	8	9	10	
10	0.01225	0.016083	0.0237	0.012996	0.010271	0.009681	0.007572	0.029036	0.020059	0.008891	0.0150539
20	0.08497	0.152621	0.099825	1	0.106529	0.123517	0.115062	0.249644	0.094411	0.11376	0.2140339
30	0.207804	0.12064	0.105899	0.119563	0.114307	0.140354	0.139951	0.738035	0.182587	0.111007	0.1980147
40	0.100627	0.194559	0.143898	0.139801	0.227892	0.814921	0.128687	0.152053	0.738406	0.126844	0.2767688
50	1	0.359383	0.168304	0.164055	0.457145	0.722304	0.156256	0.217603	0.210204	0.978333	0.4433587



Gambar 4 Grafik pengujian probabilitas *crossover* dan probabilitas mutasi



Gambar 5 Grafik Pengaruh Jumlah Generasi terhadap Nilai *Fitness*



Gambar 6. Grafik Pengaruh Ukuran Populasi Terhadap Nilai *Fitness*