

SKRIPSI

PEMBANGUNAN SIMULATOR PENYELESAIAN MASALAH
STRIP PACKING 2D DENGAN HEURISTIK *BOTTOM-UP*
LEFT-JUSTIFIED (BL), *NEXT-FIT DECREASING-HEIGHT*
(NFDH), DAN *FIRST-FIT DECREASING-HEIGHT* (FFDH)



Christopher William Sanjaya

NPM: 2017730034

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2022

UNDERGRADUATE THESIS

CONSTRUCTION OF 2D STRIP PACKING PROBLEM
SOLVING SIMULATOR WITH *BOTTOM-UP*
LEFT-JUSTIFIED (BL), *NEXT-FIT DECREASING-HEIGHT*
(NFDH), DAN *FIRST-FIT DECREASING-HEIGHT* (FFDH)
HEURISTICS



Christopher William Sanjaya

NPM: 2017730034

DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2022

LEMBAR PENGESAHAN

PEMBANGUNAN SIMULATOR PENYELESAIAN MASALAH
STRIP PACKING 2D DENGAN HEURISTIK *BOTTOM-UP*
LEFT-JUSTIFIED (BL), *NEXT-FIT DECREASING-HEIGHT*
(NFDH), DAN *FIRST-FIT DECREASING-HEIGHT* (FFDH)

Christopher William Sanjaya

NPM: 2017730034

Bandung, 18 Januari 2022

Menyetujui,

Pembimbing

Digitally signed
by Luciana
Abednego

Luciana Abednego, M.T.

Ketua Tim Penguji

Digitally signed
by Vania Natali

Vania Natali, M.T.

Anggota Tim Penguji

Digitally signed
by Maria V.
Claudia M.

Maria Veronica, M.T.

Mengetahui,

Ketua Program Studi

Digitally signed
by Mariskha Tri
Adithia

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

PEMBANGUNAN SIMULATOR PENYELESAIAN MASALAH STRIP PACKING 2D DENGAN HEURISTIK *BOTTOM-UP LEFT-JUSTIFIED* (BL), *NEXT-FIT DECREASING-HEIGHT* (NFDH), DAN *FIRST-FIT DECREASING-HEIGHT* (FFDH)

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 18 Januari 2022



Christopher William Sanjaya
NPM: 2017730034

ABSTRAK

Salah satu masalah pada dunia industri adalah masalah penempatan barang, masalah ini dapat disebut sebagai masalah *Strip Packing*, Masalah *Strip Packing* merupakan permasalahan peletakan sekumpulan objek ke dalam sebuah kontainer persegi (panjang), dan dibutuhkan simulasi untuk menyelesaikan masalah tersebut. Simulasi membutuhkan biaya, waktu, dan tenaga kerja yang besar jika dilakukan secara manual(coba-coba) sehingga menimbulkan permasalahan lain seperti rusaknya kualitas barang, untuk itu dibutuhkan simulator yang dapat menangani masalah-masalah tersebut. Simulator yang dibangun dapat memvisualisasikan peletakan sekumpulan objek di dalam kontainer dan dapat membandingkan hasil akhir setiap heuristik.

Pada skripsi ini digunakan 3 heuristik yang berbeda dalam menyelesaikan masalah tersebut, yaitu *Bottom-Up Left-Justified* (BL), *Next-Fit Decreasing-Height* (NFDH), dan *First-Fit Decreasing-Height* (FFDH), yang digunakan untuk mencari letak item pada kontainer yang digunakan untuk menampung item tersebut. Untuk membandingkan kualitas dari setiap heuristik, digunakan perbandingan menggunakan tinggi kontainer akhir yang dihasilkan oleh masing-masing heuristik.

Dari pengujian yang telah dilakukan, heuristik *Bottom-Up Left-Justified* (BL) cocok digunakan untuk memproses dataset yang digunakan pada penelitian ini karena persentase nilai normalisasi dari perbedaan tinggi optimum dengan hasil akhir program pada heuristik ini merupakan nilai yang paling kecil dibandingkan dengan heuristik lain yang digunakan pada penelitian ini, selain hal tersebut simulator yang dibangun dapat menghasilkan nilai akhir yang dibutuhkan. Beberapa hal yang dapat dilakukan selanjutnya adalah menggunakan heuristik lain untuk pemroses dataset masukan, dan memodifikasi masukan ketika memproses masukan data

Kata-kata kunci: simulator, simulasi, *Strip Packing 2D*, heuristik, *Bottom-Up Left-Justified* (BL), *Next-Fit Decreasing-Height* (NFDH), *First-Fit Decreasing-Height* (FFDH)

ABSTRACT

One of the problems in the industrial world is how to place goods, this problem can be referred as Strip Packing problem, Strip Packing problem is the problem of placing a set of objects into a rectangular container, simulation is needed to solve this problem. Simulation requires large costs, time, and labor if done manually (trial and error) and it can cause other problems such as damage to the quality of the goods, for that we need a simulator that can simulate these problems. The simulator that is built can visualize the placement of a set of objects in a container and can compare the final results of each heuristic. In this thesis, 3 different heuristics are used to solve this problem, namely Bottom-Up Left-Justified (BL), Next-Fit Decreasing-Height (NFDH), and First-Fit Decreasing-Height (FFDH), which are used to locate items in the container that used to hold the item. To compare the quality of each heuristic, a comparison using the final container height that generated by each heuristic. From the tests that have been carried out, the Bottom-Up Left-Justified (BL) heuristic is good for processing the dataset used in this study, and the simulator that already made can produce the final value.

Keywords: simulator, simulation, Strip packing 2d, heuristics, Bottom-Up Left-Justified (BL), Next-Fit Decreasing-Height (NFDH), First-Fit Decreasing-Height (FFDH)

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena atas karunia-Nya, penulis dapat menyelesaikan penyusunan skripsi yang berjudul ‘Pembangunan Simulator Penyelesaian Masalah Strip Packing 2D dengan Heuristik BL, NFDH, FFDH’. Penulis juga berterimakasih kepada pihak-pihak yang telah memberikan dukungan dan bantuan kepada penulis selama penyusunan skripsi ini, yaitu:

1. Tuhan Yesus atas Anugerah, Berkah, dan Rahmat-Nya.
2. Orang tua, saudara, dan keluarga yang selalu memberikan dukungan kepada penulis.
3. Ibu Luciana Abednego, S.Kom., M.T. yang telah memberikan bimbingan dan arahan selama proses penyusunan skripsi.
4. Ibu Vania Natali, M.T. dan Ibu Maria Veronica, M.T. sebagai dosen penguji yang telah membantu dalam menguji skripsi ini.
5. Teman-teman Teknik Informatika UNPAR yang telah berbagi ilmu dan selalu memberikan semangat kepada penulis.
6. Pacar penulis yang telah menemani dan memberikan semangat juga bantuan kepada penulis.
7. Teman-teman SMA dan lainnya yang telah menemani dan mendukung penulis.
8. Pihak-pihak lain yang telah membantu penulisan skripsi ini, yang terus memberikan doa, semangat, dan kesempatan kepada penulis.

Penulis berharap skripsi ini dapat berguna bagi segenap pihak yang berkepentingan. Akhir kata, penulis memohon maaf apabila terdapat kekurangan dalam hasil penyusunan skripsi ini.

Bandung, Januari 2022

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
DAFTAR KODE PROGRAM	xxiii
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi	3
1.6 Sistematika Pembahasan	3
2 LANDASAN TEORI	5
2.1 <i>Strip Packing Problem</i>	5
2.2 Heuristik	6
2.2.1 <i>Bottom-up left-justified</i> (BL)	6
2.2.2 <i>Next-fit decreasing-height</i> (NFDH)	7
2.2.3 <i>First-fit decreasing-height</i> (FFDH)	8
2.3 Simulator	8
2.4 <i>Benchmark</i>	9
2.5 <i>Framework Tkinter</i>	12
3 ANALISIS	13
3.1 Analisis Masalah	13
3.2 Studi Kasus Setiap Heuristik	13
3.2.1 Deskripsi Masalah	13
3.2.2 Studi Kasus penyelesaian masalah menggunakan <i>Bottom-Up Left-Justified</i> (BL)	14
3.2.3 Studi Kasus penyelesaian masalah menggunakan <i>Next-Fit Decreasing-Height</i> (NFDH)	19
3.2.4 Studi Kasus penyelesaian masalah menggunakan <i>First-Fit Decreasing-Height</i> (FFDH)	24
3.3 Analisis masukan dan keluaran perangkat lunak	29
3.3.1 Masukan perangkat lunak	29
3.3.2 Keluaran perangkat lunak	32
3.4 Analisis Cara Kerja Heuristik <i>Bottom-Up Left-Justified</i> (BL), <i>Next-Fit Decreasing-Height</i> (NFDH), dan <i>First-Fit Decreasing-Height</i> (FFDH)	32

3.4.1	<i>Bottom-Up Left-Justified (BL)</i>	33
4	PERANCANGAN PERANGKAT LUNAK	37
4.1	Perancangan <i>Use Case Diagram</i> Perangkat Lunak	37
4.1.1	Analisis <i>Use Case Scenario</i>	37
4.2	Perancangan Antarmuka	38
4.3	<i>Pseudocode</i> Heuristik <i>Bottom-Up Left-Justified (BL)</i> , <i>Next-Fit Decreasing-Height (NFDH)</i> , dan <i>First-Fit Decreasing-Height (FFDH)</i>	44
4.4	Perancangan Diagram Alir pada Aplikasi	49
4.4.1	Diagram alir aplikasi simulator	49
4.5	Perancangan Diagram Kelas	50
5	IMPLEMENTASI DAN PENGUJIAN	57
5.1	Implementasi	57
5.1.1	Lingkungan Implementasi	57
5.1.2	Hasil Implementasi	57
5.2	Pengujian heuristik terhadap data dummy yang dibuat	64
5.3	Pengujian heuristik terhadap benchmark yang digunakan	64
6	KESIMPULAN DAN SARAN	67
6.1	Kesimpulan	67
6.2	Saran	67
	DAFTAR REFERENSI	69
	A KODE PROGRAM	71

DAFTAR GAMBAR

2.1	Contoh visualisasi <i>Strip Packing Problem</i>	5
2.2	Contoh hasil penyelesaian menggunakan heuristik <i>Bottom-up left-justified</i> (BL)	6
2.3	Contoh hasil penyelesaian menggunakan heuristik <i>Next-fit decreasing-height</i> (NFDH)	7
2.4	Contoh hasil penyelesaian menggunakan heuristik <i>First-fit decreasing-height</i> (FFDH)	8
2.5	Contoh <i>Flight simulator</i>	9
2.6	Contoh dataset N1 dan N2 pada benchmark Hopper(2000)	10
2.7	Contoh dataset beng06 pada benchmark Martello(2003)	11
3.1	Kontainer dengan lebar 20	14
3.2	Visualisasi <i>item</i> ke-1 setelah dimasukkan kedalam kontainer	15
3.3	Visualisasi <i>item</i> ke-2 setelah dimasukkan kedalam kontainer	15
3.4	Visualisasi <i>item</i> ke-3 setelah dimasukkan kedalam kontainer	15
3.5	Visualisasi <i>item</i> ke-4 setelah dimasukkan kedalam kontainer	16
3.6	Visualisasi <i>item</i> ke-5 setelah dimasukkan kedalam kontainer	16
3.7	Visualisasi <i>item</i> ke-6 setelah dimasukkan kedalam kontainer	16
3.8	Visualisasi <i>item</i> ke-7 setelah dimasukkan kedalam kontainer	17
3.9	Visualisasi <i>item</i> ke-8 setelah dimasukkan kedalam kontainer	17
3.10	Visualisasi <i>item</i> ke-9 setelah dimasukkan kedalam kontainer	18
3.11	Visualisasi <i>item</i> ke-10 setelah dimasukkan kedalam kontainer	18
3.12	Visualisasi <i>item</i> ke-11 setelah dimasukkan kedalam kontainer	19
3.13	Visualisasi <i>item</i> ke-1 setelah dimasukkan kedalam kontainer	20
3.14	Visualisasi <i>item</i> ke-2 setelah dimasukkan kedalam kontainer	20
3.15	Visualisasi <i>item</i> ke-3 setelah dimasukkan kedalam kontainer	20
3.16	Visualisasi <i>item</i> ke-4 setelah dimasukkan kedalam kontainer	21
3.17	Visualisasi <i>item</i> ke-5 setelah dimasukkan kedalam kontainer	21
3.18	Visualisasi <i>item</i> ke-6 setelah dimasukkan kedalam kontainer	21
3.19	Visualisasi <i>item</i> ke-7 setelah dimasukkan kedalam kontainer	22
3.20	Visualisasi <i>item</i> ke-8 setelah dimasukkan kedalam kontainer	22
3.21	Visualisasi <i>item</i> ke-9 setelah dimasukkan kedalam kontainer	23
3.22	Visualisasi <i>item</i> ke-10 setelah dimasukkan kedalam kontainer	23
3.23	Visualisasi <i>item</i> ke-11 setelah dimasukkan kedalam kontainer	24
3.24	Visualisasi <i>item</i> ke-1 setelah dimasukkan kedalam kontainer	25
3.25	Visualisasi <i>item</i> ke-2 setelah dimasukkan kedalam kontainer	25
3.26	Visualisasi <i>item</i> ke-3 setelah dimasukkan kedalam kontainer	25
3.27	Visualisasi <i>item</i> ke-4 setelah dimasukkan kedalam kontainer	26
3.28	Visualisasi <i>item</i> ke-5 setelah dimasukkan kedalam kontainer	26
3.29	Visualisasi <i>item</i> ke-6 setelah dimasukkan kedalam kontainer	26
3.30	Visualisasi <i>item</i> ke-7 setelah dimasukkan kedalam kontainer	27
3.31	Visualisasi <i>item</i> ke-8 setelah dimasukkan kedalam kontainer	27
3.32	Visualisasi <i>item</i> ke-9 setelah dimasukkan kedalam kontainer	28
3.33	Visualisasi <i>item</i> ke-10 setelah dimasukkan kedalam kontainer	28
3.34	Visualisasi <i>item</i> ke-11 setelah dimasukkan kedalam kontainer	29

3.35	Contoh tangkapan layar untuk dataset yang digunakan Hopper et al.	30
3.36	Contoh tangkapan layar untuk dataset yang digunakan Martello et al.	31
3.37	Contoh tangkapan layar untuk hasil keluaran dari perangkat lunak	32
3.38	Contoh tangkapan layar untuk hasil keluaran dari perangkat lunak	32
3.39	Diagram alir heuristik <i>Bottom-Up Left-Justified</i> (BL)	33
3.40	Diagram alir heuristik <i>Next-Fit Decreasing-Height</i> (NFDH)	34
3.41	Diagram alir heuristik <i>First-Fit Decreasing-Height</i> (FFDH)	35
4.1	<i>Use Case Diagram Simulator Strip Packing 2D</i>	37
4.2	Halaman Awal	39
4.3	Halaman Inisialisasi Atribut Kontainer	39
4.4	Halaman Info	40
4.5	Halaman Visualisasi dan Perbandingan	40
4.6	Halaman Pilih Heuristik	41
4.7	Halaman Hasil Penyelesaian	41
4.8	Halaman Visualisasi	42
4.9	Halaman Perbandingan setiap Heuristik	42
4.10	Halaman Informasi Perangkat Lunak	43
4.11	Halaman Lain-lain	43
4.12	Diagram Alir pada Aplikasi	49
4.13	Diagram Kelas	50
4.14	Fungsi <i>visualize</i>	51
4.15	Fungsi pada kelas BL	51
4.16	Fungsi pada kelas FFDH	52
4.17	Fungsi pada kelas NFDH	53
4.18	Fungsi pada kelas main	54
5.1	Halaman Awal simulator ketika program dijalankan	58
5.2	Halaman untuk memasukkan data	58
5.3	Halaman visualisasi data	59
5.4	Halaman untuk memilih salah satu heuristik atau membandingkan hasil setiap heuristik	60
5.5	Halaman memilih heuristik	60
5.6	Halaman menampilkan hasil program	61
5.7	Halaman menampilkan hasil visualisasi	62
5.8	Halaman menampilkan perbandingan hasil program	62
5.9	Halaman Petunjuk	63
5.10	Halaman Lain-lain	64

DAFTAR TABEL

3.1	List ukuran setiap <i>item</i>	14
3.2	List ukuran setiap <i>item</i> setelah diurutkan berdasarkan tingginya.	19
4.1	Tabel skenario saat pengguna menentukan heuristik yang digunakan untuk mengolah data masukan	38
4.2	Tabel skenario saat pengguna menentukan untuk membandingkan hasil akhir dari setiap heuristik	38
5.1	Hasil pengujian terhadap data dummy	64
5.2	List hasil proses program pada data benchmark Hopper(2000)	65
5.3	List hasil proses program pada data benchmark Martello(2003)[1]	66

DAFTAR KODE PROGRAM

A.1	<code>BL.py</code>	71
A.2	<code>FFDH.py</code>	73
A.3	<code>NFDH.py</code>	76
A.4	<code>visualize.py</code>	78
A.5	<code>txtloader.py</code>	78
A.6	<code>main.py</code>	78

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Salah satu masalah pada dunia industri adalah pengaturan modal dalam membuat sebuah produk, dan pengaturan modal yang kurang baik mengakibatkan pengeluaran yang tidak perlu. Salah satu contoh alokasi modal pada sebuah produk yaitu tempat penyimpanan produk, urutan penyimpanan dan bagaimana cara menyimpan produk tersebut sehingga dapat meminimalisir tempat yang dibutuhkan. Dengan meminimalisir tempat penyimpanan barang maka tidak dibutuhkan tempat penyimpanan yang luas dan biaya yang besar. Dilihat dari masalah yang sudah disebutkan, masalah ini dapat di simulasikan dengan melakukan *strip packing* dua dimensi.

Simulasi adalah metode pelatihan yang meragakan sesuatu dalam bentuk tiruan yang mirip dengan keadaan yang sesungguhnya. Dalam melakukan simulasi tersebut diperlukan sebuah alat, perangkat, atau objek yang biasa disebut simulator. Simulator adalah alat, perangkat, atau objek yang berfungsi untuk menyimulasikan suatu peralatan, alat untuk melakukan simulasi, atau alat yang dapat menyimulasikan. Simulator diperlukan karena dalam dunia industri kita tidak dapat menggunakan objek secara bebas sebab dapat mengubah kualitas dari benda tersebut dimana hal tersebut merupakan hal yang tidak diinginkan.

Strip packing 2D adalah salah satu cara untuk mengetahui bagaimana menempatkan objek berbentuk persegi atau persegi panjang ke dalam sebuah bidang dua dimensi tanpa adanya perpotongan antara objek objek tersebut. Beberapa algoritma heuristik yang sudah cukup sering digunakan dalam kasus ini adalah *Bottom-Up Left-Justified* (BL), *Next-Fit Decreasing-Height* (NFDH), *First-Fit Decreasing-Height* (FFDH), *split-fit*, *Sleator's*, *Reverse-fit*, dan heuristik lainnya.

Dari berbagai heuristik yang ada, digunakan heuristik *Bottom-Up Left-Justified* (BL), *Next-Fit Decreasing-Height* (NFDH), *First-Fit Decreasing-Height* (FFDH) pada penelitian ini. Heuristik *Bottom-Up Left-Justified* (BL) bekerja dengan cara meletakkan objek yang disusun, secara berurutan dimulai dari sisi terbawah dan ter kiri dari sebuah bidang penempatan. Heuristik *Next-Fit Decreasing-Height*(NFDH) bekerja dengan cara meletakkan objek yang disusun, secara berurutan dari sisi terbawah dan ter kiri dari sebuah bidang dengan catatan ketinggian antar objek sudah disusun sedemikian rupa sehingga terurut dari yang tertinggi hingga terendah, dan ketika lebar bidang sudah tidak cukup untuk menyimpan objek tersebut maka objek tersebut akan diletakkan pada level berikutnya, dan level sebelumnya tidak akan dicek kembali. Heuristik *First-Fit Decreasing-Height*(FFDH) bekerja dengan cara yang cukup mirip dengan heuristik *Next-Fit Decreasing-Height*(NFDH), perbedaan kedua heuristik itu ada pada penempatan objek pada baris berikutnya, dimana heuristik NFDH tidak melakukan pengecekan kembali pada baris sebelumnya, sedangkan heuristik FFDH akan melakukan pengecekan apakah baris sebelumnya dapat menerima objek tersebut tanpa bertindihan dengan *item* lain atau melewati batas bidang yang ditentukan.

Pada skripsi ini dibuat sebuah perangkat lunak berupa simulator yang dapat digunakan untuk mensimulasikan masalah *strip packing* 2D dan menyelesaikannya dengan menggunakan beberapa heuristik yaitu *Bottom-Up Left-Justified* (BL), *Next-Fit Decreasing-Height* (NFDH), dan *First-Fit Decreasing-Height* (FFDH). Perangkat lunak ini juga dapat digunakan untuk mencari heuristik yang paling optimal untuk kasus-kasus tertentu.

Program ini dibuat dengan bantuan aplikasi Spyder dalam pembuatan kodenya, menggunakan bahasa pemrograman Python, dan membuat data dummy dengan megacak nilai untuk setiap atribut persegi. Visualisasi hasil dilakukan dengan cara menampilkan bidang yang telah dijadikan parameter masukan beserta penempatan objek-objek yang telah di proses menggunakan heuristik pilihan.

Dalam melakukan evaluasi program menggunakan dataset pada penelitian yang dilakukan oleh Hopper (2000) [2] dan penelitian yang dilakukan oleh Martello (2003) [1]. Evaluasi dilakukan dengan membandingkan nilai tinggi *strip*/kontainer optimal dari benchmark yang digunakan [3] untuk setiap dataset dengan hasil keluaran dari perangkat lunak yang dibangun.

1.2 Rumusan Masalah

Berikut merupakan beberapa rumusan masalah pada pembuatan program untuk menyelesaikan masalah *strip packing* 2D:

1. Bagaimana cara kerja heuristik *Bottom-Up Left-Justified* (BL), *Next-Fit Decreasing-Height* (NFDH), *First-Fit Decreasing-Height* (FFDH)?
2. Bagaimana cara menilai atau mengevaluasi hasil dari setiap heuristik?
3. Bagaimana cara membangun perangkat lunak berupa simulator yang dapat menyelesaikan masalah *strip packing* 2D dan memvisualisasikan hasilnya, juga membandingkan nilai/hasil dari setiap heuristik yang berbeda?

1.3 Tujuan

Tujuan dari pembuatan program ini adalah sebagai berikut:

1. Mempelajari cara kerja *Bottom-Up Left-Justified* (BL), *Next-Fit Decreasing-Height* (NFDH), *First-Fit Decreasing-Height* (FFDH).
2. Mempelajari cara menilai atau mengevaluasi hasil dari setiap algoritma.
3. Membangun perangkat lunak berupa simulator yang dapat menyelesaikan masalah *strip packing* 2D dan memvisualisasikan hasilnya, juga membandingkan nilai/hasil dari setiap algoritma yang berbeda.

1.4 Batasan Masalah

Batasan-batasan masalah pada penelitian ini adalah sebagai berikut:

1. Data yang diterima berupa file text yang harus sesuai dengan format yang diberikan.
2. Objek yang menjadi masukan perangkat lunak berbentuk persegi (panjang) dan lebar dari persegi tersebut tidak dapat melebihi lebar kontainer yang digunakan.
3. Objek yang diletakkan kedalam kontainer tidak boleh saling bertumpuk dan melebihi batas kontainer yang digunakan.
4. Objek tidak dapat diputar.

1.5 Metodologi

Metodologi yang digunakan dalam penyusunan penelitian ini adalah:

1. Mempelajari heuristik *Bottom-Up Left-Justified* (BL), *Next-Fit Decreasing-Height* (NFDH), *First-Fit Decreasing-Height* (FFDH).
2. Mempelajari cara menilai atau mengevaluasi setiap heuristik yang digunakan.
3. Menyiapkan data dummy yang digunakan untuk melakukan tes pada setiap heuristik.
4. Mengimplementasikan heuristik yang akan digunakan kedalam program.
5. Merancang arsitektur perangkat lunak.
6. Melakukan test pada setiap heuristik yang telah diimplementasikan.
7. Merancang tampilan perangkat lunak.
8. Membangun perangkat lunak yang telah dirancang.
9. Melakukan test pada perangkat lunak yang telah dibangun.

1.6 Sistematika Pembahasan

Penelitian ini ditulis dalam beberapa topik pembahasan yang disusun dengan sistematika sebagai berikut:

- **Bab 1 Pendahuluan**
Berisi latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian, dan sistematika pembahasan.
- **Bab 2 Landasan Teori**
Berisi definisi *strip packing*, definisi heuristik, definisi setiap heuristik yang digunakan pada penelitian ini, definisi simulator, penjelasan *benchmark* yang digunakan, dan penjelasan mengenai framework Tkinter.
- **Bab 3 Analisis**
Berisi analisis masalah, studi kasus untuk setiap heuristik yang digunakan pada penelitian ini yaitu *Bottom-Up Left-Justified* (BL), *Next-Fit Decreasing-Height* (NFDH), *First-Fit Decreasing-Height* (FFDH), analisis masukan dan keluaran perangkat lunak, dan analisis cara kerja heuristik yang digunakan pada penelitian ini menggunakan diagram alir (*flowchart*).
- **Bab 4 Perancangan Perangkat Lunak**
Berisi perancangan *use case diagram* perangkat lunak, perancangan antarmuka, pseudocode setiap heuristik yang digunakan pada penelitian ini, perancangan diagram alir aplikasi, dan perancangan diagram kelas.
- **Bab 5 Implementasi dan Pengujian**
Berisi implementasi perangkat lunak, pengujian heuristik terhadap data dummy yang dibuat, dan pengujian heuristik terhadap dataset pada benchmark yang digunakan.
- **Bab 6 Kesimpulan dan Saran**
Berisi kesimpulan dari penelitian ini, dan saran dari peneliti untuk penelitian selanjutnya.