

SKRIPSI

PEMBANGUNAN SIMULATOR PENJADWALAN *FLOW SHOP*



Tedi Tri Ramdani

NPM: 2015730001

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2019

UNDERGRADUATE THESIS

*CONSTRUCTION OF A FLOW SHOP SCHEDULING
SIMULATOR*



Tedi Tri Ramdani

NPM: 2015730001

DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2019

LEMBAR PENGESAHAN

PEMBANGUNAN SIMULATOR PENJADWALAN *FLOW SHOP*

Tedi Tri Ramdani

NPM: 2015730001

Bandung, 17 Mei 2019

Menyetujui,

Pembimbing

Luciana Abednego, M.T.

Ketua Tim Penguji

Anggota Tim Penguji

Raymond Chandra Putra, M.T.

Dr.rer.nat. Cecilia Esti Nugraheni

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

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

PEMBANGUNAN SIMULATOR PENJADWALAN *FLOW SHOP*

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 17 Mei 2019

Meterai Rp. 6000

Tedi Tri Ramdani
NPM: 2015730001

ABSTRAK

Penjadwalan atau penjadwalan produksi merupakan suatu kegiatan yang sangat penting pada proses produksi. Penjadwalan produksi merupakan kegiatan yang menentukan efisiensi proses produksi. Dengan penjadwalan produksi yang baik, proses produksi akan berjalan dengan lancar dan menghemat biaya. Dengan perkembangan jumlah job dan *machine*, maka kegiatan penjadwalan tidak mungkin dilakukan secara manual. Maka dari itu diperlukan perangkat lunak yang mampu melakukan proses penjadwalan dan mensimulasikan hasil dari proses penjadwalan.

Penjadwalan adalah kegiatan pengalokasian sumber-sumber atau *machine* yang ada untuk menjalankan sekumpulan tugas dalam jangka waktu tertentu [1]. Hasil dari penjadwalan adalah urutan *job* yang menjadi acuan untuk melaksanakan proses produksi. Terdapat banyak kriteria untuk melakukan proses penjadwalan, salah satunya adalah kriteria untuk meminimasi nilai *make span*[2] atau total waktu untuk memproses semua *job*. Terdapat dua aliran produksi yang sering digunakan pada proses produksi yaitu alur produksi *flow shop*[2] dan alur produksi *job shop*[2]. Alur produksi *flow shop* adalah alur produksi dimana sekumpulan *job* akan diproses pada sekumpulan *machine* secara berurutan. Pada alur *flow shop* semua *job* akan diproses pada semua *machine*, sedangkan pada alur produksi *job shop* semua *job* hanya akan melewati *machine* sesuai dengan kebutuhan dan karakteristik dari *job*.

Penjadwalan pada proses produksi *flow shop* dapat dilakukan dengan berbagai cara. Penjadwalan *flow shop* dapat dilakukan dengan metode Nawaz Ensore Ham (NEH)[3], Palmer [4], dan Gupta [5]. Ketiga metode ini memiliki kemiripan yaitu memprioritaskan *job* dengan total waktu proses terbesar untuk dikerjakan terlebih dahulu. Pada penelitian ini dibangun sebuah perangkat lunak simulator penjadwalan *flow shop* yang mengimplementasikan ketiga algoritma penjadwalan tersebut. Perangkat lunak menggambarkan simulasi hasil penjadwalan *flow shop* dalam bentuk *gant chart*. *Gantt chart* berfungsi untuk menggambarkan jalannya proses produksi setelah dilakukan penjadwalan.

Berdasarkan hasil pengujian yang didapatkan dari perangkat lunak yang telah dibangun, perangkat lunak mampu mensimulasikan hasil penjadwalan *flow shop* dengan algoritma NEH, Palmer, dan Gupta. Ketiga algoritma menghasilkan nilai *make span* dan urutan antrian *job* yang bervariasi. Algoritma NEH[3] menjadi algoritma yang menghasilkan nilai *make span*[2] paling minimal dibandingkan dengan algoritma Palmer[4] dan Gupta[5].

Kata-kata kunci: Penjadwalan, *Flow Shop*, *Make Span*, NEH, Palmer, Gupta, Simulator

ABSTRACT

Scheduling or production scheduling is an important activity in production process. Production scheduling is an activity that determine the efficiency of production process. With a good production scheduling, the production process will be running smooth and benefit in cost. With the growing scale in number of jobs and machines on production floor, manual scheduling is irrelevant these days. So a software is required to do a scheduling activity and simulating the result.

Scheduling is an activity of allocating resources or machines to run some tasks in specific time range[1]. The result of scheduling is a sequence of jobs that used for reference in determining on how to run a production process. There are several criteria in scheduling process, one of them is minimizing make span[2] or the total amount of time to process all the jobs. The production process has two major flow type, they are flow shop and job shop. Flow shop production flow is a production flow type which a sequence of jobs processed on an array of machine. In flow shop production flow every job on the sequence will be processed on every machine on the machine's array, while on job shop, every job is processed based on the requirement and characteristic of the job.

Scheduling in flow shop production flow can be done using many methods. The flow shop can be done using Nawaz Ensore Ham (NEH)[3] method, Palmer[4] method, and Gupta[5] method. Those method prioritizing a job with biggest total of processing time to be processed first. In this research, a scheduling simulator software that implement the three algorithms will be build. The software will simulating the result of flow shop scheduling process in a form of gantt chart. Gantt chart is used for simulating the flow of production process after scheduling process.

Based on the result from testing the build software, the software is able to simulate the result of scheduling form NEH, Palmer, and Gupta algorithm. The three algorithms result vary make spans and sequences. The NEH[3] algorithm is resulting the most minimal make span[2] compared to Palmer[4] and Gupta[5] algorithm .

Keywords: *Scheduling, Flow Shop, Make Span, NEH, Palmer, Gupta, Simulator*

Dipersembahkan untuk kedua orang tua tercinta...

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa karena atas berkat, rahmat, dan karunia-Nya penulis dapat menyelesaikan skripsi dengan judul "Pembangunan Simulator Penjadwalan *Flow Shop*". Penulis juga ingin mengucapkan terima kasih kepada dukungan dan bantuan dari pihak-pihak yang membantu dalam proses penyelesaian skripsi ini. Penulis ingin berterima kasih kepada :

- Kedua orang tua yang selalu memberikan dukungan untuk menyelesaikan skripsi ini.
- Dosen pembimbing, ibu Luciana Abednego yang memberikan bimbingan, masukan, dan tambahan wawasan selama proses pembuatan skripsi ini.
- Rekan-rekan asisten laboratorium komputer di fakultas ekonomi UNPAR, Muhamad Rifqi Mulyadi, Aditya Putra Leksono, Vincent Joel Sinatra, serta kepala staf laboratorium Bapak Iqbal yang telah membantu penulis mengembangkan keahlian penulis di bidang IT support dan memberikan dukungan kepada penulis untuk menyelesaikan skripsi ini.
- Dosen pengajar dan staf tata usaha FTIS yang sudah membantu proses perkuliahan penulis selama menimba ilmu di jurusan teknik informatika UNPAR.

Semoga seluruh pihak yang membantu dalam penyusunan skripsi ini mendapat berkah dan rahmat dari Tuhan Yang Maha Esa. Akhir kata, penulis memohon maaf apabila terdapat kesalahan dan kekurangan pada skripsi ini. Semoga skripsi ini dapat berguna bagi semua pihak yang membutuhkan.

Bandung, Mei 2019

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xxi
DAFTAR TABEL	xxv
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	3
1.4 Batasan Masalah	3
1.5 Metodologi	3
1.6 Sistematika Pembahasan	4
2 LANDASAN TEORI	5
2.1 Penjadwalan	5
2.1.1 Fungsi utama penjadwalan	5
2.1.2 Tujuan Penjadwalan Produksi	5
2.1.3 Komponen pada Penjadwalan Produksi	6
2.1.4 Kriteria Penjadwalan	6
2.2 Tipe Penjadwalan	6
2.2.1 Penjadwalan <i>Flow Shop</i>	6
2.2.2 Penjadwalan <i>Job Shop</i>	7
2.3 Aturan Sequencing	7
2.4 Algoritma Penjadwalan <i>Flow Shop</i>	8
2.4.1 Algoritma Nawaz Enscore Ham	8
2.4.2 Algoritma Palmer	8
2.4.3 Algoritma Gupta	9
2.5 Model	9
2.5.1 Karakterisasi model	10
2.6 Simulasi	10
2.6.1 Tipe Simulasi	11
3 ANALISIS	13
3.1 Analisis Masalah	13
3.1.1 Penjadwalan <i>Flow Shop</i> dengan Algoritma Nawaz Enscore Ham	13
3.1.2 Penjadwalan <i>Flow Shop</i> dengan Algoritma Palmer	18
3.1.3 Penjadwalan <i>Flow Shop</i> dengan Algoritma Gupta	19
3.2 Analisis Kebutuhan Perangkat Lunak	20
3.2.1 Analisis Data <i>Input</i>	20
3.2.2 Diagram Kelas	21

3.2.3	Kelas <i>Job</i>	21
3.2.4	Kelas <i>Machine</i>	22
3.2.5	Kelas <i>Algorithm</i>	23
3.2.6	Kelas <i>NEHAlgorithm</i>	23
3.2.7	Kelas <i>PalmerAlgorithm</i>	23
3.2.8	Kelas <i>GuptaAlgorithm</i>	23
3.2.9	Kelas <i>FlowShopScheduler</i>	23
3.2.10	Kelas <i>FSSSimulatorController</i>	24
3.3	Analisis Penggambaran <i>Gantt Chart</i>	25
4	PERANCANGAN	27
4.1	Perancangan Antarmuka	27
4.1.1	<i>Tab Data Input</i>	30
4.1.2	<i>Tab Simulation View</i>	35
4.1.3	<i>Tab Result View</i>	36
4.1.4	<i>Tab Statistics</i>	37
4.2	Perancangan <i>File Data Input</i>	37
4.3	Perancangan Kelas	39
4.4	Deskripsi dan Fungsi Setiap Kelas	39
4.4.1	Kelas <i>Job</i>	40
4.4.2	Kelas (<i>Machine</i>)	41
4.4.3	Kelas <i>Algorithm</i>	42
4.4.4	Kelas <i>NEHAlgorithm</i>	42
4.4.5	Kelas <i>PalmerAlgorithm</i>	43
4.4.6	Kelas <i>GuptaAlgorithm</i>	45
4.4.7	Kelas <i>FlowShopScheduler</i>	47
4.4.8	Kelas <i>FSSSimulatorViewController</i>	49
4.5	Diagram Aktivitas	54
5	IMPLEMENTASI DAN PENGUJIAN	57
5.1	Tampilan Antarmuka	57
5.1.1	Tampilan Awal/Tampilan <i>Data Input</i>	57
5.1.2	Tampilan <i>Simulation View</i>	62
5.1.3	Tampilan <i>Tab Result View</i>	67
5.1.4	<i>Tab Statistics</i>	68
5.2	Pengujian Perangkat Lunak	69
5.2.1	Pengujian Fungsional	69
5.2.2	Pengujian Eksperimental	83
5.2.3	Statistik berdasarkan jumlah <i>job</i>	84
5.2.4	Statistik berdasarkan jumlah <i>machine</i>	86
5.2.5	Kesimpulan Pengujian	88
6	KESIMPULAN DAN SARAN	89
6.1	Kesimpulan	89
6.2	Saran	90
	DAFTAR REFERENSI	91
	A KODE PROGRAM	93
	B HASIL EKSPERIMEN	121
B.1	Tabel Hasil Penjadwalan <i>Flow Shop</i> dengan <i>Input File 20 Job</i> dan <i>5 Machine</i>	121
B.1.1	Algoritma <i>NEH</i>	122

B.1.2	Algoritma Palmer	123
B.1.3	Algoritma Gupta	124
B.2	Tabel Hasil Penjadwalan <i>Flow Shop</i> dengan <i>Input File 50 Job</i> dan <i>10 Machine</i>	125
B.2.1	Algoritma NEH	126
B.2.2	Algoritma Palmer	127
B.2.3	Algoritma Gupta	128
B.3	Tabel Hasil Penjadwalan <i>Flow Shop</i> dengan <i>Input File 100 Job</i> dan <i>20 Machine</i>	128
B.3.1	Algoritma NEH	129
B.3.2	Algoritma Palmer	130
B.3.3	Algoritma Gupta	131
B.4	Tabel Hasil Penjadwalan <i>Flow Shop</i> dengan <i>Input File 200 Job</i> dan <i>20 Machine</i>	132
B.4.1	Algoritma NEH	134
B.4.2	Algoritma Palmer	134
B.4.3	Algoritma Gupta	136
B.5	Tabel Hasil Penjadwalan <i>Flow Shop</i> dengan <i>Input Random 10 Job, 5 Machine</i> dan <i>225 Due Date</i>	137
B.5.1	Algoritma NEH	137
B.5.2	Algoritma Palmer	137
B.5.3	Algoritma Gupta	138
B.6	Tabel Hasil Penjadwalan <i>Flow Shop</i> dengan <i>Input Random 25 Job, 10 Machine</i> dan <i>450 Due Date</i>	138
B.6.1	Algoritma NEH	139
B.6.2	Algoritma Palmer	139
B.6.3	Algoritma Gupta	140

DAFTAR GAMBAR

1.1	Proses alur produksi <i>Flow Shop</i> dengan 4 <i>Machine</i>	1
1.2	Proses Produksi pada Industri Konveksi	2
2.1	Penjadwalan <i>flow shop</i> dengan M(4) <i>machine</i> dan N(5) <i>job</i>	7
2.2	Penjadwalan <i>job shop</i> dengan M(3) <i>machine</i> dan N(3) <i>job</i>	7
2.3	Perbandingan perubahan variabel <i>state</i> pada simulasi kejadian diskret dan simulasi kontinu	11
3.1	Hubungan antar Kelas	21
3.2	Penggambaran <i>gantt chart</i>	25
4.1	Tampilan antarmuka awal	27
4.2	Tampilan antarmuka <i>tab simulation view</i>	28
4.3	Tampilan antarmuka <i>tab result view</i>	29
4.4	Tampilan antarmuka <i>tab statistics</i>	29
4.5	Tampilan antarmuka <i>tab data input</i> untuk tipe <i>input file</i>	30
4.6	Tampilan antarmuka <i>file selector</i>	31
4.7	Tampilan antarmuka pemberitahuan keberhasilan membaca <i>file data input</i>	32
4.8	Tampilan antarmuka input data random	33
4.9	Tampilan antarmuka pesan pemberitahuan keberhasilan atau kesalahan	34
4.10	Tampilan antarmuka <i>tab simulation view</i>	35
4.11	Tampilan antarmuka <i>tab result view</i>	36
4.12	Tampilan antarmuka <i>tab result view</i>	37
4.13	Format <i>file data input .txt</i>	38
4.14	Contoh <i>file data input .txt</i>	38
4.15	Format <i>file data input .xlsx</i>	38
4.16	Contoh <i>file data input .xlsx</i>	39
4.17	Gambar diagram kelas beserta asosiasinya	39
4.18	Diagram Kelas <i>Job</i>	40
4.19	Diagram Kelas <i>Machine</i>	41
4.20	Diagram Kelas <i>Algorithm</i>	42
4.21	Diagram Kelas <i>NEHAlgorithm</i>	42
4.22	Diagram Kelas <i>PalmerAlgorithm</i>	43
4.23	Diagram Kelas <i>PalmerAlgorithm</i>	45
4.24	Diagram Kelas <i>FlowShopScheduler</i>	47
4.25	Diagram Kelas <i>FSSSimulatorViewController</i>	49
4.26	Diagram aktivitas proses penjadwalan dengan algoritma NEH	55
4.27	Diagram aktivitas proses penjadwalan dengan algoritma Palmer	55
4.28	Diagram aktivitas proses penjadwalan dengan algoritma Gupta	56
5.1	Halaman antarmuka awal/ <i>tab data input 1</i>	57
5.2	Halaman antarmuka awal/ <i>tab data input 2</i>	58
5.3	Tampilan antarmuka <i>file selector</i>	58

5.4	Tampilan antarmuka pemberitahuan berhasil membuka <i>file data input</i>	59
5.5	Tampilan antarmuka <i>input random</i>	59
5.6	Tampilan antarmuka pesan pemberitahuan dan kesalahan jika <i>input</i> kosong	60
5.7	Tampilan antarmuka pesan pemberitahuan dan kesalahan jika <i>input</i> kurang	61
5.8	Tampilan antarmuka pesan pemberitahuan dan kesalahan jika <i>input</i> kurang	61
5.9	Halaman antarmuka <i>tab simulation view</i> yang belum memiliki data	62
5.10	Tampilan awal simulasi dengan algoritma NEH	62
5.11	Tampilan simulasi selesai dijalankan pada algoritma NEH	63
5.12	Tampilan akhir simulasi pada algoritma NEH	63
5.13	Tampilan awal simulasi dengan algoritma Palmer	64
5.14	Tampilan simulasi selesai dijalankan pada algoritma Palmer	64
5.15	Tampilan akhir simulasi pada algoritma Palmer	65
5.16	Tampilan awal simulasi dengan algoritma Gupta	65
5.17	Tampilan simulasi selesai dijalankan pada algoritma Gupta	66
5.18	Tampilan akhir simulasi pada algoritma Gupta	66
5.19	Tampilan antarmuka <i>result view</i> saat data <i>input</i> kosong	67
5.20	Tampilan antarmuka <i>result view</i> saat data <i>input</i> telah diisi dan penjadwalan telah dilakukan	67
5.21	Tampilan antarmuka pesan pemberitahuan dan kesalahan	68
5.22	Tampilan antarmuka statistik dengan parameter 20 <i>job</i>	69
5.23	Tampilan antarmuka statistik dengan parameter 5 <i>machine</i>	69
5.24	Hasil simulasi algoritma Neh dengan 20 <i>job</i> dan 5 <i>machine</i>	71
5.25	Hasil simulasi algoritma Palmer dengan 20 <i>job</i> dan 5 <i>machine</i>	71
5.26	Hasil simulasi algoritma Gupta dengan 20 <i>job</i> dan 5 <i>machine</i>	72
5.27	Hasil simulasi penjadwalan <i>flow shop</i> dengan 20 <i>job</i> dan 5 <i>machine</i>	72
5.28	Hasil simulasi algoritma Neh dengan 50 <i>job</i> dan 10 <i>machine</i>	73
5.29	Hasil simulasi algoritma Palmer dengan 50 <i>job</i> dan 10 <i>machine</i>	73
5.30	Hasil simulasi algoritma Gupta dengan 50 <i>job</i> dan 10 <i>machine</i>	74
5.31	Hasil simulasi penjadwalan <i>flow shop</i> dengan 50 <i>job</i> dan 10 <i>machine</i>	74
5.32	Hasil simulasi algoritma Neh dengan 100 <i>job</i> dan 20 <i>machine</i>	75
5.33	Hasil simulasi algoritma Palmer dengan 100 <i>job</i> dan 20 <i>machine</i>	75
5.34	Hasil simulasi algoritma Gupta dengan 100 <i>job</i> dan 20 <i>machine</i>	76
5.35	Hasil simulasi penjadwalan <i>flow shop</i> dengan 100 <i>job</i> dan 20 <i>machine</i>	76
5.36	Hasil simulasi algoritma Neh dengan 200 <i>job</i> dan 20 <i>machine</i>	77
5.37	Hasil simulasi algoritma Palmer dengan 200 <i>job</i> dan 20 <i>machine</i>	77
5.38	Hasil simulasi algoritma Gupta dengan 200 <i>job</i> dan 20 <i>machine</i>	78
5.39	Hasil simulasi penjadwalan <i>flow shop</i> dengan 200 <i>job</i> dan 20 <i>machine</i>	78
5.40	Hasil simulasi algoritma Neh dengan 10 <i>job</i> dan 5 <i>machine</i>	79
5.41	Hasil simulasi algoritma Palmer dengan 10 <i>job</i> dan 5 <i>machine</i>	80
5.42	Hasil simulasi algoritma Gupta dengan 10 <i>job</i> dan 5 <i>machine</i>	80
5.43	Hasil simulasi penjadwalan <i>flow shop</i> dengan 10 <i>job</i> dan 5 <i>machine</i>	81
5.44	Hasil simulasi algoritma Neh dengan 25 <i>job</i> dan 10 <i>machine</i>	82
5.45	Hasil simulasi algoritma Palmer dengan 25 <i>job</i> dan 10 <i>machine</i>	82
5.46	Hasil simulasi algoritma Gupta dengan 25 <i>job</i> dan 10 <i>machine</i>	83
5.47	Hasil simulasi penjadwalan <i>flow shop</i> dengan 25 <i>job</i> dan 10 <i>machine</i>	83
5.48	Grafik nilai <i>make span</i> terhadap 20 <i>jobs</i>	84
5.49	Grafik nilai <i>make span</i> terhadap 50 <i>jobs</i>	84
5.50	Grafik nilai <i>make span</i> terhadap 100 <i>jobs</i>	85
5.51	Grafik nilai <i>make span</i> terhadap 200 <i>jobs</i>	85
5.52	Grafik nilai <i>make span</i> terhadap 500 <i>jobs</i>	86
5.53	Grafik nilai <i>make span</i> terhadap 5 <i>machine</i>	86

5.54 Grafik nilai <i>make span</i> terhadap 10 <i>machine</i>	87
5.55 Grafik nilai <i>make span</i> terhadap 20 <i>machine</i>	87

DAFTAR TABEL

3.1	Tabel <i>processing time</i> dari 5 <i>job</i> dan 3 <i>machine</i>	13
3.2	Tabel total waktu proses untuk masing-masing <i>job</i>	14
3.3	Tabel <i>make span</i> parsial kandidat urutan antrian II 1	14
3.4	Tabel <i>make span</i> parsial kandidat urutan antrian II 2	14
3.5	Tabel <i>make span</i> parsial kandidat urutan antrian III 1	15
3.6	Tabel <i>make span</i> parsial kandidat urutan antrian III 2	15
3.7	Tabel <i>make span</i> parsial kandidat urutan antrian III 3	15
3.8	Tabel <i>make span</i> parsial kandidat urutan antrian IV 1	16
3.9	Tabel <i>make span</i> parsial kandidat urutan antrian IV 2	16
3.10	Tabel <i>make span</i> parsial kandidat urutan antrian IV 3	16
3.11	Tabel <i>make span</i> parsial kandidat urutan antrian IV 4	16
3.12	Tabel <i>make span</i> parsial kandidat urutan antrian V 1	17
3.13	Tabel <i>make span</i> parsial kandidat urutan antrian V 2	17
3.14	Tabel <i>make span</i> parsial kandidat urutan antrian V 3	17
3.15	Tabel <i>make span</i> parsial kandidat urutan antrian V 4	17
3.16	Tabel <i>make span</i> parsial kandidat urutan antrian V 5	18
3.17	Tabel nilai e untuk masing-masing <i>job</i>	19
3.18	Tabel Kelas <i>Job</i>	22
3.19	Tabel Kelas <i>Machine</i>	22
3.20	Tabel Kelas <i>FlowShopScheduler</i>	24
3.21	Tabel Kelas <i>Controller</i>	24
B.1	Tabel <i>processing pime</i> untuk data <i>input</i> 20 <i>Job</i> dan 5 <i>Machine</i>	121
B.2	Tabel <i>Make Span</i> Hasil Penjadwalan NEH dengan 20 <i>Job</i> dan 5 <i>Machine</i>	122
B.3	Tabel <i>Make Span</i> Hasil Penjadwalan Palmer dengan 20 <i>Job</i> dan 5 <i>Machine</i>	123
B.4	Tabel <i>Make Span</i> Hasil Penjadwalan Gupta dengan 20 <i>Job</i> dan 5 <i>Machine</i>	124
B.5	Tabel <i>processing time</i> untuk data <i>input</i> 50 <i>Job</i> dan 10 <i>Machine</i>	125
B.6	Tabel <i>Make Span</i> Hasil Penjadwalan NEH dengan 50 <i>Job</i> dan 10 <i>Machine</i>	126
B.7	Tabel <i>Make Span</i> Hasil Penjadwalan Palmer dengan 50 <i>Job</i> dan 10 <i>Machine</i>	127
B.8	Tabel <i>Make Span</i> Hasil Penjadwalan Gupta dengan 50 <i>Job</i> dan 10 <i>Machine</i>	128
B.9	Tabel <i>processing time</i> untuk data <i>input</i> 100 <i>Job</i> dan 20 <i>Machine</i>	129
B.10	Tabel <i>Make Span</i> Hasil Penjadwalan NEH dengan 100 <i>Job</i> dan 20 <i>Machine</i>	130
B.11	Tabel <i>Make Span</i> Hasil Penjadwalan Palmer dengan 100 <i>Job</i> dan 20 <i>Machine</i>	131
B.12	Tabel <i>Make Span</i> Hasil Penjadwalan Gupta dengan 100 <i>Job</i> dan 20 <i>Machine</i>	132
B.13	Tabel <i>processing time</i> untuk data <i>input</i> 200 <i>Job</i> dan 20 <i>Machine</i>	133
B.14	Tabel <i>Make Span</i> Hasil Penjadwalan NEH dengan 200 <i>Job</i> dan 20 <i>Machine</i>	134
B.15	Tabel <i>Make Span</i> Hasil Penjadwalan Palmer dengan 200 <i>Job</i> dan 20 <i>Machine</i>	135
B.16	Tabel <i>Make Span</i> Hasil Penjadwalan Gupta dengan 200 <i>Job</i> dan 20 <i>Machine</i>	136
B.17	Tabel <i>processing time</i> untuk data <i>input</i> 10 <i>Job</i> dan 5 <i>Machine</i>	137
B.18	Tabel <i>Make Span</i> Hasil Penjadwalan NEH dengan 10 <i>Job</i> dan 5 <i>Machine</i>	137
B.19	Tabel <i>Make Span</i> Hasil Penjadwalan Palmer dengan 10 <i>Job</i> dan 5 <i>Machine</i>	137
B.20	Tabel <i>Make Span</i> Hasil Penjadwalan Gupta dengan 10 <i>Job</i> dan 5 <i>Machine</i>	138

B.21	Tabel <i>processing time</i> untuk data <i>input</i> 25 <i>Job</i> dan 10 <i>Machine</i>	138
B.22	Tabel <i>Make Span</i> Hasil Penjadwalan NEH dengan 25 <i>Job</i> dan 10 <i>Machine</i>	139
B.23	Tabel <i>Make Span</i> Hasil Penjadwalan Palmer dengan 25 <i>Job</i> dan 10 <i>Machine</i>	139
B.24	Tabel <i>Make Span</i> Hasil Penjadwalan Gupta dengan 25 <i>Job</i> dan 10 <i>Machine</i>	140

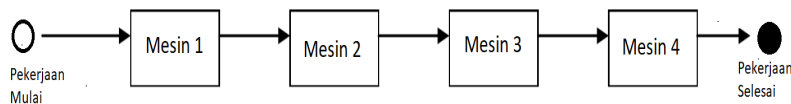
BAB 1

PENDAHULUAN

1.1 Latar Belakang

Penjadwalan pekerjaan merupakan salah satu keputusan yang sangat penting pada sistem kontrol produksi. Penjadwalan merupakan pengaturan jumlah dan tipe produksi yang akan dilaksanakan dalam kurun waktu tertentu. Penjadwalan[6] berhubungan dengan waktu penggunaan fasilitas dan bahan material untuk pelaksanaan produksi. Selain itu, penggunaan perangkat fasilitas produksi seperti mesin sangat berpengaruh pada biaya produksi. Penjadwalan pekerjaan yang buruk akan mengakibatkan proses produksi memakan banyak waktu dan biaya produksi. Penjadwalan produksi yang baik akan meningkatkan penggunaan sumber daya atau mengurangi waktu tunggu, sehingga total waktu proses dapat berkurang, dan produktivitas dapat meningkat. Dengan menentukan penjadwalan pekerjaan yang tepat dalam sistem kontrol produksi maka proses produksi akan menjadi lebih efisien dan cepat. Hal tersebut berdampak pada berkurangnya biaya produksi.

Terdapat dua jenis alur produksi yang sering dijumpai pada proses produksi. Jenis alur tersebut adalah alur produksi *flow shop* dan alur produksi *job shop*. Alur produksi yang sering digunakan pada proses produksi adalah alur produksi *flow shop*. Alur produksi *flow shop* adalah alur produksi dari n buah *job* pada m buah *machine* dengan pola proses yang searah dari *machine* pertama hingga *machine* ke- m (Gambar 1.1). N buah *job* pada alur produksi *flow shop* memiliki m buah proses yang akan dikerjakan pada m buah *machine* mulai dari proses pertama hingga proses ke- m untuk setiap *job*.



Gambar 1.1: Proses alur produksi *Flow Shop* dengan 4 *Machine*

Contoh dari proses alur produksi *flow shop* adalah proses pada industri konveksi [7] (Gambar 1.2). Pada industri konveksi, produk yang dihasilkan bervariasi mulai dari, baju, celana, jaket, kemeja, topi, dan sebagainya. Masing-masing produk memiliki karakteristik dan waktu pembuatan yang berbeda-beda. Setiap produk akan diproses pada urutan stasiun kerja yang sama (*machine*/manusia).



Gambar 1.2: Proses Produksi pada Industri Konveksi

Penjadwalan pekerjaan yang dilakukan dengan cara manual tanpa bantuan perangkat lunak memiliki beberapa kelemahan. Penjadwalan pekerjaan secara manual dapat memakan cukup banyak waktu untuk dilakukan. Selain itu, kesalahan perhitungan juga dapat terjadi jika dilakukan dengan kurang teliti. Dengan semakin pentingnya penjadwalan pekerjaan pada proses produksi dan kelemahan yang ada pada penjadwalan pekerjaan yang dilakukan secara manual, maka dibutuhkan dukungan perangkat lunak yang dapat melakukan penjadwalan pekerjaan. Selain dapat melakukan penjadwalan pekerjaan, perangkat lunak juga mampu mensimulasikan proses dari pekerjaan yang sudah dijadwalkan sebelumnya.

Perangkat lunak simulator penjadwalan pekerjaan sangat penting untuk menentukan penjadwalan proses produksi. Perangkat lunak simulator penjadwalan mampu membantu seorang pemimpin produksi untuk mengambil keputusan penjadwalan pekerjaan dalam proses produksi. Dengan mensimulasikan penjadwalan pada perangkat lunak simulator penjadwalan, seorang pemimpin produksi dapat mengambil keputusan untuk menjadwalkan pekerjaan berdasarkan hasil simulasi yang dilakukan oleh perangkat lunak dengan cepat dan tepat. Dengan adanya perangkat lunak simulator penjadwalan tersebut, pekerjaan pemimpin produksi menjadi lebih mudah dalam menentukan proses produksi. Oleh karena itu, pada penelitian ini akan dibangun sebuah perangkat lunak simulator penjadwalan *flow shop* yang mensimulasikan penjadwalan *flow shop*. Perangkat lunak simulator penjadwalan *flow shop* mensimulasikan hasil penjadwalan dengan menggunakan algoritma Nawaz Enscore Hame (NEH) [3], Palmer [4], dan Gupta [5]. Ketiga algoritma tersebut dipilih karena ketiga algoritma tersebut memiliki kemiripan dalam melakukan proses penjadwalan *flow shop*, namun menghasilkan hasil penjadwalan yang berbeda, sehingga hasil dari penjadwalan dari ketiga algoritma tersebut dapat dibandingkan untuk digunakan sebagai referensi dalam menentukan proses produksi. Dengan banyaknya kriteria untuk melakukan penjadwalan, maka perangkat lunak simulator penjadwalan *flow shop* yang dibangun untuk menghasilkan total nilai *make span*[2] minimal sebagai kriteria agar ruang lingkup masalah tidak terlalu luas.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, maka dirumuskan masalah pada penelitian ini, yaitu:

1. Bagaimana cara kerja dari algoritma penjadwalan NEH, Palmer, dan Gupta ?
2. Bagaimana rancangan model yang akan digunakan untuk penjadwalan *flow shop* dengan ketiga algoritma penjadwalan tersebut ?

3. Bagaimana rancangan simulator penjadwalan *flow shop* yang dapat mensimulasikan hasil penjadwalan dari ketiga algoritma tersebut ?
4. Bagaimana cara mengimplementasikan algoritma penjadwalan *flow shop* NEH, Palmer, dan Gupta pada perangkat lunak simulator penjadwalan *flow shop* ?

1.3 Tujuan

Adapun tujuan yang hendak dicapai oleh skripsi ini antara lain :

1. Mempelajari cara kerja dari algoritma penjadwalan NEH, Palmer, dan Gupta.
2. Merancang model yang dapat mendukung algoritma penjadwalan NEH, Palmer, dan Gupta pada perangkat lunak simulator penjadwalan *flow shop*.
3. Merancang simulator penjadwalan *flow shop* yang dapat mensimulasikan penjadwalan dari ketiga algoritma penjadwalan yang digunakan.
4. Mengimplementasikan algoritma penjadwalan NEH, Palmer dan Gupta pada perangkat lunak simulator penjadwalan *flow shop*.

1.4 Batasan Masalah

Batasan masalah pada penelitian ini meliputi :

1. Perangkat lunak simulator penjadwalan *flow shop* berfokus pada meminimasi nilai *make span*.
2. Karena keterbatasan ruang untuk visualisasi simulasi, data set pelatihan yang digunakan pada pengujian menggunakan data set pelatihan skala kecil-menengah agar menghasilkan tampilan simulasi yang optimal.
3. Input jumlah *machine* dan *job* ditentukan berdasarkan *file* data set pelatihan dan dibatasi hingga 200 *job* dan 20 *machine*.
4. Penjadwalan dilakukan secara kontinu tanpa ada jeda atau *delay*. Setiap *job* akan langsung diproses setelah *job* sebelumnya selesai diproses.

1.5 Metodologi

Langkah-langkah yang akan dilakukan dalam melakukan penelitian ini, yaitu :

1. Melakukan studi literatur mengenai penjadwalan *flow shop* dan algoritma-algoritma yang akan digunakan pada perangkat lunak (NEH, Palmer, Gupta).
2. Melakukan studi literatur mengenai pemodelan dan simulasi.
3. Melakukan analisis dan merancang struktur data yang cocok untuk perangkat lunak penjadwalan *flow shop*.
4. Mengimplementasikan keseluruhan rancangan algoritma dan struktur data pada perangkat lunak.
5. Melakukan pengujian (dan eksperimen) simulasi penjadwalan *flow shop* pada perangkat lunak.
6. Menulis dokumen skripsi.

1.6 Sistematika Pembahasan

Sistematika penelitian ini secara garis besar terdiri dari enam bab. Setiap bab pada skripsi ini saling berkaitan dan berhubungan. Berikut ini akan dikemukakan uraian dari sistematika pembahasan :

1. Bab 1. Pendahuluan, membahas tentang latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, metode penelitian, dan sistematika pembahasan mengenai skripsi ini.
2. Bab 2. Dasar Teori, membahas pengertian penjadwalan, penjadwalan *flow shop*, pemodelan dan simulasi, algoritma penjadwalan (NEH, Palmer, dan Gupta).
3. Bab 3. Analisis, membahas tentang analisis pemecahan rumusan masalah pada penelitian ini, cara kerja algoritma penjadwalan *flow shop* NEH, Palmer, dan Gupta, dan analisis kebutuhan perangkat lunak
4. Bab 4. Perancangan, membahas mengenai perancangan aplikasi simulator penjadwalan *flow shop*, mulai dari perancangan diagram kelas, diagram aktivitas, perancangan struktur data, perancangan algoritma, perancangan data pelatihan, dan perancangan antarmuka.
5. Bab 5. Implementasi dan Pengujian Perangkat Lunak, membahas mengenai implementasi dari hasil rancangan perangkat lunak dan pengujian terhadap perangkat lunak yang telah dibuat.
6. Bab 6. Kesimpulan dan Saran, sebagai penutup berisi tentang kesimpulan pembangunan perangkat lunak serta saran untuk pengembangan penelitian selanjutnya