

**SIMULASI 3D OPTIMASI PENATAAN BARANG  
PADA KONTAINER MENGGUNAKAN  
ALGORITMA GENETIKA**

**SKRIPSI**

**Oleh:**

**LAFNIDITA FAROSANTI**

**NIM. 11650048**



**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK  
IBRAHIM  
MALANG  
2015**

**SIMULASI 3D OPTIMASI PENATAAN BARANG PADA KONTAINER  
MENGUNAKAN ALGORITMA GENETIKA**

**SKRIPSI**

Diajukan Kepada:  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang  
Untuk Memenuhi Salah Satu Persyaratan Dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)

**Oleh:**

**Lafnidita Farosanti**

**NIM. 11650048**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG**

**2015**

**SIMULASI 3D OPTIMASI PENATAAN BARANG PADA KONTAINER  
MENGUNAKAN ALGORITMA GENETIKA**

**Oleh:**

**Lafnidita Farosanti**

**NIM. 11650048**

**Telah disetujui oleh:**

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**Fachrul Kurniawan, M.M.T**

**19710722 201101 1 001**

**Hani Nurhayati, M.T**

**19780625 200801 2 006**

**Tanggal,**

**Oktober 2015**

**Mengetahui dan Mengesahkan,**

**Ketua Jurusan Teknik Informatika**

**Dr. Cahyo Crysdiان**

**NIP. 19740424 200901 1 008**

**SIMULASI 3D OPTIMASI PENATAAN BARANG PADA KONTAINER  
MENGUNAKAN ALGORITMA GENETIKA**

**SKRIPSI**

**Oleh:**

**Lafnidita Farosanti**

**NIM. 11650048**

**Telah Dipertahankan di Depan Dewan Penguji Tugas akhir dan  
Dinyatakan Diterima Sebagai Salah Satu Persyaratan Untuk  
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Tanggal, Oktober 2015**

<b>Susunan Dewan Penguji</b>		<b>Tanda Tangan</b>
<b>1. Penguji Utama</b>	<b>: <u>Yunifa Miftachul Arif, M.T</u> NIP. 19830616 201101 1 004</b>	<b>(            )</b>
<b>2. Ketua</b>	<b>: <u>Fatchurrohman, M.Kom</u> NIP. 19700731 200501 1 002</b>	<b>(            )</b>
<b>3. Sekretaris</b>	<b>: <u>Fachrul Kurniawan, M. MT</u> NIP. 19771020 200901 1 001</b>	<b>(            )</b>
<b>4. Anggota</b>	<b>: <u>Hani Nurhayati, M.T</u> NIP. 19780625 200801 2 006</b>	<b>(            )</b>

**Mengetahui dan Mengesahkan,  
Ketua Jurusan Teknik Informatika**

**Dr. Cahyo Crysdian**

**NIP. 19740424 200901 1 008**

## PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini, saya :

Nama : Lafnidita Farosanti

NIM : 11650048

Dengan penuh kesadaran dan rasa tanggung jawab terhadap pengembangan keilmuan, saya menyatakan bahwa skripsi dengan judul :

**“Simulasi 3D Optimasi Penataan Barang Pada Kontainer Menggunakan Algoritma Genetika”**

adalah benar-benar merupakan karya ilmiah yang disusun sendiri, bukan duplikat atau memindah data milik orang lain dan di dalam skripsi ini terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan pada suatu perguruan tinggi, dan sepanjang pengetahuan saya, juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dan teracu dalam naskah ini serta disebutkan dalam daftar pustaka.

Demikian surat pernyataan ini saya buat dengan sebenarnya dan tanpa paksaan dari siapapun.

Malang, 9 September 2015

Hormat Saya,

**Lafnidita Farosanti**

**NIM. 1650048**

## MOTTO

*“Waktu tidak akan bisa berputar kembali, oleh karena itu Manfaatkan Waktu dengan Sebaik-baiknya”*

*“Tidak ada yang bisa kita ubah sebelum kita mengubah diri sendiri. Tak bisa kita mengubah diri sendiri sebelum mengenal diri sendiri. Tak kan kenal pada diri sendiri sebelum mampu menerima diri apa adanya”*



## KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya kepada kita semua sehingga skripsi ini dapat terselesaikan dengan judul : “Simulasi 3D Optimasi Penataan Barang Pada Kontainer Menggunakan Algoritma Genetika”. Sholawat serta salam mari kita tujukan kepada junjungan kita Nabi Muhammad SAW yang telah menuntun kita dari jalan yang gelap menuju jalan yang terang benderang yakni agama islam.

Proses dalam menyelesaikan skripsi ini tentunya tak luput dari bantuan, motivasi, petunjuk, dan bimbingan dari berbagai pihak. Oleh karena itu penulis ingin mengucapkan banyak terimakasih kepada :

1. Bapak Fachrul Kurniawan, M, M.T selaku Dosen Pembimbing I, yang telah memberi masukan, saran, serta kesediaannya dalam membimbing penulis dalam proses mengerjakan skripsi ini.
2. Ibu Hani Nurhayati, M.T selaku Dosen Pembimbing II, yang telah memberikan masukan serta saran dalam menyelesaikan skripsi ini.
3. Bapak Dr. Cahyo Crysdiyan, selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
4. Kedua orang tuaku, M.Mufakhir dan Roisatu Masruhah, yang dengan ketulusan memberikan kasih sayang penuh, motivasi, dan doa restu.
5. Teman-teman seperjuangan TI Angkatan 2011, Wim Sonevel, Mas Risqi Kurnia, yang sangat membantu dalam proses pengerjaan skripsi ini.
6. Sahabat-sahabatku, Maya Siti Maysaroh, Ajeng Gian Aprilliya, Nurul Avivah, Siti Nurjannah, dan semua yang tak dapat disebutkan satu per satu.

Terimakasih karena sudah menemani dan mendampingi dalam mengerjakan skripsi ini.

7. Teman-teman kost ku, adek Kiki, Oni, Ana, dan Sari, yang terus memberiku semangat untuk menyelesaikan skripsi.

Penulis juga menyadari bahwa tidak ada yang sempurna di muka bumi ini melainkan hanya Allah SWT yang memberi jalan dan petunjuk dalam melancarkan proses pengerjaan skripsi ini. Oleh karena itu, penulis mengharapkan kritik serta saran yang membangun demi kemajuan penelitian ini. Penulis juga berharap agar penelitian ini dapat bermanfaat bagi mahasiswa pada khususnya dan pembaca pada umumnya.

Malang, 8 September 2015

Penulis

Lafnidita Farosanti



## DAFTAR ISI

<b>HALAMAN JUDUL</b> .....	i
<b>HALAMAN PERSETUJUAN</b> .....	ii
<b>HALAMAN PENGESAHAN</b> .....	iii
<b>PERNYATAAN SKRIPSI</b> .....	iv
<b>MOTTO</b> .....	v
<b>KATA PENGANTAR</b> .....	vi
<b>DAFTAR ISI</b> .....	viii
<b>DAFTAR GAMBAR</b> .....	xi
<b>DAFTAR TABEL</b> .....	xiii
<b>ABSTRAK</b> .....	xiv
<b>ABSTRACT</b> .....	xv
<b>BAB I PENDAHULUAN</b> .....	1
1.1 Latar Belakang .....	1
1.2 Identifikasi Masalah .....	3
1.3 Batasan Masalah .....	4
1.4 Tujuan Penelitian .....	5
1.5 Manfaat Penelitian .....	5
<b>BAB II KAJIAN PUSTAKA</b> .....	6
2.1 Teori Penunjang .....	6
A. <i>Container</i> atau Peti Kemas .....	6
B. Algoritma Genetika .....	15

2.2 Optimasi dengan GA.....	28
2.3 Simulasi .....	29
2.4 Penelitian Terkait .....	32
<b>BAB III PERANCANGAN SISTEM .....</b>	<b>37</b>
3.1 Desain Penelitian.....	37
3.2 Analisis Data.....	39
3.3 Identifikasi Kebutuhan .....	41
3.4 Perancangan Sistem.....	41
3.4.1 Program Java .....	42
3.4.2 Program Matlab .....	42
3.5 Model Genetika.....	44
3.6 Konsep Penataan Barang .....	54
3.7 Konsep Penentuan Titik Koordinat.....	59
3.8 Desain Database .....	61
3.9 Desain <i>User Interface</i> .....	65
<b>BAB IV IMPLEMENTASI DAN PEMBAHASAN .....</b>	<b>68</b>
4.1 Deskripsi Program.....	68
4.2 Penerapan Aplikasi.....	69
4.2.1 Implementasi Algoritma Genetika.....	69
4.2.2 Proses Penataan Barang pada Kontainer .....	77
4.2.3 Visualisasi 3D Penataan Barang pada Kontainer .....	78
4.3 Uji Coba.....	79
4.4 Analisa Hasil .....	81
4.5 Integrasi Prrogram dalam Islam .....	86

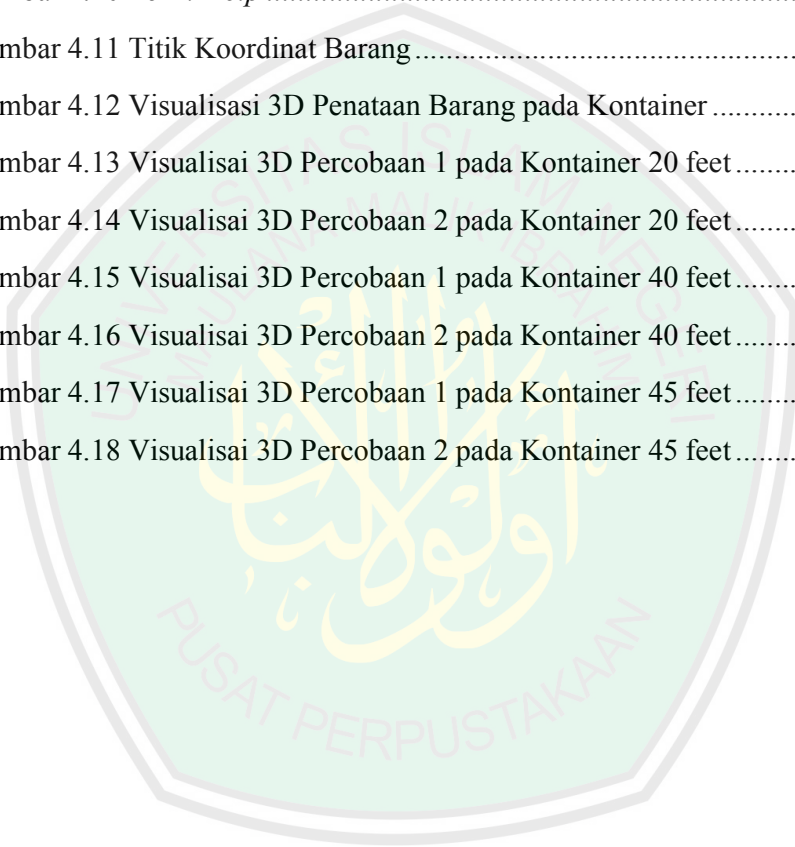
<b>BAB V PENUTUP</b> .....	89
5.1 Kesimpulan .....	89
5.2 Saran .....	90
<b>DAFTAR PUSTAKA</b> .....	91



## DAFTAR GAMBAR

Gambar 2.1 Sistem Pengiriman <i>FCL</i> .....	10
Gambar 2.2 Sistem Pengiriman <i>LCL</i> .....	11
Gambar 2.3 <i>Shipping Mark</i> .....	13
Gambar 2.4 Flowchart Algoritma Genetika .....	17
Gambar 2.5 Contoh <i>Partial-Mapped Crossover (PMX)</i> .....	24
Gambar 2.6 Contoh <i>Order Crossover (OX)</i> .....	25
Gambar 2.7 Contoh <i>Cycle Crossover (CX)</i> .....	25
Gambar 2.8 Posisi Barang pada Kromosom .....	33
Gambar 2.9 Susunan Barang dengan Perbedaan Tinggi Ruang .....	33
Gambar 2.10 Posisi Barang dengan Koordinat Berbeda .....	35
Gambar 3.1 Desain Penelitian Sistem .....	39
Gambar 3.2 <i>Flowchart</i> Sistem .....	43
Gambar 3.3 Siklus Algoritma Genetika oleh David Goldberg .....	44
Gambar 3.4 Proses <i>Crossover</i> .....	52
Gambar 3.5 Proses Mutasi .....	54
Gambar 3.6 Cara Penataan Barang dengan Parameter Tinggi .....	56
Gambar 3.7 Cara Penataan Barang dengan Parameter Panjang .....	57
Gambar 3.8 Peletakan Barang pada Kontainer .....	58
Gambar 3.9 Posisi Barang Jenis Kulkas dalam Kontainer .....	59
Gambar 3.10 Posisi Barang Jenis Televisi dalam Kontainer .....	59
Gambar 3.11 Posisi Barang Jenis Makanan dalam Kontainer .....	59
Gambar 3.12 Penentuan Titik Koordinat Barang .....	60
Gambar 3.13 <i>Form Login</i> .....	65
Gambar 3.14 <i>Form</i> Implementasi Algoritma Genetika .....	66
Gambar 3.15 <i>Form</i> Detail Hasil .....	66
Gambar 3.16 <i>Form</i> Visualisasi 3D .....	67
Gambar 4.1 Struktur Program Java .....	68

Gambar 4.2 <i>Form Login</i> .....	70
Gambar 4.3 <i>Form About</i> .....	71
Gambar 4.4 Input Data Barang.....	72
Gambar 4.5 Tampilan Tabel Data Barang .....	72
Gambar 4.6 Hasil Inisialisasi Kromosom .....	73
Gambar 4.7 <i>Output</i> Algoritma Genetika.....	74
Gambar 4.8 Hasil Statistik Generasi.....	75
Gambar 4.9 <i>Form Admin</i> .....	76
Gambar 4.10 <i>Form Help</i> .....	76
Gambar 4.11 Titik Koordinat Barang .....	77
Gambar 4.12 Visualisasi 3D Penataan Barang pada Kontainer .....	78
Gambar 4.13 Visualisai 3D Percobaan 1 pada Kontainer 20 feet .....	82
Gambar 4.14 Visualisai 3D Percobaan 2 pada Kontainer 20 feet .....	82
Gambar 4.15 Visualisai 3D Percobaan 1 pada Kontainer 40 feet .....	83
Gambar 4.16 Visualisai 3D Percobaan 2 pada Kontainer 40 feet .....	84
Gambar 4.17 Visualisai 3D Percobaan 1 pada Kontainer 45 feet .....	85
Gambar 4.18 Visualisai 3D Percobaan 2 pada Kontainer 45 feet .....	85



## DAFTAR TABEL

Tabel 2.1 Contoh Mutasi Pengkodean Biner .....	26
Tabel 2.2 Contoh Mutasi Pengkodean Nilai .....	27
Tabel 2.3 Contoh Mutasi Pengkodean Permutasi.....	27
Tabel 2.4 Relevansi Algoritma Genetika dengan Teknik Pemrograman .....	32
Tabel 3.1 Data Barang .....	39
Tabel 3.2 Pengelompokan Data Barang.....	40
Tabel 4.1 Muatan Barang pada Percobaan 1 .....	80
Tabel 4.2 Muatan Barang pada Percobaan 2.....	81



## ABSTRAK

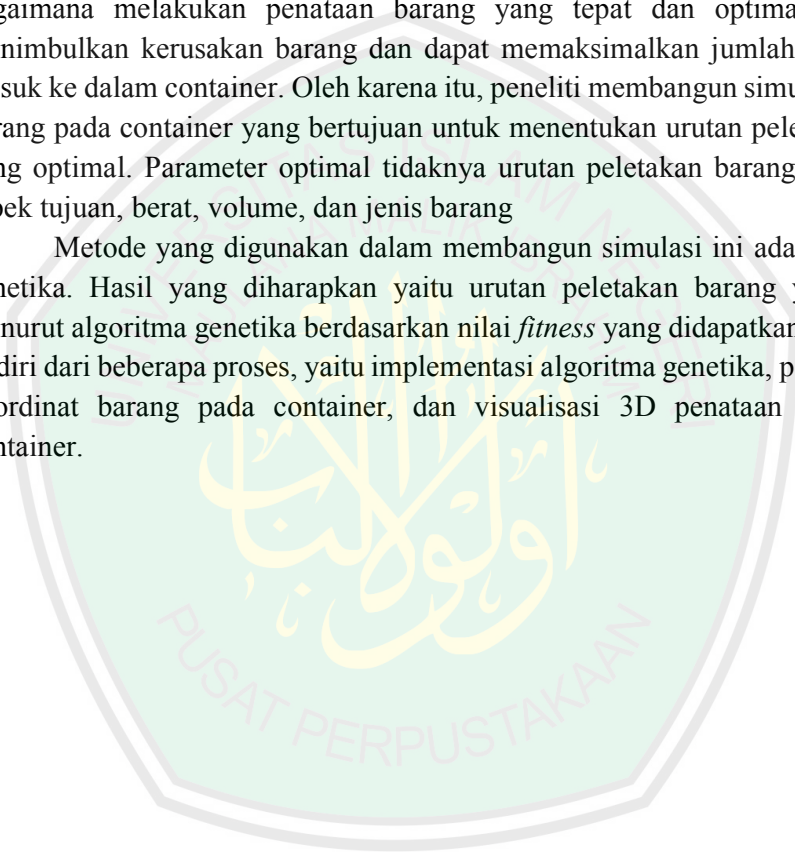
Farosanti, Lafnidita. 2015. **Simulasi 3D Optimasi Penataan Barang Pada Kontainer Menggunakan Algoritma Genetika**. Dosen Pembimbing : (I) Fachrul Kurniawan, M, M.T. (II) Hani Nurhayati, M.T.

Kata kunci : Algoritma Genetika, Penataan Barang, Nilai *fitness*.

---

Perusahaan jasa pengiriman barang berkembang dengan pesat seiring dengan tuntutan kebutuhan perusahaan dalam kegiatan distribusi barang. Permasalahan yang dialami oleh perusahaan jasa pengiriman barang adalah bagaimana melakukan penataan barang yang tepat dan optimal agar tidak menimbulkan kerusakan barang dan dapat memaksimalkan jumlah barang yang masuk ke dalam container. Oleh karena itu, peneliti membangun simulasi penataan barang pada container yang bertujuan untuk menentukan urutan peletakan barang yang optimal. Parameter optimal tidaknya urutan peletakan barang ditinjau dari aspek tujuan, berat, volume, dan jenis barang

Metode yang digunakan dalam membangun simulasi ini adalah algoritma genetika. Hasil yang diharapkan yaitu urutan peletakan barang yang optimal menurut algoritma genetika berdasarkan nilai *fitness* yang didapatkan. Program ini terdiri dari beberapa proses, yaitu implementasi algoritma genetika, penentuan titik koordinat barang pada container, dan visualisasi 3D penataan barang pada container.



## ABSTRACT

Farosanti, Lafnidita. 2015. **3D Simulation Optimization Arrangement of Goods in Containers Using Genetic Algorithms**. Supervisor : (I) Fachrul Kurniawan, M, M.T. (II) Hani Nurhayati, M.T.

**Keywords** : *Genetic Algorithms, Arrangement of Goods, Fitness Cost*

---

A freight company growing rapidly along with the demands of companies in the activities of the distribution of goods. The problem happen in a freight company is how to make arrangement of goods optimum that is not to make damage to the goods and can maximize the amount of goods entering into the container. Therefore, researchers built a simulation arrangement of goods in containers that aims to determine the optimal order of placement of goods. Optimal parameter whether the order placements of goods in terms of aspects destination, weight, volume, and type of goods.

The method in built this simulation is genetic algorithms. The expected result is the optimal order of placement of goods according to genetic algorithms based on fitness cost. This program consist of some process that is implementation of genetic algorithms, determining the coordinates of goods in containers, and 3D visualization arrangement of goods in the containers.

## مخلص

فاروسنتي، لافنيديتا. 2015. إعداد عناصر 3D محاكاة الأمتل في حاويات عن طريق الخوارزميات الجينية.

مشريف (I): فخر الكورنياوان، الماجستير، (II). هاني نور حياتي، الماجستير

الكلمة الرئيسية: الخوارزميات الجينية، هيكله السلع، القيمة اللياقة البدنية

شركة الشحن لتنمو بسرعة بما يتماشى مع متطلبات الشركات في أنشطة توزيع السلع. المشاكل التي يعاني منها شركة الشحن هو كيفية جعل ترتيب البضائع المناسبة والمثلى لكي لا تسبب ضررا على السلع ويمكن زيادة كمية البضائع التي تدخل في وعاء. ولذلك، تبنى الباحثون ترتيب محاكاة البضائع في الحاويات التي تهدف إلى تحديد النظام الأمتل للوضع البضائع. المعلومات الأمتل أم لا ترتيب وضع البضائع من حيث الجوانب جهة والوزن والحجم، ونوع البضاعة. ولذلك، تبنى الباحثون ترتيب محاكاة البضائع في الحاويات التي تهدف إلى تحديد النظام الأمتل للوضع البضائع. المعلومات الأمتل أم لا ترتيب وضع البضائع من حيث الجوانب جهة والوزن والحجم، ونوع البضاعة.



## مخلص

فاروسنتي، لافنيديتا. 2015. إعداد عناصر 3D محاكاة الأمتل في حاويات عن طريق الخوارزميات الجينية.  
مشريف (I): فخر الكورنياوان، الماجستير، (II). هاني نور حياتي، الماجستير

الكلمة الرئيسية: الخوارزميات الجينية، هيكله السلع، القيمة اللياقة البدنية

شركة الشحن لتنمو بسرعة بما يتماشى مع متطلبات الشركات في أنشطة توزيع السلع. المشاكل التي يعاني منها شركة الشحن هو كيفية جعل ترتيب البضائع المناسبة والمثلى لكي لا تسبب ضررا على السلع ويمكن زيادة كمية البضائع التي تدخل في وعاء. ولذلك، تبنى الباحثون ترتيب محاكاة البضائع في الحاويات التي تهدف إلى تحديد النظام الأمتل للوضع البضائع. المعلمات الأمتل أم لا ترتيب وضع البضائع من حيث الجوانب جهة والوزن والحجم، ونوع البضاعة.

ولذلك، تبنى الباحثون ترتيب محاكاة البضائع في الحاويات التي تهدف إلى تحديد النظام الأمتل للوضع البضائع. المعلمات الأمتل أم لا ترتيب وضع البضائع من حيث الجوانب جهة والوزن والحجم، ونوع البضاعة

## ABSTRAK

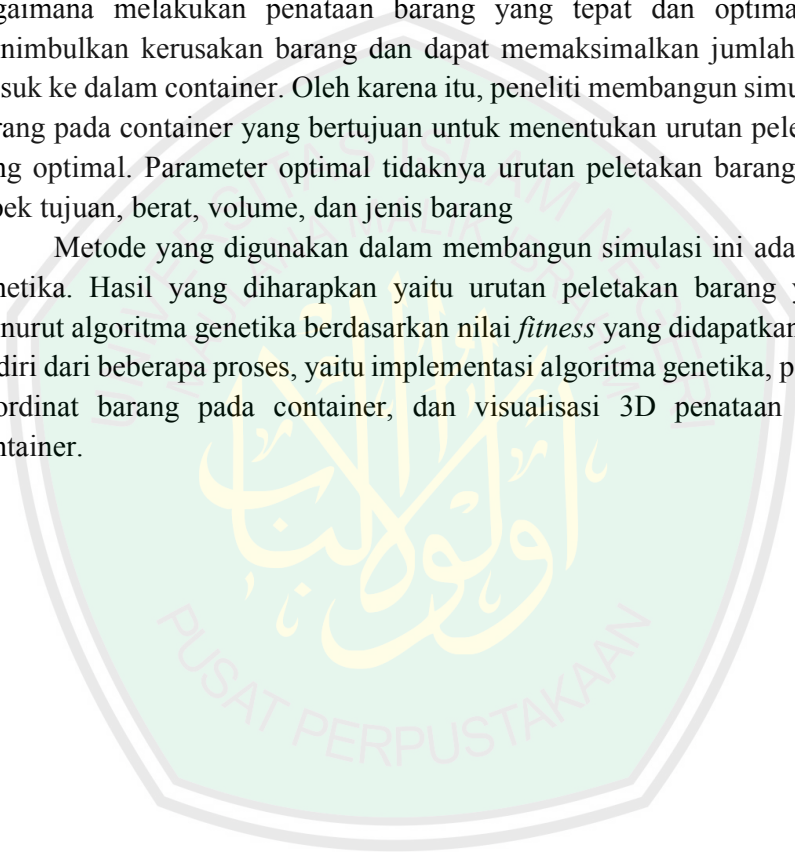
Farosanti, Lafnidita. 2015. **Simulasi 3D Optimasi Penataan Barang Pada Kontainer Menggunakan Algoritma Genetika**. Dosen Pembimbing : (I) Fachrul Kurniawan, M, M.T. (II) Hani Nurhayati, M.T.

Kata kunci : Algoritma Genetika, Penataan Barang, Nilai *fitness*.

---

Perusahaan jasa pengiriman barang berkembang dengan pesat seiring dengan tuntutan kebutuhan perusahaan dalam kegiatan distribusi barang. Permasalahan yang dialami oleh perusahaan jasa pengiriman barang adalah bagaimana melakukan penataan barang yang tepat dan optimal agar tidak menimbulkan kerusakan barang dan dapat memaksimalkan jumlah barang yang masuk ke dalam container. Oleh karena itu, peneliti membangun simulasi penataan barang pada container yang bertujuan untuk menentukan urutan peletakan barang yang optimal. Parameter optimal tidaknya urutan peletakan barang ditinjau dari aspek tujuan, berat, volume, dan jenis barang

Metode yang digunakan dalam membangun simulasi ini adalah algoritma genetika. Hasil yang diharapkan yaitu urutan peletakan barang yang optimal menurut algoritma genetika berdasarkan nilai *fitness* yang didapatkan. Program ini terdiri dari beberapa proses, yaitu implementasi algoritma genetika, penentuan titik koordinat barang pada container, dan visualisasi 3D penataan barang pada container.



## ABSTRACT

Farosanti, Lafnidita. 2015. **3D Simulation Optimization Arrangement of Goods in Containers Using Genetic Algorithms**. Supervisor : (I) Fachrul Kurniawan, M, M.T. (II) Hani Nurhayati, M.T.

**Keywords** : *Genetic Algorithms, Arrangement of Goods, Fitness Cost*

---

A freight company growing rapidly along with the demands of companies in the activities of the distribution of goods. The problem happen in a freight company is how to make arrangement of goods optimum that is not to make damage to the goods and can maximize the amount of goods entering into the container. Therefore, researchers built a simulation arrangement of goods in containers that aims to determine the optimal order of placement of goods. Optimal parameter whether the order placements of goods in terms of aspects destination, weight, volume, and type of goods.

The method in built this simulation is genetic algorithms. The expected result is the optimal order of placement of goods according to genetic algorithms based on fitness cost. This program consist of some process that is implementation of genetic algorithms, determining the coordinates of goods in containers, and 3D visualization arrangement of goods in the containers.

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Perusahaan yang bergerak di bidang jasa pengiriman barang mengalami perkembangan yang cukup pesat di era modern ini. Hal itu disebabkan oleh tuntutan kebutuhan berbagai perusahaan dalam proses pendistribusian barang/produk yang dihasilkan. Beberapa perusahaan memakai jasa pengiriman barang agar barang tetap terjaga kualitas dan keamanannya ketika hendak dikirimkan ke tujuan. Selain itu, mungkin perusahaan tersebut tidak mempunyai peralatan yang memadai dalam proses pengiriman barang. Salah satu alat yang digunakan dalam proses pengiriman barang suatu perusahaan yakni container.

Dalam melakukan proses ekspedisi barang, sebuah perusahaan selalu berupaya untuk meningkatkan kualitas kinerja dan pelayanannya. Kerja sama antar perusahaan merupakan bentuk proses jual beli yang dilakukan dalam jumlah besar dan interaksi yang lebih kompleks serta mengikutsertakan banyak pihak di dalamnya. Dalam melakukan proses perdagangan terdapat kaidah atau hukum yang berlaku sesuai syariat Islam. Hal tersebut tertuang dalam dalil Al-Qur'an yang berbunyi :

يَا أَيُّهَا الَّذِينَ ءَامَنُوا لَا تَأْكُلُوا أَمْوَالَكُم بَيْنَكُم بِالْبَاطِلِ إِلَّا أَنْ تَكُونَ تِجَارَةً عَنْ تَرَاضٍ

مِّنْكُمْ

Artinya : *“Hai orang-orang yang beriman janganlah kamu makan harta sesamamu dengan jalan yang bathil, kecuali dengan jalan perniagaan yang berlaku dengan suka sama suka diantara kamu”* (QS. An-Nisa : 29).

Kegiatan niaga mempunyai maksud atau bentuk yang amat luas, yaitu segala jual beli, sewa menyewa, import dan export, upah mengupah, dan semua yang menimbulkan peredaran harta benda. Ayat tersebut dengan tegas melarang adanya pihak atau orang yang memakan harta orang lain. Istilahnya dalam kehidupan sehari-hari adalah mencegah adanya salah satu pihak yang dirugikan dalam interaksi perdagangan. Perdagangan atau kegiatan niaga harus dilakukan atas persetujuan dan ke-*ridhaan* antara satu sama lain (penjual dan pembeli). Keridhaan merupakan sesuatu yang tersembunyi dalam lubuk hati karena jika tidak ridha maka akan mempengaruhi tindakan dari pihak atau orang yang melakukannya. Oleh karena itu, mencegah terjadinya ketidak-*ridhaan* antara kedua belah pihak dalam proses perniagaan barang yang dilakukan maka pelayanan harus dilakukan sebaik mungkin agar tidak menimbulkan kekecewaan.

Dalam hal ini kegiatan perniagaan yang dilakukan adalah proses distribusi atau pengiriman barang. Agar dalam proses pengiriman tersebut mencapai hasil yang optimal, maka diperlukan adanya proses optimasi. Optimasi penataan barang di dalam container bertujuan agar barang dapat disusun secara tepat dalam sebuah container dengan mempertimbangkan beberapa aspek, yakni tujuan, berat, jenis, dan volume barang. Jika dalam proses penyusunan barang tidak tepat, maka akan mengakibatkan kerusakan barang yang dikirimkan dan mengalami kesulitan dalam proses pengambilan barang. Penataan barang yang dilakukan harus bisa memaksimalkan jumlah barang yang masuk ke container. Hal itu bertujuan agar

perusahaan bisa menekan biaya pengiriman sekecil mungkin bila pengisian barang ke dalam container dapat dilakukan secara maksimal. Misalnya, pengiriman barang bisa dilakukan oleh sebuah container saja apabila pengisian dilakukan secara maksimal. Jika tidak, bisa jadi pengiriman barang diangkut oleh 2 kontainer untuk menampung sisa barang yang tidak masuk pada 1 kontainer. Melihat realita demikian maka diperlukan adanya simulasi yang dapat menentukan proses penataan barang dalam container secara tepat dan optimal serta dapat memaksimalkan jumlah barang yang masuk ke dalam container.

Metode yang digunakan dalam membangun program Simulasi 3D Penataan Barang pada Kontainer ini adalah Algoritma Genetika yang digunakan untuk menentukan urutan penyusunan barang yang optimal dengan nilai fitness sebagai acuannya berdasarkan parameter tujuan, berat, jenis, dan volume barang. Algoritma genetika merupakan metode yang cukup tepat dan cukup baik dalam pencarian solusi karena prosesnya didasarkan atas mekanisme evolusi biologis dimana individu yang paling fit mempunyai peluang besar untuk dipilih pada generasi selanjutnya. Selain itu, algoritma genetika menggunakan prinsip OPP (*Object Oriented Programming*) dimana hal-hal yang digunakan dalam prosesnya adalah sebuah objek. Hal itu akan mempermudah dalam membangun program simulasi penataan barang ini untuk mencari solusi yang optimal.

## **1.2 Identifikasi Masalah**

Berdasarkan penjelasan dari latar belakang di atas, maka identifikasi masalah dalam penelitian ini adalah :

1. Bagaimana membangun Simulasi 3D Optimasi Penataan Barang Pada Kontainer Menggunakan Algoritma Genetika?
2. Apakah algoritma genetika mampu memberikan solusi penataan barang pada container secara optimal?

### 1.3 Batasan Masalah

Dalam membangun program Simulasi 3D Penataan Barang pada Kontainer Menggunakan Algoritma Genetika terdapat batasan-batasan dalam penelitian, antara lain :

1. Program ini merupakan simulasi bagaimana menentukan urutan peletakan barang pada container yang tepat dan optimal menurut algoritma genetika berdasarkan parameter tujuan, berat, jenis, dan volume barangnya.
2. Barang-barang yang disusun dalam simulasi ini meliputi barang yang sudah dipacking dalam bentuk kubus dan balok.
3. Ukuran container yang digunakan dalam simulasi ini terdiri dari 3 ukuran, yaitu 20 feet, 40 feet, dan 45 feet.
4. Jenis kontainer yang digunakan dalam program simulasi ini adalah *general cargo*.
5. Barang tidak mengalami perubahan posisi ketika ditata dalam container.
6. Penelitian ini dilakukan pada jumlah barang yang diinputkan sebanyak 50 data barang.

#### **1.4 Tujuan Penelitian**

1. Membangun simulasi optimasi penataan barang dalam container agar diperoleh solusi urutan peletakan barang pada container yang tepat dan optimal menurut algoritma genetika.
2. Membuktikan bahwa algoritma genetika mampu memberikan solusi penataan barang pada container secara tepat dan optimal.

#### **1.5 Manfaat Penelitian**

1. Membantu menyelesaikan permasalahan khususnya pada perusahaan yang bergerak di bidang jasa pengiriman barang dalam menentukan letak barang secara tepat dan optimal berdasarkan tujuan, berat, jenis, dan volume barang.
2. Memberikan solusi terbaik untuk permasalahan penataan barang pada container agar proses pengiriman barang berjalan dengan lancar.
3. Memudahkan para pegawai di perusahaan jasa pengiriman barang dalam menentukan penataan barang pada container secara tepat, optimal dan tidak memakan banyak waktu.



## BAB II

### KAJIAN PUSTAKA

#### 2.1 Teori Penunjang

##### A. *Container* atau Peti Kemas

###### 1. Pengertian

Peti kemas atau container diibaratkan sebagai gudang kecil yang berjalan untuk mengangkut barang dari satu tempat ke tempat lain bersama-sama dengan alat pengangkutnya, yakni truk atau kereta api sampai ke tempat yang dituju, biasanya ke gudang pemilik barang (exporter dan importer). (Samidjan H.D, 1991:70) . Sedangkan menurut Ella Ernawati (2011 : 14) petikemas adalah suatu peti berbentuk empat persegi panjang, tahan cuaca, digunakan untuk mengangkut dan menyimpan sejumlah muatan kemasan dan barang-barang curah yang melindungi isinya dari kehilangan dan kerusakan, dapat dipisahkan dari alat transportasinya, diperlakukan sebagai satuan muat dan tanpa harus membongkar isinya jika akan pindah kapal.

Petikemas (container adalah suatu kemasan yang dirancang secara khusus dengan ukuran tertentu, dapat dipakai berulang kali, dipergunakan untuk menyimpan sekaligus mengangkat muatan yang ada di dalamnya (Suyono R.P, 2003:179). Berdasarkan beberapa pengertian mengenai container seperti disebutkan di atas yakni container dikenal juga dengan sebutan peti kemas dengan bentuk empat persegi panjang yang berfungsi memuat barang-barang di dalamnya yang kemudian dibawa ke tempat tujuan tertentu. Biasanya untuk kepentingan perusahaan khususnya di bidang jasa pengiriman barang. Kemudian Koleangan,

menambahkan bahwa berdasarkan international standard organization (ISO) (2008 : 6) menjelaskan peti kemas yakni alat untuk mengangkut barang dimana :

- a. Sifatnya cukup kuat untuk digunakan berkali-kali.
- b. Dirancang secara khusus sebagai fasilitas untuk membawa barang dengan berbagai macam transport yang ada.
- c. Dipasang alat-alat yang memungkinkan sewaktu-waktu digunakan untuk menangani lebih dari satu alat transport ke alat transport yang lain.
- d. Dirancang sedemikian rupa untuk memudahkan mengisi maupun mengosongkan.
- e. Mempunyai isi ruangan dalam (interval volume) sekurang-kurangnya  $1 \text{ m}^3 = 35,3 \text{ CUFF}$ .

Jenis container menurut Koleangan (2008 : 6-7) adalah :

**a. General cargo container**

General cargo container adalah petikemas yang dipakai untuk mengangkut muatan umum, misal: kayu, kain, dll.

**b. Thermal container**

Thermal container adalah petikemas yang dilengkapi dengan pengatur suhu untuk muatan tertentu.

**c. Tank container**

Tank container adalah tangki yang ditempatkan dalam kerangka petikemas yang digunakan untuk muatan cair (bulk liquid) maupun gas (bulk gas).

**d. Dry bulk**

Dry bulk adalah general purpose container yang dipergunakan khusus untuk mengangkut muatan curah. Untuk memasukkan muatan melalui lubang bagian atas petikemas sedangkan mengeluarkan muatan melalui lubang atau pintu di bagian bawah petikemas.

**e. Platform container**

Platform container adalah petikemas yang terdiri dari lantai dasar.

**f. Specials container**

Specials container adalah petikemas yang khusus dibuat untuk muatan tertentu, seperti petikemas untuk muatan ternak (cattle container) atau muatan kendaraan (car container).

Ukuran Peti Kemas menurut dalam penelitian yang dilakukan oleh Ella Ernawati (2011), yakni sesuai dengan standard ISO telah menetapkan ukuran-ukuran dari petikemas sebagai berikut :

a. Container 20' Dry Freight (20 feet)

Ukuran luar : 20'(p) x 8'(l) x 8'6" (l)

Ukuran dalam : 5.919 x 2.340 x 2.380 m

Kapasitas : Cubic Capacity : 33 Cbm

Pay load : 22,1 ton

b. Container 40' Dry Freight (40 feet)

Ukuran luar : 40'(p) x 8'(l) x 8'6" (l)

Ukuran dalam : 12.045 x 2.309 x 2.379 m

Kapasitas : Cubic Capacity : 67,3 Cbm

Pay load : 27,396 ton

## c. Container 45 feet

Ukuran luar	: 40'(p) x 8'(l) x 9'6" (l)
Ukuran dalam	: 12.056 x 2.347 x 2.684 m
Kapasitas	: Cubic Capacity: 76 Cbm
	Pay load : 29,6 ton

Ukuran muatan dalam pembongkaran/pemuatan kapal petikemas dinyatakan dalam TEU (Twenty Footer Equivalent Unit) dan FEU (Forty Footer Equivalent Unit). Oleh karena itu, ukuran standar dari petikemas dimulai dari panjang 20 feet, untuk satu petikemas 20 feet dinyatakan sebagai 1 TEU dan petikemas 40 feet dinyatakan dengan 2 TEU atau 1 FEU (Forty Footer Equivalent Unit).

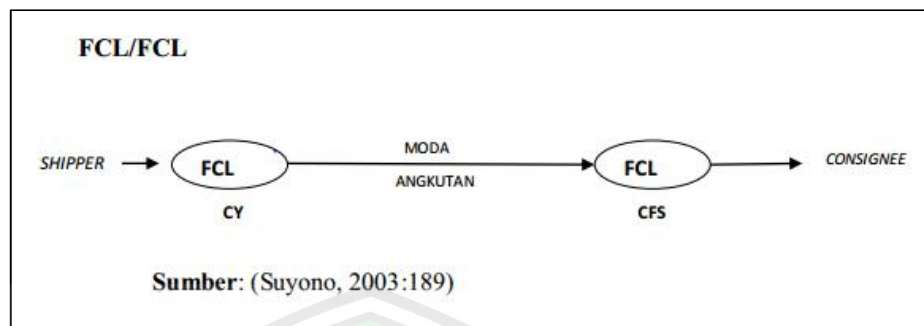
Pengangkutan barang ke dalam petikemas atau container terdiri dari 2 sistem (Suyono, 2003:188) yaitu :

### 1. Full Container Load (FCL)

FCL adalah shipper menggunakan satu atau lebih petikemas untuk digunakan mengirim barangnya sendiri. Sistem ini mempunyai ciri-ciri sebagai berikut:

- a. Petikemas berisi muatan dari satu shipper dan dikirim untuk satu consignee.
- b. Petikemas diisi (stuffing) oleh shipper (shipper load and count) atau dapat melalui perantara forwarder dan petikemas yang sudah diisi diserahkan di container yard (CY) pelabuhan muat.
- c. Di pelabuhan bongkar petikemas di ambil oleh consignee di container yard (CY) dan di un-stuffing oleh consignee.

- d. Perusahaan pelayaran tidak bertanggung jawab atas kerusakan dan kehilangan barang yang ada dalam petikemas.

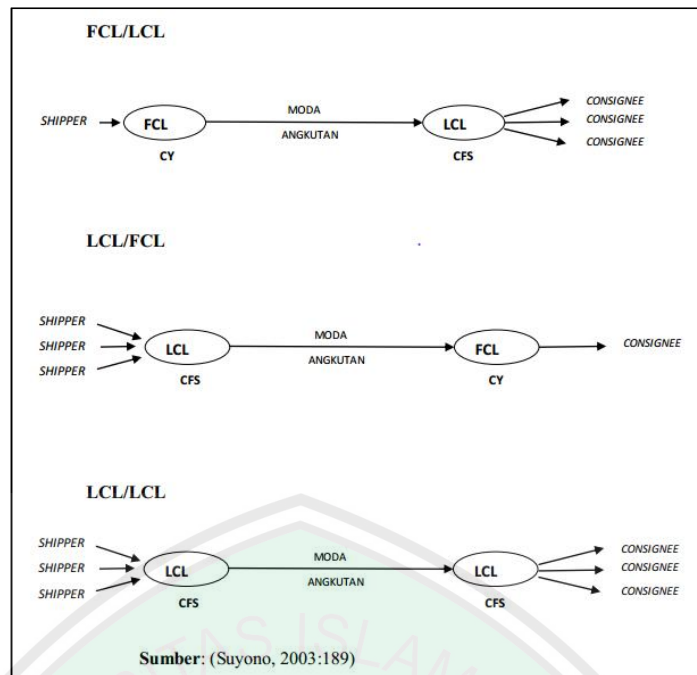


**Gambar 2.1 Sistem Pengiriman FCL**

## 2. Less Than Container Load (LCL)

LCL adalah shipper mengkonsolidasi/mencampur barangnya dengan barang shipper lain dalam satu petikemas. Sistem ini mempunyai ciri -ciri sebagai berikut:

- a. Petikemas berisi muatan dari beberapa shipper dan ditujukan kepada beberapa consignee.
- b. Muatan diterima dalam keadaan breakbulk dan diisi (stuffing) di container freight station (CFS) oleh perusahaan pelayaran.
- c. Di pelabuhan bongkar, petikemas di un-stuffing di CFS oleh perusahaan pelayaran dan diserahkan kepada beberapa consignee dalam keadaan breakbulk.
- d. Perusahaan pelayaran tidak bertanggung jawab atas kerusakan dan kehilangan barang yang ada dalam petikemas.



**Gambar 2.2 Sistem Pengiriman LCL**

Berdasarkan hal tersebut maka disimpulkan bahwa menggunakan fasilitas container dalam proses pemuatan barang mempunyai banyak keuntungan diantaranya barang-barang yang dimuat dalam container terjaga keamanan dan kualitas barang. Dapat dilakukan pemindahan barang secara efektif dan praktis, sehingga pada umumnya perusahaan-perusahaan memakai container untuk proses pengangkutan barang-barang produksinya.

## 2. Proses Pemuatan Barang ke Dalam Container

Pemuatan barang produksi pada suatu perusahaan umumnya terdiri dari beberapa proses diantaranya yaitu *packing*, *shipping mark*, dan *stuffing*. Uraian berikut merupakan penjelasan mengenai beberapa proses tersebut dari penelitian yang dilakukan oleh Danang Rosyid (2010) dengan judul “Proses Stuffing Produk

Furniture Pada Rakabu Furniture Di Surakarta”. Barang produksi dari perusahaan tersebut merupakan produk-produk furniture lemari, meja, kursi, dan lainnya.

#### ❖ **Packing**

Packing merupakan kegiatan membungkus barang ekspor dengan menggunakan berbagai jenis alat bungkus sesuai dengan kebutuhan keamanan barang (Suyono, 2003:162). Syarat pembungkusan barang harus memenuhi 3K yaitu keamanan, keaslian, kepuasan. Secara umum, jenis bungkus yang diperlukan dalam berbagai muatan adalah sebagai berikut (Sudijono & Sarjiyanto, 2007: 36) :

##### 1. Karung

Bahan karung dapat menekan isi yang dalam tetapi tidak melindungi kerusakan yang datang dari luar. Dapat digunakan untuk muatan misalkan : pupuk, beras, jagung, kopi, dll.

##### 2. Fiber dan karton

Bahan fiber dan karton dapat menahan tekanan dan bantingan dan relatif murah.

##### 3. Peti kayu

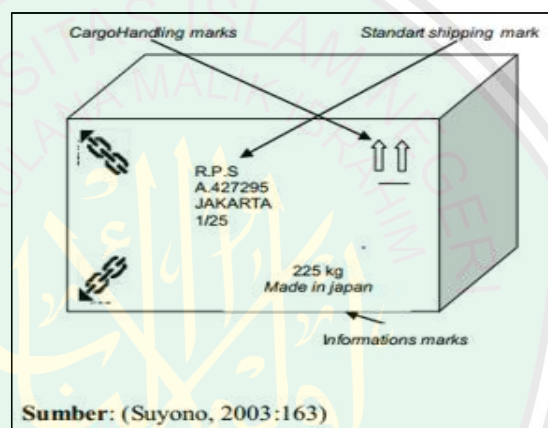
Peti kayu merupakan bahan bungkus yang paling baik dan sesuai untuk pengangkutan barang secara konvensional, tahan terhadap panas atau kelembapan.

#### ❖ **Shipping Mark**

Shipping mark merupakan tanda pengenalan barang ekspor (Suyono, 2003:163). Tujuan dari shipping mark adalah agar barang lebih mudah

dikenal untuk cepat mengenal barang dan karena itu dapat cepat sampai ke tempat tujuan, dengan demikian tulisan pada pembungkus harus jelas. Dalam pemberian shipping mark harus diperhatikan letak dari merk barang dan segala keterangan yang sesuai keadaan barang dan perlu ditulis secara berurut, sebagai contoh:

1. Initial atau kependekan : RPS
2. Nomor Referensi : A. 427295
3. Tujuan : JAKARTA
4. Nomor Pembungkus : 1/25



**Gambar 2.3 Shipping Mark**

#### ❖ Stuffing

Stuffing merupakan proses pemindahan produk yang sudah di packing ke dalam kontainer dengan diberi kode-kode yang ditentukan dan dihitung untuk pembuatan packing list. Ada beberapa cara stuffing container (Suyono, 2003:198):

1. Untuk peti karton

Bila berat peti/karton tidak sama, maka peti/karton yang lebih berat diletakkan dan disusun di bawah. Bila susunan peti kartonnya seragam, maka tumpukan pertama disusun dari kanan ke kiri dan tumpukan dua dari kiri ke kanan.

2. Untuk muatan karung yang tidak dapat di palet

Susunlah karung pada tumpukan pertama dengan baris melintang petikemas dan paling ujung membujur petikemas. Selanjutnya, pada tumpukan kedua, dua baris melintang dimulai dari atas yang membujur dan yang paling ujung disusun membujur.

3. Untuk muatan drum/barrels

Drum atau barrel harus selalu disusun berdiri, selang satu baris dipergunakan dunnage, mulai dari kiri ke kanan atau dari depan ke belakang. Pergunakan dunnage diatas tumpukan/ susunan pertama untuk mulai tumpukan/susunan kadua. Untuk mengurangi broken space, gunakan alas papan pada baris urutan ganjil agar benjolan drum tidak saling bersentuhan.

4. Untuk muatan yang dipalet

Muatan diatas palet harus diikat kuat menggunakan ban, ikatan baja atau plastik, dan diikat pada palet. Bila petikemas hanya diisi dengan satu atau dua palet saja maka letakkan susunan palet di tengah-tengah petikemas dan diperkuat letaknya dengan ganjal (chocking) agar muatan palet tidak bergoyang.

5. Untuk long length cargo

Lebih baik menggunakan petikemas jenis flat-rack atau opentop untuk memudahkan pemuatan dan pembongkarannya. Pasang chocking di ujung-ujung petikemas. Agar mudah mengeluarkan muatan, gunakan *dunnage* agar sling dapat mudah dimasukkan atau di-presling dahulu.

## **B. Algoritma Genetika (*Genetic Algorithms-GA*)**

### **1. Pengertian**

Algoritma genetika adalah algoritma pencarian heuristik yang didasarkan atas mekanisme evolusi biologis. Keberagaman pada evolusi biologis adalah variasi dari kromosom antar individu organisme. Variasi kromosom ini akan mempengaruhi laju reproduksi dan tingkat kemampuan organisme untuk tetap hidup. Algoritma genetika pertama kali dikembangkan oleh John Holland dari Universitas Michigan (1975). John Holland mengatakan bahwa setiap masalah yang berbentuk adaptasi (alami maupun buatan) dapat diformulasikan dalam terminologi genetika. Algoritma genetika adalah simulasi dari proses evolusi Darwin dan operasi genetika atas kromosom (Kusumadewi, 2003: 279).

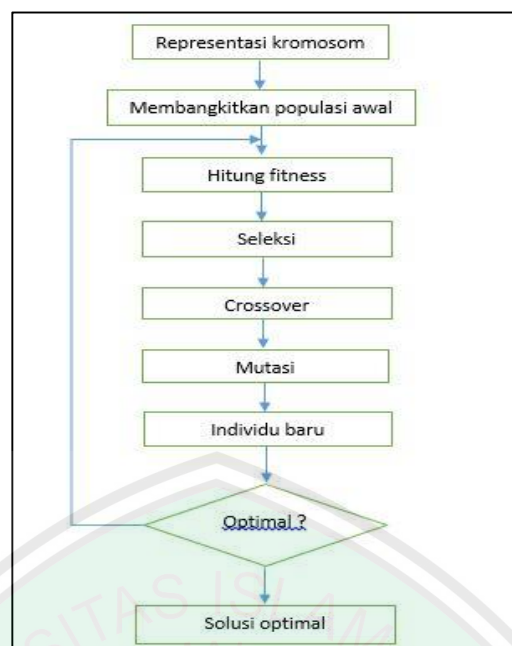
### **2. Struktur Umum Algoritma Genetika**

Sebuah solusi yang dibangkitkan dalam algoritma genetika disebut sebagai *chromosome*, sedangkan kumpulan chromosome-chromosome tersebut disebut sebagai *populasi*. Sebuah chromosome dibentuk dari komponen-komponen penyusun yang disebut sebagai *gen* dan nilainya dapat berupa bilangan numerik, biner, simbol ataupun karakter tergantung dari permasalahan yang ingin diselesaikan. Chromosome-chromosome tersebut akan berevolusi secara

berkelanjutan yang disebut dengan *generasi*. Dalam tiap generasi chromosome-chromosome tersebut dievaluasi tingkat keberhasilan nilai solusinya terhadap masalah yang ingin diselesaikan (*fungsi\_objektif*) menggunakan ukuran yang disebut dengan *fitness*. Untuk memilih chromosome yang tetap dipertahankan untuk generasi selanjutnya dilakukan proses yang disebut dengan *seleksi*. Proses seleksi chromosome menggunakan konsep aturan evolusi Darwin yang telah disebutkan sebelumnya yaitu chromosome yang mempunyai nilai fitness tinggi akan memiliki peluang lebih besar untuk terpilih lagi pada generasi selanjutnya.

Chromosome-chromosome baru yang disebut dengan *offspring*, dibentuk dengan cara melakukan perkawinan antar chromosome-chromosome dalam satu generasi yang disebut sebagai proses *crossover*. Jumlah chromosome dalam populasi yang mengalami crossover ditentukan oleh parameter yang disebut dengan *crossover\_rate*. Mekanisme perubahan susunan unsur penyusun makhluk hidup akibat adanya faktor alam yang disebut dengan *mutasi* direpresentasikan sebagai proses berubahnya satu atau lebih nilai gen dalam chromosome dengan suatu nilai acak. Jumlah gen dalam populasi yang mengalami mutasi ditentukan oleh parameter yang dinamakan *mutation\_rate*. Setelah beberapa generasi akan dihasilkan chromosome-chromosome yang nilai gen-gennya konvergen ke suatu nilai tertentu yang merupakan solusi terbaik yang dihasilkan oleh algoritma genetika terhadap permasalahan yang ingin diselesaikan.

Berikut flowchart secara umum Algoritma Genetika :



**Gambar 2.4 Flowchart Algoritma Genetika**

### 3. Komponen-Komponen Utama Algoritma Genetika

Terdapat 6 komponen utama dalam Algoritma Genetika menurut Kusumadewi, 2003: 280-283 yaitu :

#### a. Teknik Pengkodean

Teknik pengkodean disini meliputi pengkodean gen dari kromosom. Gen merupakan bagian dari kromosom. Satu gen biasanya akan mewakili satu variabel. Gen dapat direpresentasikan dalam bentuk: string bit, pohon, array bilangan real, daftar aturan, elemen permutasi, elemen program, atau representasi lainnya yang dapat diimplementasikan untuk operator genetika. Demikian juga kromosom dapat direpresentasikan dengan menggunakan:

String bit : 10011, 01101, 11101, dst.

Bilangan real : 65.65, -67.98, 562.88, dst.

Elemen permutasi : E2, E10, E5, dst.

Daftar aturan : R1, R2, R3, dst.

Elemen program : pemrograman genetika

Berikut adalah beberapa jenis pengkodean yang umum digunakan.

- a. Pengkodean Biner
- b. Pengkodean Permutasi
- c. Pengkodean Nilai
- d. Pengkodean Pohon

### **b. Membangkitkan Populasi Awal**

Membangkitkan populasi awal adalah proses membangkitkan sejumlah individu secara acak atau melalui prosedur tertentu. Ukuran untuk populasi tergantung pada masalah yang akan diselesaikan dan jenis operator genetika yang akan diimplementasikan. Setelah ukuran populasi ditentukan, kemudian dilakukan pembangkitan populasi awal. Syarat-syarat yang harus dipenuhi untuk menunjukkan suatu solusi harus benar-benar diperhatikan dalam pembangkitan setiap individu. Teknik dalam pembangkitan populasi awal ini ada beberapa cara, diantaranya sebagai berikut :

#### 1. Random Generator

Inti dari cara ini adalah melibatkan pembangkitan bilangan random untuk nilai setiap gen sesuai dengan representasi kromosom yang digunakan. Jika menggunakan representasi biner, salah satu contoh penggunaan random generator adalah penggunaan rumus berikut untuk pembangkitan populasi awal :

$$IPOP = \text{round} \{ \text{random}(N_{ipop}, N_{bits}) \}$$

Dimana IPOP adalah gen yang nantinya berisi pembulatan dari bilangan random yang dibangkitkan sebanyak  $N_{ipop}$  (Jumlah populasi)  $\times N_{bits}$  (Jumlah Gen dalam tiap kromosom).

## 2. Pendekatan Tertentu (Memasukkan Nilai Tertentu ke Dalam Gen)

Cara ini adalah dengan memasukkan nilai tertentu ke dalam gen dari populasi awal yang dibentuk.

## 3. Permutasi Gen

Salah satu cara permutasi gen dalam pembangkitan populasi awal adalah penggunaan permutasi Josephus dalam permasalahan kombinatorial seperti TSP. misalkan ada kota dari 1 sampai 9. Permutasi dari lintasan dapat dilakukan dengan menentukan titik awal dan selang. Misalnya titik awal adalah 6 dan selang adalah 5. Maka lintasan berangkat dari kota 6, selang dari 6 kota adalah kota 2 (dengan asumsi kota 1 sampai 9 membentuk *circular list*). Kota 2 dihapus dari list. Selang 5 kemudian adalah kota 7. Proses ini diulang hingga ada satu lintasan dalam list. Hasil dari permutasi ini adalah 2-7-3-8-4-9-5-1-6.

### c. Prosedur Inisialisasi

Ukuran populasi tergantung pada masalah yang akan dipecahkan dan jenis operator genetika yang akan diimplementasikan. Setelah ukuran populasi ditentukan, kemudian harus dilakukan inisialisasi terhadap kromosom yang terdapat pada populasi tersebut. Inisialisasi kromosom dilakukan secara acak,

namun demikian harus tetap memperhatikan domain solusi dan kendala permasalahan yang ada.

#### **d. Fungsi Evaluasi**

Ada 2 hal yang harus dilakukan dalam melakukan evaluasi kromosom, yaitu: evaluasi fungsi objektif (fungsi tujuan) dan konversi fungsi objektif dengan nilai yang tidak negatif. Apabila ternyata fungsi objektif memiliki nilai negatif, maka perlu ditambahkan suatu konstanta  $C$  agar nilai fitness yang terbentuk menjadi tidak negatif.

#### **e. Seleksi**

Seleksi bertujuan memberikan kesempatan reproduksi yang lebih besar bagi anggota populasi yang paling fit (Anita dan Muhammad, 2006:193). Seleksi akan menentukan individu-individu mana saja yang akan dipilih untuk dilakukan rekombinasi dan bagaimana offspring terbentuk dari individu-individu terpilih tersebut. Langkah pertama yang dilakukan dalam seleksi ini adalah pencarian nilai fitness. Masing masing individu dalam suatu wadah seleksi akan menerima probabilitas reproduksi yang tergantung pada nilai objektif dirinya sendiri terhadap nilai objektif dari semua individu dalam wadah seleksi tersebut. Nilai fitness inilah yang nantinya akan digunakan pada tahap-tahap seleksi berikutnya (Kusumadewi dan Hari, 2005: 235).

#### **✓ Rank-based Fitness**

Populasi diurutkan menurut nilai objektifnya. Nilai fitness dari tiap-tiap individu hanya tergantung pada posisi individu tersebut dalam urutan, dan tidak dipengaruhi oleh nilai objektifnya.

✓ **Seleksi Roda Roulette**

Istilah lainnya adalah stochastic sampling with replacement. Individu - individu dipetakan dalam suatu segmen garis secara berurutan sedemikian hingga tiap-tiap segmen individu memiliki ukuran yang sama dengan ukuran fitnessnya. Sebuah bilangan random dibangkitkan dan individu yang memiliki segmen dalam kawasan segmen dalam kawasan bilangan random tersebut akan terseleksi. Proses ini berulang hingga didapatkan sejumlah individu yang diharapkan.

✓ ***Stochastic Universal Sampling***

Memiliki nilai bias nol dan penyebaran yang minimum. Individu-individu dipetakan dalam suatu segmen garis secara berurut sedemikian hingga tiap-tiap segmen individu memiliki ukuran yang sama dengan ukuran fitnessnya seperti halnya pada seleksi roda roulette.

✓ **Seleksi Local**

Setiap individu yang berada di dalam konstrain tertentu disebut dengan nama lingkungan lokal. Interaksi antar individu hanya dilakukan di dalam wilayah tersebut. Lingkungan tersebut ditetapkan sebagai struktur dimana populasi tersebut terdistribusi. Lingkungan tersebut juga dapat dipandang sebagai kelompok pasangan-pasangan yang potensial.

✓ **Seleksi dengan Pemotongan**

Merupakan seleksi buatan yang digunakan oleh populasi yang jumlahnya sangat besar. Individu-individu diurutkan berdasarkan nilai fitnessnya. Hanya individu yang terbaik saja yang akan diseleksi sebagai induk.

#### ✓ **Seleksi dengan Turnamen**

Ditetapkan suatu nilai tour untuk individu-individu yang dipilih secara random dari suatu populasi. Individu-individu yang terbaik dalam kelompok ini akan diseleksi sebagai induk. Parameter yang digunakan adalah ukuran tour yang bernilai antara 2 sampai N (jumlah individu dalam populasi).

#### **f. Operator Genetika**

Ada 2 operator Genetika, yaitu: perkawinan silang (*crossover*) dan mutasi.

##### **1. Crossover**

Crossover (perkawinan silang) adalah operator genetik yang utama. Operator bekerja dengan mengambil dua individu dan memotong string kromosom mereka pada posisi yang terpilih secara acak, untuk memproduksi dua segment head dan dua segment tail (Son, 2007: 185). Crossover bertujuan menambah keanekaragaman string dalam satu produksi dengan penyilangan antar string yang diperoleh dari reproduksi sebelumnya (Anita dan Muhammad, 2006: 196).

#### ✓ **Crossover satu titik**

Crossover satu titik dan banyak titik biasanya digunakan untuk representasi kromosom dalam biner. Pada crossover satu titik, posisi crossover  $k$  ( $k=1,2,\dots,N-1$ ) dengan  $N$ =panjang kromosom diseleksi secara random.

Variabel-variabel ditukar antar kromosom pada titik tersebut untuk menghasilkan anak.

#### ✓ **Crossover banyak titik**

Pada crossover banyak titik,  $m$  posisi penyilangan  $k_i$  ( $k=1,2,\dots,N-1$ ,  $i=1,2,\dots,m$ ) dengan  $N$ =panjang kromosom diseleksi secara random dan tidak diperbolehkan ada posisi yang sama, serta diurutkan naik. Variable-variabel ditukar antar kromosom pada titik tersebut untuk menghasilkan anak.

#### ✓ **Crossover Aritmatika**

Crossover aritmatika digunakan untuk representasi berupa bilangan float (pecahan). Crossover ini dilakukan dengan menentukan nilai  $r$  sebagai bilangan random lebih dari 0 dan kurang dari 1. Selain itu juga ditentukan posisi dari gen yang dilakukan crossover menggunakan bilangan random.

#### ✓ **Crossover untuk Representasi Kromosom Permutasi**

##### - ***Partial-Mapped Crossover (PMX)***

Prosedur PMX

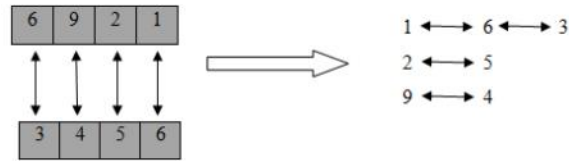
Langkah 1: Pilih posisi untuk menentukan substring secara acak

Induk 1	1	2	3	4	5	6	7	8	9
Induk 2	5	4	6	9	2	1	7	8	3

Langkah 2: tukar dua substring antar induk untuk menghasilkan *proto-child*.

Proto-child 1	1	2	6	9	2	1	7	8	9
Proto-child 2	5	4	3	4	5	6	7	8	3

Langkah 3: tentukan hubungan pemetaan diantara dua daerah pemetaan.



Langkah 4: tentukan kromosom keturunan mengacu pada hubungan pemetaan.



**Gambar 2.5 Contoh *Partial-Mapped Crossover (PMX)***

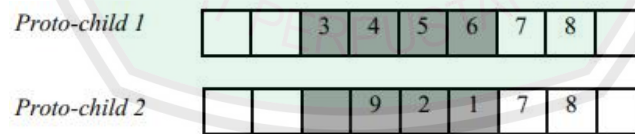
- ***Order Crossover (OX)***

Prosedur OX

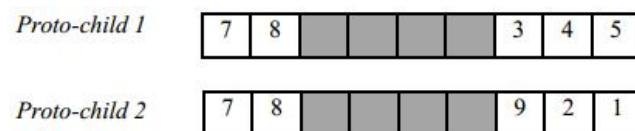
Langkah 1: pilih substring dari sebuah induk secara acak.



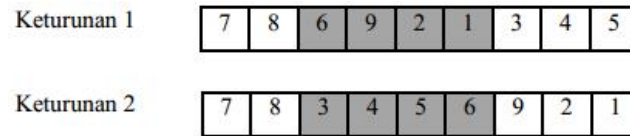
Langkah 2: bangkitkan sebuah proto-child dengan mengkosongkan tempat substring induk 2 pada induk 1.



Langkah 3: SHR allele dari substring pada tempat yang bersesuaian.



Langkah 4: tukar substring antara 2 induk.

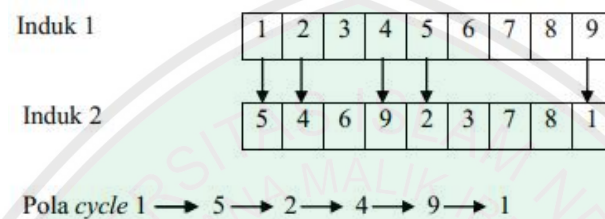


**Gambar 2.6 Contoh *Order-Crossover* (OX)**

- ***Cycle Crossover* (CX)**

Prosedur CX

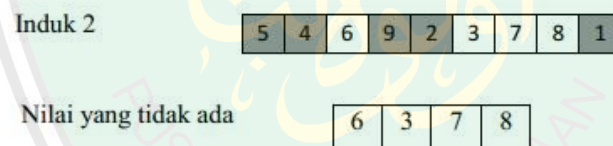
Langkah 1: tentukan pola cycle



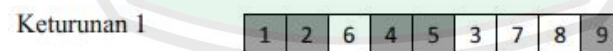
Langkah 2: copy nilai dalam *cycle* pada *proto-child*



Langkah 3: tentukan nilai yang diingat dari induk yang lain



Langkah 4: mengisi nilai yang diingat dalam keturunan



Dengan cara yang sama memperoleh keturunan 2



**Gambar 2.7 Contoh *Cycle Crossover* (CX)**

## 2. Mutasi

Mutasi merupakan proses mengubah nilai dari satu atau beberapa gen dalam suatu kromosom (Anita dan Muhammad, 2006: 197). Mutasi ini berperan untuk menggantikan gen yang hilang dari populasi akibat proses seleksi yang memungkinkan munculnya kembali gen yang tidak muncul pada inisialisasi populasi (Kusumadewi, 2003: 296). Mutasi diterapkan dengan probabilitas sangat kecil. Peluang mutasi ( $P_m$ ) mengendalikan banyaknya gen baru yang akan dimunculkan untuk dievaluasi. Jika peluang mutasi terlalu kecil, banyak gen yang mungkin berguna tidak pernah dievaluasi. Tetapi bila peluang mutasi terlalu besar, maka akan terlalu banyak gangguan acak, sehingga anak akan kehilangan kemiripan dari induknya.

### ✓ Mutasi Pengkodean Biner

Mutasi pengkodean biner merupakan operasi yang sangat sederhana. Proses mutasi pengkodean biner dilakukan dengan cara menginversi nilai bit pada kromosom yang terpilih secara acak (atau menggunakan skema tertentu) dengan diubah nilainya menjadi nilai lawannya (0 ke 1, atau 1 ke 0). Sebagai contoh, dapat dilihat pada tabel berikut ini :

**Tabel 2.1** Contoh Mutasi Pengkodean Biner

Keadaan Kromosom	Proses Mutasi
Kromosom sebelum mutasi	1000 1111 1011 0110
Kromosom sesudah mutasi	10011 1011 0110

### ✓ Mutasi Pengkodean Nilai

Mutasi pengkodean nilai adalah proses yang terjadi pada saat pengkodean nilai. Proses mutasi dalam pengkodean nilai dapat dilakukan dengan cara memilih sembarang posisi gen pada

kromosom, dan nilai yang ada kemudian ditambahkan atau dikurangkan dengan suatu nilai kecil tertentu yang diambil secara acak. Sebagai contoh, dapat dilihat pada tabel berikut ini, yaitu nilai riil ditambahkan dan dikurangkan dengan nilai 0 dan 1.

**Tabel 2.2** Contoh Mutasi Pengkodean Nilai

Keadaan Kromosom	Proses Mutasi
Kromosom sebelum mutasi	1,45 2,67 1,87 2,56
Kromosom sesudah mutasi	1,55 2,67 1,77 2,56

#### ✓ Mutasi Pengkodean Permutasi

Proses mutasi pengkodean permutasi tidak sama halnya dengan proses mutasi yang dilakukan pada pengkodean biner dengan mengubah langsung bit-bit pada kromosom. Salah satu cara yang dapat dilakukan adalah dengan memilih dua posisi (locus) dari kromosom dan kemudian nilainya saling dipertukarkan. Orangtua yang berada dibawah titik crossover dipertukarkan untuk menghasilkan anak baru. Contoh Mutasi pada pengkodean permutasi, dapat dilihat pada tabel di berikut ini :

**Table 2.3** Contoh Mutasi Pengkodean Permutasi

Keadaan Kromosom	Proses Mutasi
Kromosom sebelum mutasi	1 2 3 4 5 6 7 8 9
Kromosom sesudah mutasi	1 2 7 4 6 5 8 3 9

#### g. Penentuan Parameter

Yang disebut dengan parameter di sini adalah parameter kontrol algoritma genetika, yaitu: ukuran populasi (*popsi*), peluang crossover ( $P_c$ ), dan peluang

mutasi ( $P_m$ ). Nilai parameter ini ditentukan juga berdasarkan permasalahan yang akan dipecahkan. Ada beberapa rekomendasi yang bisa digunakan, antara lain (Kusumadewi, 2003: 283):

- Untuk permasalahan yang memiliki kawasan solusi cukup besar, De Jong merekomendasikan untuk nilai parameter kontrol:

$$(P_{\text{popsize}}; P_c; P_m) = (50; 0,6; 0,001)$$

- Bila rata-rata fitness setiap generasi digunakan sebagai indikator, maka Grefensette merekomendasikan:

$$(P_{\text{popsize}}; P_c; P_m) = (30; 0,95; 0,01).$$

- Bila *fitness* dari individu terbaik dipantau pada setiap generasi maka usulannya adalah:

$$(P_{\text{popsize}}; P_c; P_m) = (80; 0,45; 0,01).$$

- Ukuran populasi sebaiknya tidak lebih kecil dari 30, untuk sembarang jenis permasalahan.

#### **h. Nilai *Fitness***

Nilai *fitness* adalah nilai yang menyatakan baik tidaknya suatu solusi (individu). Nilai *fitness* ini yang dijadikan acuan dalam mencapai nilai optimal dalam algoritma genetika. Algoritma genetika bertujuan mencari individu dengan nilai *fitness* yang paling tinggi.

### **C. Optimasi dengan GA**

GA secara khusus dapat diterapkan untuk memecahkan masalah optimasi yang kompleks. Optimal disini dapat diartikan sebagai kasus minimasi atau

maksimasi. Karena itu GA baik untuk aplikasi yang memerlukan strategi pemecahan masalah secara adaptif, maka dari itu GA adalah suatu teknik optimasi yang terkenal. Secara umum ada tiga golongan besar teknik optimasi, yang berbasis pada kalkulus, enumerative, dan pencarian acak terarah (*guided random search*) (Son, 2007:168).

Karena GA banyak digunakan pada masalah praktis yang berfokus pada pencarian parameter-parameter optimal (Suyanto, 2005:3), maka hal ini membuat banyak orang mengira bahwa GA hanya bisa digunakan untuk masalah optimasi. Pada kenyataannya, GA juga memiliki performansi yang bagus untuk masalah-masalah selain optimasi.

#### **D. Simulasi**

Simulasi dapat diartikan sebagai meniru proses riil yang disebut system dengan sebuah model untuk memahami bagaimana system tersebut bekerja. Simulasi dengan computer merupakan model dievaluasi secara numeric, dan data dikumpulkan untuk mengestimasi karakteristik yang sebenarnya dari model.

Pemodelan Sistem dalam simulasi meliputi :

- Sistem yang memodelkan symbol (identitas) permasalahan secara simbolik.
- System identitas semu, dimana kondisi dari luar diberikan secara rekaan.
- System laboratorik, dipergunakan berbagai komponen seperti adanya operator, software, hardware, computer, prosedur operasi, fungsi matematis, statistic, dll.

➤ **Aturan Verifikasi dan Validasi dalam Simulasi**

Ketika membangun model simulasi sistem nyata, kita harus melewati beberapa tahapan atau level pemodelan. Pertama kita harus membangun model konseptual yang memuat elemen sistem nyata. Dari model konseptual ini dibangun model logika yang memuat relasi logis antara elemen sistem juga variabel eksogenus yang mempengaruhi sistem. Model kedua ini sering disebut sebagai model diagram alur. Menggunakan model diagram alur ini, lalu dikembangkan program komputer, yang disebut juga sebagai model simulasi, yang akan mengeksekusi model diagram alur.

Pengembangan model simulasi merupakan proses iteratif dengan beberapa perubahan kecil pada setiap tahap. Dasar iterasi antara model yang berbeda adalah kesuksesan atau kegagalan ketika verifikasi dan validasi setiap model. Ketika validasi model dilakukan, kita mengembangkan representasi kredibel sistem nyata, ketika verifikasi dilakukan kita memeriksa apakah logika model diimplementasikan dengan benar atau tidak. Karena verifikasi dan validasi berbeda, teknik yang digunakan untuk yang satu tidak selalu bermanfaat untuk yang lain.

Baik untuk verifikasi atau validasi model, kita harus membangun sekumpulan kriteria untuk menilai apakah diagram alur model dan logika internal adalah benar dan apakah model konseptual representasi valid dari sistem nyata. Bersamaan dengan kriteria evaluasi model, kita harus spesifikasikan siapa yang akan mengaplikasikan kriteria dan menilai seberapa dekat kriteria itu memenuhi apa yang sebenarnya.

Simulasi adalah eksperimentasi secara abstrak tentang suatu system, maka dimungkinkan diperoleh suatu kesimpulan berkenaan dengan sistem tersebut dengan ciri :

- a. Tanpa harus membangun sistem, jika kita ingin mengevaluasi suatu sistem yang belum ada.
- b. Tanpa mengganggu sistem, jika kita ingin mempelajari sistem yang tengah beroperasi dan melakukan suatu eksperimen pada sistem amatlah melakukan suatu eksperimen pada sistem amatlah mahal ataupun berbahaya.
- c. Tanpa harus menghancurkan sistem, misalnya kita mempunyai tujuan untuk menentukan limit tekanan mempunyai tujuan untuk menentukan limit tekanan pada suatu sistem.

Keuntungan dari system simulasi yakni :

- Menghemat waktu
- Dapat melebar luaskan waktu prediksi analisis
- Dapat mencakup berbagai variasi sumber
- Dapat mengoreksi kesalahan perhitungan
- Dapat dihentikan sesuai dengan jumlah data yang dikehendaki
- Mudah diulang-ulang dengan random number yang berubah

## E. Penelitian Terkait

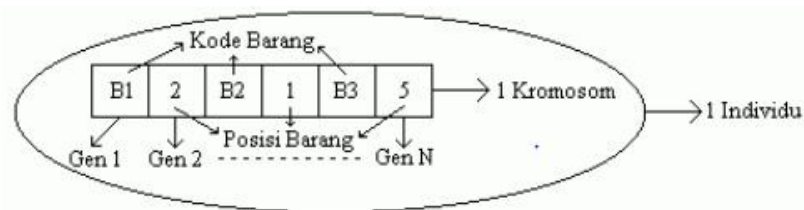
### 1. Optimasi Pola Penyusunan Barang Dalam Ruang Tiga Dimensi Menggunakan Metode Genetic Algorithms

Penelitian ini dilakukan oleh Beny Hariyanto, mahasiswa jurusan Teknik Informatika, Universitas Kristen Petra. Dalam penelitian ini ia menggunakan 3 operator genetic algorithm, yakni operator reproduksi (*Reproduction*), operator pindah silang (*Crossover*), dan operator mutasi (*Mutation*). Di dalam penelitiannya tersebut ia memaparkan relevansi antara algoritma genetika dan teknik pemrograman yang ia lakukan dan menampilkan table yang berisi istilah-istilah algoritma genetic yang digunakan dalam optimasi pola penyusunan barang.

**Tabel 2.4 Relevansi Algoritma Genetika dengan Teknik Pemrograman**

1.	Gen	Setiap gen mewakili satu barang beserta posisinya didalam kromosom
2.	Alele	Nilai dari gen, terdiri dari kode barang, panjang, lebar, dan tinggi barang
3.	Lokus	Posisi dari gen pada suatu kromosom
4.	Genotip	Penulisan rangkaian kode barang dalam kromosom
5.	Kromosom	Terdiri dari gen-gen sebanyak input dari user
6.	Penotip	Pembacaan rangkaian kode barang dalam kromosom untuk disusun dalam ruang tiga dimensi
7.	Evaluasi	Proses penyusunan barang (gen) dalam ruang untuk mendapatkan <i>fitness</i> dari setiap individu.
8.	Individu	Terdiri dari sebuah kromosom yang memiliki <i>fitness</i> tertentu
9.	Populasi	Terdiri dari beberapa individu
10.	Seleksi	Proses seleksi suatu individu yang ada dalam populasi berdasarkan nilai probabilitasnya
11.	Fitness	Nilai dari individu yang mewakili satuan ruang yang terisi oleh barang (semakin tinggi nilai fitness semakin optimal pula pola penyusunan barang)

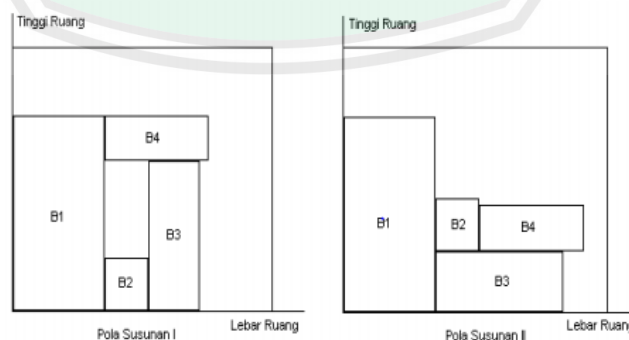
Berikut diberikan contoh ilustrasi penempatan kode-kode (coding) barang beserta posisinya sebagai gen-gen dalam suatu rangkaian kromosom pada satu individu:



**Gambar 2.8 Posisi Barang Pada Kromosom**

Pada gambar tersebut dapat dijelaskan bahwa sebuah individu memiliki satu kromosom yang terdiri dari banyak gen, dimana setiap gen yang ada mewakili satu barang dengan kode barang sebagai identitasnya. Sedangkan setiap gen yang berada disamping gen yang berisikan kode barang mewakili satu dari enam posisi penempatan barang yang ada dalam ruang tiga dimensi.

Perbedaan mendasar dalam penelitian ini yaitu pada saat melakukan perhitungan fitness dimana perhitungan tersebut dibagi menjadi 3 kelompok, yaitu  $\frac{1}{4}$ ,  $\frac{1}{2}$ , dan  $\frac{3}{4}$  tinggi ruang pada container. Pembagian perhitungan tersebut dilakukan untuk memperoleh hasil fitness yang terbaik dan mengantisipasi apabila terjadi fitness yang sama besar dari dua individu. Keadaan tersebut digambarkan sebagai berikut :



**Gambar 2.19 Susunan Barang dengan Perbedaan Tinggi Ruang**

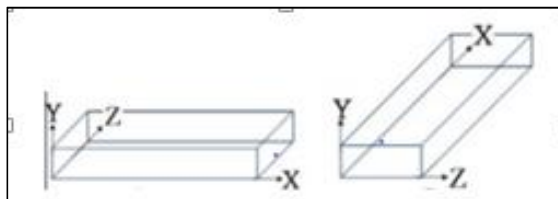
Pada gambar diatas dapat dilihat bahwa fitness yang didapat dari pola susunan I dan pola susunan II adalah sama besar jika dipakai teknik menghitung jumlah satuan ruang yang terisi oleh barang, karena seluruh barang dapat dimasukkan ke dalam ruang. Tetapi jika dilihat secara tampilan tiga dimensi, maka pola susunan II lebih baik untuk diterapkan daripada pola susunan I karena tidak terdapat rongga kosong diantara tumpukan barang.

## **2. Penyusunan Barang Dalam Kontainer Dengan Memperhatikan Aspek Berat Benda Menggunakan Metode Hybrid Algoritma Genetik**

Penelitian ini difokuskan untuk menangani permasalahan pemanfaatan maksimal pada tahap *3D bin packing* pada kasus penyusunan barang karena pada kasus sebelumnya hanya diterapkan pada pola penyusunan dalam bentuk 2D saja. Operator algoritma yang digunakan yakni operator seleksi, *crossover*, dan mutasi. Dalam penelitian ini tidak hanya digunakan algoritma genetic saja, melainkan terdapat algoritma lain yang digunakan dalam proses penentuan penyusunan barang, yakni algoritma DBLF yang merupakan hasil perkembangan dari algoritma BLF. Pada algoritma DBLF, benda digerakkan sejauh mungkin ke arah dalam (sejajar sumbu z), kemudian sejauh mungkin ke arah bawah (sejajar sumbu y), lalu sejauh mungkin ke arah kiri (sejajar sumbu x).

Pada penelitian ini dimensi barang diasumsikan dalam sebuah table yang di dalamnya terdiri dari kode barang, panjang, lebar, dan tinggi yang mempunyai masing-masing nilai. Kemudian disesuaikan dengan asumsi panjang, lebar, dan tinggi container agar diperoleh posisi penempatan barang menurut koordinat  $x$ ,  $y$ , dan  $z$ . Teknik penyusunan barangnya dilakukan satu per satu yakni penentuan posisi

barang dengan kode 0, 1, dan seterusnya. Pada penempatan barang terdapat 2 cara yakni seperti dalam gambar berikut :



**Gambar 2.10 Posisi Barang dengan Koordinat Berbeda**

Pertama, pengaturan pola peletakan sesuai dengan posisi awal barang sesuai dengan koordinatnya, misal  $x$  (panjang),  $y$  (tinggi), dan  $x$  (lebar). Apabila diperlukan perubahan posisi maka barang tersebut akan diputar atau diganti asumsi koordinatnya menjadi  $z$  (panjang), dan  $x$  (lebar), dengan  $y$  tetap sebagai tinggi barang.

Pada proses perhitungan fitnessnya dibagi menjadi dua, yakni fitness utilitas dan fitness berat. Untuk fitness utilitas menggunakan algoritma DBLF dalam penyelesaiannya. Sedangkan fitness berat menggunakan metode konvensional dalam penyelesaiannya. Penerapan metode dalam penelitian ini sangat kompleks sehingga hasil yang dicapai lebih kompleks pula dalam menyelesaikan suatu permasalahan.

### **3. Optimasi Penataan Silinder Dalam Kontainer Dengan Algoritma**

#### **Genetika**

Penelitian ini dilakukan oleh mahasiswa Politeknik Elektronika Negeri Surabaya. Namun objek yang digunakan berupa silinder dimana penelitian ini dilakukan untuk menyelesaikan suatu permasalahan mengenai proses optimasi penempatan silinder dalam sebuah container dalam berbagai macam ukuran. Asumsi silinder yang dipakai dalam penelitian ini yakni silinder pejal. Metode atau

solusi yang digunakan yakni pencarian koordinat dan algoritma genetika untuk menentuka pola penyusunan silinder yang tepat. Pada proses pencarian koordinat, dilakukan perhitungan sesuai dengan diameter masing-masing silinder menggunakan rumus Paul Bourke. Sedangkan pada proses implemntasi algoritma genetic terdapat beberapa tahap, yakni representasi kromosom, proses inisialisasi, evaluasi, operator tukar silang, operator mutasi, dan seleksi. Program ini dibangun menggunakan bahasa pemrograman Java dan diaplikasikan dalam bentuk objek 3 Dimensi. Implementasi algoritma genetic dalam penelitian ini bersifat secara umum dalam arti tidak menggunakan algoritma lain dalam pencarian solusinya, sehingga masih diperlukan pengembangan penelitian lain untuk mencapai hasil yang lebih kompleks.





## **BAB III**

### **PERANCANGAN SISTEM**

#### **3.1 Desain Penelitian**

Pembuatan program simulasi optimasi penataan barang pada container ini dibagi menjadi beberapa tahap penelitian. Hal ini dilakukan agar proses pengerjaan program dapat terkonsep dan terjadwal dengan baik, sehingga peneliti tidak kesulitan dalam melalui proses pengerjaan. Berikut tahap-tahap penelitian yang dilakukan :

##### **1. Identifikasi Masalah**

Tahap ini merupakan awal darimana suatu permasalahan muncul dan bagaimana solusi untuk mengatasinya menurut metode/algortima tertentu. Permasalahan dari penelitian ini yaitu bagaimana menentukan urutan penataan barang pada container yang tepat dan optimal berdasarkan tujuan, berat, dan jenis barang menggunakan Algoritma Genetika.

##### **2. Kajian Pustaka**

Dalam hal ini dilakukan kajian terhadap literature yang membahas tentang Algoritma Genetika dan Prosedur Pengiriman Barang yang dilakukan oleh perusahaan yang bergerak di bidang Jasa Pengiriman Barang.

##### **3. Analisis Data**

Tahap analisis data merupakan tahap pengelompokan data, yakni mengambil beberapa data dari file data mentah untuk kemudian diproses sesuai kebutuhan implementasi algoritma genetika.

##### **4. Desain Sistem**

Desain sistem merupakan penjelasan secara umum bagaimana sistem itu dibuat, komponen apa saja yang ada didalamnya. Dalam program ini desain sistemnya yaitu bagaimana implementasi algoritma genetika digunakan dalam membangun sistem ini.

#### 5. Desain Aplikasi

Tahap ini merupakan tahap perincian bagaimana prosedur menjalankan program, bagaimana desain *interface*-nya, dan apa output yang diharapkan.

#### 6. Membangun Aplikasi

Tahap ini merupakan tahap pembuatan program atau proses melakukan *coding* dalam mengimplementasikan algoritma genetika menggunakan dua bahasa pemrograman, yakni Java dan Matlab.

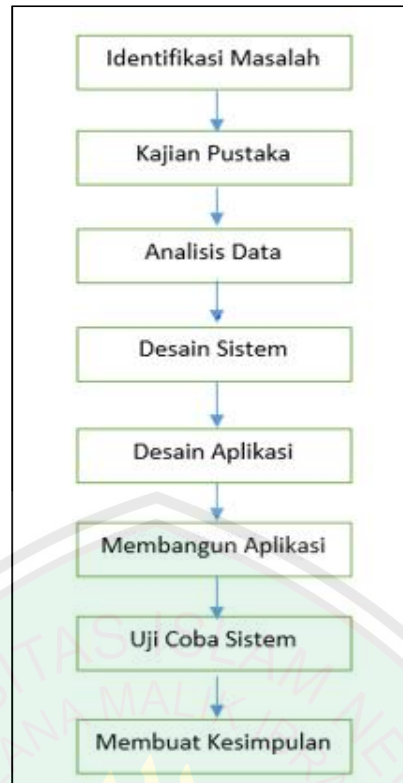
#### 7. Uji Coba Sistem

Tahap ini merupakan tahap melakukan *testing* terhadap program yang sudah selesai dibuat. Apakah sudah benar dalam implementasi algoritma genetika dan apakah outputnya sesuai dengan yang diharapkan.

#### 8. Membuat Kesimpulan

Tahap ini merupakan tahap penjelasan proses secara keseluruhan dalam sistem yang dibuat dan dikemas dalam suatu kalimat yang ringkas.

Berikut diagram alur pembuatan program :



**Gambar 3.1 Desain Penelitian Sistem**

### 3.2 Analisa Data

Data yang akan diproses berupa data mentah dalam bentuk file Excel. Data tersebut diperoleh dari studi kasus perusahaan yang bergerak di bidang jasa pengiriman barang, yakni PT.ANTESS yang berlokasi di daerah Sidoarjo. Data yang didapatkan lebih kompleks karena keterangan barang dijelaskan secara terperinci. Berikut isi dari file Excel tersebut :

**Tabel 3.1. Data Barang**

<b>Kode barang</b>	<b>Dimensi barang</b>	<b>Berat barang</b>	<b>Keterangan</b>
- Berupa rangkaian/gabungan dari huruf dan angka	- Nilainya berupa bilangan	- Nilainya berupa bilangan	- LED TV, Lemari es, dan berbagai



- Nomor, merupakan data **nomor urut** barang yang dimasukkan ke dalam Excel dan dijadikan sebagai *primary key* pada inisialisasi kromosom nantinya.
- Untuk nilai dimensi dan berat barang dilakukan pembulatan untuk mempermudah dalam proses perhitungannya.

### **3.3 Identifikasi Kebutuhan**

Program Simulasi 3D Optimasi Penataan Barang pada Kontainer ini dibuat dengan 3 software, yaitu Netbeans IDE 7.1, Matlab R2008a, dan XAMPP 1.7.3. Netbeans IDE 7.1 digunakan untuk implementasi algoritma genetika dan program menentukan titik koordinat barang di container. Matlab R2008a digunakan untuk membuat program visualisasi objek 3D. XAMPP 1.7.3 sebagai aplikasi database MySQL sebagai tempat untuk menyimpan hasil dari algoritma genetika dan program penentuan titik koordinat barang.

### **3.4 Perancangan Sistem**

Program Simulasi 3D Penataan Barang pada Kontainer Menggunakan Algoritma Genetika ini terbagi menjadi 2 program, yaitu program yang dibangun menggunakan bahasa pemrograman Java dan Matlab. Proses yang ada dalam program Java merupakan proses bagaimana implementasi algoritma genetika dalam menentukan urutan peletakan barang yang tepat agar pengisian barang dalam container dapat dilakukan secara maksimal. Selain itu juga terdapat proses menentukan titik koordinat barang dalam container. Program Matlab hanya digunakan untuk memvisualisasikan objek barang dan kontainernya dalam bentuk 3D.

### 1.4.1 Program Java

Tahap awal yang dilakukan dalam program ini adalah input data barang ke database. Data barang yang diinputkan berupa file *excel*. File tersebut merupakan file data barang yang sudah diatur komponen datanya dari data mentah. Setelah file tersebut diinputkan, maka secara otomatis akan tersimpan di dalam database.

Tahap selanjutnya menerapkan data barang tersebut ke dalam algoritma genetika. Pertama dilakukan inisialisasi kromosom untuk mendapatkan beberapa solusi yang dibangkitkan secara acak. Hasilnya disimpan dalam database kemudian diproses sesuai konsep algoritma genetika untuk mendapatkan urutan peletakan barang di container secara optimal. Optimal tidaknya ditinjau dari nilai *fitness* yang dihasilkan dalam algoritma genetika. Hasil urutan peletakan barang yang optimal tersebut ditampilkan melalui table di program Java kemudian disimpan ke database.

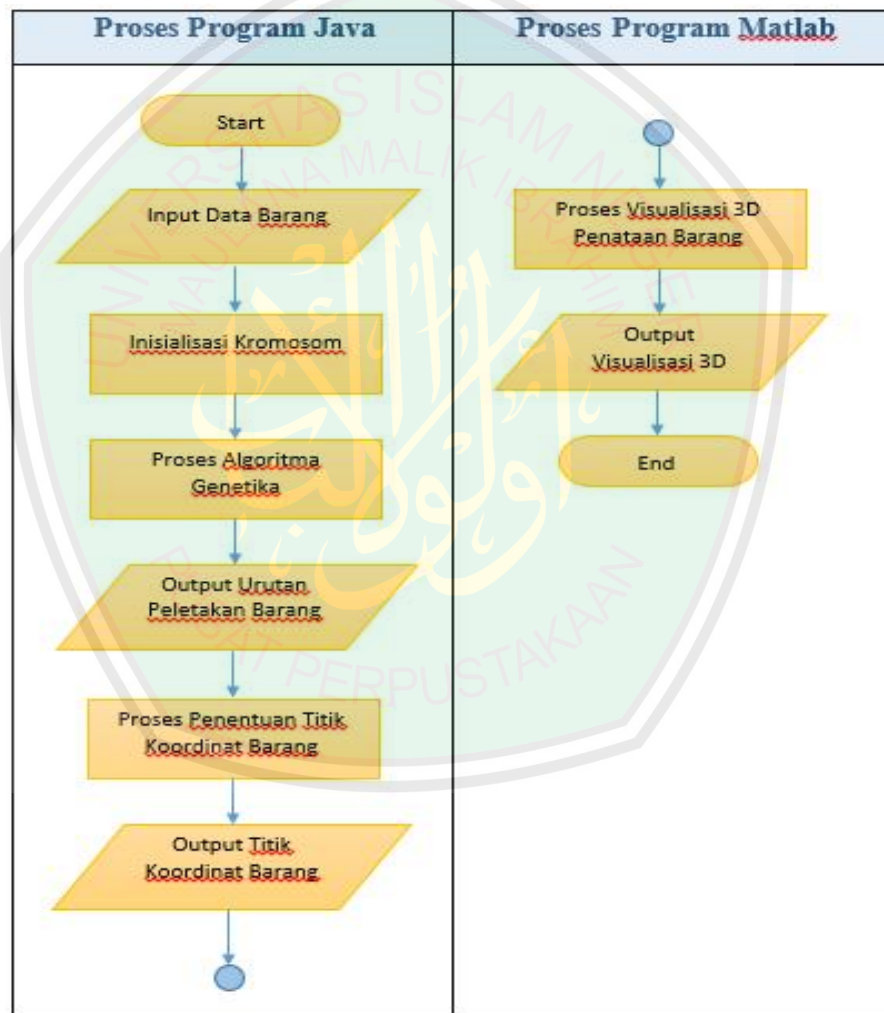
Tahap terakhir dari program Java adalah proses penentuan titik koordinat barang dalam container berdasarkan output dari algoritma genetika yaitu urutan peletakan barang yang paling optimal. Titik koordinat yang dihasilkan disimpan di database dan digunakan untuk proses visualisasi 3D barang pada container yang dilakukan pada program Matlab. Jenis container yang digunakan yaitu container 20 feet, 40 feet, dan 45 feet.

### 3.4.2 Program Matlab

Tahap awal yang dilakukan dalam program ini adalah melakukan koneksi database untuk mendapatkan komponen data output urutan peletakan barang yang optimal menurut algoritma genetika beserta titik koordinat barang dalam container

yang dihasilkan dari program sebelumnya. Komponen yang diambil adalah identitas barang (nomor dan tujuan barang), dimensi barang (panjang, lebar, tinggi), dan titik-titik koordinat barang sesuai dengan dimensi container yang dipilih. Selanjutnya komponen tersebut diproses untuk mendapatkan bentuk 3 dimensi penataan barang dalam container.

Dari penjelasan proses yang ada dalam masing-masing program di atas, maka dapat diperoleh diagram alur sistem secara keseluruhan yaitu sebagai berikut :

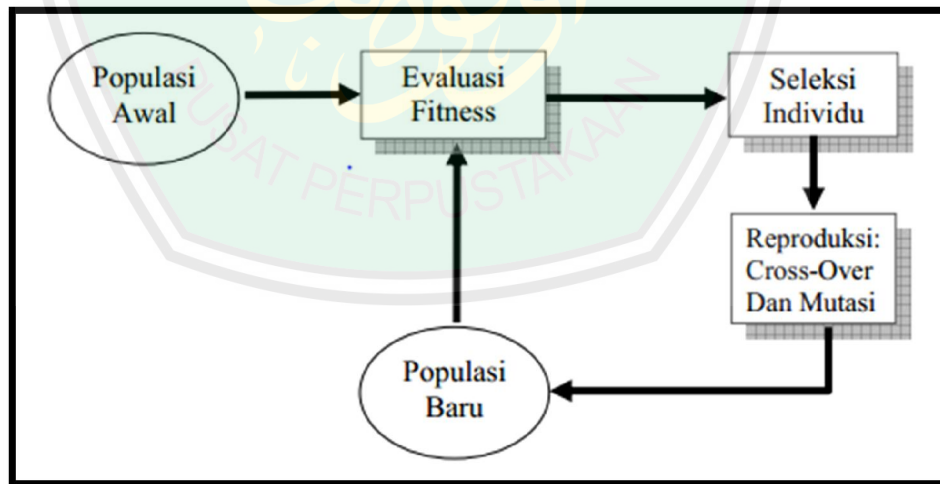


Gambar 3.2 Flowchart Sistem

### 3.5 Model Genetika

Dalam pembuatan program ini, algoritma genetika diimplementasikan menggunakan bahasa pemrograman Java untuk menentukan urutan peletakan barang yang tepat dan optimal. Parameter optimal tidaknya yaitu dilihat dari nilai *fitness* yang didapatkan pada tiap kromosomnya. Nilai *fitness* merupakan nilai *fitness* secara keseluruhan, yakni gabungan dari *fitness volume*, *fitness berat*, dan *fitness jenis* tiap kromosom.

Teknik pengkodean yang digunakan dalam masalah ini adalah pengkodean permutasi. Solusi atau kromosom dibangkitkan secara *random*. Metode seleksi yang digunakan adalah seleksi roda roulette (*Roulette Wheel*). Sedangkan untuk metode *crossover* yang digunakan adalah *crossover* untuk representasi kromosom permutasi dan metod mutasi yang digunakan adalah *swapping mutation*. Berikut siklus secara umum algoritma genetika yang pertama kali dikenalkan oleh David Goldberg :



Gambar 3.3 Siklus Algoritma Genetika oleh David Goldberg

### a. Inisialisasi Kromosom

Pada program ini inisialisasi kromosom menunjukkan urutan barang yang dihasilkan secara random yang komponennya berupa nomor urut barang yang terdapat pada file *excel*. Pertama mengambil beberapa file pada database data barang yaitu **nomor, berat, jenis, tujuan, dan dimensi barang (panjang, lebar, tinggi)**. Kemudian masing-masing komponen diberi nilai dan dihitung volume barangnya untuk mempermudah melakukan proses selanjutnya.

#### - Jenis

Data barang yang diinputkan terdiri dari 3 jenis, yaitu Televisi, Kulkas, dan Makanan. Ketiga jenis barang diberi bobot nilai dengan tipe bilangan *integer*. Pemberian nilai ini bertujuan agar barang yang mempunyai berat paling besar dimasukkan terlebih dahulu ke dalam container.

- Kulkas = 1
- Televisi = 2
- Makanan = 3

#### - Tujuan

Data barang yang diinputkan terdiri dari 3 tujuan, yaitu Jakarta, Bandung, dan Semarang. Ketiga tujuan tersebut juga diberi bobot nilai dengan tipe bilangan *integer*. Pemberian nilai tersebut bertujuan agar barang yang mempunyai tujuan paling jauh dimasukkan terlebih dahulu ke dalam container.

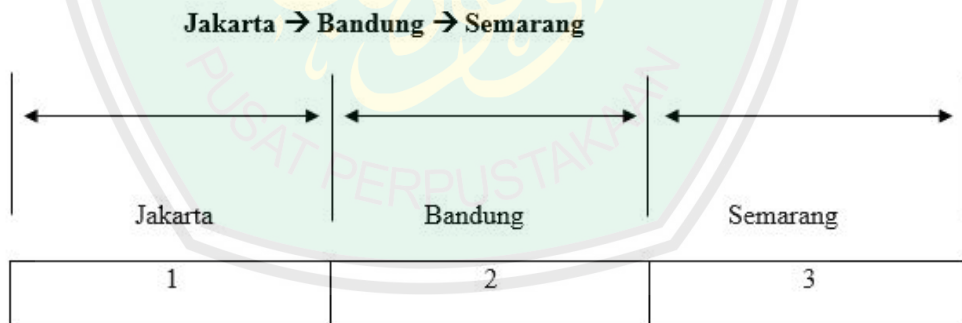
- Jakarta = 1
- Bandung = 2
- Semarang = 3

- **Volume**

Melakukan proses perhitungan volume terhadap data dimensi barang (panjang, lebar, tinggi). Perhitungan volume barang sesuai dengan rumus :

$$Volume = p \times l \times t$$

Tahap selanjutnya adalah penempatan gen-gen dalam kromosom. Tahap ini merupakan tahap pemebentukan individu. Individu disini merupakan solusi urutan peletakan barang pada container. Gen yang dimaksud adalah data barang. Gen-gen tersebut berupa **nomor** barang yang urutannya dibangkitkan secara acak, namun tetap memperhatikan parameter **tujuan barang**. Peletakan barang tersebut dilakukan sesuai urutan lokasi tujuan barang yaitu dari terjauh sampai terdekat dari lokasi pengiriman. Pengurutan letak barang dalam kromosom tersebut dilakukan untuk mempermudah proses pengambilan barang ketika tiba di tujuan. Berikut ilustrasi kromosom yang diberikan :



❖ **Kromosom 1**

	← Jakarta					← Bandung					← Semarang →				
Nomor	13	2	21	14	6	7	9	10	16	20	4	1	15	23	22
Berat															
Tujuan															
Jenis															
Volume															

Kromosom yang ditampilkan hanya **nomor** barang dengan aturan **tidak boleh ada angka yang sama**. Hal itu dikarenakan **nomor** barang bertindak sebagai *primary key* yang akan digunakan pada proses selanjutnya.

**b. Pembangkitan Populasi Awal**

Populasi terdiri dari beberapa individu, dan setiap individu terdiri dari satu kromosom. Individu merupakan solusi urutan peletakan barang pada container. Populasi dibangkitkan sebanyak inputan dari *user* yang disebut dengan *Popsiz*e. Bilangan acak random yang dibangkitkan digunakan untuk mengacak **posisi** gen (barang) dalam kromosom. Berikut source code membangkitkan bilangan random :

```
Random rnd =new Random();
tmp = rnd.nextInt(gen)+1;
```

### c. Perhitungan Nilai *Fitness*

Nilai *fitness* menyatakan optimal atau tidaknya solusi yang dihasilkan. Nilai *fitness* tersebut diperoleh dari rumus yang sudah ditentukan berdasarkan masalah yang akan diselesaikan. Dalam penelitian ini parameter optimal tidaknya berdasarkan berat, dimensi barang (volume), dan jenis, yaitu *fitness berat*, *fitness utilitas* (volume), dan *fitness keseluruhan*.

#### 1. *Fitness Berat*

Penentuan nilai *fitness berat* suatu susunan barang dilihat dari penalty yang dihasilkan. Penalty merupakan nilai kesalahan atau pelanggaran. Semakin banyak nilai penalty yang dihasilkan, maka semakin kecil nilai *fitness berat* yang diperoleh, begitu juga sebaliknya. Berikut aturan pemberian nilai penalty terhadap kesalahan urutan peletakan barang :

$$P = \begin{cases} 1, (M_{Bawah} < \sum M_{Atas}) \\ 0, (M_{Bawah} \geq \sum M_{Atas}) \end{cases}$$

Ket : P = Penalti

$M_{bawah}$  = Berat pada susunan bawah

$M_{atas}$  = Berat pada susunan atas

Jika barang yang berada di bawah lebih berat daripada barang yang berada di atasnya maka tidak dikenakan penalty. Akan tetapi jika sebaliknya maka dikenakan penalti sebesar 1. Hal itu dikarenakan barang yang ditata pada bagian bawahnya harus lebih berat daripada barang di atasnya.

Apabila seluruh nilai penalty sudah didapatkan maka *fitness berat* dapat dihitung dengan persamaan di bawah ini :

$$F_b = \frac{B_{tersusun} - \sum P}{B_{tersusun}}$$

Ket :  $F_b$  = Fitness berat

$B_{tersusun}$  = Banyaknya barang yang tersusun

$P$  = Penalti yang diperoleh

## 2. *Fitness Utilitas*

Perhitungan perbandingan antara volume container dengan jumlah volume barang yang masuk akan menghasilkan suatu perkiraan dan kemungkinan jumlah barang yang masuk ke dalam container. Untuk penentuan fitness utilitas atau menghitung fitness volume yakni sebagai berikut :

$$F_u = \frac{\sum V_{Barang}}{V_{Kontainer}}$$

Ket :  $F_u$  = Fitness utilitas

$V_{Barang}$  = Volume barang yang tersusun

$V_{Kontainer}$  = Volume container

## 3. *Fitness Jenis*

Perhitungan dari *fitness* jenis sama rumusnya dengan perhitungan *fitness* berat. Hanya nilai yang diolah berbeda. Nilai dalam perhitungan *fitness* ini adalah hasil inisialisasi kromosom pada data jenis, yaitu Kulkas = 1, Televisi = 2, dan Makanan = 3. Pemberian nilai penaltynya merupakan kebalikan dari cara perhitungan fitness berat, yaitu jika nilai barang ke- $k$  lebih kecil atau sama dengan nilai barang ke- $(k+1)$ , maka penalty = 0. Selain itu nilai penalty nya sama dengan 1.

#### 4. *Fitness* Keseluruhan

Nilai *fitness* keseluruhan merupakan nilai *fitness objektif* yang digunakan sebagai parameter apakah suatu kromosom (solusi) yang dihasilkan optimal/tidak dalam menentukan urutan peletakan barang dalam container. Nilai *fitness* ini merupakan gabungan dari beberapa nilai *fitness* yang sudah dihitung tadi, yaitu *fitness berat*, *fitness volume*, dan *fitness jenis*. Berikut rumus mencari nilai *fitness* keseluruhan :

$$F = (P_u \times F_u) + (P_b \times F_b) + (P_j \times F_j)$$

Ket :

F = *Fitness* keseluruhan

P<sub>b</sub> = Porsi *fitness* berat

P<sub>u</sub> = Porsi *fitness* utilitas

F<sub>b</sub> = *Fitness* berat

F<sub>u</sub> = *Fitness* utilitas

P<sub>j</sub> = Porsi *fitness* jenis

F<sub>j</sub> = *Fitness* jenis

Penempatan gen-gen (barang) yang disusun dalam kromosom (solusi) mempunyai aturan, yakni sesuai dengan urutan tujuan barang. Sehingga nilai *fitness* (*fitness* keseluruhan) yang diberikan dihitung per tujuan dahulu. Kemudian dari ketiga tujuan tersebut digabung dalam 1 kromosom. Jadi, urutan perhitungannya dalam satu kromosom mulai dari tujuan **Jakarta** → **Bandung** → **Semarang**.

$$F_{kromosom} = F_{Jakarta} + F_{Bandung} + F_{Semarang}$$

Semakin tinggi nilai *Fitness* Kromosom, semakin optimal solusi urutan peletakan barang dalam container yang dihasilkan.

#### d. Seleksi

Seleksi bertujuan menentukan individu-individu mana saja yang akan dipilih untuk dilakukan rekombinasi (crossover dan mutasi) dan bagaimana *offspring* terbentuk dari individu-individu terpilih tersebut. Metode seleksi yang dipilih adalah *Roulette Wheel Selection*. Berikut algoritma seleksi dengan roda roulette atau *Roulette Wheel Selection* :

1. Hitung total fitness (F):

$$\text{TotFitness} = \sum F_k; k = 1, 2, \dots, \text{popsize}$$

2. Hitung fitness relatif setiap individu:

$$P_k = F_k / \text{TotFitness}$$

3. Hitung fitness kumulatif:

$$q_1 = p_1$$
$$q_k = q_{k-1} + p_k; k = 2, 3, \dots, \text{popsize}$$

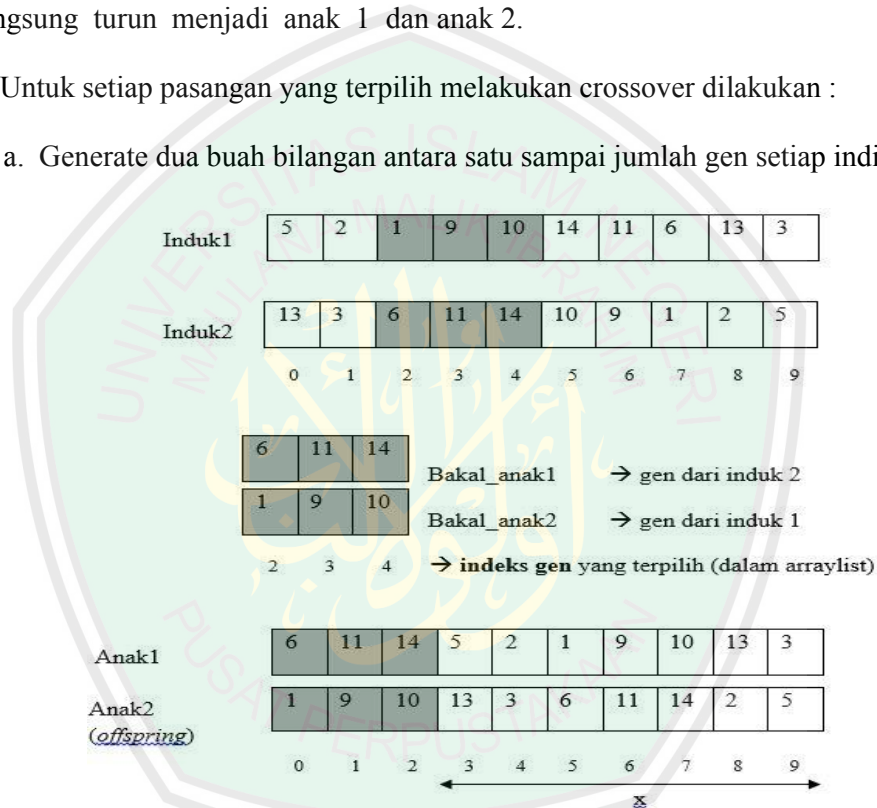
4. Pilih induk yang akan menjadi kandidat untuk di-crossover dengan cara:

- Bangkitkan bilangan random  $r$
- Jika  $q_k \leq r$  dan  $q_{k+1} > r$  maka pilih kromosom ke  $k+1$  sebagai kandidat induk.

#### e. Perkawinan Silang (*Crossover*)

*Crossover* (perkawinan silang) dilakukan atas 2 kromosom untuk menghasilkan kromosom anak (*offspring*). Kromosom anak yang terbentuk akan mewarisi sebagian sifat dari kromosom induknya. Dalam permasalahan ini metode *crossover* yang digunakan adalah *crossover* untuk representasi kromosom permutasi. Berikut algoritma *crossover* :

1. Tentukan probabilitas crossover yang diinginkan (biasanya antara range 0.75 sampai 0.9).
2. Untuk setiap populasi sebanyak jumlah populasi lakukan:
  - a. Generate sebuah bilangan random antara nol sampai satu.
  - b. Jika bilangan tersebut lebih kecil daripada probabilitas crossover, maka lakukan crossover.
  - c. Sedangkan apabila bilangan tersebut lebih besar, maka induk 1 dan induk 2 langsung turun menjadi anak 1 dan anak 2.
3. Untuk setiap pasangan yang terpilih melakukan crossover dilakukan :
  - a. Generate dua buah bilangan antara satu sampai jumlah gen setiap individu.



**Gambar 3.4 Proses Crossover**

Pemilihan gen-gen yang akan di-*crossover* pada permasalahan ini ditentukan berdasarkan tujuan barang. Sehingga bilangan random yang dibangkitkan sesuai

dengan panjang gen tiap tujuan yang ada dalam sebuah kromosom. Berikut ilustrasi pemilihan gen yang diberikan :

- Titik potong 1 : **Bangkitkan bilangan random dari 0 sampai panjang gen Jakarta - 1.**
- Titik potong 2 : **Bangkitkan bilangan random dari panjang gen Jakarta sampai (panjang gen Jakarta + panjang gen Bandung) - 1.**
- Titik potong 3 : **Bangkitkan bilangan random dari (panjang gen Jakarta + panjang gen Bandung) sampai panjang kromosom - 1.**

**Panjang gen Jakarta** = banyaknya barang dengan tujuan Jakarta dalam suatu kromosom.

**Panjang gen Bandung** = banyaknya barang dengan tujuan Bandung dalam suatu kromosom.

**Panjang gen Semarang** = banyaknya barang dengan tujuan Semarang dalam suatu kromosom.

**Panjang kromosom** = banyaknya barang yang diinputkan.

#### f. Mutasi

Mutasi merupakan proses mengubah nilai dari satu atau beberapa gen dalam suatu kromosom. Individu yang telah melewati proses seleksi dan crossover akan menghasilkan individu baru (offspring) yang akan dimutasi untuk membantu mempercepat terjadinya perbedaan individu pada populasi.

Berikut algoritma *swapping mutation/exchange mutation* :

1. Bangkitkan dua bilangan random (antara 0 sampai panjang kromosom) untuk setiap individu, sebanyak jumlah individu dalam populasi.

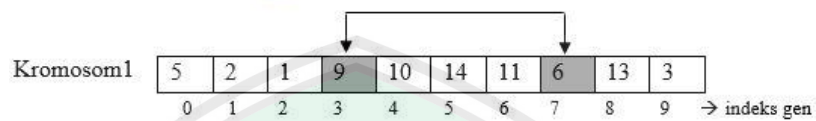
2. Tukarkan gen dengan lokasi kedua bilangan tersebut dengan cara:

bilanganTemp = gen1;

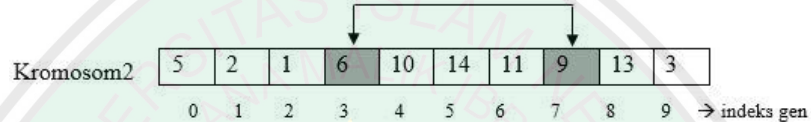
gen1 = gen2;

gen2 = bilanganTemp;

Sebelum mutasi (*Parent 1*):



Sesudah mutasi (*Offspring*):



**Gambar 3.5 Proses Mutasi**

Pemilihan gen-gen yang akan dimutasi pada permasalahan ini sama halnya pada proses crossover yaitu ditentukan berdasarkan tujuan barang. Sehingga bilangan random yang dibangkitkan sesuai dengan panjang gen tiap tujuan yang ada dalam sebuah kromosom.

### 3.6 Konsep Penataan Barang

Konsep penataan barang bertujuan agar barang dapat ditata maupun diambil dengan mudah dari dalam container. Proses penataan barang menggunakan konsep dari perusahaan ekspedisi barang dan peneliti memodifikasinya sesuai dengan algoritma genetika. Berikut beberapa aturan yang ada dalam konsep penataan barang dalam container :

1. Manipulasi Data

Data dimensi barang dan berat yang diinputkan oleh *user* harus bilangan bulat. Pembulatan yang dilakukan adalah pembulatan puluhan. Satuan yang digunakan dalam dimensi barang adalah cm, sedangkan berat barang adalah Kg.

- Dimensi Barang

Panjang 83.5 cm menjadi 84 cm

- Berat Barang

Makanan = 3,71 kg menjadi 4 kg

## 2. Penempatan Barang dalam Kontainer

Penempatan barang dilakukan sesuai dengan urutan peletakan barang yang dihasilkan dari pencarian solusi optimal menggunakan algoritma genetika. Urutan penataan barang dalam container berawal dari penataan ke arah **Lebar → Tinggi → Panjang Kontainer**. Dilakukan penataan barang ke dalam container jika memenuhi kondisi sebagai berikut :

1. Jika **(jumlah berat barang yang tertata + berat barang yang akan masuk) < berat maks container**

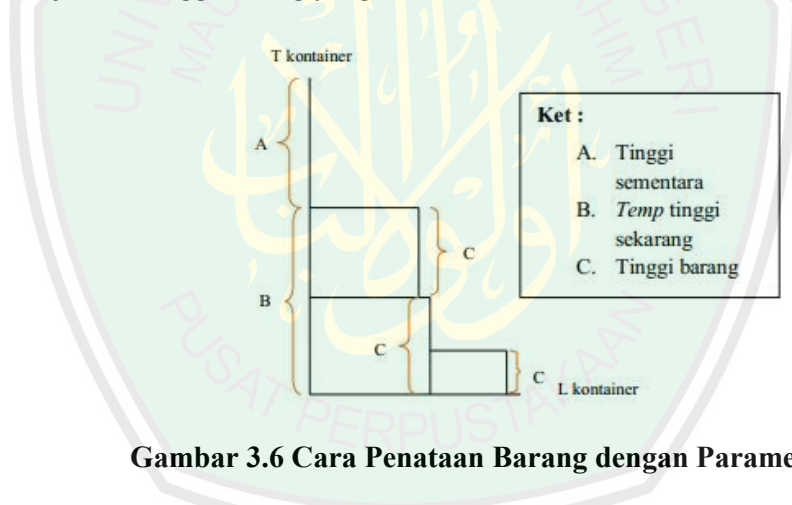
Dilakukan proses penataan barang jika berat barang sekarang ditambah dengan barang yang akan masuk kurang dari berat maksimum container. berat barang sekarang diperoleh dari jumlah berat barang yang sudah tertata di container. Berat container beragam sesuai dengan jenis container yang dipilih.

2. Jika **(jumlah lebar barang yang tertata + lebar barang yang masuk) < lebar container**

Dilakukan proses penataan jika lebar barang sekarang kurang dari lebar sementara. Lebar barang sekarang merupakan lebar barang yang akan masuk. Sedangkan lebar sementara diperoleh dari lebar maksimum container dikurangi *temp* lebar sekarang. *Temp* lebar sekarang diperoleh dari penambahan lebar barang yang tertata.

3. Jika **(jumlah tinggi barang yang tertata + tinggi barang yang masuk) < tinggi container**

Dilakukan proses penataan barang jika tinggi barang sekarang kurang dari tinggi sementara. Tinggi barang sekarang merupakan tinggi barang yang akan masuk. Tinggi sementara diperoleh dari maksimal tinggi container dikurangi *temp* tinggi sekarang. *Temp* tinggi sekarang diperoleh dari jumlah tinggi barang yang telah tertata.

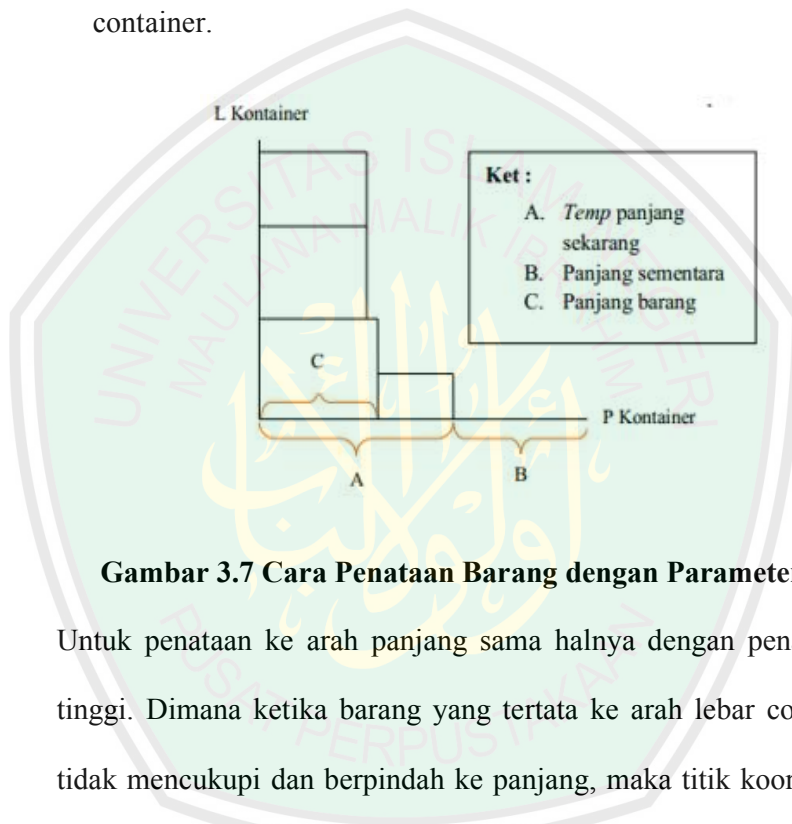


**Gambar 3.6 Cara Penataan Barang dengan Parameter Tinggi**

Untuk penataan ke arah tinggi, diasumsikan bahwa di dalam container terdapat penyangga (pallet) untuk menampung barang di atasnya. Letak penyangga tersebut diambil dari tinggi maksimal barang yang tertata didalam container, sehingga penataanya selalu berada di ketinggian yang sama. Tidak berada tepat di atas barang satu per satu.

4. Jika (**jumlah panjang barang yang tertata + panjang barang yang masuk**) < **panjang container**

Dilakukan proses penataan barang jika panjang barang sekarang kurang dari panjang sementara. Panjang barang sekarang merupakan panjang barang yang akan masuk. Panjang sementara diperoleh dari maksimal panjang container dikurangi *temp* panjang sekarang. *Temp* panjang sekarang diperoleh dari jumlah panjang barang yang telah tertata dalam container.

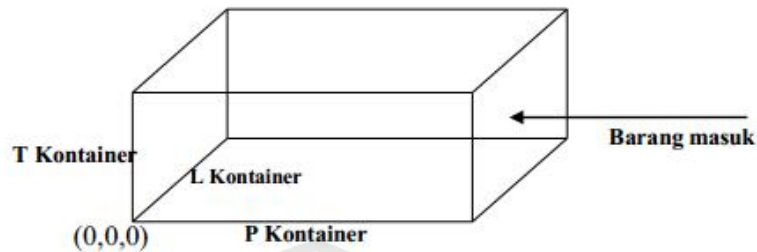


**Gambar 3.7 Cara Penataan Barang dengan Parameter Panjang**

Untuk penataan ke arah panjang sama halnya dengan penataan ke arah tinggi. Dimana ketika barang yang tertata ke arah lebar container sudah tidak mencukupi dan berpindah ke panjang, maka titik koordinat panjang diambil dari ukuran barang paling panjang yang sudah tertata sebelumnya di container.

Barang yang ditata sesuai dengan urutan yang ada dalam output algoritma genetika. Barang yang mendapatkan urutan pertama masuk menempati posisi titik koordinat [0,0,0]. Hal itu dikarenakan barang disusun sejauh mungkin ke arah dalam (sejajar

sumbu  $x$ ), kemudian sejauh mungkin ke arah bawah (sejajar sumbu  $y$ ), lalu sejauh mungkin ke arah kiri (sejajar sumbu  $x$ ). Berikut ilustrasi cara penataan barang ke dalam container :



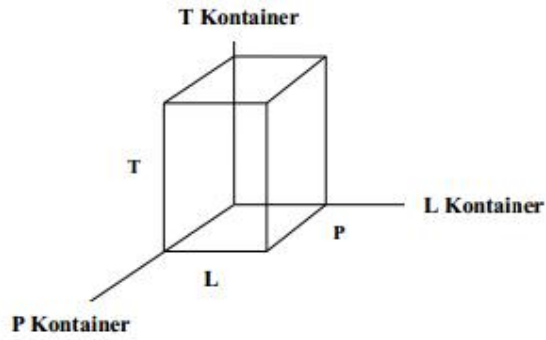
**Gambar 3.8 Peletakan Barang dalam Kontainer**

### 3. Letak Barang dalam Kontainer berdasarkan Tujuan Barang

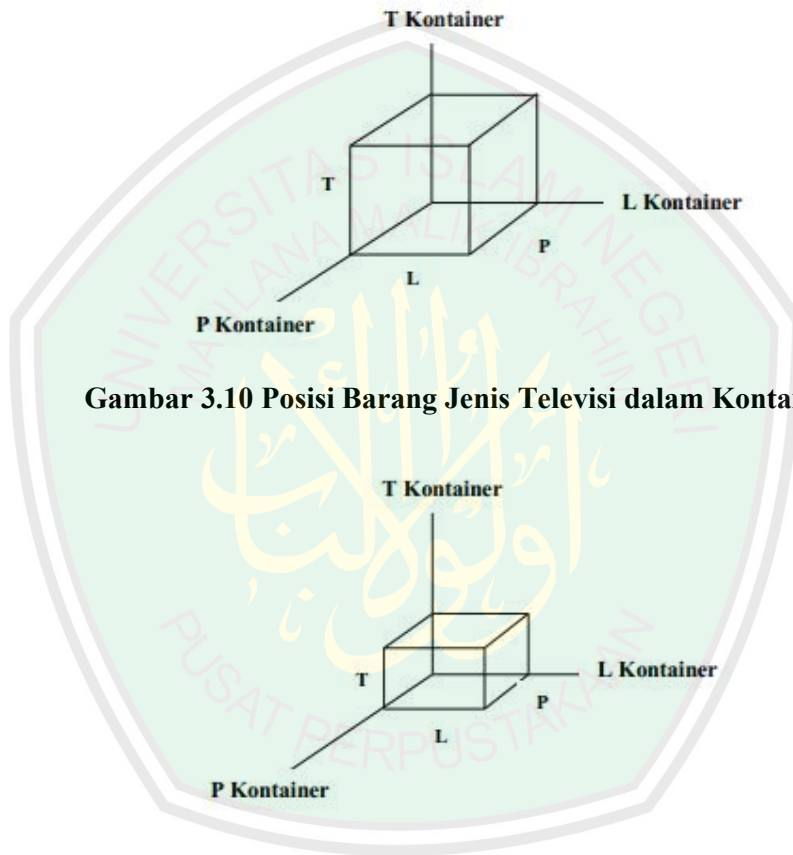
Barang yang ditata dimulai dari barang dengan tujuan paling jauh dahulu sampai yang paling dekat dari lokasi pengiriman. Urutan peletakan barang dimulai dari barang tujuan Jakarta, kemudian Bandung, dan selanjutnya Semarang. Namun, urutan sesuai tujuan barang tersebut sudah ditentukan dalam implementasi algoritma genetika sehingga urutan penataan barang sudah diurutkan berdasarkan tujuan barang.

### 4. Posisi Barang dalam Kontainer berdasarkan Jenis Barang

Dalam penelitian ini hanya terdapat 3 barang, yaitu kulkas, televisei, dan makanan. Barang yang akan ditata dalam container posisinya tidak dapat diubah dari posisi awal barang. Posisi barang dengan jenis kulkas diletakkan berdiri tidak boleh dalam posisi tidur. Begitu juga dengan posisi jenis barang lainnya.



**Gambar 3.9 Posisi Barang Jenis Kulkas dalam Kontainer**



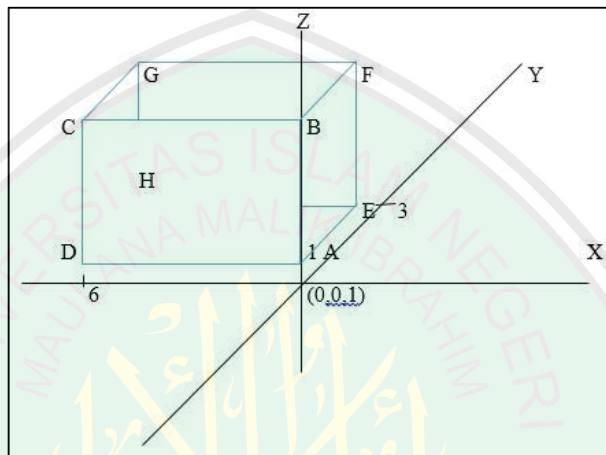
**Gambar 3.10 Posisi Barang Jenis Televisi dalam Kontainer**

**Gambar 3.11 Posisi Barang Jenis Makanan dalam Kontainer**

### **3.7 Konsep Penentuan Titik Koordinat Barang**

Nilai titik koordinat barang didapatkan dari program Java pada tahap proses penataan barang dan hasilnya sudah disimpan dalam database. Nilai titik koordinat

tersebut didapatkan sesuai aturan yang ada pada konsep penataan barang. Nilai titik koordinat ini digunakan untuk proses pembentukan objek 3D barang pada container. Objek 3D yang ingin dibentuk adalah objek kubus atau balok sesuai dengan ukuran barang. Metode yang dipakai dalam membuat objek 3D ini adalah *polygon* yaitu menentukan beberapa titik berdasarkan sumbu  $x$ ,  $y$ ,  $z$  kemudian menghubungkan titik-titik tersebut dengan garis. Berikut ilustrasi penentuan titik koordinat barang :



**Gambar 3.12 Penentuan Titik Koordinat Barang**

Dapat diasumsikan bahwa masing-masing sumbu koordinat menunjukkan :

Sumbu X = Panjang Barang

Sumbu Y = Lebar Barang

Sumbu Z = Tinggi Barang

Kubus/balok terdiri dari 6 sisi permukaan. Masing-masing dicari nilai titik koordinat  $(x,y,z)$  pada setiap titik sudut balok/kubus. Pada contoh gambar diatas posisi barang diletakkan pada titik koordinat  $(0,0,1)$  yang berarti  $X=0$ ,  $Y=0$ ,  $Z=1$  dengan panjang = 6, lebar = 3, dan tinggi = 5.

Sisi pertama yang dicari yaitu sisi permukaan ABCD.

Nilai titik A = (0,0,1)

Nilai titik C = (6,0,6)

Nilai titik B = (0,0,6)

Nilai titik D = (6,0,1)

Pada titik B dan titik C nilai koordinat  $z$  bernilai 6. Hal ini karena posisi barang ini berada pada titik (0,0,1). Sehingga titik koordinat  $z = 1 + \text{tinggi barang} = 6$ . Begitu seterusnya hingga semua titik terhubung membentuk kubus/balok.

Proses dari program Matlab yaitu melakukan koneksi database terlebih dahulu untuk mengakses dimensi barang, ukuran container yang dipilih, dan nilai titik koordinat barang. Kemudian langsung di-generate objek 3D barang dan container dengan posisi sesuai nilai titik koordinat masing-masing barang.

### 3.8 Desain Database

Aplikasi database yang digunakan dalam penelitian ini adalah MySQL Database, nama databasenya adalah “**simulasi\_optimasi**”. Berikut ini nama-nama table yang digunakan beserta *field-field* yang terdapat pada masing-masing table.

1. Tabel data\_barang, table ini berfungsi sebagai data barang yang diinputkan oleh *admin* dari file Excel. Berikut struktur dari table data\_barang.

No.	Field Name	Data Type	Field Size
1.	Nomor	Int	50
2.	kode_barang	Text	-
3.	Panjang	Text	-
4.	Lebar	Text	-
5.	Tinggi	Text	-
6.	Berat	Text	-
7.	Jenis	Text	-
8.	tujuan	varchar	100

2. Tabel db1, table ini berfungsi menyimpan hasil inialisasi kromosom dari data yang ada dalam table data\_barang. Dalam table ini ditambahkan pula hasil perhitungan volume barang.

No.	Field Name	Data Type	Field Size
1.	nomor	Text	-
2.	Berat	Text	-
3.	Jenis	Text	-
4.	tujuan	Text	-
5.	volume	Text	-

3. Table login, table ini berfungsi menyimpan data *username* dan *password* yang dimiliki oleh admin.

No.	Field Name	Data Type	Field Size
1.	User	varchar	30
2.	Pass	varchar	30

4. Table menu\_help, merupakan table yang berisi petunjuk penggunaan program maupun informasi lain yang berkaitan dengan program.

No.	Field Name	Data Type	Field Size
1.	algoritma_genetika	Text	-
2.	Inialisasi	Text	-
3.	petunjuk_pengguna	Text	-

5. Table posisi\_barang, merupakan table yang berisi titik-titik koordinat barang yang ada didalam container yang merupakan output dari program *penataan*.

No.	Field Name	Data Type	Field Size
1.	nomor	int	5
2.	panjang	int	10
3.	Lebar	int	10
4.	Tinggi	int	10
5.	Jenis	varchar	20
6.	tujuan	varchar	15
7.	posisi_x_awal	int	10
8.	posisi_y_awal	int	10
9.	posisi_z_awal	int	10
10.	posisi_x_akhir	int	50
11.	posisi_y_akhir	int	50
12.	posisi_z_akhir	int	50

6. Table solusi, table ini berfungsi menyimpan output/hasil dari proses implementasi optimasi melalui algoritma genetika.

No.	Field Name	Data Type	Field Size
1.	nomor	Text	-
2.	kode_barang	Text	-
3.	Berat	Text	-
4.	Jenis	Text	-
5.	tujuan	Text	-

7. Tabel statistik\_gen, table ini berfungsi untuk merekap hasil nilai *fitness* tertinggi suatu kromosom yang dipilih setiap iterasi / generasinya.

No.	Field Name	Data Type	Field Size
1.	Generasi	Text	-
2.	Fitness	Text	-
3.	Indeks	Text	-

8. Table temp\_posisi\_barang, digunakan untuk mengetahui tinggi barang yang paling maksimal pada barang yang sudah tertata di container. Hal ini bertujuan untuk menentukan titik koordinat tinggi (z) untuk barang selanjutnya.

No.	Field Name	Data Type	Field Size
1.	nomor	int	50
2.	panjang	int	50
3.	Lebar	int	50
4.	tinggi	int	50
5.	posisi_x_awal	int	50
6.	posisi_y_awal	int	50
7.	posisi_z_awal	int	50
8.	posisi_x_akhir	int	50
9.	posisi_y_akhir	int	50
10.	posisi_z_akhir	int	50

9. Table temp\_posisi\_barang\_panjang, digunakan untuk mengetahui panjang barang yang paling maksimal pada barang yang sudah tertata di container. Hal ini bertujuan untuk menentukan titik koordinat panjang (x) untuk barang selanjutnya.

No.	Field Name	Data Type	Field Size
1.	nomor	int	10
2.	panjang	int	10
3.	Lebar	int	10
4.	tinggi	int	10
5.	posisi_x_awal	int	10
6.	posisi_y_awal	int	10

7.	posisi_z_awal	int	10
8.	posisi_x_akhir	int	10
9.	posisi_y_akhir	int	10
10.	posisi_z_akhir	int	10

### 3.9 Desain User Interface

*User interface* adalah bagian yang paling tampak dari sebuah program computer yang memungkinkan terjadinya interaksi antara pengguna dengan program computer. Desain interface dibuat berdasarkan proses pengolahan database dari tahap awal input data barang, kemudian menampilkan outputnya dalam bentuk table di Java. Berikut tampilan dari masing-masing form beserta kegunaannya :

#### ✓ Form Login

Form yang digunakan oleh admin untuk menginputkan username dan password terlebih dahulu.

Help ...

SIMULASI 3D OPTIMASI PENATAAN BARANG  
PADA KONTAINER MENGGUNAKAN ALGORITMA GENETIKA

Login Admin

Username

Password

LOGIN

**Gambar 3.13 Form Login**

✓ Form Pencarian Implementasi Algoritma Genetika

Form yang digunakan admin untuk menginputkan data barang yang akan diolah, dan mencari solusi optimal menurut algoritma genetika.

title1	title2	title3	title4	title5	title6	title7	title8

title1	title2	title3	title4	title5	title6	title7	title8

Gambar 3.14 Form Implementasi Algoritma Genetika

✓ Form Detail Hasil

Form untuk melihat hasil inisialisasi kromosom dan statistik generasi.

title1	title2	title3	title4

title1	title2	title3

Solusi terbesar yang dipilih :

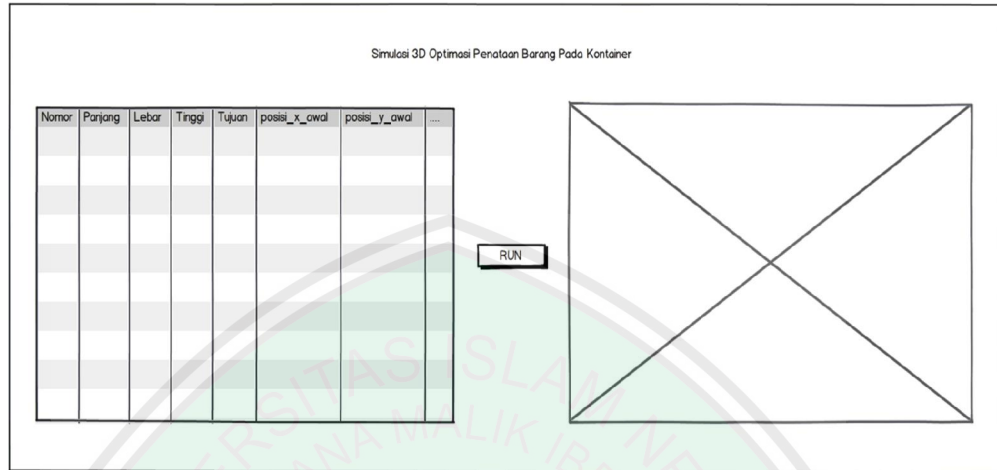
Generasi ke-

Nilai Fitness

Gambar 3.15 Form Detail Hasil

✓ *Form* Visualisasi 3D Penataan Barang di Kontainer

Form ini termasuk dalam program Matlab, berfungsi untuk menampilkan visualisasi objek 3D penataan barang di container.



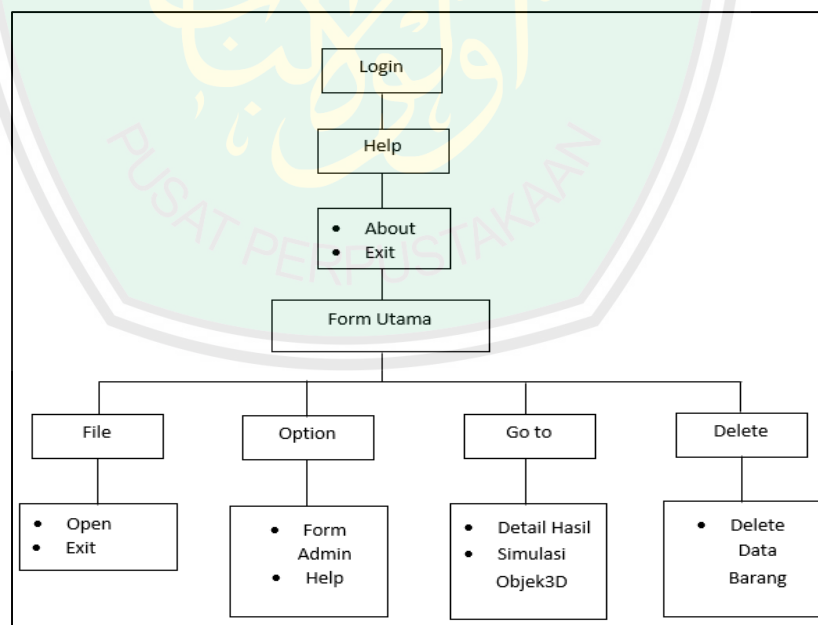
**Gambar 3.16** *Form* Visualisasi 3D

## BAB IV

### IMPLEMENTASI DAN PEMBAHASAN

#### 4.1 Deskripsi Program

Program Simulasi 3D Optimasi Penataan Barang pada Kontainer menggunakan algoritma genetika ini dibangun menggunakan 2 bahasa pemrograman, yaitu Java dan Matlab. Proses yang ada pada program Java yaitu implementasi algoritma genetika untuk mencari solusi urutan peletakan barang yang tepat pada container. Selain itu, juga terdapat proses penataan barang, yaitu menentukan titik koordinat barang yang ditata pada container berdasarkan urutan peletakan barang yang dihasilkan algoritma genetika. Titik koordinat yang dihasilkan dipengaruhi oleh dimensi container yang dipilih, yaitu 20 feet, 40 feet, dan 45 feet. Berikut struktur menu pada program Java :



**Gambar 4.1 Struktur Program Java**

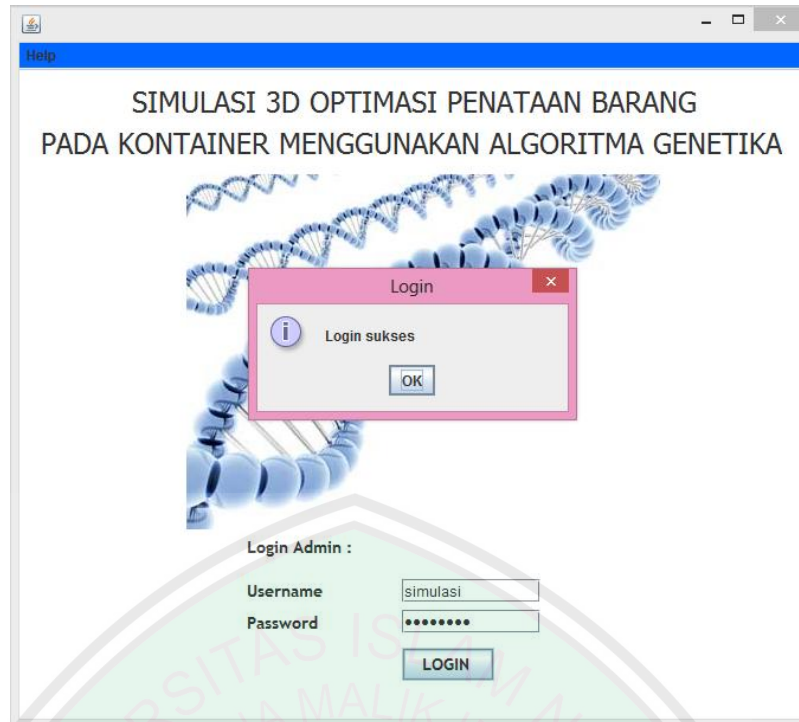
Proses yang ada pada program Matlab hanya visualisasi barang yang ditata di container dalam objek 3D. Hasilnya adalah objek 3D penataan barang pada container berdasarkan titik koordinat yang diperoleh dari proses penataan barang menurut urutan peletakan barang yang dihasilkan algoritma genetika pada program Java.

## **4.2 Penerapan Aplikasi**

Berikut akan dijelaskan tahap-tahap penggunaan program Simulasi 3D Penataan Barang pada Kontainer untuk mendapatkan urutan peletakan barang yang optimal menurut algoritma genetika serta bagaimana proses mendapatkan objek 3D-nya.

### **4.2.1 Implementasi Algoritma Genetika**

Tahap pertama yaitu membuka program Java sehingga halaman utama yang tampil adalah *form login*. Setelah itu *user* menginputkan *username* dan *password* untuk melanjutkan ke proses berikutnya. Jika berhasil maka muncul tampilan sebagai berikut :



**Gambar 4.2 Form Login**

Pada pojok kiri atas terdapat menu *Help*. Menu *Help* ini terdiri dari submenu *About* yang berisi tentang identitas dari pembuat program dan submenu *Exit* jika ingin keluar dari program.



**Gambar 4.3 Form About**

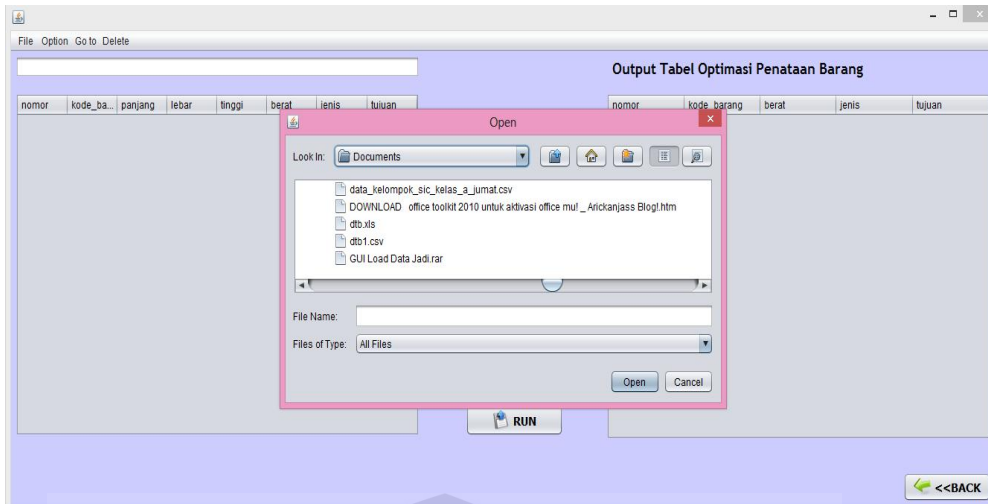
Tahap kedua yaitu muncul *form* utama. *Form* utama ini merupakan form untuk mencari solusi peletakan barang yang optimal menurut algoritma genetika.

Langkah-langkah yang dilakukan yaitu sebagai berikut :

1. Input Data Barang

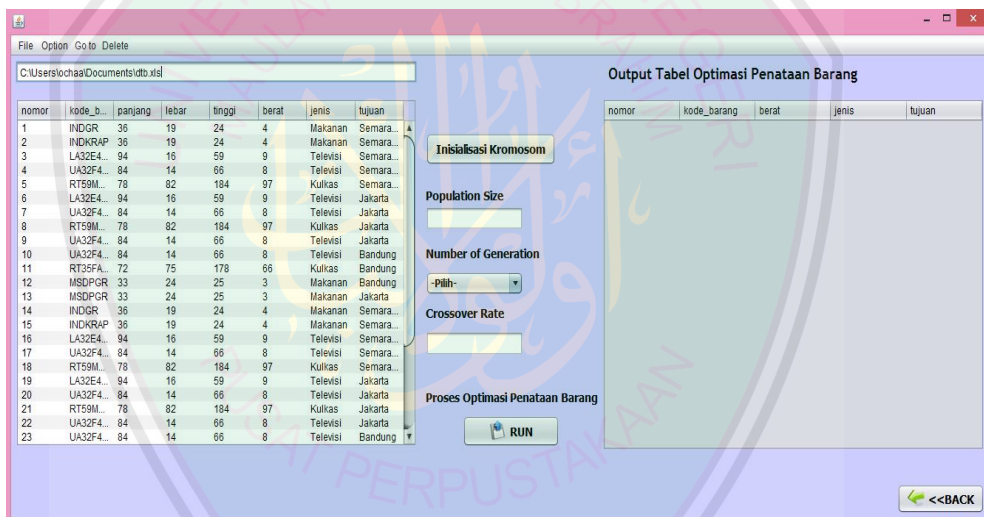
Data barang yang diinputkan berupa file *Excel*. Dalam uji coba ini filenya bernama *dtb.xls*. Cara menginputkannya yaitu :

- Klik menu File → Open. Pilih file excel yang diinginkan.



**Gambar 4.4 Input Data Barang**

Maka otomatis data barang akan tampil pada table sebelah kiri dari form utama. Data barang ini juga langsung disimpan ke database .

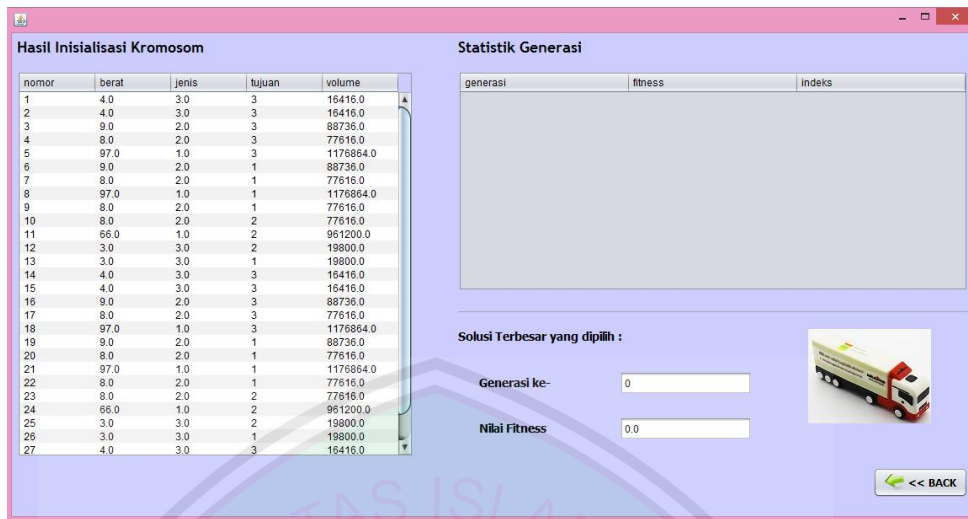


**Gambar 4.5 Tampilan Table Data Barang**

## 2. Inisialisasi Kromosom

- Klik button Inisialisasi Kromosom. Hasilnya akan tampak pada form Detail Hasil..

- Pilih menu Go to → Detail Hasil. Maka hasilnya tampil pada table sebelah kiri.



**Gambar 4.6 Hasil Inisialisasi Kromosom**

Hasil inisialisasi kromosom tersebut juga otomatis tersimpan dalam database. Hasil ini kemudian diproses dalam algoritma genetika.

### 3. Input Parameter Algoritma Genetika

- Pada *form* utama isi parameter-parameter algoritma genetika seperti berikut :

*Population Size* = 30

*Number of Generation* = 5

*Crossover Rate* = 0.45

Pada parameter algoritma genetika tidak terdapat *mutation rate* karena kromosom yang dicrossover langsung dimutasi. Setelah selesai mengisi parameter algoritma genetika, klik button RUN. Tunggu beberapa saat hingga muncul hasilnya pada table sebelah kanan.

The screenshot shows a software window titled "Output Tabel Optimasi Penataan Barang". It features a central control panel with fields for "Population Size", "Number of Generation", and "Crossover Rate", along with a "Pilih" dropdown menu and a "RUN" button. On the left, a table lists 25 items with columns for "nomor", "kode\_b...", "panjang", "lebar", "tinggi", "berat", "jenis", and "tujuan". On the right, another table titled "Output Tabel Optimasi Penataan Barang" shows the optimized sequence of items, with columns for "nomor", "kode\_barang", "berat", "jenis", and "tujuan".

nomor	kode_b...	panjang	lebar	tinggi	berat	jenis	tujuan
1	INDGR	36.0	19.0	24.0	4.0	Makanan	Semara...
2	INDKRAP	36.0	19.0	24.0	4.0	Makanan	Semara...
3	LA32E4...	94.0	16.0	59.0	9.0	Televisi	Semara...
4	UA32F4...	84.0	14.0	66.0	8.0	Televisi	Semara...
5	RT59M...	78.0	82.0	184.0	97.0	Kulkas	Semara...
6	LA32E4...	94.0	16.0	59.0	9.0	Televisi	Jakarta
7	UA32F4...	84.0	14.0	66.0	8.0	Televisi	Jakarta
8	RT59M...	78.0	82.0	184.0	97.0	Kulkas	Jakarta
9	UA32F4...	84.0	14.0	66.0	8.0	Televisi	Jakarta
10	UA32F4...	84.0	14.0	66.0	8.0	Televisi	Bandung
11	RT35FA...	72.0	75.0	178.0	66.0	Kulkas	Bandung
12	MSDPGR	33.0	24.0	25.0	3.0	Makanan	Bandung
13	MSDPGR	33.0	24.0	25.0	3.0	Makanan	Jakarta
14	INDGR	36.0	19.0	24.0	4.0	Makanan	Semara...
15	INDKRAP	36.0	19.0	24.0	4.0	Makanan	Semara...
16	LA32E4...	94.0	16.0	59.0	9.0	Televisi	Semara...
17	UA32F4...	84.0	14.0	66.0	8.0	Televisi	Semara...
18	RT59M...	78.0	82.0	184.0	97.0	Kulkas	Semara...
19	LA32E4...	94.0	16.0	59.0	9.0	Televisi	Jakarta
20	UA32F4...	84.0	14.0	66.0	8.0	Televisi	Jakarta
21	RT59M...	78.0	82.0	184.0	97.0	Kulkas	Jakarta
22	UA32F4...	84.0	14.0	66.0	8.0	Televisi	Bandung
23	RT35FA...	72.0	75.0	178.0	66.0	Kulkas	Bandung
24	MSDPGR	33.0	24.0	25.0	3.0	Makanan	Bandung

nomor	kode_barang	berat	jenis	tujuan
13	MSDPGR	3.0	Makanan	Jakarta
21	RT59MBSL1X...	97.0	Kulkas	Jakarta
9	RT59MBSL1X...	97.0	Kulkas	Jakarta
22	UA32F4510A...	8.0	Televisi	Jakarta
20	UA32F4510A...	8.0	Televisi	Jakarta
9	UA32F4510A...	8.0	Televisi	Jakarta
6	LA32E420E2...	9.0	Televisi	Jakarta
19	LA32E420E2...	9.0	Televisi	Jakarta
7	UA32F4510A...	8.0	Televisi	Jakarta
26	MSDPGR	3.0	Makanan	Jakarta
10	UA32F4510A...	8.0	Televisi	Bandung
25	MSDPGR	3.0	Makanan	Bandung
12	MSDPGR	3.0	Makanan	Bandung
11	RT35FACDS...	66.0	Kulkas	Bandung
24	RT35FACDS...	66.0	Kulkas	Bandung
23	UA32F4510A...	8.0	Televisi	Bandung
28	INDKRAP	4.0	Makanan	Semarang
15	INDKRAP	4.0	Makanan	Semarang
1	INDGR	4.0	Makanan	Semarang
27	INDGR	4.0	Makanan	Semarang
5	RT59MBSL1X...	97.0	Kulkas	Semarang
16	LA32E420E2...	9.0	Televisi	Semarang
14	INDGR	4.0	Makanan	Semarang
18	RT59MBSL1X...	97.0	Kulkas	Semarang
4	UA32F4510A...	8.0	Televisi	Semarang

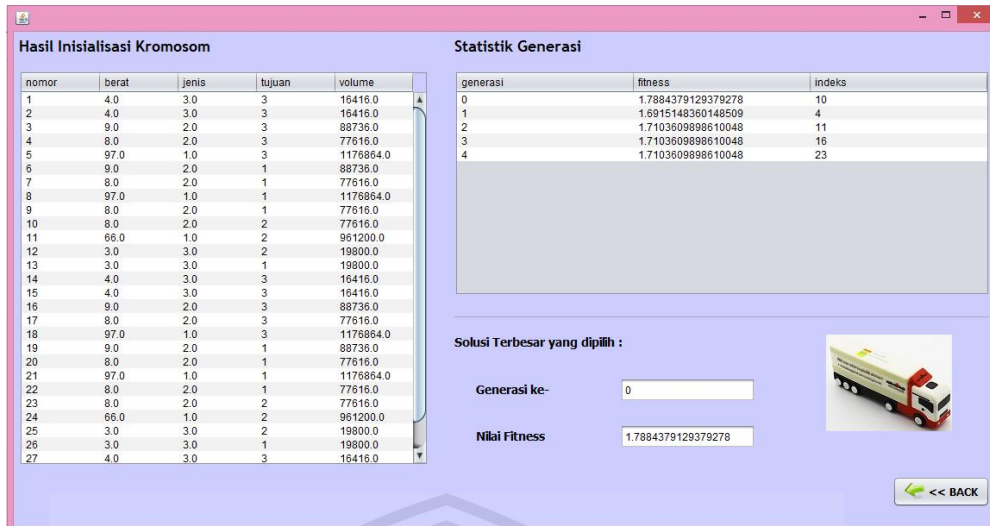
Gambar 4.7 Output Algoritma Genetika

Pada table tersebut merupakan output urutan peletakan barang yang optimal menurut algoritma genetika berdasarkan nilai parameter yang diinputkan sebelumnya. Output tersebut langsung disimpan ke dalam database.

Selama proses pencarian solusi algoritma genetika berlangsung, program menyimpan data yang disebut *statistic generasi*. Statistic generasi merupakan data kromosom-kromosom yang mempunyai nilai *fitness* tertinggi dan dipilih pada tiap generasinya. Lalu dari beberapa generasi tersebut dipilih kromosom yang nilai *fitness*nya paling tinggi. Kromosom (solusi) itulah yang dijadikan sebagai hasil optimasi penataan barang pada container.

Cara mengakses data Statistik Generasi yaitu :

- Pilih menu Go to → Detail Hasil. Maka akan tampil database statistic generasi pada table sebelah kanan.



**Gambar 4.8 Hasil Statistik Generasi**

Data statistic generasi tersebut juga tersimpan otomatis ke database.

#### 4. Form tambahan

Pada program ini terdapat menu untuk menambah 1 atau beberapa data barang. Menu ini disebut *form admin*. Dalam form ini terdapat fasilitas CRUD (*Create, Update, Delete*) data barang. Selain itu pengguna dapat mencari data barang untuk mengetahui identitasnya secara rinci. Berikut form admin.

Form Admin

Nomor:

Kode Barang:

Dimensi Barang :

Panjang:  cm

Lebar:  cm

Tinggi:  cm

Berat:  cm

Jenis:

Tujuan:

**Gambar 4.9 Form Admin**

Apabila user kebingungan dalam menggunakan aplikasi, maka terdapat menu *Help* yang menunjukkan langkah-langkah penggunaan program ini, deskripsi algoritma genetika, dan keterangan mengenai inisialisasi kromosom. Pilih menu **Option** → **Help**.

Help

Petunjuk Pengguna

Petunjuk menggunakan/me-run program ini adalah terbagi menjadi dua bagian, yaitu mencari nilai optimal urutan penataan barang dan menentukan titik koordinat barang pada kontainer.

#Mencari Nilai Optimal Urutan Penataan Barang :

1. Silahkan login sesuai dengan username dan password yang anda miliki sebagai admin.
2. Klik File->Open . Kemudian pilih file data barang yang akan dimasukkan kedalam database dalam format excel.
3. Klik inisialisasi kromosom.
4. Isi textfield pada pilihan population size, pilih number of generation, isi textfield crossover rate.
5. Klik RUN.

#Menentukan Titik Koordinat Penataan Barang di Kontainer :

1. Buka form 'Simulasi Objek3D'. Klik Option --> Simulasi Objek3D
2. Pilih Jenis Kontainer
3. Klik RUN

**Gambar 4.10 Form Help**

Jika ingin menghapus data barang atau menghapus data hasil dari proses lain, maka terdapat menu *Delete* kemudian pilih data apa yang ingin dihapus.

#### 4.2.2 Proses Penataan Barang pada Kontainer

Proses ini bertujuan untuk mendapatkan titik koordinat barang yang ditata ke dalam container sesuai dengan dimensi container yang dipilih. Urutan barang yang diproses untuk mendapatkan titik koordinat ini sesuai dengan urutan peletakan barang yang dihasilkan algoritma genetika. Cara menjalankan proses ini yaitu :

- Pilih menu Go to → Simulasi Objek3D
- Tentukan jenis container yang dipilih → Klik button RUN

Hasilnya akan tampil seperti berikut :

**Tabel Hasil Penataan / Posisi Barang**

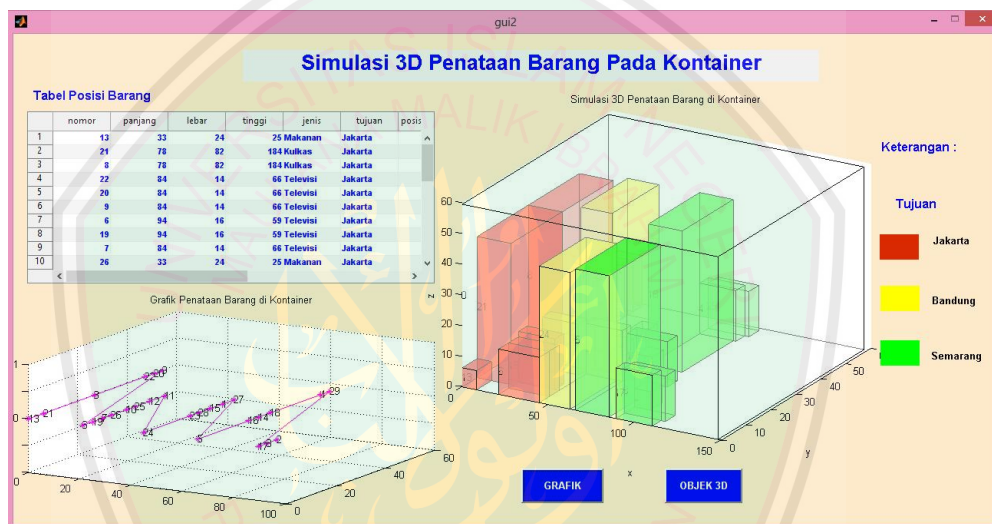
nomor	panjang	lebar	tinggi	jenis	tujuan	x_awal	y_awal	z_awal	x_akhir	y_akhir	z_akhir
13	33	24	25	Makanan	Jakarta	0	0	0	33	24	25
21	78	82	184	Kulkas	Jakarta	0	24	0	78	106	184
8	78	82	184	Kulkas	Jakarta	0	106	0	78	188	184
22	84	14	66	Televisi	Jakarta	0	188	0	84	202	66
20	84	14	66	Televisi	Jakarta	0	202	0	84	216	66
9	84	14	66	Televisi	Jakarta	0	216	0	84	230	66
6	94	16	59	Televisi	Jakarta	84	0	0	178	16	59
19	94	16	59	Televisi	Jakarta	84	16	0	178	32	59
7	84	14	66	Televisi	Jakarta	84	32	0	168	46	66
26	33	24	25	Makanan	Jakarta	84	46	0	117	70	25
10	84	14	66	Televisi	Bandung	84	70	0	168	84	66
25	33	24	25	Makanan	Bandung	84	84	0	117	108	25
12	33	24	25	Makanan	Bandung	84	108	0	117	132	25
11	72	75	178	Kulkas	Bandung	84	132	0	156	207	178
24	72	75	178	Kulkas	Bandung	178	0	0	250	75	178
23	84	14	66	Televisi	Bandung	178	75	0	262	89	66
28	36	19	24	Makanan	Semarang	178	89	0	214	108	24
15	36	19	24	Makanan	Semarang	178	108	0	214	127	24
11	36	19	24	Makanan	Semarang	178	127	0	214	146	24
27	36	19	24	Makanan	Semarang	178	146	0	214	165	24
5	78	82	184	Kulkas	Semarang	262	0	0	340	82	184
16	94	16	59	Televisi	Semarang	262	82	0	356	98	59
14	36	19	24	Makanan	Semarang	262	98	0	298	117	24
18	78	82	184	Kulkas	Semarang	262	117	0	340	199	184
4	84	14	66	Televisi	Semarang	262	199	0	346	213	66
29	94	16	59	Televisi	Semarang	262	213	0	356	229	59
17	84	14	66	Televisi	Semarang	356	0	0	440	14	66

**Gambar 4.11 Titik Koordinat Barang**

Titik koordinat yang dihasilkan tersebut disimpan langsung ke database untuk diproses pada program visualisasi 3D penataan barang pada container di Matlab.

### 4.2.3 Visualisasi 3D Penataan Barang pada Kontainer

Setelah output titik koordinat barang menurut urutan yang dihasilkan algoritma genetika sudah didapatkan, maka selanjutnya menjalankan program visualisasi 3D untuk mengetahui bagaimana objek barang yang ditata ke dalam container dalam bentuk 3D. Program ini terdapat di Matlab. Cara penggunaannya yaitu meng-*klik* button Objek3D untuk mendapatkan hasil objek 3D penataan barang pada container dan button Grafik untuk melihat bagaimana alur penataan barang yang dilakukan. Berikut hasil yang ditampilkan dari program visualisasi 3D :



**Gambar 4.12 Visualisasi 3D Penataan Barang pada Kontainer**

Komponen pada form ini terdiri dari :

- Tabel database titik koordinat barang
- Gambar visualisasi 3D
- Grafik titik koordinat penataan barang

Pada setiap gambar diberi keterangan mengenai tujuan dan nomor barang sebagai identitas barang. Berikut perbedaan warna barang berdasarkan tujuan :

- Jakarta → Merah

- Bandung → Kuning
- Semarang → Hijau

Konsep penataan sendiri jika suatu barang tidak muat pada suatu tempat akan dicarikan tempat lain, bukan dicarikan barang lain karena proses penataan sesuai dengan urutan peletakan barangnya.

### 4.3 Uji Coba

Uji coba dilakukan 2 kali pada proses pencarian solusi penataan barang menggunakan algoritma genetika. Uji coba tersebut dilakukan pada input nilai parameter algoritma genetika yang berbeda untuk mengetahui bagaimana nilai *fitness* yang dihasilkan dan output urutan peletakan barangnya. Selanjutnya dilakukan proses penataan barang pada container yang bertujuan untuk menentukan titik koordinat barang pada container berdasarkan urutan peletakan barang yang optimal menurut algoritma genetika. Jenis kontainer yang digunakan adalah kontainer 20 feet, 40 feet, dan 45 feet. Hasil penataan barang tersebut akan diperjelas dengan visualisasi objek 3D pada Matlab. Kemudian pada masing-masing container akan dilakukan perbandingan penataan barang yang dihasilkan dengan nilai *fitness* yang berbeda.

Pada uji coba yang akan dilakukan jumlah populasi yang diinputkan adalah 30 karena untuk semua jenis permasalahan sebaiknya jumlah populasi tidak kurang dari 30. Sedangkan jumlah generasi diisi sesuai keinginan. Input nilai *crossover rate* sebesar 0.45 dan 0.75. Bila *fitness* dari individu terbaik dipantau pada setiap generasi, maka usulan  $P_c$  (*crossover rate*) direkomendasikan sebesar 0.45. Pernyataan tersebut sesuai dengan implementasi algoritma genetika pada program

ini dalam mencari nilai *fitness* yang optimal. Sedangkan angka 0.75 peneliti mengambilnya karena biasanya nilai *crossover rate* yang direkomendasikan antara range 0.75-0.9. Selan itu, pada implementasi algoritma genetika ini kromosom yang mengalami crossover akan langsung dimutasi. Sehingga apabila nilai parameter mutasi terlalu tinggi, dikhawatirkan akan merusak komponen kromosom anak dan menghilangkan sifat dari induknya.

*Percobaan 1 :*

Uji coba dilakukan dengan input parameter algoritma genetika sebagai berikut :

*Population Size* = 30

*Number of Generation* = 5

*Crossover Rate* = 0.45

Nilai *fitness* yang dihasilkan = 1.7285435178399013

Berikut jumlah barang yang bisa muat ke dalam container sesuai jenis container yang dipilih :

**Table 4.1 Muatan Barang pada Percobaan 1**

Jenis Kontainer	Jumlah Barang Masuk	Jumlah Barang Tidak Masuk
container 20 feet	38	12
container 40 feet	50	0
container 45 feet	50	0

*Percobaan 2 :*

Uji coba dilakukan dengan input parameter algoritma genetika sebagai berikut :

*Population Size* = 30

*Number of Generation* = 5

*Crossover Rate* = 0.75

Nilai *fitness* yang dihasilkan = 1.7990010341797706

Berikut jumlah barang yang bisa muat ke dalam container sesuai jenis container yang dipilih :

**Table 4.2 Muatan Barang pada Percobaan 2**

Jenis Kontainer	Jumlah Barang Masuk	Jumlah Barang Tidak Masuk
container 20 feet	44	6
container 40 feet	50	0
container 45 feet	50	0

#### 4.4 Analisa Hasil

Nilai *fitness* yang dihasilkan dipengaruhi oleh nilai parameter algoritma genetika. Dari hasil tersebut menunjukkan bahwa semakin besar *crossover rate*, semakin besar nilai *fitness* yang dihasilkan. Hal tersebut juga menunjukkan bahwa solusi yang dihasilkan lebih optimal.

Selain itu, penataan barang yang dihasilkan juga dipengaruhi nilai *fitness* yang dihasilkan. Pada uji coba kedua, jumlah barang yang masuk lebih banyak dibandingkan dengan jumlah barang yang masuk pada uji coba pertama dengan jenis container yang sama, yaitu 20 feet. Untuk jenis container 40 feet dan 45 feet jumlah barang yang masuk sama pada uji coba 1 dan 2 yaitu sebanyak 50 barang yang merupakan jumlah seluruh barang.

Perbedaan susunan penataan barang juga semakin terlihat jelas dari visualisasi 3D yang dihasilkan. Hal itu dapat dijabarkan sebagai berikut :

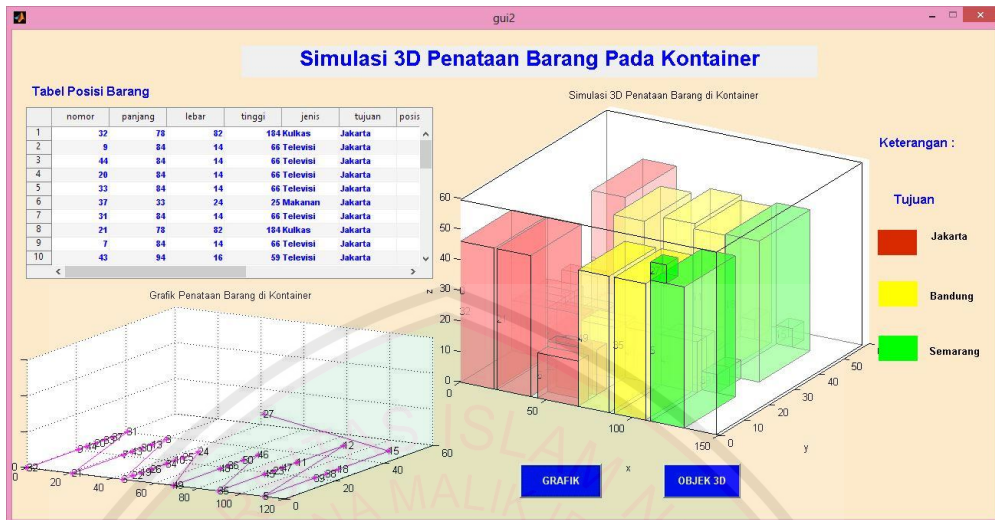
- a. Container 20 feet

Dimensi container : P kontainer = 591.9

L kontainer = 234.0

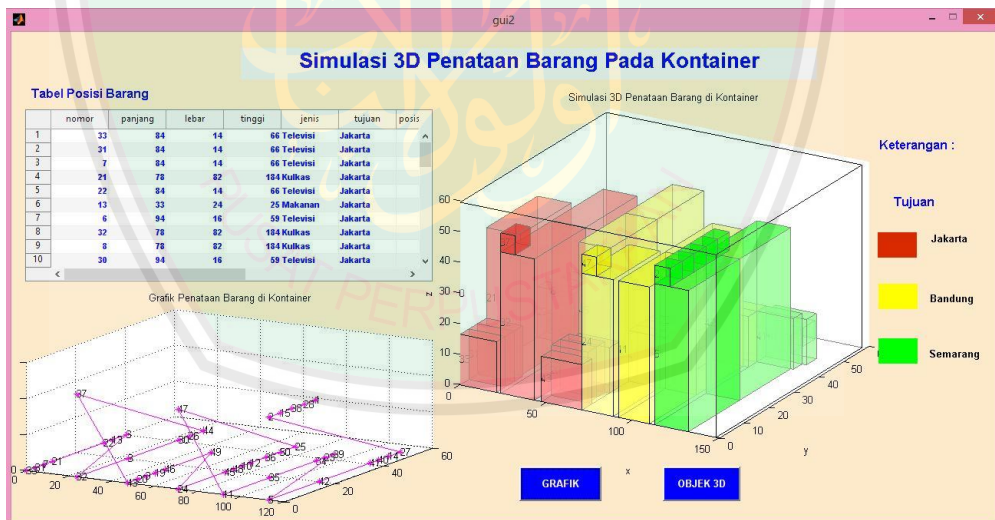
T kontainer = 238.0

- Nilai *fitness* = 1.7285435178399013



Gambar 4.13 Visualisasi 3D Percobaan 1 pada Kontainer 20 feet

- Nilai *fitness* yang dihasilkan = 1.7990010341797706



Gambar 4.14 Visualisasi 3D Percobaan 2 pada Kontainer 20 feet

Dari kedua gambar tersebut dapat dilihat bahwa terdapat perbedaan dalam penataan barang ke arah tinggi. Pada gambar kedua barang yang

berhasil ditata ke arah tinggi lebih banyak daripada gambar pertama yaitu sebanyak 7 buah. Sedangkan pada gambar pertama 1 buah. Hal tersebut menunjukkan bahwa urutan barang yang dihasilkan lebih baik pada percobaan kedua dengan nilai fitness yang lebih besar. Jika penataan barang ke arah tinggi lebih banyak, maka semakin sedikit tempat yang terpakai untuk menata barang. Pada kedua gambar tersebut penataan ke arah panjang sama banyaknya yaitu 6 kolom.

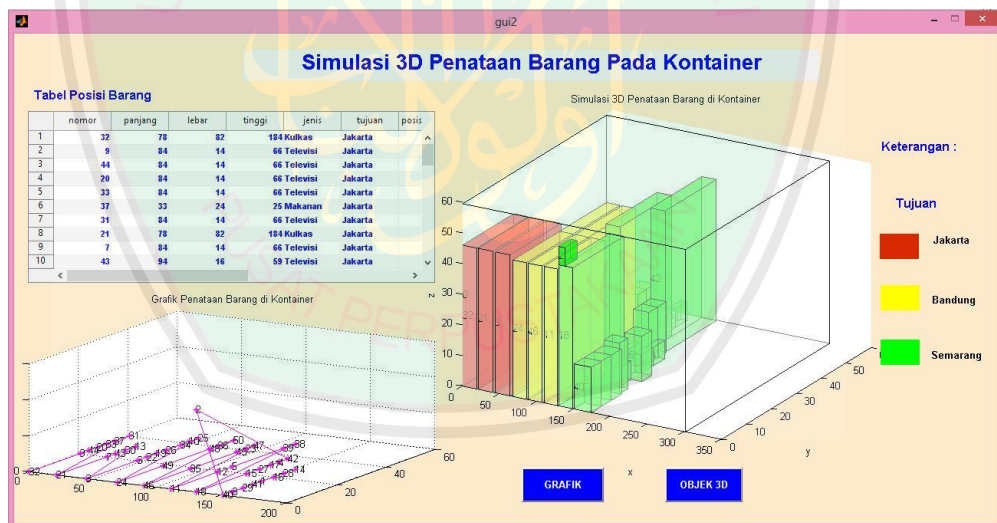
b. Container 40 feet

Dimensi container 40 feet : P kontainer = 1204.5

L kontainer = 230.9

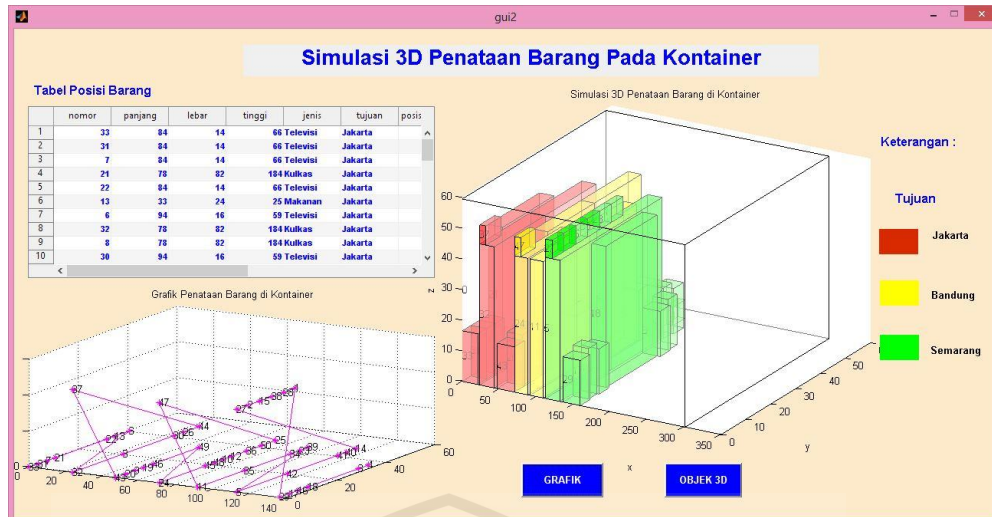
T kontainer = 237.9

- Nilai *fitness* = 1.7285435178399013



Gambar 4.15 Visualisasi 3D Percobaan 1 pada Kontainer 40 feet

- Nilai *fitness* yang dihasilkan = 1.7990010341797706



**Gambar 4.16 Visualisasi 3D Percobaan 2 pada Kontainer 40 feet**

Dari kedua gambar tersebut terlihat bahwa barang yang berhasil ditata ke arah tinggi lebih banyak pada gambar kedua daripada gambar pertama dengan jumlah 1 : 8 buah. Selain itu, untuk penataan ke arah panjang lebih banyak pada gambar pertama dengan jumlah 8 : 7 kolom. Hal itu menunjukkan bahwa tempat yang terpakai untuk menata barang lebih banyak daripada gambar kedua sehingga pengisian lebih optimal pada gambar kedua. Namun, untuk penataan barang ke arah panjang lebih banyak dibandingkan container 20 feet karena lebar container 20 feet lebih besar daripada 40 feet.

Gambar yang dihasilkan pada container 40 feet ini terlihat lebih kecil dan ramping. Hal itu karena dimensi container yang lebih besar sehingga terjadi perbedaan sudut pandang.

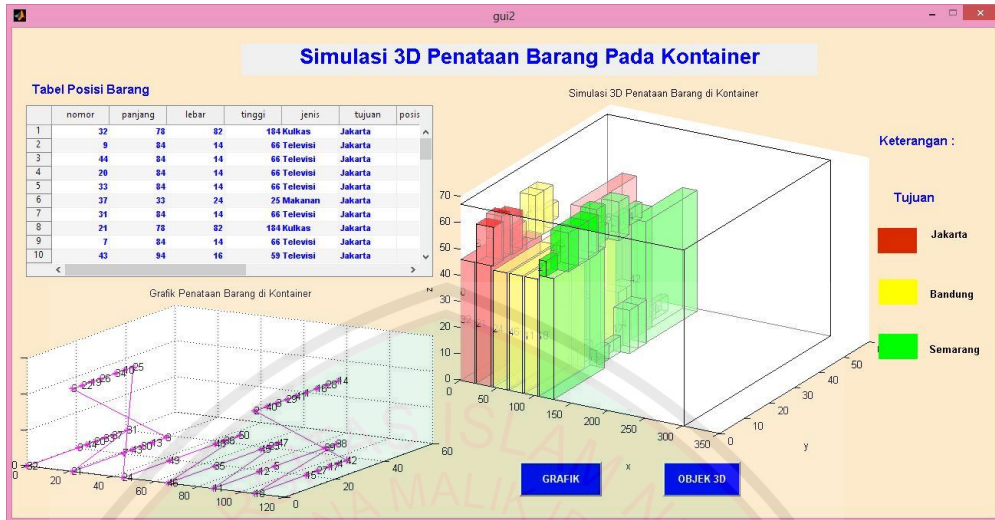
c. Container 45 feet

Dimensi container 40 feet : P kontainer = 591.9

L kontainer = 234.0

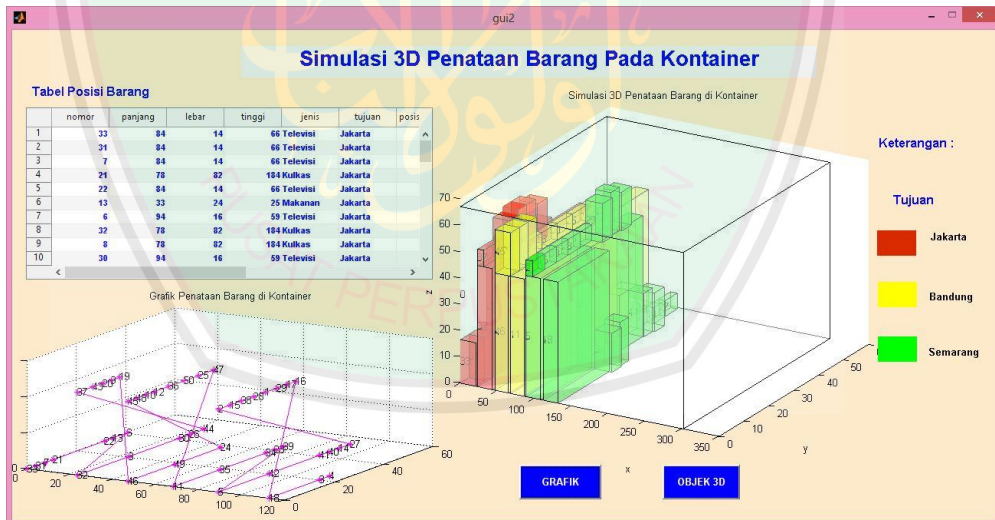
T kontainer = 238.0

- Nilai *fitness* = 1.7285435178399013



Gambar 4.17 Visualisasi 3D Percobaan 1 pada Kontainer 45 feet

- Nilai *fitness* yang dihasilkan = 1.7990010341797706



Gambar 4.18 Visualisasi 3D Percobaan 2 pada Kontainer 45 feet

Dari kedua gambar tersebut dapat dilihat bahwa penataan barang ke arah tinggi pada gambar kedua lebih banyak daripada gambar pertama yaitu

dengan jumlah 16 : 21. Hal ini menunjukkan bahwa pemakaian tempat pada container untuk gambar kedua lebih sedikit daripada gambar pertama. Jumlah penataan barang ke arah panjang lebih sedikit jika dibandingkan pada container 40 feet, namun jumlahnya sama dengan container 20 feet.

Dari hasil visualisasi 3D penataan barang pada container tersebut kadang ditemukan celah antara barang satu dengan barang lain pada penataan ke arah panjang. Pengambilan titik koordinat panjang selanjutnya diambil dari barang terpanjang dalam satu kolom. Begitu juga dengan penataan ke arah tinggi, pengambilan titik koordinat tinggi selanjutnya diambil dari barang tertinggi yang sudah tertata dalam satu kolom.

#### **4.5 Integrasi Program dalam Islam**

Seperti yang sudah disebutkan pada bab sebelumnya, yaitu pada surah An-Nisa' ayat 29 dimana ayat tersebut dengan tegas melarang adanya pihak atau orang yang memakan harta orang lain. Hal itu menunjukkan bahwa dalam kegiatan transaksi tersebut diupayakan untuk mencegah hal-hal yang merugikan salah satu pihak. Kedua belah pihak harus memperoleh persetujuan dan ke-*ridhaan* satu sama lain. Oleh karena itu, untuk memperoleh ke-*ridhaan* pada kedua belah pihak, maka pelayanan harus dilakukan sebaik mungkin agar tidak menimbulkan kekecewaan.

Kegiatan niaga yang dimaksud disini adalah kegiatan transaksi pada perusahaan jasa pengiriman barang. Kegiatan transaksi ini mencakup skala besar karena barang titipannya dalam jumlah banyak yaitu barang hasil produksi suatu perusahaan untuk dikirimkan ke suatu tujuan atau yang dikenal dengan proses distribusi. Di dalam islam, titipan barang disebut juga *wadi'ah*. *Wadi'ah* dapat

diartikan pelimpahan kekuasaan oleh seseorang penitip kepada orang yang menjaga hartanya.

Dalam firman Allah SWT QS. An-Nissa' ayat 58 :

إِنَّ اللَّهَ يَأْمُرُكُمْ أَنْ تُؤَدُّوا الْأَمَانَاتِ إِلَىٰ أَهْلِهَا وَإِذَا حَكَمْتُمْ بَيْنَ النَّاسِ أَنْ تَحْكُمُوا

بِالْعَدْلِ ۚ إِنَّ اللَّهَ نِعِمَّا يَعِظُكُمْ بِهِ ۗ إِنَّ اللَّهَ كَانَ سَمِيعًا بَصِيرًا

Artinya : *“Sesungguhnya Allah menyuruh kamu menyampaikan amanat kepada yang berhak menerimanya, dan (menyuruh kamu) apabila menetapkan hukum di antara manusia supaya kamu menetapkan dengan adil. Sesungguhnya Allah Maha memberi pengajaran yang sebaik-baiknya kepadamu. Sesungguhnya Allah adalah Maha Mendengar lagi Maha Adil.”*

Perusahaan lain menggunakan jasa pengiriman barang berarti mempercayakan barang produksinya untuk menyampaikan kepada tempat yang dituju. Begitu juga perusahaan jasa pengiriman barang harus bertanggungjawab atas keselamatan dan kebaikan barang yang dititipkan (diamanatkan). Dalam hukum islam sikap tanggung jawab merupakan masalah yang penting dan telah terikat pada diri setiap manusia.

Firman Allah SWT QS. Al-Mudatsir (74) ayat 38 :

كُلُّ نَفْسٍ بِمَا كَسَبَتْ رَهِينَةٌ

Artinya : *“Tiap-tiap diri bertanggung jawab atas apa yang telah diperbuatnya.”*

Dalam hal pengangkutan dan pengiriman barang sangat membutuhkan kehati-hatian dalam menjaga barang supaya selamat sampai tujuan. Oleh karena

itu perusahaan berupaya agar dalam proses penataan barang dalam container harus dilakukan secara tepat dan benar untuk mencegah adanya barang yang rusak. Dengan adanya program Simulasi Penataan Barang pada Kontainer Menggunakan Algoritma Genetika ini diharapkan dapat memenuhi kriteria agar barang yang dikirimkan tidak mengalami kerusakan dan proses penataannya dapat dilakukan seoptimal mungkin.



## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Hal-hal yang perlu dilakukan dalam membangun program Simulasi 3D Optimasi Penataan Barang pada Kontainer menggunakan Algoritma Genetika ini adalah :

1. Menentukan model genetika atau menentukan inialisasi kromosomnya berdasarkan permasalahan yang akan diselesaikan.
2. Menentukan parameter optimal/tidaknya urutan peletakan barang. Pada program ini parameter optimal tidaknya ditinjau dari aspek tujuan, berat, volume, dan jenis barang.
3. Menentukan rumus perhitungan nilai *fitness* berdasarkan parameter optimal tidaknya urutan peletakan barang.
4. Menentukan konsep penataan barang pada container.

Program ini terdiri dari beberapa 3 proses utama, yaitu :

1. Pencarian solusi optimal menggunakan algoritma genetika untuk menentukan urutan peletakan barang yang tepat dan optimal.
2. Proses penataan barang pada container untuk menentukan nilai titik koordinat barang sesuai dengan jenis container yang dipilih.
3. Proses Visualisasi 3D penataan barang sesuai jenis container yang dipilih.

Nilai *fitness* dipengaruhi oleh nilai parameter algoritma genetika yang diinputkan. Dari uji coba yang dilakukan terbukti bahwa semakin besar nilai *fitness* yang diperoleh, semakin optimal penataan barang yang dihasilkan. Semakin optimal penataan barang semakin sedikit tempat yang terpakai untuk menata barang pada container. Selain itu, hal yang mempengaruhi optimal tidaknya penataan barang adalah dimensi container yang dipilih. Semakin besar ukuran container semakin banyak barang yang ditampung.

## 5.2 Saran

Perlu dilakukan penelitian lebih lanjut untuk menentukan kombinasi nilai parameter algoritma genetika untuk menghasilkan nilai *fitness* yang lebih optimal lagi. Untuk penataan barang ke arah tinggi perlu dilakukan penelitian agar barang bisa ditata tepat di atas barang lainnya karena dalam penelitian ini diasumsikan terdapat *pallet* atau penyangga untuk menampung barang di atasnya.

## DAFTAR PUSTAKA

Ernawati, Ella. 2011. *Proses Pemuatan Barang ke Dalam Container (Stuffing) pada CV. Manggala Java Art di Klaten*. Universitas Sebelas Maret, Surakarta.

Azizah, Nurul. *Implementasi Algoritma Cat Swarm Optimization untuk Optimasi Pengepakan Barang*. Institut Teknologi Telkom, Bandung.

Evarda Syuman Abhrisa, dkk. *Penyusunan Barang dalam Container dengan Memperhatikan Aspek Berat Menggunakan Metode Hybrid Algoritma Genetik*. Universitas Brawijaya, Malang.

Djajadi, Irawati, dkk. *Perancangan Aplikasi Optimalisasi Muatan pada Kontainer dengan Algoritma Metaheuristic*. Universitas Tarumanegara.

Beny Hariyanto, Gunadi, Kartika, dkk. *Optimalisasi Penyusunan Barang dalam Ruang Tiga Dimensi Menggunakan Metode Genetic Algorithms*. Universitas Kristen Petra.

Ridwan, Mujib. 2009. *Optimasi Penempatan Mahasiswa Baru Di Ma'had Sunan Ampel Al-Ali Universitas Islam Negeri (UIN) Malang Menggunakan Algoritma Genetika*.

Andhika, Tri Hendry. 2012. *PENGLASIFIKASIAN KELAS TA'LIM AFKAR MAHASANTRI BARU MA'HAD SUNAN AMPEL AL-ALI UIN MALIKI MALANG MENGGUNAKAN ALGORITMA GENETIKA*. Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Nur Anwari, Febrianto, dkk. *Perancangan System Optimasi Penjadwalan Pengangkutan Sampah di Surabaya Secara Adaptif Menggunakan Metode Algoritma Genetika*. PENS-ITS Surabaya.

Valensia, Shinta Ayu, dkk. 2013. OPTIMASI RUTE PENGIRIMAN LAUNDRY DENGAN TIME WINDOWS (VRPTW) MENGGUNAKAN ALGORITMA GENETIKA. Malang. Universitas Brawijaya Program Studi Informatika.

Risky Firmansyah, Eka, dkk. 2012. *Algoritma Genetika*. Universitas Islam Negeri Syarif Hidayatullah Jakarta.

Nur Anwari, Febrianto, dkk. PERANCANGAN SISTEM OPTIMASI PENJADWALAN PENGANGKUTAN SAMPAH DI SURABAYA SECARA ADAPTIF MENGGUNAKAN METODE ALGORITMA GENETIKA. PENS - ITS Surabaya.

Rosyid, Danang. 2010. PROSES STUFFING PRODUK FURNITURE PADA RAKABU FURNITURE DI SURAKARTA. Universitas Sebelas Maret, Surakarta.

## LAMPIRAN

- **Table Output Algoritma Genetika Percobaan 1**

nomor	kode_barang	berat	jenis	tujuan
32	RT59MBSL1/XSE	97	Kulkas	Jakarta
9	UA32F4510AMXXD	8	Televisi	Jakarta
44	UA32F4510AMXXD	8	Televisi	Jakarta
20	UA32F4510AMXXD	8	Televisi	Jakarta
33	UA32F4510AMXXD	8	Televisi	Jakarta
37	MSDPGR	3	Makanan	Jakarta
31	UA32F4510AMXXD	8	Televisi	Jakarta
21	RT59MBSL1/XSE	97	Kulkas	Jakarta
7	UA32F4510AMXXD	8	Televisi	Jakarta
43	LA32E420E2MXXD	9	Televisi	Jakarta
30	LA32E420E2MXXD	9	Televisi	Jakarta
13	MSDPGR	3	Makanan	Jakarta
8	RT59MBSL1/XSE	97	Kulkas	Jakarta
6	LA32E420E2MXXD	9	Televisi	Jakarta
22	UA32F4510AMXXD	8	Televisi	Jakarta
19	LA32E420E2MXXD	9	Televisi	Jakarta
26	MSDPGR	3	Makanan	Jakarta
34	UA32F4510AMXXD	8	Televisi	Bandung
10	UA32F4510AMXXD	8	Televisi	Bandung
25	MSDPGR	3	Makanan	Bandung
24	RT35FAACDSA/SE	66	Kulkas	Bandung
49	RT35FAACDSA/SE	66	Kulkas	Bandung
48	UA32F4510AMXXD	8	Televisi	Bandung
36	MSDPGR	3	Makanan	Bandung
50	MSDPGR	3	Makanan	Bandung
46	RT35FAACDSA/SE	66	Kulkas	Bandung

nomor	kode_barang	berat	jenis	tujuan
35	RT35FAACDSA/SE	66	Kulkas	Bandung
45	UA32F4510AMXXD	8	Televisi	Bandung
23	UA32F4510AMXXD	8	Televisi	Bandung
47	MSDPGR	3	Makanan	Bandung
11	RT35FAACDSA/SE	66	Kulkas	Bandung
12	MSDPGR	3	Makanan	Bandung
5	RT59MBSL1/XSE	97	Kulkas	Semarang
39	INDKRAP	4	Makanan	Semarang
38	INDGR	4	Makanan	Semarang
18	RT59MBSL1/XSE	97	Kulkas	Semarang
15	INDKRAP	4	Makanan	Semarang
27	INDGR	4	Makanan	Semarang
17	UA32F4510AMXXD	8	Televisi	Semarang
4	UA32F4510AMXXD	8	Televisi	Semarang
42	RT59MBSL1/XSE	97	Kulkas	Semarang
2	INDKRAP	4	Makanan	Semarang
40	LA32E420E2MXXD	9	Televisi	Semarang
3	LA32E420E2MXXD	9	Televisi	Semarang
29	LA32E420E2MXXD	9	Televisi	Semarang
41	UA32F4510AMXXD	8	Televisi	Semarang
1	INDGR	4	Makanan	Semarang
16	LA32E420E2MXXD	9	Televisi	Semarang
28	INDKRAP	4	Makanan	Semarang
14	INDGR	4	Makanan	Semarang

- Tabel Titik Koordinat Barang Percobaan 1

**Container 20 feet**

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
33	84	14	66	Televisi	Jakarta	0	0	0	84	14	66
31	84	14	66	Televisi	Jakarta	0	14	0	84	28	66
7	84	14	66	Televisi	Jakarta	0	28	0	84	42	66
21	78	82	184	Kulkas	Jakarta	0	42	0	78	124	184
22	84	14	66	Televisi	Jakarta	0	124	0	84	138	66
13	33	24	25	Makanan	Jakarta	0	138	0	33	162	25
6	94	16	59	Televisi	Jakarta	0	162	0	94	178	59
32	78	82	184	Kulkas	Jakarta	94	0	0	172	82	184
8	78	82	184	Kulkas	Jakarta	94	82	0	172	164	184
30	94	16	59	Televisi	Jakarta	94	164	0	188	180	59
26	33	24	25	Makanan	Jakarta	94	180	0	127	204	25
44	84	14	66	Televisi	Jakarta	94	204	0	178	218	66
37	33	24	25	Makanan	Jakarta	94	0	184	127	24	209
43	94	16	59	Televisi	Jakarta	188	0	0	282	16	59
20	84	14	66	Televisi	Jakarta	188	16	0	272	30	66

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
9	84	14	66	Televisi	Jakarta	188	30	0	272	44	66
19	94	16	59	Televisi	Jakarta	188	44	0	282	60	59
46	72	75	178	Kulkas	Bandung	188	60	0	260	135	178
49	72	75	178	Kulkas	Bandung	188	135	0	260	210	178
24	72	75	178	Kulkas	Bandung	282	0	0	354	75	178
45	84	14	66	Televisi	Bandung	282	75	0	366	89	66
48	84	14	66	Televisi	Bandung	282	89	0	366	103	66
10	84	14	66	Televisi	Bandung	282	103	0	366	117	66
12	33	24	25	Makanan	Bandung	282	117	0	315	141	25
36	33	24	25	Makanan	Bandung	282	141	0	315	165	25
50	33	24	25	Makanan	Bandung	282	165	0	315	189	25
25	33	24	25	Makanan	Bandung	282	189	0	315	213	25
47	33	24	25	Makanan	Bandung	282	0	178	315	24	203
11	72	75	178	Kulkas	Bandung	366	0	0	438	75	178
35	72	75	178	Kulkas	Bandung	366	75	0	438	150	178
34	84	14	66	Televisi	Bandung	366	150	0	450	164	66
23	84	14	66	Televisi	Bandung	366	164	0	450	178	66
39	36	19	24	Makanan	Semarang	366	178	0	402	197	24
5	78	82	184	Kulkas	Semarang	450	0	0	528	82	184

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
42	78	82	184	Kulkas	Semarang	450	82	0	528	164	184
41	84	14	66	Televisi	Semarang	450	164	0	534	178	66
40	94	16	59	Televisi	Semarang	450	178	0	544	194	59
14	36	19	24	Makanan	Semarang	450	194	0	486	213	24
27	36	19	24	Makanan	Semarang	450	213	0	486	232	24
2	36	19	24	Makanan	Semarang	450	0	184	486	19	208
15	36	19	24	Makanan	Semarang	450	19	184	486	38	208
38	36	19	24	Makanan	Semarang	450	38	184	486	57	208
28	36	19	24	Makanan	Semarang	450	57	184	486	76	208
1	36	19	24	Makanan	Semarang	450	76	184	486	95	208

### Container 40 feet

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akh
33	84	14	66	Televisi	Jakarta	0	0	0	84	14	66
31	84	14	66	Televisi	Jakarta	0	14	0	84	28	66
7	84	14	66	Televisi	Jakarta	0	28	0	84	42	66
21	78	82	184	Kulkas	Jakarta	0	42	0	78	124	184
22	84	14	66	Televisi	Jakarta	0	124	0	84	138	66
13	33	24	25	Makanan	Jakarta	0	138	0	33	162	25
6	94	16	59	Televisi	Jakarta	0	162	0	94	178	59
32	78	82	184	Kulkas	Jakarta	94	0	0	172	82	184
8	78	82	184	Kulkas	Jakarta	94	82	0	172	164	184
30	94	16	59	Televisi	Jakarta	94	164	0	188	180	59
26	33	24	25	Makanan	Jakarta	94	180	0	127	204	25
44	84	14	66	Televisi	Jakarta	94	204	0	178	218	66
37	33	24	25	Makanan	Jakarta	94	0	184	127	24	209
43	94	16	59	Televisi	Jakarta	188	0	0	282	16	59
20	84	14	66	Televisi	Jakarta	188	16	0	272	30	66

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
9	84	14	66	Televisi	Jakarta	188	30	0	272	44	66
19	94	16	59	Televisi	Jakarta	188	44	0	282	60	59
46	72	75	178	Kulkas	Bandung	188	60	0	260	135	178
49	72	75	178	Kulkas	Bandung	188	135	0	260	210	178
24	72	75	178	Kulkas	Bandung	282	0	0	354	75	178
45	84	14	66	Televisi	Bandung	282	75	0	366	89	66
48	84	14	66	Televisi	Bandung	282	89	0	366	103	66
10	84	14	66	Televisi	Bandung	282	103	0	366	117	66
12	33	24	25	Makanan	Bandung	282	117	0	315	141	25
36	33	24	25	Makanan	Bandung	282	141	0	315	165	25
50	33	24	25	Makanan	Bandung	282	165	0	315	189	25
25	33	24	25	Makanan	Bandung	282	189	0	315	213	25
47	33	24	25	Makanan	Bandung	282	0	178	315	24	203
11	72	75	178	Kulkas	Bandung	366	0	0	438	75	178
35	72	75	178	Kulkas	Bandung	366	75	0	438	150	178
34	84	14	66	Televisi	Bandung	366	150	0	450	164	66

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
23	84	14	66	Televisi	Bandung	366	164	0	450	178	66
39	36	19	24	Makanan	Semarang	366	178	0	402	197	24
5	78	82	184	Kulkas	Semarang	450	0	0	528	82	184
42	78	82	184	Kulkas	Semarang	450	82	0	528	164	184
41	84	14	66	Televisi	Semarang	450	164	0	534	178	66
40	94	16	59	Televisi	Semarang	450	178	0	544	194	59
14	36	19	24	Makanan	Semarang	450	194	0	486	213	24
27	36	19	24	Makanan	Semarang	450	0	184	486	19	208
2	36	19	24	Makanan	Semarang	450	19	184	486	38	208
15	36	19	24	Makanan	Semarang	450	38	184	486	57	208
38	36	19	24	Makanan	Semarang	450	57	184	486	76	208
28	36	19	24	Makanan	Semarang	450	76	184	486	95	208
1	36	19	24	Makanan	Semarang	450	95	184	486	114	208
29	94	16	59	Televisi	Semarang	544	0	0	638	16	59
17	84	14	66	Televisi	Semarang	544	16	0	628	30	66
16	94	16	59	Televisi	Semarang	544	30	0	638	46	59

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
18	78	82	184	Kulkas	Semarang	544	46	0	622	128	184
3	94	16	59	Televisi	Semarang	544	128	0	638	144	59
4	84	14	66	Televisi	Semarang	544	144	0	628	158	66

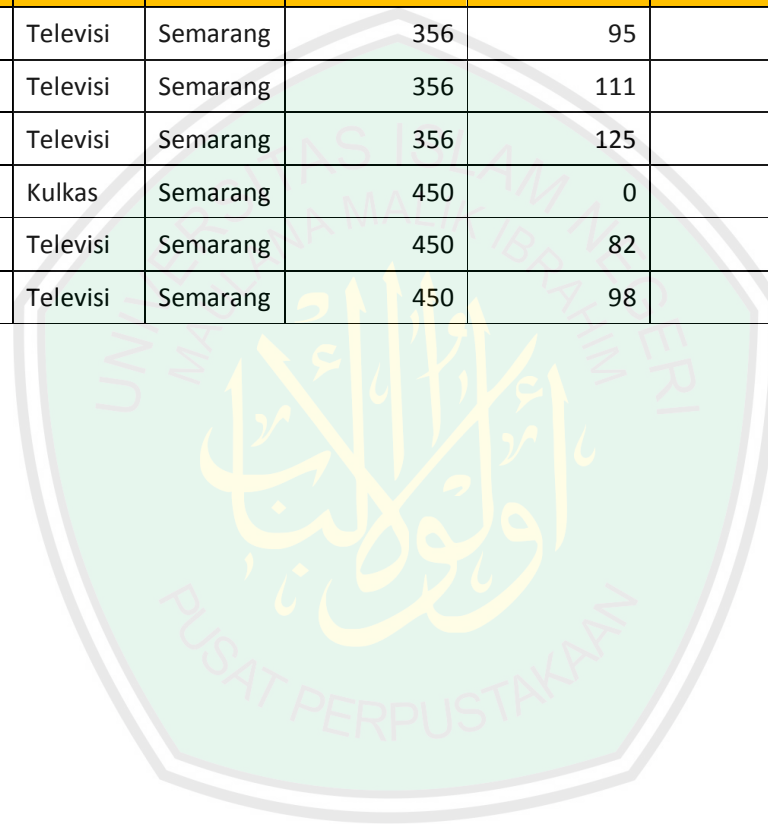
#### Container 45 feet

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
33	84	14	66	Televisi	Jakarta	0	0	0	84	14	66
31	84	14	66	Televisi	Jakarta	0	14	0	84	28	66
7	84	14	66	Televisi	Jakarta	0	28	0	84	42	66
21	78	82	184	Kulkas	Jakarta	0	42	0	78	124	184
22	84	14	66	Televisi	Jakarta	0	124	0	84	138	66
13	33	24	25	Makanan	Jakarta	0	138	0	33	162	25
6	94	16	59	Televisi	Jakarta	0	162	0	94	178	59
32	78	82	184	Kulkas	Jakarta	94	0	0	172	82	184
8	78	82	184	Kulkas	Jakarta	94	82	0	172	164	184
30	94	16	59	Televisi	Jakarta	94	164	0	188	180	59

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
26	33	24	25	Makanan	Jakarta	94	180	0	127	204	25
44	84	14	66	Televisi	Jakarta	94	204	0	178	218	66
37	33	24	25	Makanan	Jakarta	94	0	184	127	24	209
43	94	16	59	Televisi	Jakarta	94	24	184	188	40	243
20	84	14	66	Televisi	Jakarta	94	40	184	178	54	250
9	84	14	66	Televisi	Jakarta	94	54	184	178	68	250
19	94	16	59	Televisi	Jakarta	94	68	184	188	84	243
46	72	75	178	Kulkas	Bandung	188	0	0	260	75	178
49	72	75	178	Kulkas	Bandung	188	75	0	260	150	178
24	72	75	178	Kulkas	Bandung	188	150	0	260	225	178
45	84	14	66	Televisi	Bandung	188	0	178	272	14	244
48	84	14	66	Televisi	Bandung	188	14	178	272	28	244
10	84	14	66	Televisi	Bandung	188	28	178	272	42	244
12	33	24	25	Makanan	Bandung	188	42	178	221	66	203
36	33	24	25	Makanan	Bandung	188	66	178	221	90	203
50	33	24	25	Makanan	Bandung	188	90	178	221	114	203
25	33	24	25	Makanan	Bandung	188	114	178	221	138	203

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
47	33	24	25	Makanan	Bandung	188	138	178	221	162	203
11	72	75	178	Kulkas	Bandung	272	0	0	344	75	178
35	72	75	178	Kulkas	Bandung	272	75	0	344	150	178
34	84	14	66	Televisi	Bandung	272	150	0	356	164	66
23	84	14	66	Televisi	Bandung	272	164	0	356	178	66
39	36	19	24	Makanan	Semarang	272	178	0	308	197	24
5	78	82	184	Kulkas	Semarang	356	0	0	434	82	184
42	78	82	184	Kulkas	Semarang	356	82	0	434	164	184
41	84	14	66	Televisi	Semarang	356	164	0	440	178	66
40	94	16	59	Televisi	Semarang	356	178	0	450	194	59
14	36	19	24	Makanan	Semarang	356	194	0	392	213	24
27	36	19	24	Makanan	Semarang	356	213	0	392	232	24
2	36	19	24	Makanan	Semarang	356	0	184	392	19	208
15	36	19	24	Makanan	Semarang	356	19	184	392	38	208
38	36	19	24	Makanan	Semarang	356	38	184	392	57	208
28	36	19	24	Makanan	Semarang	356	57	184	392	76	208
1	36	19	24	Makanan	Semarang	356	76	184	392	95	208

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
29	94	16	59	Televisi	Semarang	356	95	184	450	111	243
17	84	14	66	Televisi	Semarang	356	111	184	440	125	250
16	94	16	59	Televisi	Semarang	356	125	184	450	141	243
18	78	82	184	Kulkas	Semarang	450	0	0	528	82	184
3	94	16	59	Televisi	Semarang	450	82	0	544	98	59
4	84	14	66	Televisi	Semarang	450	98	0	534	112	66



- **Tabel Output Algoritma Genetika Percobaan 2**

nomor	kode_barang	berat	jenis	tujuan
33	UA32F4510AMXXD	8	Televisi	Jakarta
31	UA32F4510AMXXD	8	Televisi	Jakarta
7	UA32F4510AMXXD	8	Televisi	Jakarta
21	RT59MBSL1/XSE	97	Kulkas	Jakarta
22	UA32F4510AMXXD	8	Televisi	Jakarta
13	MSDPGR	3	Makanan	Jakarta
6	LA32E420E2MXXD	9	Televisi	Jakarta
32	RT59MBSL1/XSE	97	Kulkas	Jakarta
8	RT59MBSL1/XSE	97	Kulkas	Jakarta
30	LA32E420E2MXXD	9	Televisi	Jakarta
26	MSDPGR	3	Makanan	Jakarta
44	UA32F4510AMXXD	8	Televisi	Jakarta
37	MSDPGR	3	Makanan	Jakarta
43	LA32E420E2MXXD	9	Televisi	Jakarta
20	UA32F4510AMXXD	8	Televisi	Jakarta
9	UA32F4510AMXXD	8	Televisi	Jakarta
19	LA32E420E2MXXD	9	Televisi	Jakarta
46	RT35FAACDSA/SE	66	Kulkas	Bandung
49	RT35FAACDSA/SE	66	Kulkas	Bandung
24	RT35FAACDSA/SE	66	Kulkas	Bandung
45	UA32F4510AMXXD	8	Televisi	Bandung
48	UA32F4510AMXXD	8	Televisi	Bandung
10	UA32F4510AMXXD	8	Televisi	Bandung
12	MSDPGR	3	Makanan	Bandung
36	MSDPGR	3	Makanan	Bandung
50	MSDPGR	3	Makanan	Bandung
25	MSDPGR	3	Makanan	Bandung
47	MSDPGR	3	Makanan	Bandung
11	RT35FAACDSA/SE	66	Kulkas	Bandung

nomor	kode_barang	berat	jenis	tujuan
35	RT35FAACDSA/SE	66	Kulkas	Bandung
34	UA32F4510AMXXD	8	Televisi	Bandung
23	UA32F4510AMXXD	8	Televisi	Bandung
39	INDKRAP	4	Makanan	Semarang
5	RT59MBSL1/XSE	97	Kulkas	Semarang
42	RT59MBSL1/XSE	97	Kulkas	Semarang
41	UA32F4510AMXXD	8	Televisi	Semarang
40	LA32E420E2MXXD	9	Televisi	Semarang
14	INDGR	4	Makanan	Semarang
27	INDGR	4	Makanan	Semarang
2	INDKRAP	4	Makanan	Semarang
15	INDKRAP	4	Makanan	Semarang
38	INDGR	4	Makanan	Semarang
28	INDKRAP	4	Makanan	Semarang
1	INDGR	4	Makanan	Semarang
29	LA32E420E2MXXD	9	Televisi	Semarang
17	UA32F4510AMXXD	8	Televisi	Semarang
16	LA32E420E2MXXD	9	Televisi	Semarang
18	RT59MBSL1/XSE	97	Kulkas	Semarang
3	LA32E420E2MXXD	9	Televisi	Semarang
4	UA32F4510AMXXD	8	Televisi	Semarang

- Tabel Titik Koordinat Barang Percobaan 2

**Container 20 feet**

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
32	78	82	184	Kulkas	Jakarta	0	0	0	78	82	184
9	84	14	66	Televisi	Jakarta	0	82	0	84	96	66
44	84	14	66	Televisi	Jakarta	0	96	0	84	110	66
20	84	14	66	Televisi	Jakarta	0	110	0	84	124	66
33	84	14	66	Televisi	Jakarta	0	124	0	84	138	66
37	33	24	25	Makanan	Jakarta	0	138	0	33	162	25
31	84	14	66	Televisi	Jakarta	0	162	0	84	176	66
21	78	82	184	Kulkas	Jakarta	84	0	0	162	82	184
7	84	14	66	Televisi	Jakarta	84	82	0	168	96	66
43	94	16	59	Televisi	Jakarta	84	96	0	178	112	59
30	94	16	59	Televisi	Jakarta	84	112	0	178	128	59
13	33	24	25	Makanan	Jakarta	84	128	0	117	152	25
8	78	82	184	Kulkas	Jakarta	84	152	0	162	234	184
6	94	16	59	Televisi	Jakarta	178	0	0	272	16	59
22	84	14	66	Televisi	Jakarta	178	16	0	262	30	66
19	94	16	59	Televisi	Jakarta	178	30	0	272	46	59
26	33	24	25	Makanan	Jakarta	178	46	0	211	70	25

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
34	84	14	66	Televisi	Bandung	178	70	0	262	84	66
10	84	14	66	Televisi	Bandung	178	84	0	262	98	66
25	33	24	25	Makanan	Bandung	178	98	0	211	122	25
24	72	75	178	Kulkas	Bandung	178	122	0	250	197	178
49	72	75	178	Kulkas	Bandung	272	0	0	344	75	178
48	84	14	66	Televisi	Bandung	272	75	0	356	89	66
36	33	24	25	Makanan	Bandung	272	89	0	305	113	25
50	33	24	25	Makanan	Bandung	272	113	0	305	137	25
46	72	75	178	Kulkas	Bandung	272	137	0	344	212	178
35	72	75	178	Kulkas	Bandung	356	0	0	428	75	178
45	84	14	66	Televisi	Bandung	356	75	0	440	89	66
23	84	14	66	Televisi	Bandung	356	89	0	440	103	66
47	33	24	25	Makanan	Bandung	356	103	0	389	127	25
11	72	75	178	Kulkas	Bandung	356	127	0	428	202	178
12	33	24	25	Makanan	Bandung	356	202	0	389	226	25
5	78	82	184	Kulkas	Semarang	440	0	0	518	82	184
39	36	19	24	Makanan	Semarang	440	82	0	476	101	24
38	36	19	24	Makanan	Semarang	440	101	0	476	120	24
18	78	82	184	Kulkas	Semarang	440	120	0	518	202	184
15	36	19	24	Makanan	Semarang	440	202	0	476	221	24

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
27	36	19	24	Makanan	Semarang	440	0	184	476	19	208

### Container 40 feet

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
32	78	82	184	Kulkas	Jakarta	0	0	0	78	82	184
9	84	14	66	Televisi	Jakarta	0	82	0	84	96	66
44	84	14	66	Televisi	Jakarta	0	96	0	84	110	66
20	84	14	66	Televisi	Jakarta	0	110	0	84	124	66
33	84	14	66	Televisi	Jakarta	0	124	0	84	138	66
37	33	24	25	Makanan	Jakarta	0	138	0	33	162	25
31	84	14	66	Televisi	Jakarta	0	162	0	84	176	66
21	78	82	184	Kulkas	Jakarta	84	0	0	162	82	184
7	84	14	66	Televisi	Jakarta	84	82	0	168	96	66
43	94	16	59	Televisi	Jakarta	84	96	0	178	112	59
30	94	16	59	Televisi	Jakarta	84	112	0	178	128	59

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
13	33	24	25	Makanan	Jakarta	84	128	0	117	152	25
8	78	82	184	Kulkas	Jakarta	178	0	0	256	82	184
6	94	16	59	Televisi	Jakarta	178	82	0	272	98	59
22	84	14	66	Televisi	Jakarta	178	98	0	262	112	66
19	94	16	59	Televisi	Jakarta	178	112	0	272	128	59
26	33	24	25	Makanan	Jakarta	178	128	0	211	152	25
34	84	14	66	Televisi	Bandung	178	152	0	262	166	66
10	84	14	66	Televisi	Bandung	178	166	0	262	180	66
25	33	24	25	Makanan	Bandung	178	180	0	211	204	25
24	72	75	178	Kulkas	Bandung	272	0	0	344	75	178
49	72	75	178	Kulkas	Bandung	272	75	0	344	150	178
48	84	14	66	Televisi	Bandung	272	150	0	356	164	66
36	33	24	25	Makanan	Bandung	272	164	0	305	188	25
50	33	24	25	Makanan	Bandung	272	188	0	305	212	25
46	72	75	178	Kulkas	Bandung	356	0	0	428	75	178

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
35	72	75	178	Kulkas	Bandung	356	75	0	428	150	178
45	84	14	66	Televisi	Bandung	356	150	0	440	164	66
23	84	14	66	Televisi	Bandung	356	164	0	440	178	66
47	33	24	25	Makanan	Bandung	356	178	0	389	202	25
11	72	75	178	Kulkas	Bandung	440	0	0	512	75	178
12	33	24	25	Makanan	Bandung	440	75	0	473	99	25
5	78	82	184	Kulkas	Semarang	440	99	0	518	181	184
39	36	19	24	Makanan	Semarang	440	181	0	476	200	24
38	36	19	24	Makanan	Semarang	440	200	0	476	219	24
18	78	82	184	Kulkas	Semarang	518	0	0	596	82	184
15	36	19	24	Makanan	Semarang	518	82	0	554	101	24
27	36	19	24	Makanan	Semarang	518	101	0	554	120	24
17	84	14	66	Televisi	Semarang	518	120	0	602	134	66
4	84	14	66	Televisi	Semarang	518	134	0	602	148	66
42	78	82	184	Kulkas	Semarang	518	148	0	596	230	184

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
2	36	19	24	Makanan	Semarang	518	0	184	554	19	208
40	94	16	59	Televisi	Semarang	602	0	0	696	16	59
3	94	16	59	Televisi	Semarang	602	16	0	696	32	59
29	94	16	59	Televisi	Semarang	602	32	0	696	48	59
41	84	14	66	Televisi	Semarang	602	48	0	686	62	66
1	36	19	24	Makanan	Semarang	602	62	0	638	81	24
16	94	16	59	Televisi	Semarang	602	81	0	696	97	59
28	36	19	24	Makanan	Semarang	602	97	0	638	116	24
14	36	19	24	Makanan	Semarang	602	116	0	638	135	24

### Container 45 feet

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akh
32	78	82	184	Kulkas	Jakarta	0	0	0	78	82	184
9	84	14	66	Televisi	Jakarta	0	82	0	84	96	66
44	84	14	66	Televisi	Jakarta	0	96	0	84	110	66
20	84	14	66	Televisi	Jakarta	0	110	0	84	124	66
33	84	14	66	Televisi	Jakarta	0	124	0	84	138	66
37	33	24	25	Makanan	Jakarta	0	138	0	33	162	25
31	84	14	66	Televisi	Jakarta	0	162	0	84	176	66
21	78	82	184	Kulkas	Jakarta	84	0	0	162	82	184
7	84	14	66	Televisi	Jakarta	84	82	0	168	96	66
43	94	16	59	Televisi	Jakarta	84	96	0	178	112	59
30	94	16	59	Televisi	Jakarta	84	112	0	178	128	59
13	33	24	25	Makanan	Jakarta	84	128	0	117	152	25
8	78	82	184	Kulkas	Jakarta	84	152	0	162	234	184
6	94	16	59	Televisi	Jakarta	84	0	184	178	16	243
22	84	14	66	Televisi	Jakarta	84	16	184	168	30	250
19	94	16	59	Televisi	Jakarta	84	30	184	178	46	243

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
26	33	24	25	Makanan	Jakarta	84	46	184	117	70	209
34	84	14	66	Televisi	Bandung	84	70	184	168	84	250
10	84	14	66	Televisi	Bandung	84	84	184	168	98	250
25	33	24	25	Makanan	Bandung	84	98	184	117	122	209
24	72	75	178	Kulkas	Bandung	178	0	0	250	75	178
49	72	75	178	Kulkas	Bandung	178	75	0	250	150	178
48	84	14	66	Televisi	Bandung	178	150	0	262	164	66
36	33	24	25	Makanan	Bandung	178	164	0	211	188	25
50	33	24	25	Makanan	Bandung	178	188	0	211	212	25
46	72	75	178	Kulkas	Bandung	262	0	0	334	75	178
35	72	75	178	Kulkas	Bandung	262	75	0	334	150	178
45	84	14	66	Televisi	Bandung	262	150	0	346	164	66
23	84	14	66	Televisi	Bandung	262	164	0	346	178	66
47	33	24	25	Makanan	Bandung	262	178	0	295	202	25
11	72	75	178	Kulkas	Bandung	346	0	0	418	75	178
12	33	24	25	Makanan	Bandung	346	75	0	379	99	25
5	78	82	184	Kulkas	Semarang	346	99	0	424	181	184
39	36	19	24	Makanan	Semarang	346	181	0	382	200	24

nomor	panjang	lebar	tinggi	jenis	tujuan	posisi_x_awal	posisi_y_awal	posisi_z_awal	posisi_x_akhir	posisi_y_akhir	posisi_z_akhir
38	36	19	24	Makanan	Semarang	346	200	0	382	219	24
18	78	82	184	Kulkas	Semarang	424	0	0	502	82	184
15	36	19	24	Makanan	Semarang	424	82	0	460	101	24
27	36	19	24	Makanan	Semarang	424	101	0	460	120	24
17	84	14	66	Televisi	Semarang	424	120	0	508	134	66
4	84	14	66	Televisi	Semarang	424	134	0	508	148	66
42	78	82	184	Kulkas	Semarang	424	148	0	502	230	184
2	36	19	24	Makanan	Semarang	424	0	184	460	19	208
40	94	16	59	Televisi	Semarang	424	19	184	518	35	243
3	94	16	59	Televisi	Semarang	424	35	184	518	51	243
29	94	16	59	Televisi	Semarang	424	51	184	518	67	243
41	84	14	66	Televisi	Semarang	424	67	184	508	81	250
1	36	19	24	Makanan	Semarang	424	81	184	460	100	208
16	94	16	59	Televisi	Semarang	424	100	184	518	116	243
28	36	19	24	Makanan	Semarang	424	116	184	460	135	208
14	36	19	24	Makanan	Semarang	424	135	184	460	154	208