

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan tentang gambaran umum PT. Barata Indonesia (Persero), penjelasan mengenai data-data yang dikumpulkan dan melakukan pengolahan data serta pembahasan dari hasil pengolahan data untuk menjawab rumusan masalah dan tujuan penelitian yang telah ditetapkan sebelumnya.

4.1 Pengumpulan Data

Pengumpulan data dilakukan melalui pengamatan secara langsung di lapangan dan data yang diambil dalam penelitian ini adalah data tentang penjadwalan produksi untuk permintaan-permintaan yang diterima *workshop* III PT. Barata Indonesia pada bulan Januari-Maret 2014 yang meliputi produk kualiti timah dan *laddle*. Dalam pelaksanaan penjadwalan produksi diperlukan data-data yang digunakan sebagai *input*, yaitu data profil perusahaan, data permintaan produk terkait beserta kuantitasnya, data alur proses produksi, waktu baku proses produksi, dan lain-lain.

4.1.1 Profil Perusahaan

PT BARATA INDONESIA (Persero) didirikan pada tahun 1971 dengan nama PT. BARATA METALWORKS & ENGINEERING yang merupakan *merger* dari:

1. PN. BARATA dahulu NV. BRAAT *Machinefabriek*, didirikan pada tahun 1901 untuk memberikan jasa pemugaran kepada pabrik - pabrik gula, manufaktur jembatan, dan konstruksi baja lainnya.
2. PN. SABANG MERAUKE dahulu *Machinefabriek & Scheepswerf* NV. MOLENVLIET, didirikan pada tahun 1920 untuk memberikan jasa pemugaran pada industri budidaya gunung dan perkapalan pantai.
3. PN. PEPRIDA, yaitu perusahaan milik pemerintah yang didirikan pada tahun 1962 untuk melaksanakan pembangunan proyek-proyek industri dasar.

Pada awal berdirinya, PT BARATA INDONESIA berpusat di Surabaya menempati area seluas 6.7 Ha di jalan Ngagel No. 109 dengan perkembangannya dari waktu ke waktu telah menjadi wilayah pusat kota yang padat penduduk. Sehingga mempertimbangkan pengembangan ke depan yang membutuhkan ketersediaan lahan yang lebih luas, maka PT. BARATA INDONESIA (Persero) melakukan relokasi kantor

dan pabrik ke Gresik di Jl. Veteran No. 241 pada tahun 2006 dengan menempati lahan seluas 22 Ha.

4.1.1.1 Visi dan Misi Perusahaan

Setiap organisasi pasti mempunyai visi dan misi, begitu halnya perusahaan dari yang skala kecil hingga internasional. Berikut merupakan visi dan misi yang dimiliki oleh PT. Barata Indonesia.

1. Visi Perusahaan

Visi dari PT. Barata Indonesia (Persero) ini adalah menjadi perusahaan *Engineering, Procurement, dan Construction (EPC)* dan *Manufacturing* yang tangguh.

2. Misi Perusahaan

Sedangkan misi yang dimiliki oleh PT. Barata Indonesia (Persero) sebagai berikut:

1. Melakukan kegiatan usaha *manufacturing* peralatan industri dan komponen untuk bidang Industri, Agro, Oil dan Gas, *Power Plant*, Peralatan Pembuatan Jalan dan Pengairan, dengan mengoptimalkan sumber daya dan fasilitas yang ada, sehingga memberikan nilai tambah bagi karyawan, pemesan, pemegang saham, dan *stakeholder* lainnya.
2. Melakukan kegiatan *Engineering, Procurement, Construction (EPC)* untuk bidang “Industri Agro” dan Industri Pembangkit Tenaga Listrik”.

❖ Strategi Perusahaan

1. Mendukung kemajuan dan kemandirian Industri Nasional.
2. Memberikan produk dan pelayanan yang berkualitas kepada pemesan dalam rangka menciptakan nilai yang prima.
3. Menjadi keuntungan bagi para pemegang saham.
4. Menciptakan kesejahteraan dan peningkatan kualitas kerja karyawan.

4.1.1.2 Bidang Usaha

Dalam menjalankan proses produksinya, PT. Barata Indonesia (Persero) pada saat ini bergerak dalam bidang usaha yang diantaranya sebagai berikut:

A. *ENGINEERING PROCUREMENT AND CONSTRUCTION (EPC)*

1. Peralatan Penanganan Pelabuhan
2. Pabrik Gula (Baru dan Rekondisi)

3. Pabrik Minyak Kelapa Sawit
4. Peralatan Lapangan Terbang
5. Pembangkit Tenaga Listrik

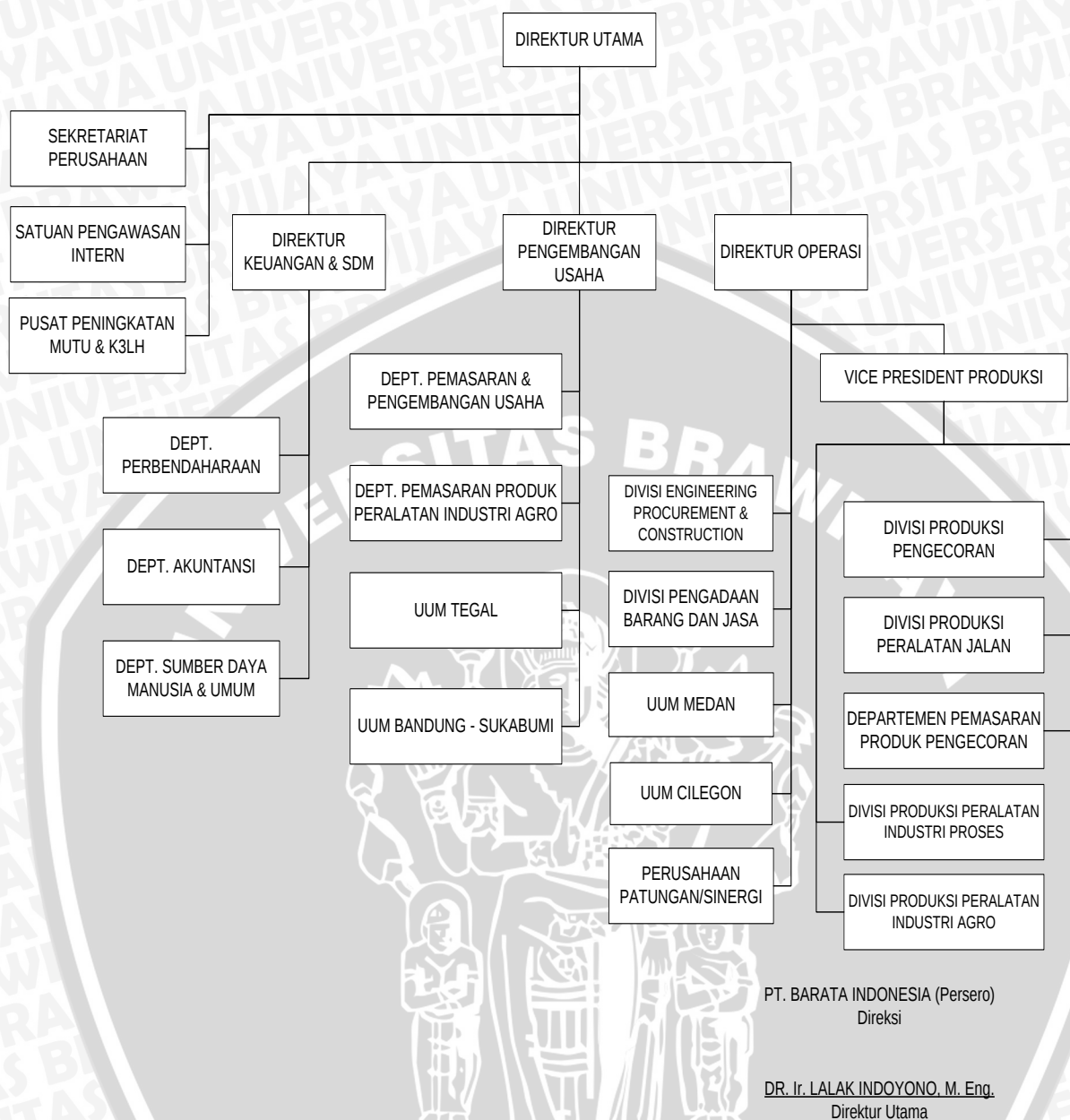
B. MANUFACTURE

1. Minyak dan Gas: *Heat Exchanger, Pressure Vessel, Boiler, HRSG, Kondensor, Pipa Saluran, Deaerator, Tangki.*
2. Peralatan Pembangunan Jalan: *Mesin Gilas, Penyemprot Aspal, Stamper, Penghancur Batuan.*
3. *Hydro Mechanic: Pintu Air, Trash Rack, Penstock, Hidro Turbin.*
4. Industri Argo: *Peralatan dan Komponen Pabrik Gula.*
5. Proses Industri: *Pabrik Semen dan Peralatannya, Peralatan Industri Pertambangan, Peralatan Industri Kimia.*
6. Konstruksi Baja: *Jembatan Kereta Api Baja, Jembatan Jalan Baja.*
7. *Heavy Machining: Gear Making, Heavy Machining.*

C. PENGECORAN (CASTING)

1. Peralatan Kereta Api: *Bogie, Automatic Coppler, Shoulder/Rail, Clip Housing, Knukle, Axle Box.*
2. Industri Semen: *Liners, Hammer Mills, Grate Plates, Wobblers, Nose Ring, Grinding Balls.*
3. Industri Pertambangan: *Dredge Bucket, Bucket Teeth, Jaw Crusher, Communion Part, WSlurry Pump.*
4. Komponen Otomotif: *Trunnion, Hanger, Piston Ring, Cam Shaft.*
5. Industri Kimia: *Distributor Pipers, Impeller, Pump Casing, Komponen Pabrik Kertas.*
6. Industri Perkapalan: *Rudden Horn, Rudder Frame, Jangkar, Gear kapal, Bollards, Gypsy.*

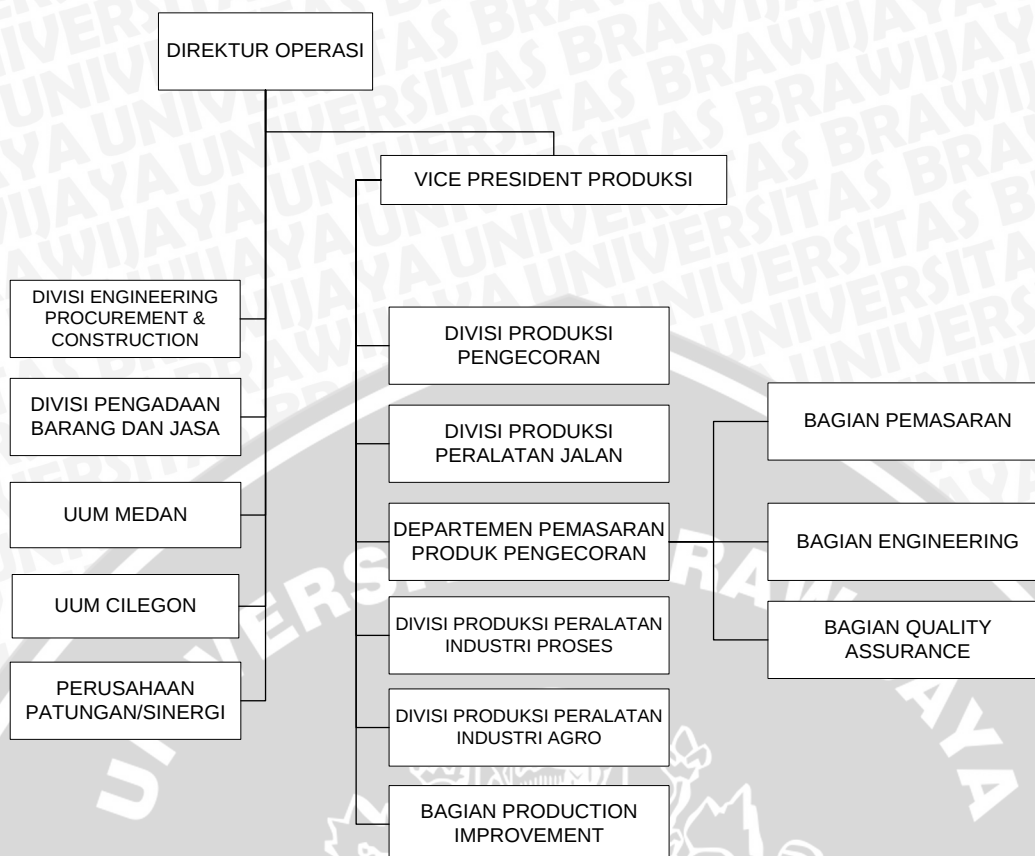
4.1.1.3 Struktur Organisasi



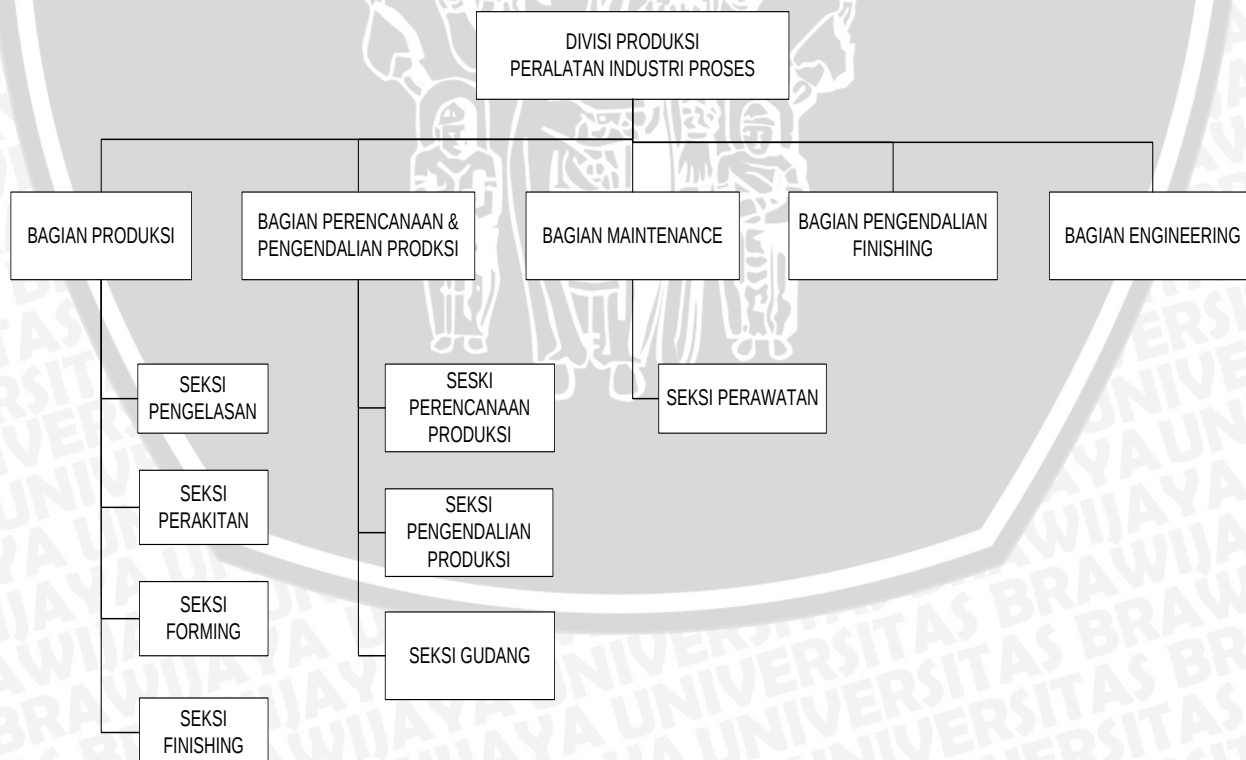
Gambar 4.1 Struktur organisasi perusahaan

Keterangan Gambar 4.1:

- SDM = Sumber Daya Manusia
- Dept. = Departemen
- UUM = Unit Usaha Mandiri
- K3LH = Kesehatan dan Keselamatan Kerja dan Lingkungan Hidup



Gambar 4.2 Struktur organisasi direktur operasi



Gambar 4.3 Struktur organisasi divisi Produksi Peralatan Industri Proses (PPIP)

4.1.2 Manajemen Personalia

Latar pendidikan buruh/karyawan pabrik PT. Barata Indonesia (Persero) minimal berpendidikan SMP hingga SMA, sedangkan untuk karyawan kantor memiliki sebaran pendidikan mulai SMA/SMK hingga S1. Untuk peningkatan kualitas dan kompetensi dari seorang manajer, maka terdapat persyaratan yang mengharuskan seorang manajer berpendidikan minimal S2.

4.1.2.1 Jumlah Tenaga Kerja

Karyawan yang dimiliki oleh kantor pusat PT. Barata Indonesia (Persero) di Gresik ini kurang lebih adalah sebanyak 336 karyawan (per Januari 2013) yang dapat dibedakan menjadi beberapa golongan sebagai berikut :

1. Tenaga Kerja Organik, merupakan tenaga kerja yang asli bekerja di PT. Barata Indonesia (Persero). Tenaga kerja organik dibagi lagi menjadi dua jenis, yaitu :
 - a. Tenaga Kerja Organik Aktif, merupakan tenaga kerja (karyawan) tetap yang aktif bekerja setiap harinya, terbagi lagi menjadi :
 - Karyawan tetap, sejumlah 321 orang.
 - CKT (Calon Karyawan Tetap), merupakan karyawan yang bekerja di PT. Barata Indonesia (Persero) namun masih dalam masa training dan masih dalam masa pengangkatan menjadi pegawai tetap, saat ini sebanyak 5 orang.
 - b. Tenaga Kerja Organik Non-aktif MPPB (Masa Persiapan Purnabakti) merupakan masa persiapan karyawan selama satu tahun sebelum masa pensiun, saat ini sebanyak 10 orang karyawan.
2. Tenaga Kerja Non-Organik (Sub-Kon), merupakan tenaga kerja yang berasal dari pihak luar PT. Barata Indonesia (Persero) yang ikut membantu dalam menyelesaikan suatu proyek dimana jumlah karyawan non-organik ini selalu tidak tetap bergantung pada kekurangan tenaga kerja dalam suatu proyek.

4.1.2.2 Jam Kerja Karyawan

PT. Barata Indonesia (Persero) menetapkan jam kerja sebagai berikut:

1. Hari Senin sampai Jumat
Pukul 07.30 – 16.30 WIB
2. Jam Lembur (Senin - Jumat)
Pukul 16.30 – 20.00 WIB

3. Hari Sabtu, (*Optional* untuk beberapa karyawan Organik dan wajib bagi karyawan bengkel apabila ada keterlambatan pengerjaan)

Pukul 07.30 – 16.00

4.1.3 Proses Produksi *Workshop* III PT. Barata Indonesia

Secara garis besar proses produksi di PT. Barata Indonesia Gresik terdiri dari dua kegiatan produksi (*work centre*) yakni *forming* dan *assembly*. Proses produksi yang terjadi di *work shop* ini juga memungkinkan untuk terjadi secara bolak-balik antar *work centre* karena PT. Barata Indonesia memproduksi produk-produk yang mempunyai spesifikasi berbeda-beda sesuai dengan keinginan dari masing-masing pemesan. Ditinjau dari beberapa kondisi tersebut maka tipe aliran produksinya sesuai dengan alur produksi *flexible job shop/hybrid jobshop*.

Work centre yang digunakan di *workshop* ini mempunyai fungsi yang berbeda sehingga tipe mesin dan jumlahnya pun juga akan berbeda. Pada *work centre forming*, kegiatan produksi yang dilakukan meliputi pembentukan *raw material* yaitu plat besi hingga menjadi komponen-komponen yang siap untuk dirakit lebih lanjut. Sedangkan pada *work centre assembly* terjadi kegiatan perakitan dari komponen-komponen berupa *head*, *shell*, *ring*, *support*, dan *lift* yang dibutuhkan oleh masing-masing produk yang berbeda. Penjelasan dari masing-masing proses di *work centre* akan dijelaskan sebagai berikut:

1. *Work centre 1 (forming)*.

Forming merupakan proses mengubah bentuk logam dengan suatu gaya pada arah tertentu tanpa menyisakan serpih. Pada *work centre* ini terjadi proses produksi dari seluruh komponen masing-masing produk yang berbahan pelat besi untuk akhirnya akan dirakit menjadi produk jadi. Kegiatan produksinya sebagai berikut:

- a. *Cutting*

Proses ini merupakan proses pemotongan pelat besi sesuai desain yang diinginkan oleh pemesan dan dengan spesifikasi tertentu. Pemotongan dapat dilakukan dengan menggunakan mesin *Copier Gas Cutting* yang secara otomatis akan memotong pelat sesuai dengan desain yang di-input oleh operator. Sumbu api akan memotong pelat sesuai dengan koordinat polanya dan lamanya waktu pemotongan akan berbanding lurus dengan tebal pelat dan panjang pelat.

b. *Bending*

Proses ini menekuk pelat menjadi bentuk siku atau lengkung sesuai dengan kebutuhan. Terkadang proses ini diperlukan untuk melengkungan pelat yang memiliki spesifikasi diameter lebih kecil dibandingkan diameter mesin *roll*. Produk *laddle* yang berdiameter kecil menggunakan mesin *bending* untuk membentuk *shell* menjadi bentuk lengkung.

c. *Rolling*

Proses ini adalah proses melengkungkan pelat sehingga dapat melengkung menjadi komponen *shell* sesuai dengan spesifikasi diameter yang diinginkan. Terdapat beberapa mesin *rolling* yang dapat digunakan selama proses produksi bergantung pada besarnya diameter komponen dan tebal pelat yang digunakan.

d. *Dishing*

Proses ini merupakan proses penempaan pelat pada titik-titik tertentu yang telah dibentuk menjadi lingkaran agar dapat membentuk *head* yang mirip seperti wajan. Mesin *Head dishing* diatur sedemikian rupa sehingga matras (tempat pelat diletakkan) dan bola pejal dapat menempa pelat beberapa kali hingga sesuai dengan diameter yang diinginkan. Umumnya proses ini memakan waktu 3-6 jam untuk menghasilkan 1 *head* bergantung pada besar/kecilnya diameter *head*.

e. *Flanging*

Flanging merupakan kegiatan produksi yang membentuk mulut *head* agar bisa membentuk radius sehingga akan memudahkan proses perakitan dengan komponen lain, yaitu biasanya dengan komponen *shell*. Mulut *head* diapit pada 2 bola pejal (dibagian dalam dan luar *head*) yang kemudian *head* akan diputar sehingga ujung *head* lama kelamaan akan melunak membentuk radius sesuai dengan kebutuhan.

f. *Bevelling*

Bevelling adalah suatu proses memotong ujung masing-masing item komponen berbahan pelat membentuk huruf “v” atau serong. Proses ini dibutuhkan untuk seluruh item yang akan disambungkan dengan item lain menggunakan mesin las. Pemotongan bergantung pada tebal pelat yang akan disambungkan. Apabila pelat mempunyai tebal hingga 20mm, maka ujung pelat tersebut akan dipotong serong saja sudah cukup. Namun, untuk pelat yang mempunyai tebal melebihi 20mm maka lebih baik ujung pelat dipotong bentuk huruf “v”.

Penentuan jenis pemotongan ini akan menentukan proses las selanjutnya. Pelat-pelat yang tebal akan lebih mudah disambung apabila ujung pelatnya berpotongan huruf “v” karena akan meminimalisir gas yang masuk di daerah lasan dan hasil lasan akan mempunyai kualitas yang lebih baik. Berbeda dengan pelat tipis, potongan serong sudah cukup menghasilkan daerah lasan yang kuat dan berkualitas baik.

g. *Grinding*

Gerinda merupakan proses penggerindaan komponen, yaitu penghalusan komponen dari bentuk yang tidak sesuai dengan desain yang ada. Bentuk yang tidak diinginkan ini misalnya seperti hasil las yang meluber, perbaikan hasil lasan, penghalusan komponen setelah pemotongan, dsb. Mata pisau yang berbentuk silinder diputar pada sumbu pusatnya menggunakan motor sehingga apabila didekatkan dengan benda kerja akan memakan benda kerja tersebut memanfaatkan gesekan yang terjadi. Gesekan akan terus diberikan selama benda kerja masih belum sesuai dengan desain yang ada. Prinsip kerja mesin ini hampir sama dengan kasa saat digesekkan pada kayu, lama kelamaan gaya gesek yang terjadi akan dapat menghaluskan kayu.

2. *Work centre 2 (assembly).*

Assembly merupakan proses penggabungan dari beberapa bagian komponen yang terdiri dari pelat tebal dan pelat tipis untuk membentuk suatu konstruksi yang diinginkan. Kegiatan produksinya sebagai berikut:

a. *Fit-up*

Fit-up merupakan proses awal sebelum antar komponen akan di las. Ujung komponen yang akan disambung dicocokkan dan disambung di beberapa titik saja melalui proses *fit-up*. Proses ini bertujuan untuk memudahkan selama proses las berlangsung agar komponen-komponennya tidak bergeser sehingga akan menghasilkan daerah sambungan yang presisi. Pada proses ini terdapat 5 mesin yang dapat digunakan secara bersamaan untuk meringankan beban kerja.

b. *Welding*

Proses *welding* merupakan proses pengelasan atau penyambungan antar item/komponen sehingga menjadi komponen lain atau produk akhir. Proses *welding* yang dilakukan untuk produk ini menggunakan mesin las biasa (SAW) yang dipasang secara paralel, yaitu terdapat 5 mesin yang bisa digunakan bersamaan di *work centre* ini.

4.1.4 Bahan Baku Produk Terkait

Bahan baku yang digunakan untuk produk kualiti timah dan *laddle* mayoritas berasal dari banyak plat yang kemudian diukur dan dibentuk sesuai dengan desain. Plat-plat tersebut dipesan perusahaan dari *supplier* plat dalam negeri dengan beberapa ketentuan yang mencakup perbedaan spesifikasi sebagai berikut:

1. Plat A 516-70 ukuran 12 x 1500 x 6000
2. Plat A 516-70 ukuran 12 x 18 00 x 6000
3. Plat A 515 Gr.70 ukuran 12 x 1500 x 6000
4. Plat A 283 Gr. C ukuran 12 x 1800 x 6000

Dalam proyek ini perusahaan juga menggunakan beberapa bahan baku pembantu dimana bahan baku ini digunakan untuk melengkapi bahan baku utama dalam memproduksi kualiti timah dan *laddle*. Bahan baku pembantu bisa ikut habis terolah namun dalam jumlah pemakaian yang tidak begitu nampak, sehingga apabila tidak dicermati dengan baik dapat menimbulkan kesenjangan antara perencanaan dan pelaksanaannya. Dalam proyek ini bahan baku pembantu yang digunakan antara lain adalah cat, kawat busur las, kawat argon, *coolant*, kertas gosok, *thinner*, dan *Vaseline*.

4.1.5 Data Pesanan di Workshop III

Berikut merupakan data pesanan yang diterima di *workshop* III perusahaan:

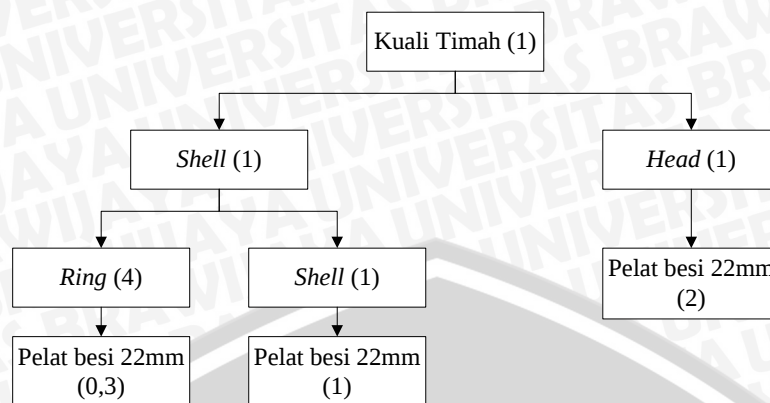
Tabel 4.1 Data Pesanan *Workshop* III PT. Barata Indonesia (Persero) Tahun 2014

No. Order	Jenis Produk	Order Qty. (unit)	Mulai	Dateline	Pemesan
PLG 9.13.326	<i>Laddle</i> Timah	10	22-12-13	22-01-14	PT. Non Ferindo Utama
PLG 9.14.007	Kualiti Timah	10	06-01-14	20-02-14	PT. Non Ferindo Utama
PLG 9.14.011	<i>Laddle</i> Timah	25	16-01-14	17-03-14	PT. Non Ferindo Utama

4.1.6 Bill of Material (BOM) Tree Produk

BOM *tree* adalah daftar kebutuhan material secara lengkap yang dibutuhkan beserta jumlah komponen penyusun, yang disajikan dalam bentuk skema secara berurutan menurut level *breakdown*. Secara umum informasi-informasi yang didapatkan dari BOM *Tree* diantaranya adalah mengetahui level pem-*breakdown*-an dari tiap komponen produk kemudian mengetahui jumlah atau banyaknya komponen yang dibutuhkan dalam pembuatan satu unit produk serta dapat mengetahui komponen-komponen mana saja yang akan dilakukan proses *assembly* (perakitan). Berikut penjabaran BOM *Tree* dari produk kualiti timah dan *laddle*.

1. BOM Tree Produk Kualiti Timah

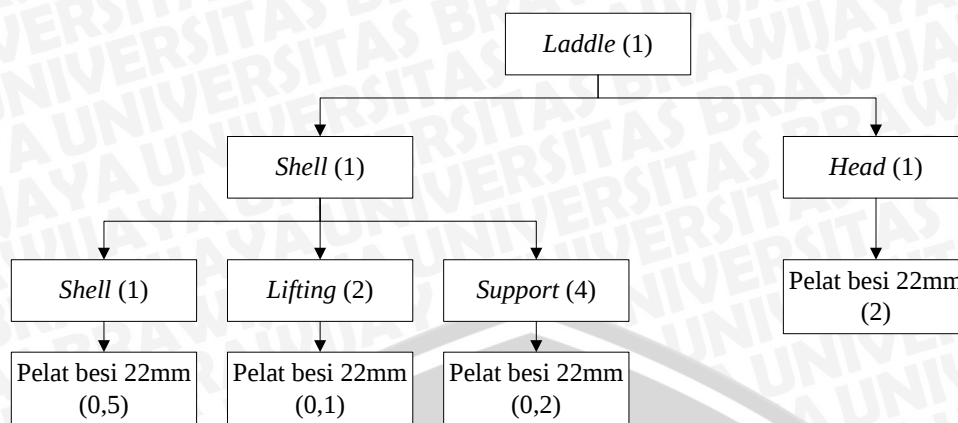


Gambar 4.4 BOM tree produk kualiti timah

Produk ini seluruhnya terbuat dari pelat besi yang sama. Pada Gambar 4.4 dijelaskan susunan level dari komponen yang dibutuhkan oleh kualiti timah. Kualiti timah membutuhkan 7 komponen yang berasal dari 4 level BOM Tree dimulai dari level 0 hingga level 3. Produk kualiti timah membutuhkan komponen *shell* sebanyak 1 item dan *head* sebanyak 1 item di level 1. Sehingga dengan kata lain, produk kualiti timah baru bisa diproduksi ketika komponen *shell* dan *head* telah tersedia dari proses sebelumnya. Komponen *ring* sebanyak 4 item dan *shell* sebanyak 1 item pada level 2 dan dikerjakan bersamaan untuk membentuk badan *shell*. Lebih jelasnya untuk kebutuhan material berupa pelat besi dari masing-masing komponen telah dijabarkan pada BOM Tree level 3.

2. BOM Tree Produk Laddle

Produk *laddle* memiliki ukuran yang lebih kecil dibandingkan dengan kualiti timah. Produk ini biasanya digunakan sebagai kualiti tempat menampung timah panas sebelum dimasukkan ke dalam cetakan. Dalam proses pembuatannya, *Laddle* memerlukan 9 komponen yang berasal dari 4 level BOM Tree. 1 unit *laddle* dapat diproduksi setelah didapat 1 komponen *shell* dan 1 komponen *head* dari proses sebelumnya, hal ini ditunjukkan pada level 1 BOM Tree. Selanjutnya pada level 2 dijelaskan komponen yang dibutuhkan untuk memproduksi 1 item *shell* yang terdiri dari 1 item *shell* itu sendiri dan 6 item atribut berupa 2 *lifting* dan 4 *support*. Sedangkan bagian *head* terbentuk dari proses *dishing* pelat besi dengan diameter yang lebih kecil dibandingkan *head* untuk produk kualiti timah. Lebih jelasnya untuk kebutuhan material berupa pelat besi dari masing-masing komponen telah dijabarkan pada BOM Tree level 3 sesuai pada Gambar 4.5.



Gambar 4.5 BOM tree produk laddle

4.1.7 Data Mesin untuk Produk Kualiti Timah dan Laddle.

Dalam memproduksi kualiti timah dan laddle terdapat beberapa mesin yang terbagi menjadi dua kategori yaitu mesin-mesin yang digunakan untuk kegiatan yang bersifat *forming* (pembentukan) dan *assembly* (perakitan). Mesin-mesin yang termasuk dalam kegiatan *forming* dipasang secara *single* untuk semua mesinnya, sedangkan untuk kegiatan perakitan mesin-mesin dipasang secara paralel. Untuk lebih jelasnya, deskripsi dan spesialisasi mesin akan dibahas sebagai berikut:

1. Kegiatan *forming* (pembentukan)

Kegiatan *forming* merupakan kegiatan membentuk pelat datar menjadi pola/ bentuk sesuai dengan desain produk yang dipesan oleh *customer*. Kegiatannya meliputi *cutting*, *dishing*, *flanging*, *bending*, dll dimana setelah diberikan proses *forming* maka pelat akan berubah bentuk meskipun perubahannya hanya sedikit saja. Mesin-mesin yang tergolong pada kegiatan ini antara lain adalah sebagai berikut :

a. Copier Gas Cutting Machine (Mesin 1)



Gambar 4.6 Mesin Copier Gas Cutting

Secara umum mesin ini digunakan untuk memotong pelat sesuai dengan pola item secara otomatis. Pelat diletakkan pada meja potong dan kemudian desain

dimasukkan secara komputerisasi. Ujung sumbu menggunakan api terfokus yang sangat panas diprogram secara otomatis sehingga berjalan teratur mengikuti pola yang sudah dimasukkan ke dalam komputer dan menghasilkan potongan pelat sesuai dengan pola yang diinginkan. Secara rinci kegunaan dari mesin ini adalah sebagai berikut :

- Pengaturan dan kontrol operasi terpusat.
- Pemotongan pelat secara otomatis.
- Prosesnya berlangsung secara terus menerus.
- Kontrol *tracer* menggunakan foto.
- Memiliki meja yang luas sehingga dapat memotong banyak pola dari 1 pelat yang berukuran besar/panjang.

b. Mesin *Bevel* (Mesin 2)

Merupakan mesin yang digunakan untuk membentuk huruf “v” atau serong pada ujung potongan pelat. Ujung yang seperti ini berguna untuk memudahkan pelat saat akan di las (*assembly*) dengan plat bagian lain. Mesin *bevel* menggunakan sumbu api yang terpusat untuk membentuk ujung pelat dengan waktu yang cepat, yaitu 16-18 menit untuk panjang plat sekitar 1-1,5 meter.

c. Mesin *Head Dishing* (Mesin 3)



Gambar 4.7 Mesin *Head Dishing*

Mesin ini berfungsi untuk membentuk lengkungan setengah bola dari plat baja yang awalnya berpermukaan datar. Proses *dishing* ini dilakukan dengan cara menempa pelat secara berulang-ulang bergantian di seluruh bagian sehingga didapatkan ukuran diameter yang sesuai dengan desain. Waktu yang diperlukan dalam proses *dishing* ini sekitar 3-6 jam untuk setiap *head*-nya. Hal ini bergantung pada diameter yang dikerjakan dan kesesuaian dimensi yang

didapatkan. Sebelum operasi dilakukan, matras dan tipe bola pejal dipasang terlebih dahulu yang disesuaikan dengan dimensi pelat yang akan dibentuk.

d. Mesin *Head Flanging* (Mesin 4)



Gambar 4.8 Mesin *Head Flanging*

Mesin ini merupakan mesin yang semi-otomatis dimana operator masih harus melakukan pengecekan dan pengoperasian di setiap tahapannya. Mesin *head flanging* digunakan untuk meluruskan ujung *head* (wajan) yang dihasilkan dari proses *dishing* sehingga dapat membentuk radius. Awalnya setelah proses *dishing* ujung wajan akan cenderung menjorok keluar, maka fungsi dari mesin ini adalah untuk meluruskan ujung *head* yang menjorok tersebut agar tegak lurus dengan tanah dengan jarak/radius tertentu.

e. *Mechanical Plate Bend Rolling Machine* (Mesin 5)



Gambar 4.9 Mesin *Roll and Bending*

Mesin *Rolling* motor adalah mesin yang berfungsi untuk membulatkan plat baja. Ketebalan lembaran plat baja yang biasa dibulatkan adalah 22 mm. Mesin ini memiliki tiga pasang *roller* dimana penggeraknya adalah motor listrik.

f. Mesin Press Hidrolik (Mesin 6)

Untuk menekuk sebatang pipa lurus dengan besar sudut yang diinginkan dalam jumlah massal, sebuah mesin diperlukan dalam proses pengerjaannya. Mesin

bending dapat menekuk sebuah plat sesuai ukuran yang ditentukan. Cara kerjanya yaitu dengan memasukkan plat dalam mesin sesuai ukuran yang dikehendaki, kemudian kencangkan pengencang lalu putar pemutar hingga plat menjadi tertekuk sesuai dengan sudut yang dibutuhkan. Selain itu, mesin ini juga dapat digunakan untuk proses penekanan, baik untuk proses *drawing*, *punching*, *blanking*, *fiting*, *shearing*, *bending*, *forging* ataupun yang lainnya.

g. Mesin Gas Turning Table (Mesin 7)



Gambar 4.10 Mesin Gas Turning Table

Mesin ini merupakan mesin gerinda yang dioperasikan secara otomatis dan cocok digunakan untuk menghaluskan permukaan benda yang berukuran besar. Fungsi dari mesin ini adalah untuk memotong ujung *head* setelah proses *flanging* agar menghasilkan ujung yang rata simetris. Perlakuan ini berguna untuk memudahkan proses las yang biasanya dilakukan setelah *head* selesai diproduksi. Permukaan yang rata juga akan menambah presisi apabila *head* akan dirakit dengan komponen lainnya lebih lanjut. Cara kerjanya adalah *head* diletakkan di atas *table*, kemudian sumbu api dipaskan agar tepat mengenai ujung *head* pada ukuran tertentu. Setelah sumbu dirasa pas maka mesin dinyalakan sehingga *table* akan bergerak memutar secara perlahan dan api juga akan memotong rata ujung *head* tersebut.

2. Kegiatan *Assembly* (perakitan)

Kegiatan *Assembly* merupakan kegiatan lanjutan setelah pola/bentuk didapatkan yang bertujuan untuk menyatukan bagian-bagian/item dari produk agar bisa

menjadi satu membentuk produk akhir. Kegiatan ini meliputi proses *welding*, *grinding*, serta inspeksi yang membutuhkan waktu relatif lebih lama dibandingkan dengan kegiatan *forming*. Mesin-mesin yang tergolong ke dalam kegiatan *assembly* antara lain adalah sebagai berikut :

a. Mesin Las (*welding*) (Mesin 8)

Proses *welding* ini merupakan proses penyambungan material, dimana plat-plat material yang diperoleh dari *customer* disambung agar dapat mencapai diameter yang diinginkan. Pemborosan waktu yang timbul pada proses *welding* ini adalah proses persiapan peralatan yang memerlukan waktu sekitar 10 menit. Mesin las yang terdapat di *workshop* cukup banyak macamnya mengingat kebanyakan proses produksi yang dilakukan pada *workshop* adalah proses las (*welding*). Dalam penelitian ini, mesin *welding* yang digunakan adalah mesin *welding* SAW yang dipasang paralel dan berjumlah 5. Beberapa jenis mesin las lainnya yang sering dipakai antara lain :

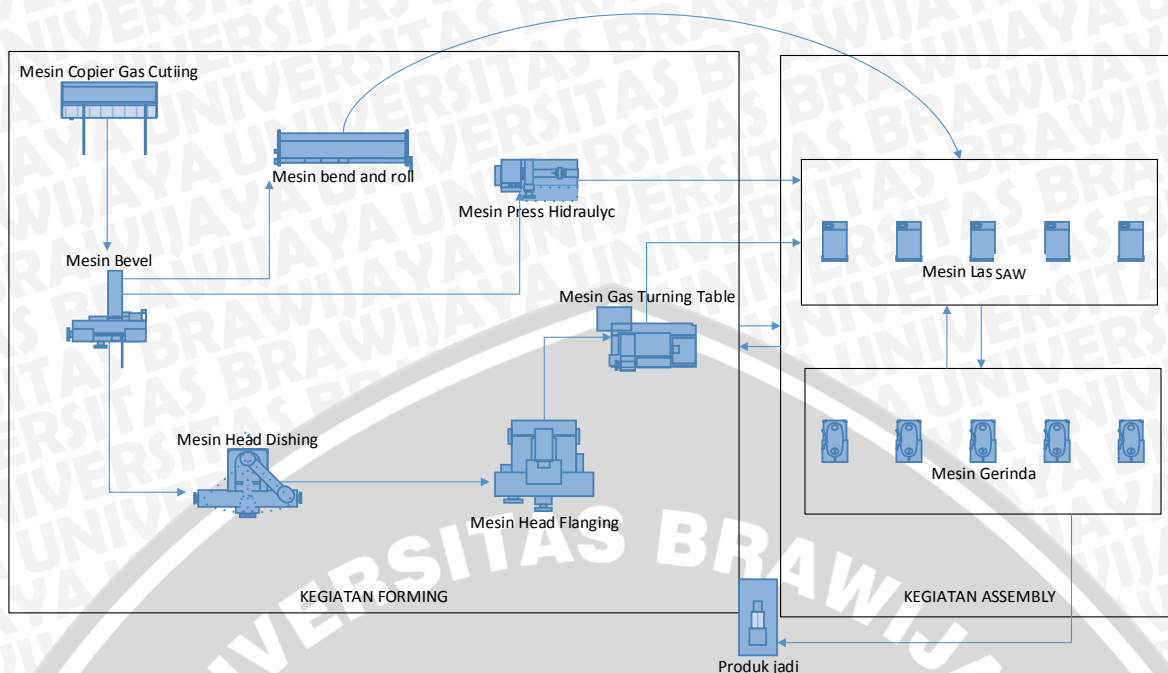
1. Mesin las busur listrik (SMAW)
2. Mesin las Argon (MTGW)
3. Mesin las Asitelin
4. Mesin las SAW

b. Mesin Gerinda (Mesin 9)

Mesin gerinda adalah salah satu mesin perkakas yang digunakan untuk memotong/mengasah benda kerja dengan tujuan seperti menghaluskan, menghilangkan das *pasca welding*, menyesuaikan produk sesuai dengan desainnya, dll. Mesin gerinda yang terdapat di *workshop* ini adalah sejumlah 5 dan semua mesin tersebut digunakan secara paralel untuk mengerjakan produk kualiti timah dan *laddle*. Prinsip kerja mesin gerinda adalah roda gerinda berputar bersentuhan dengan benda kerja dan terjadi pemotongan/pengasahan. Jenis mesin gerinda yang digunakan untuk kedua produk ini adalah gerinda motor yang ukurannya lebih kecil dibandingkan dengan gerinda pedestal.

4.1.8 Data Alur Proses Produksi, Alokasi Waktu dan Alokasi Mesin

Pada penelitian ini, khususnya bagi produk berupa kualiti timah dan *laddle* akan memiliki diagram alur proses produksi seperti pada Gambar 4.11. Sedangkan untuk alokasi waktu dan urutan pengerjaan secara rinci akan dijelaskan secara sistematis per tahap pengerjaan pada subbab berikut.



Gambar 4.11 Alur proses produksi untuk produk kualiti timah dan *ladle*

4.1.8.1 Produk Kualiti Timah

Urutan operasi dari produk kualiti timah, operasi pendahulunya (*precedence*), alokasi waktu, serta alokasi mesin yang digunakan untuk masing-masing operasi ditampilkan pada Tabel 4.2. Waktu proses yang ditampilkan pada Tabel tersebut merupakan waktu operasi yang sudah diperhitungkan terhadap kebutuhan waktu *set-up* dan *handling*-nya. Besar dari kebutuhan masing-masing waktu *set-up* dan *handling* berbeda-beda sesuai dengan mesin yang digunakan dan material yang dipindahkan. Sesuai dengan batasan dan asumsi pada penelitian ini, kedua waktu tersebut dianggap sudah menjadi satu dengan waktu prosesnya. Hal ini juga berlaku bagi produk *ladle*. Jadi dengan kata lain, waktu operasi yang terdapat pada Tabel 4.2 dan Tabel 4.3 sudah termasuk kebutuhan waktu *set-up* dan *handling*.

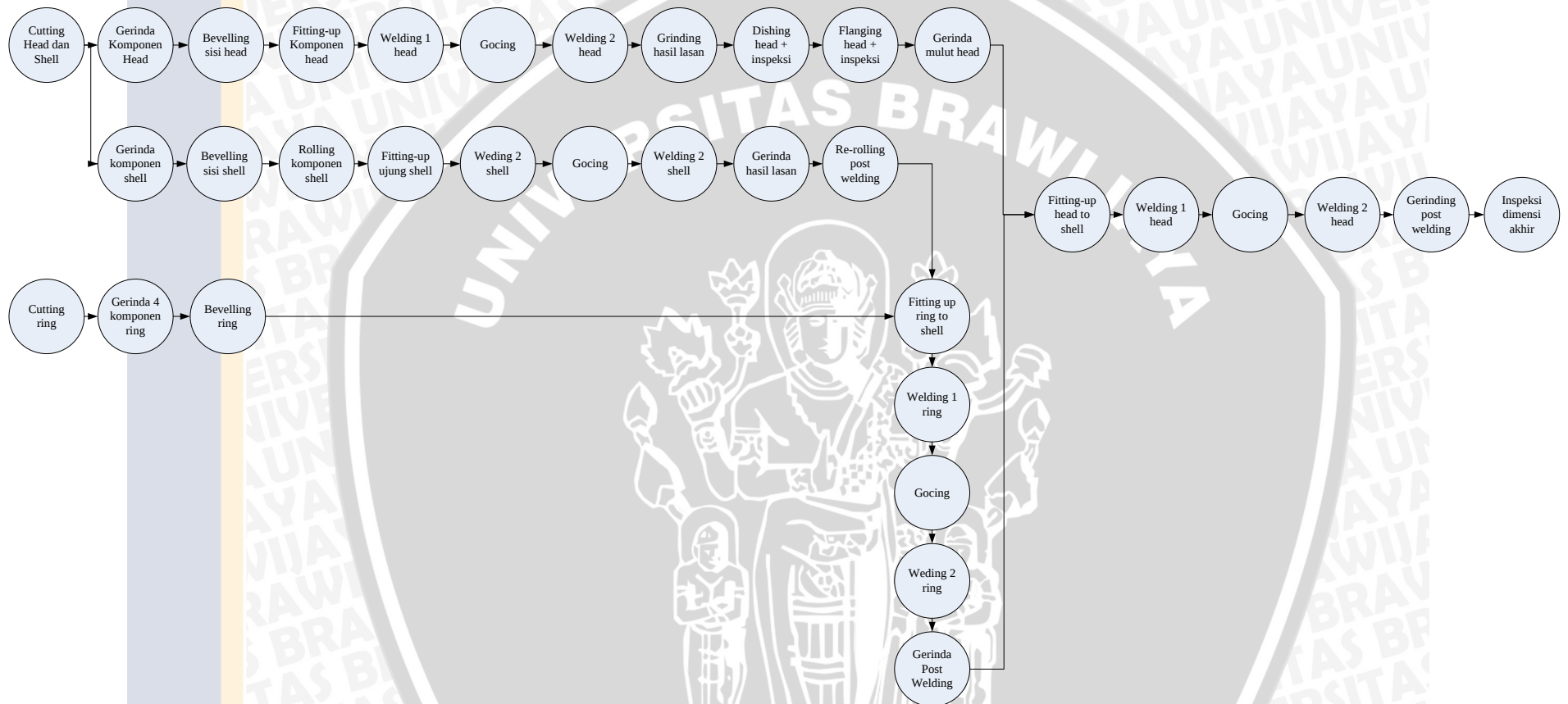
Tabel 4.2 Data Alur Proses Produksi, Alokasi Waktu, dan Alokasi Mesin Produk Kualiti Timah

Aktivitas	Deskripsi	Precedence	Mesin	Waktu Proses (menit)
(a)	(b)	(c)	(d)	(e)
1	Cutting head dan shell	-	Mesin Copier Gas Cutting	60
2	Cutting ring	-	Mesin Copier Gas Cutting	16
3	Gerinda hasil potong shell	1	Mesin Gerinda	10
4	Gerinda hasil potong head	1	Mesin Gerinda	12
5	Gerinda hasil potong ring	2	Mesin Gerinda	13
6	Bevelling sisi panjang dan 2 lebar shell	3	Mesin Bevel	30
7	Bevelling sisi head	4	Mesin Bevel	40
8	Bevelling ring	5	Mesin Bevel	35

Tabel 4.2 Data Alur Proses Produksi, Alokasi Waktu, dan Alokasi Mesin Produk Kualiti Timah (Lanjutan)

9	<i>Rolling shell</i>	6	Mesin <i>Roll and Bending</i>	240
10	<i>Fitting-up the shell</i>	9	Mesin <i>Welding</i>	90
11	<i>Welding 1 (shell)</i>	10	Mesin <i>Welding</i>	100
12	Gocing	11	Mesin Gerinda	40
13	<i>Welding 2 (shell)</i>	12	Mesin <i>Welding</i>	100
14	Gerinding hasil lasan	13	Mesin Gerinda	45
15	<i>Re-Rolling shell</i> setelah di las	14	Mesin <i>Roll and Bending</i>	135
16	<i>Fitting-up sisi-sisi head</i>	7	Mesin <i>Welding</i>	45
17	<i>Welding 1 (head)</i>	16	Mesin <i>Welding</i>	100
18	Gocing	17	Mesin Gerinda	50
19	<i>Welding 2 (head)</i>	18	Mesin <i>Welding</i>	120
20	Gerinding hasil lasan	19	Mesin Gerinda	45
21	<i>Dishing head</i> + inspeksi	20	Mesin <i>Head Dishing</i>	240
22	<i>Flanging</i> + inspeksi	21	Mesin <i>Head Flanging</i>	240
23	Gerinda mulut <i>head</i>	22	Mesin Gas Turning Table	60
24	<i>Fitting-up ring dan shell</i>	8,15	Mesin <i>Welding</i>	90
25	<i>Welding 1 ring dan shell</i>	24	Mesin <i>Welding</i>	150
26	Gocing	25	Mesin Gerinda	40
27	<i>Welding 2 ring dan shell</i>	26	Mesin <i>Welding</i>	160
28	Gerinda hasil lasan	27	Mesin Gerinda	20
29	Pencocokan <i>Head (fit-up)</i>	23,28	Mesin <i>Welding</i>	150
30	<i>Welding 1 head ke shell</i>	29	Mesin <i>Welding</i>	210
31	Gocing	30	Mesin Gerinda	50
32	<i>Welding 2 head ke shell</i>	31	Mesin <i>Welding</i>	210
33	Gerinda hasil lasan	32	Mesin Gerinda	90
34	Inspeksi dimensi	33	-	50

Dari 4.2 dapat diketahui urutan operasi serta operasi pendahulunya. Selain itu, diagram alir dari penggunaan mesin dan kegiatan produksi produk kualiti timah juga semakin terdefinisi. *Precedence* diagram dapat dilihat pada Gambar 4.12.



Gambar 4.12 Precedence diagram produk kualiti timah

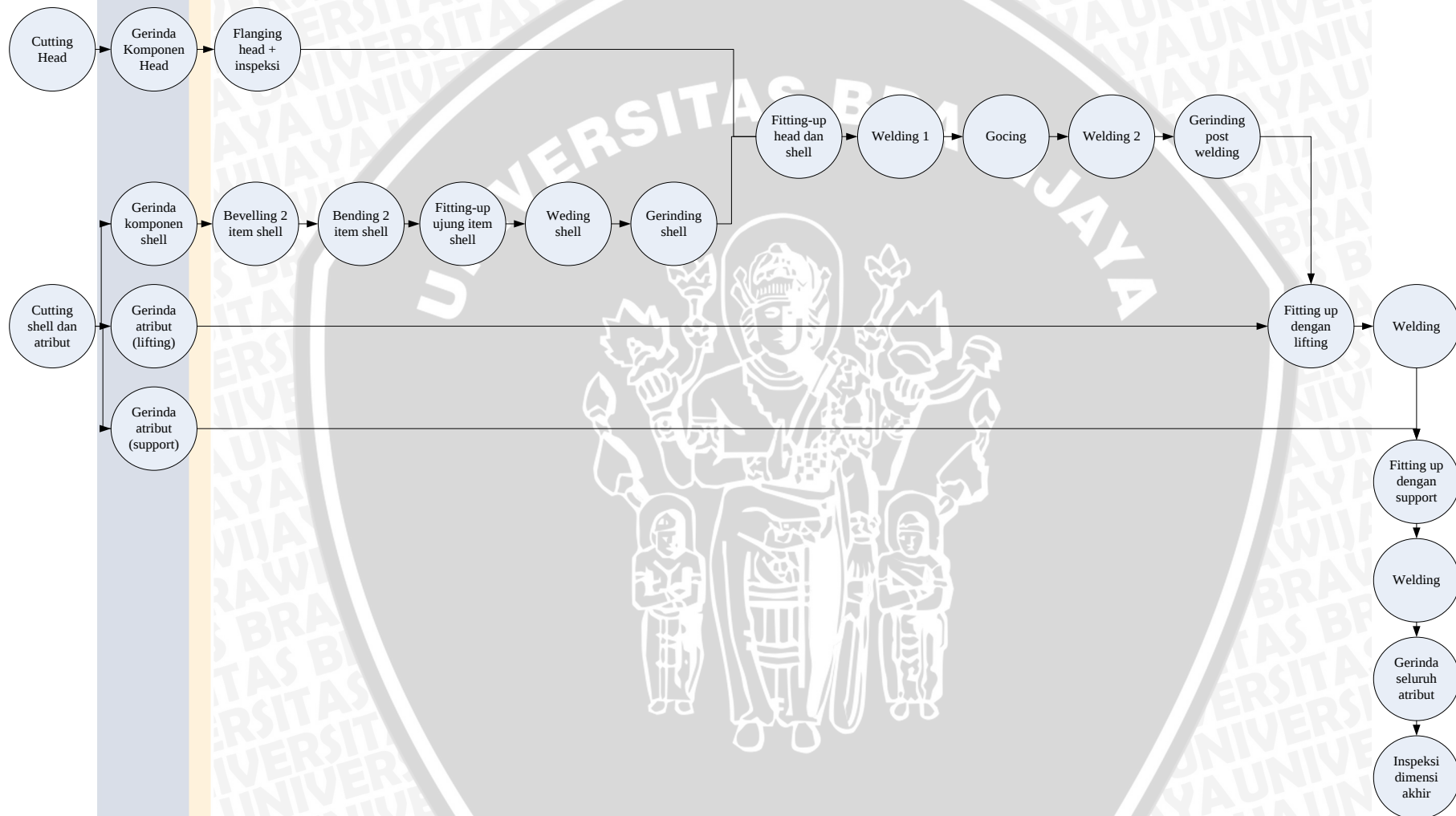
4.1.8.2 Produk *Laddle*

Berikut merupakan tabel keterangan dari urutan operasi dari produk *laddle*, operasi pendahulunya, alokasi waktu, serta alokasi mesin yang digunakan untuk masing-masing operasi ditampilkan pada Tabel 4.3.

Tabel 4.3 Data Alur Proses Produksi, Alokasi Waktu, dan Alokasi Mesin Produk *Laddle*

Aktivitas	Deskripsi	Precedence	Mesin	Waktu Proses (menit)
(a)	(b)	(c)	(d)	(e)
1	<i>Cutting head</i>	-	Mesin Copier Gas Cutting	16
2	<i>Cutting shell</i> dan atribut	-	Mesin Copier Gas Cutting	20
3	Gerinda hasil potong <i>head</i>	1	Mesin Gerinda	12
4	Gerinda hasil potong <i>shell</i>	2	Mesin Gerinda	10
5	Gerinda hasil potong <i>lifting</i>	2	Mesin Gerinda	2
6	Gerinda hasil potong <i>support</i>	2	Mesin Gerinda	2
7	<i>Bevelling</i> sisi 2 segmen <i>shell</i>	4	Mesin Bevel	35
8	<i>Bending segment shell</i>	7	Mesin Press Hidrolik	120
9	<i>Fitting-up</i> antar segmen <i>shell</i>	8	Mesin Welding	150
10	<i>Welding</i> antar <i>shell</i>	9	Mesin Welding	270
11	Gerinda <i>Shell</i>	10	Mesin Gerinda	18
12	<i>Dishing head</i> + inspeksi	3	Mesin Head Dishing	240
13	<i>Fitting-up head</i> dan <i>shell</i>	11,12	Mesin Welding	90
14	<i>Welding</i> 1 <i>head</i> dan <i>shell</i>	13	Mesin Welding	135
15	Gocing	14	Mesin Gerinda	30
16	<i>Welding</i> 2 <i>head</i> dan <i>shell</i>	15	Mesin Welding	135
17	Gerinda hasil las	16	Mesin Gerinda	18
18	<i>Fitting-up</i> dengan <i>lifting</i>	5,17	Mesin Welding	150
19	<i>Welding lifting</i> ke ladle	18	Mesin Welding	150
20	<i>Fitting-up support</i> ke ladle	6,19	Mesin Welding	120
21	<i>Welding support</i> ke ladle	20	Mesin Welding	100
22	Gerinda hasil lasan	21	Mesin Gerinda	42
23	Inspeksi dimensi total	22	-	50

Dari 4.3 dapat diketahui urutan operasi serta operasi pendahulunya. Selain itu, diagram alir dari penggunaan mesin dan kegiatan produksi *laddle* juga semakin terdefinisi. *Precedence* diagram dapat dilihat pada Gambar 4.13.



Gambar 4.13 Precedence diagram produk ladle

4.1.9 Jadwal Induk Produksi *Existing* PT. Barata Indonesia

Berdasarkan informasi yang diberikan oleh bagian PPC PT. Barata Indonesia, jadwal induk *existing* yang dikeluarkan oleh bagian PPC dibuat berdasarkan aturan FCFS dimana menurut aturan ini *job* diprioritaskan berdasarkan waktu kedatangannya. Namun dalam pelaksanaannya langsung di rantai produksi aturan ini tidak berlaku. Proses produksi yang diterapkan di rantai produksi diprioritaskan berdasarkan besarnya nilai kontrak. Selain itu, penentuan *due date* dilakukan berdasarkan perkiraan dari waktu historis produk sejenis dalam kesepakatan kontrak dengan pihak pertama.

Seharusnya, pada aturan FCFS sudah jelas memandu rantai produksi agar mengerjakan *job* sesuai dengan waktu kedatangan *order* di bagian pemasaran, bukan kedatangan *order* di rantai produksi. Terkadang waktu kedatangan kedua hal ini bisa berbeda apabila dipengaruhi oleh waktu kedatangan material yang terlambat, atau karena menunggu mesin produksi yang sedang digunakan. Sehingga ketika terdapat *order* yang datang di bagian pemasaran, maka seharusnya *order* ini diprioritaskan untuk dikerjakan di rantai produksi sesuai dengan tanggal persetujuan kontraknya. Apabila terdapat *order-order* yang datang secara bersamaan maka rantai produksi biasanya akan cenderung memprioritaskan berdasarkan besarnya nilai kontrak *order* tersebut.

Penentuan prioritas pengerjaan *job* berdasarkan nilai kontraknya akan membuat kegiatan produksi semakin tidak jelas. *Job* yang memiliki waktu kedatangan lebih awal atau *job* dengan *due date* yang lebih pendek namun nilai kontraknya kecil akan dikerjakan setelah *order* dengan nilai kontrak besar selesai. Ketika *order-order* dengan nilai kontrak yang kecil dikesampingkan hal ini mungkin akan menguntungkan perusahaan karena mengurangi penalti keterlambatan dari *job* yang memiliki nilai kontrak besar. Namun bila ditinjau kembali, *job* bernilai kontrak kecil juga akan terlambat karena waktu mulai dari *job*-nya terpakai untuk *job* yang lain sehingga hal ini akan sama-sama membuat perusahaan terkena penalti. Perusahaan mungkin tidak terlalu mempermasalahkan penalti yang harus dibayarkan, tapi meskipun nominal penalti masih dirasa wajar hal tersebut bisa mengurangi kepercayaan konsumen karena waktu penyelesaian *order* lebih lama dibandingkan dengan kontrak yang disepakati.

Berdasarkan data historis yang ada, lama keterlambatan *job* yang ditanggung perusahaan berkisar selama 5-115 hari. Angka tersebut terbilang besar mengingat peraturan penalti cukup memberatkan dan jika dibiarkan terus menerus maka hal ini akan berpengaruh tidak baik bagi kondisi keuangan perusahaan. Oleh karena itu perusahaan membutuhkan penjadwalan mesin yang mendekati optimal agar semua *job*

dapat dijadwalan tepat pada waktunya sehingga dapat mengurangi nilai penalti yang harus ditanggung oleh perusahaan. Penjadwalan yang diusulkan harus menjelaskan secara rinci pekerjaan di setiap mesin sesuai dengan harinya. Jadwal *existing* yang diterapkan oleh perusahaan terhadap tiga produk yang akan dikerjakan di triwulan awal tahun 2014 akan ditampilkan pada Lampiran 1 hingga Lampiran 3 untuk masing-masing produk sesuai dengan nomor urut waktu datang *order*.

4.1.10 Perhitungan Penalti

Berdasarkan data yang didapat dari bidang PPC *workshop* III, ketentuan besarnya penalti bergantung dari lamanya keterlambatan waktu penyelesaian kontrak yang bersangkutan. Berikut merupakan penjelasan aturan penetapan penalti yang diberlakukan terhadap perusahaan oleh pemesan.

- a. Penalti sebesar 5⁰/₀₀ dihitung setiap harinya dari keseluruhan nilai kontrak produk terkait, ketika keterlambatan yang terjadi masih kurang dari sebulan (30 hari)
- b. Penalti sebesar 5% dari keseluruhan nilai kontrak produk untuk keterlambatan melebihi waktu 30 hari. Jadi, berapa lamapun banyaknya hari yang terlambat, ketika sudah melebihi waktu 30 hari maka besarnya penalti pun akan sama sebesar nilai yang ditetapkan.

4.2 Pengolahan Data

Data-data yang terkumpul, selanjutnya digunakan sebagai *input* dalam pengolahan data untuk mendapatkan hasil dan pembahasan analisis yang sesuai dengan tujuan penulisan dari penelitian. Pengolahan data akan dijelaskan pada subbab berikut.

4.2.1 Model Penjadwalan Usulan

Penjadwalan pada penelitian ini bertujuan untuk memecahkan masalah keterlambatan atau *tardiness* di lingkungan produksi *Flexible Job Shop*. Pengertian *Flexible Job Shop* dalam sistem produksinya karena terdapat dua stasiun kerja (*work centre*) dimana pada salah satu stasiunnya terdapat mesin yang dipasang secara paralel dan terdapat alur yang bolak-balik antar stasiunnya.

Pada kondisi asli *Flexible Job Shop*, ketika terdapat operasi yang harus dikerjakan di mesin yang paralel maka penugasan operasi dari *job* yang siap akan langsung dialokasikan ke mesin manapun yang kosong. Namun, dalam penelitian ini beban setiap waktu operasi dari *job* dibagi merata di seluruh mesin yang paralel.

Sehingga meskipun terdapat mesin yang dipasang paralel, mesin tersebut dianggap sebagai mesin yang *single* (tunggal) dan setiap operasi yang akan diproses harus menunggu operasi sebelumnya diselesaikan terlebih dahulu meskipun ada beberapa mesin yang paralel tersebut dalam kondisi *eligible*.

Penyelesaian yang diusulkan untuk permasalahan pada lingkungan ini adalah dengan melakukan pendekatan kepada lingkungan *Pure Job Shop* yang berorientasi untuk meminimalkan total *tardiness* dan penalti yang mungkin akan terjadi. Model penjadwalan yang diusulkan adalah menggunakan metode pengurutan FCFS dan EDD. Metode FCFS akan memprioritaskan penjadwalan *job* berdasarkan waktu kedatangan *job*, sedangkan metode EDD memprioritaskan penjadwalan dalam rangka pemenuhan *due date*-nya. Hal ini dilakukan dengan pertimbangan bahwa waktu kedatangan *job* yang bersifat dinamis deterministik memungkinkan bahwa beberapa *job* akan datang di waktu yang bersamaan. Apabila kondisi seperti ini terjadi maka *order-order* ini tidak akan dikerjakan secara bersamaan. *Order* akan dipertimbangkan untuk dijadwalkan menggunakan aturan EDD sehingga *order* yang memiliki *due date* paling awal yang akan dikerjakan lebih dulu.

Dalam algoritma *sequencing* yang diusulkan ada beberapa penyesuaian dari metode yang digunakan dengan kondisi langsung yang terjadi dalam sistem. Sistem produksi yang berupa *Flexible Job Shop* memungkinkan terjadinya urutan operasi yang berbeda-beda untuk *job* yang akan dikerjakan di lantai produksi. Selain itu, waktu siap yang dipertimbangkan dari waktu kedatangan kontrak juga disesuaikan. *Job* yang dijadwalkan untuk diproduksi di triwulan awal tahun adalah *order* yang telah diketahui oleh perusahaan di awal tahun dan sifatnya adalah sudah pasti akan dikerjakan sesuai dengan tanggal yang disepakati. Jadi pihak pemesan hanya *indent* di awal tahun dan *order*-nya baru akan dikerjakan sesuai dengan tanggal tersebut. Kemudian, penggunaan metode EDD dan FCFS dimaksudkan untuk mengurutkan prioritas penugasan operasi masing-masing *job* pada jadwal di mesin-mesin yang sesuai, bukan berarti untuk mengurutkan pengerjaan *job* di lantai produksi. Jadi, *job* yang mendapat urutan paling awal mendapat prioritas dijadwalkan yang lebih utama dibandingkan dengan *job* urutan setelahnya.

Setiap operasi boleh dikerjakan bersamaan dengan operasi *job* yang lainnya apabila menggunakan mesin (*resource*) yang berbeda. Namun, apabila ada beberapa operasi dimana masing-masing berasal dari *job* berbeda di satu waktu mesin yang sama, maka prioritas penjadwalannya akan diutamakan untuk *job* yang memiliki urutan yang

lebih awal. Hal ini sesuai dengan sudut pandang algoritma *sequencing* bahwa setiap *job* dijadwalkan sesuai dengan urutan prioritasnya dan tidak mempengaruhi waktu siapnya (*ready time*). Ketika semua operasi dari *job* pertama telah terjadwal, operasi dari *job* selanjutnya dapat dijadwalkan dalam rantai produksi.

Job atau *order* yang datang lebih awal di awal tahun digunakan sebagai waktu acuan dimulainya penjadwalan, sehingga waktu mulai perencanaan produksi *job* yang akan dijadwalkan adalah saat $t = 0$ bersesuaian dengan tanggal kedatangan dari *job* pertama, sedangkan *job* selanjutnya memiliki *ready time* sesuai dengan waktu datang kontraknya. Nilai *ready time* dari *job* selain *job* pertama dihitung berdasarkan selisih hari aktif yang ada mulai waktu $t = 0$ hingga waktu kedatangan dari *job* itu sendiri, nilai ini kemudian dikonversikan ke dalam jam kerja perhari. Hasil yang didapat dari pengolahan data menggunakan metode EDD dan FCFS akan dibandingkan dan dipilih yang terbaik sesuai dengan fungsi tujuannya. Dari kedua metode ini bisa dipilih dan ditentukan salah satu atau keduanya sebagai solusi optimal inisiasi awal yang kemudian akan dioptimalkan lebih lanjut menggunakan Algoritma *Tabu search*, dimana dalam setiap iterasi perhitungannya diintegrasikan dengan aturan *Pairwise Interchange*.

4.2.1.1 Model Verbal dari Proses Produksi PT. Barata Indonesia

PT. Barata Indonesia menggunakan sistem produksi *Make to order* yang membuat perusahaan harus selalu menjaga kualitas produk agar sesuai dengan spesifikasi yang diinginkan konsumen serta diselesaikan dalam waktu yang tepat sesuai kontrak. Pelaksanaan proses produksinya harus diawali dengan melakukan perencanaan dan diimbangi dengan pengawasan proses produksi agar tujuan tersebut dapat dicapai. Di dalam dua kegiatan tersebut dibutuhkan suatu cara atau metode yang sesuai agar tujuan dari produksinya dapat dicapai. Pendefinisian awal dari masalah proses produksi yang dilaksanakan oleh perusahaan akan dijelaskan sebagai berikut.

Sistem yang dibahas dalam penjadwalan ini adalah sistem produksi *Make to Order* dengan alur *Flexible Job Shop* yang memproduksi beberapa produk berukuran besar. Masing-masing produk disusun oleh beberapa komponen yang berbeda-beda. Setiap komponen tersebut mempunyai proses identiknya sendiri di beberapa mesin dan membutuhkan waktu produksi yang cukup lama. Alur produksi yang berupa *Flexible Job Shop* menyebabkan peletakan mesin-mesin dikelompokkan berdasarkan prosesnya atau lebih umum disebut sebagai *process layout*.

Lantai produksi dibagi menjadi dua *work centre* yang secara umum dibedakan berdasarkan proses yang bisa dilakukan dalam *work centre* tersebut. *Work centre* pertama yaitu *forming*, merupakan *work centre* untuk membentuk material sesuai dengan spesifikasi produk. Di dalamnya terdapat mesin-mesin yang dipasang *single* dan memiliki karakteristik hanya dapat melakukan satu proses spesifik yang tidak dapat digantikan oleh mesin lain. Sedangkan *work centre* kedua yaitu *assembly*, merupakan *work centre* untuk merakit dan menggabungkan antar material atau antar komponen. Di dalamnya terdapat mesin-mesin yang dipasang paralel dan memiliki fungsi hampir sama untuk setiap jenis mesinnya. Perbedaan dari dua *work centre* ini terletak pada kekhususan proses yang bisa dilakukan dan jumlah mesin dari setiap jenis mesin yang tersedia.

Pada *work centre forming* terdapat 7 jenis mesin dengan fungsi yang berbeda-beda, sedangkan pada *work centre assembly* terdapat 2 jenis mesin dengan fungsi yang berbeda dan dipasang secara paralel masing-masing 5 mesin untuk setiap jenisnya. Sehingga dalam *work centre assembly* terdapat 10 mesin yang dapat dioperasikan bersamaan dalam satu waktu. Jenis-jenis mesin yang terdapat di dua *work centre* ini dinyatakan dalam k dimana $k = 1, 2, 3, \dots, q$. Sedangkan indeks yang menyatakan bahwa mesin jenis k merupakan mesin paralel atau tidak adalah $k(x)$ dimana $x = 1, 2, 3, \dots, m$. Setiap *job* yang dikerjakan di lantai produksi merupakan order yang harus dipenuhi oleh perusahaan, dinyatakan sebagai (i) , dan memiliki waktu *due date*-nya sendiri yang dinyatakan dalam d_i . Masing-masing *job* memiliki urutan proses produksi atau operasi yang dinyatakan sebagai j dimana $j = 1, 2, 3, \dots, p$. *Job* dinyatakan telah selesai dikerjakan ketika nilai j sama dengan p .

Secara keseluruhan, untuk dua produk ini perusahaan memiliki 9 mesin yang digunakan untuk menyelesaikannya dan terbagi ke dalam dua *work centre*. *Work centre* I (*forming*) memiliki 7 mesin dipasang secara *single* dan memiliki fungsi masing-masing yang tidak bisa digantikan oleh mesin yang lain. Alur produksi pada *work centre* ini terjadi satu arah dimana secara umum proses produksi akan diawali dengan menggunakan mesin *Coppier Gas Cutting*, selanjutnya menggunakan mesin *Bevelling*, *Head Dishing*, *Head Flanging*, *Roll and Bend*, *Press Hydraulic* dan yang terakhir adalah mesin *Gas Turning Table*. Urutan tahap proses produksi untuk dua produk ini berbeda, sehingga beberapa mesin yang digunakan pun juga berbeda. Sebaliknya pada *work centre* II (*assembly*), dua produk ini menggunakan mesin yang sama untuk semua prosesnya meskipun dalam urutan tahap yang berbeda. Proses *welding* dan *fit-up* bisa

dilakukan oleh satu mesin yang sama yaitu mesin las. Mesin yang terdapat pada *work centre* ini sebanyak dua yaitu mesin las dan mesin gerinda dimana kedua mesin ini dipasang secara paralel dengan jumlah mesin masing-masingnya sebanyak 5 mesin yang identik.

Waktu *set-up* untuk seluruh proses ini memiliki waktu yang cukup lama, yaitu sekitar 30-45 menit. Namun dalam penelitian ini waktu *set-up* dianggap sudah masuk ke dalam kebutuhan waktu setiap prosesnya karena waktu proses itu sendiri pun memiliki waktu ($t_{i,j,k(x)}$) yang sangat lama yaitu sekitar 1-5 jam. Sehingga dengan waktu *set-up* tersebut dianggap tidak terlalu berpengaruh terhadap penjadwalan produksi yang akan dilakukan.

4.2.1.2 Asumsi

Asumsi yang digunakan selama pembuatan jadwal induk usulan adalah sebagai berikut:

1. Keseluruhan mesin dinyatakan *eligible* adalah saat $t=0$, sesuai dengan tanggal *order job* pertama yang akan dikerjakan dalam sistem.
2. Pada satu titik waktu tertentu, satu mesin hanya dapat melakukan proses untuk satu operasi dari satu *job* induknya. Mesin yang digunakan bisa berupa mesin *single* ataupun mesin paralel.
3. Satu *job* dapat menggunakan beberapa mesin lebih dari 1 kali waktu dan *job* diperbolehkan diproses di dua *work centre* secara bolak balik.
4. Waktu operasi untuk mesin yang paralel dibagi berdasarkan perbandingan antara banyaknya kuantitas *order* dengan jumlah mesin paralel yang tersedia terhadap waktu operasinya.
5. *Ready time* dari masing-masing operasi mengikuti *precedence*-nya.
6. *Set-up time* bersifat *independent*. Hal ini karena belum tentu setiap *order* yang datang memiliki spesifikasi komponen yang berbeda, selain itu waktu *set-up* masih lebih cepat dibandingkan waktu prosesnya sehingga waktu *set-up* dianggap masuk ke dalam waktu proses operasi *job* i .
7. Operasi dapat diproses di mesin paralel ketika mesin tersebut berstatus *eligible*, artinya menganggur selama durasi operasi yang akan ditugaskan.
8. Pembagian waktu operasi di mesin paralel berdasarkan maksimasi dari kuantitas *order* dibagi jumlah mesin yang ada. Sehingga ketika terjadi hasil dengan nilai

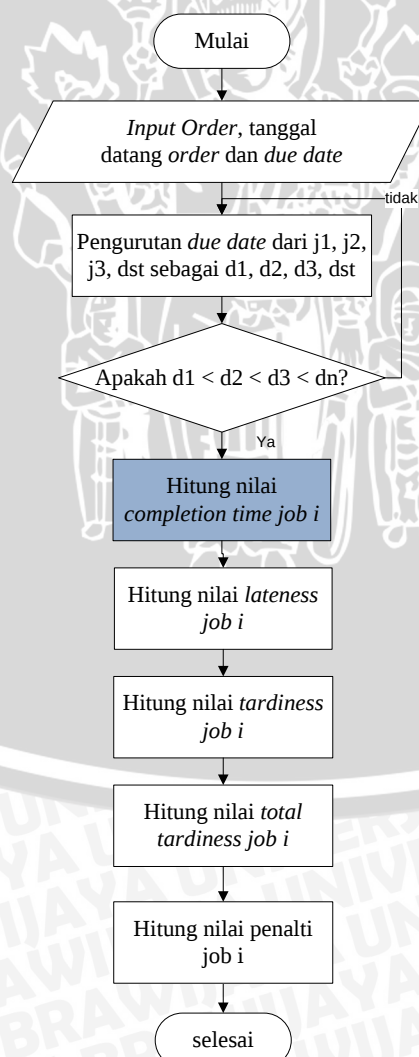
pecahan maka waktu operasi di seluruh mesin k yang parallel mengikuti lama waktu dengan hasil yang terbesar.

9. Setiap mesin memiliki kapasitas yang tidak terbatas bergantung pada spesifikasi komponen yang diproses.
10. Setiap operasi yang sedang di proses di mesin tertentu pada satu waktu tidak boleh diinterupsi atau disisipkan operasi lain (*non-preempt*).

4.2.1.3 Pembentukan dan Pendefinisian Notasi Variabel dan Parameter Model

1. *Earliest Due date* (EDD)

Konsep inti dari metode ini adalah mengurutkan sekumpulan *job* yang akan dikerjakan berdasarkan *dateline* dari waktu pengerjaannya. Jadi, *job* yang mempunyai *dateline* paling awal yang akan dikerjakan pertama di rantai produksi, begitu juga selanjutnya. Gambar 4.13 merupakan diagram alir pengerjaan menggunakan metode EDD.



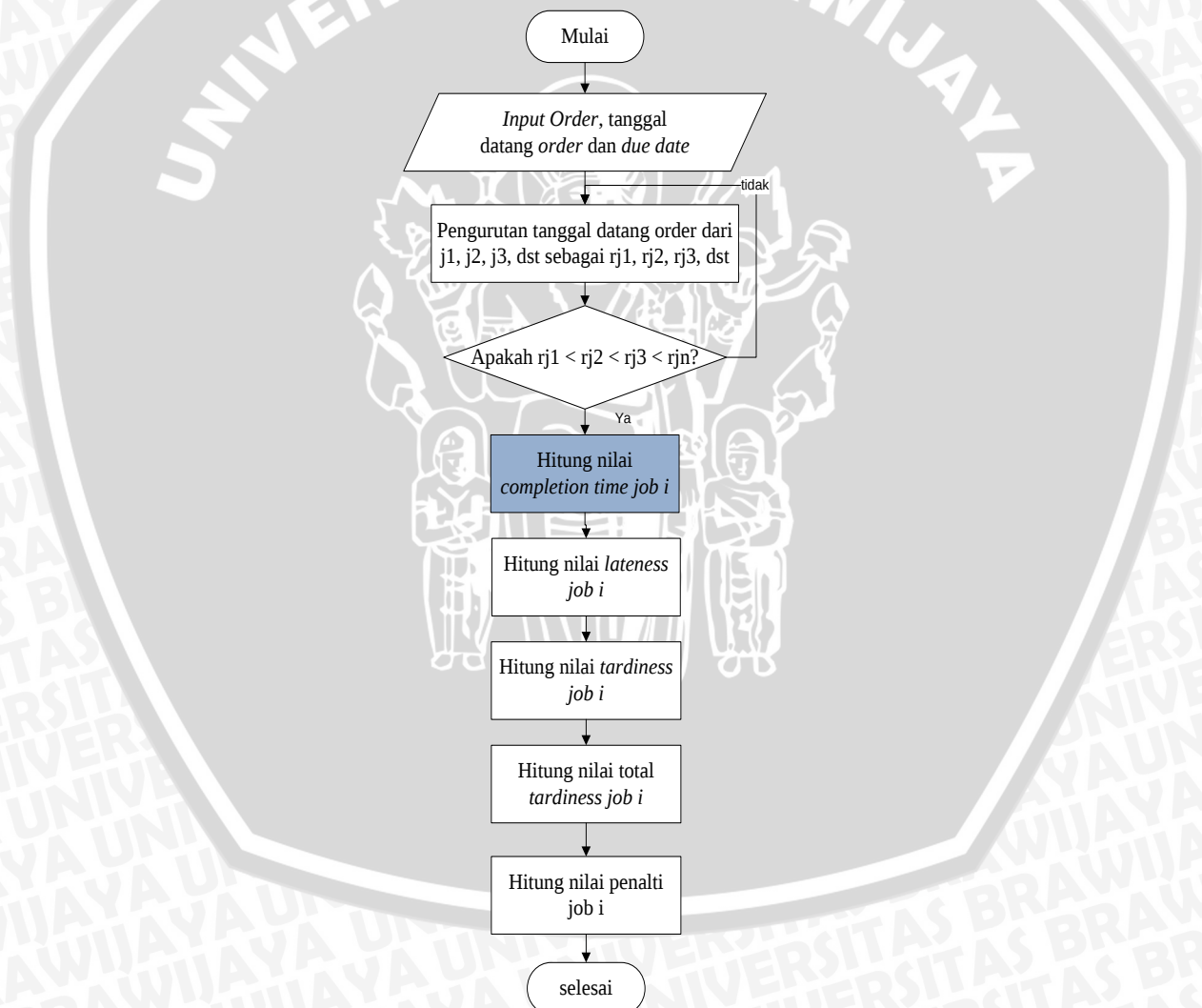
Gambar 4.14 Diagram alir metode *Earliest Due date* (EDD)

Berikut langkah-langkah yang harus dilakukan dalam menerapkan metode EDD:

- Mengurutkan *job* berdasarkan *due date* yang paling awal ($d_1 < d_2 < d_3 < \dots < d_n$) j_r
- Membuat *flowtime* masing-masing *job* dari urutan yang terbentuk
- Menghitung nilai *Lateness* ($L_j = C_j - d_j$)
- Menghitung nilai *Tardiness* ($T = \max(L_j, 0)$)

2. First Come First Serve (FCFS)

Metode ini mengurutkan penjadwalan sekumpulan *job* berdasarkan waktu kedatangan *job* sehingga *job* yang datang paling pertama juga akan diproses pertama kali juga di rantai produksi, begitupun selanjutnya. Gambar 4.14 merupakan diagram alir pengerjaan menggunakan metode FCFS.



Gambar 4.15 Diagram alir metode FCFS

Berikut merupakan langkah-langkah dari metode FCFS :

- Mengurutkan *job* berdasarkan waktu kedatangan yang paling awal ($r_{j1} < r_{j2} < r_{j3} < \dots < r_{jn}$)

- b. Membuat *flowtime* masing-masing *job* dari urutan yang terbentuk
- c. Menghitung nilai *Lateness* ($L_j = C_j - d_j$)
- d. Menghitung nilai *Tardiness* ($T = \max(L_j, 0)$)
3. Notasi dan Model Matematis pada Perhitungan *Completion time*
 - a. Notasi awal yang digunakan sebelum perhitungan adalah sebagai berikut :
 1. i = *job* atau *order*, $i = 1, 2, 3, \dots, o$.
 2. j = urutan proses/operasi dari *job*, $j = 1, 2, 3, \dots, p$.
 3. k = jenis mesin yang tersedia pada satu *workshop*, $k = 1, 2, 3, \dots, q$.
 4. $k(x)$ = jumlah (x) mesin k . $x = 1, 2, 3, \dots, m$.
 $\sum x = 1$, maka mesin k termasuk *single*,
 $\sum x \geq 1$, maka mesin k termasuk paralel.
 5. j' = *precedence* operasi j
 6. $i, j, k(x)$ = keterangan untuk *job* i , operasi ke- j , dan di mesin k .
 7. $r_{i,j,k(x)}$ = waktu siap dari operasi j *job* i untuk bisa diproses di mesin $k(x)$ yang dipengaruhi oleh waktu kedatangan *job* atau waktu selesai operasi sebelumnya dari *job* yang sama.
 8. $S_{i,j,k(x)}$ = waktu dimulainya (*start*) proses j dari *job* i di mesin $k(x)$. *Start job* memperhatikan *completion time* operasi terakhir di mesin yang akan digunakan sekarang dengan *ready time* yang dimiliki oleh operasi yang akan dijadwalkan sekarang.
 Tentukan nilai dari $C_{k(x)}$ di mesin yang akan digunakan sekarang.
 Set $r_{i,j,k(x)}$ sebagai *ready time* dari operasi sekarang.
 Sekarang bandingkan kondisi berikut untuk bisa menjadwalkan :
 - $C_{k(x)} \leq r_{i,j,k(x)}$, maka $S_{i,j,k(x)} = r_{i,j,k(x)}$, $\rightarrow$ mesin menganggur, operasi bisa langsung diproses.
 - $C_{k(x)} > r_{i,j,k(x)}$, maka $S_{i,j,k(x)} = C_{k(x)}$, $\rightarrow$ mesin masih digunakan untuk menyelesaikan operasi j .
 9. $t_{i,j,k(x)}$ = waktu proses mesin $k(x)$ untuk proses j pada *job* i .
 10. $C_{i,j,k(x)}$ = *completion time* operasi j , yaitu waktu selesai proses j *job* i di mesin $k(x)$.
 11. $C_{i,j',k(x)}$ = waktu penyelesaian operasi *precedence*
 12. $C_{k(x)}$ = merupakan waktu acuan bagi operasi/*job* selanjutnya bahwa waktu selesai paling akhir di mesin k adalah pada saat t atau

waktu saat mesin k *eligible*.

13. r_i = waktu dimulainya *job* i , sesuai dengan waktu kedatangan dari kontraknya.
14. C_i = *completion time job* i .
15. D_i = *due date job* i .
16. L_i = *lateness job* i .
17. T_i = *tardiness job* i .
18. P_i = penalti *job* i .

b. Berikut rumus matematis yang digunakan :

1. r_i = Σ hari aktif (tanggal *order job* i - tanggal *job* pertama dikerjakan) x 8 jam
2. $t_{i,j,k(x)}$ = $\max$. integer $\{\frac{qty}{\Sigma x}\}$ x waktu proses operasi tiap produk
3. $C_{i,j,k(x)}$ = $S_{i,j,k(x)} + t_{i,j,k(x)}$
4. C_i = $C_{i,j=p,k(x)} - r_i$
5. L_i = $C_i - D_i$
6. T_i = $\max.\{0; L_i\}$
7. $P_i \rightarrow T_i \leq 30$, maka $P_i = 5\% \times$ nilai kontrak *job* $i \times T_i$
 $\rightarrow T_i > 30$, maka $P_i = 5\% \times$ nilai kontrak *job* i

4. Algoritma Sequencing Job

Untuk menentukan waktu dimulai dan selesai dari *job* yang harus dikerjakan oleh perusahaan, maka digunakan suatu algoritma sebagai tahapan untuk menghitung dan mengetahui *sequencing*-nya. *Sequencing* tersebut selanjutnya akan memudahkan perhitungan *completion time*, total *tardiness* dan kemungkinan besar penalti. Algoritma *sequencing* yang digunakan adalah sebagai berikut :

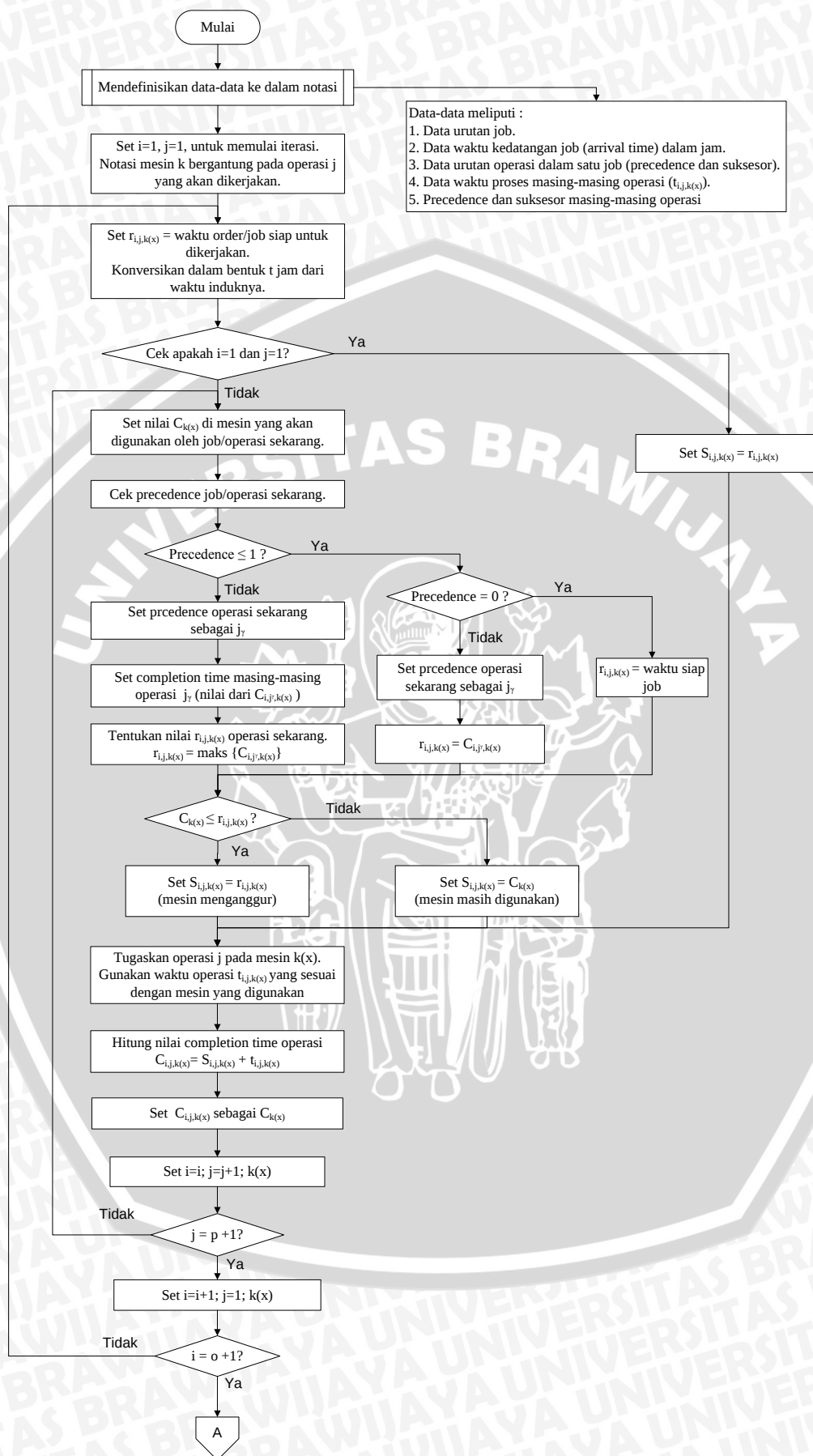
1. Definisikan data-data di lapangan ke dalam notasi pengerjaan awal.
 Data-data tersebut meliputi :
 - a. Data urutan *job*.
 - b. Data waktu kedatangan *job* (*arrival time*) dalam jam.
 - c. Data urutan operasi dalam satu *job* (*precedence* dan suksesor operasi)
 - d. Data waktu proses masing-masing operasi ($t_{i,j,k(x)}$).
 - e. *Precedence* dan suksesor masing-masing operasi
2. Set $i = 1, j = 1$ untuk memulai iterasi. Notasi tersebut dibaca sebagai *job* yang mempunyai urutan pertama (*job* 1) sehingga $i = 1$ dan operasi pertamanya

ditugaskan dalam penjadwalan $j=1$. Notasi dari mesin k bergantung pada operasi j .

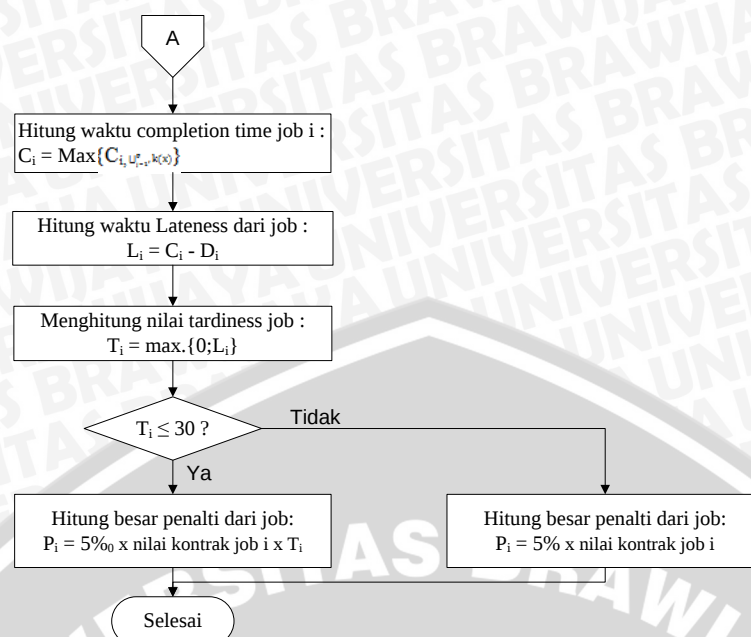
3. Set $r_{i,j,k(x)}$ sebagai waktu datang *order* atau tanggal dimana *order* disetujui sesuai kontraknya. Waktu kedatangan *order* (*arrival time*) ini masih dalam bentuk tanggal dimana selanjutnya akan dikonversikan ke dalam jam sesuai hari aktif yang dihitung mulai dari jadwal induk ini dimulai hingga tanggal datang awal kontraknya.
4. Cek apakah $i = 1$ dan $j = 1$.
 - Jika ya, maka set nilai *ready time* tersebut sebagai waktu dimulainya operasi/*job* dikerjakan sehingga $S_{i,j,k(x)} = r_{i,j,k(x)}$, karena operasi pertama *job* pertama adalah tugas yang dikerjakan pertama. Lanjut ke langkah 12.
 - Jika tidak, maka lanjut ke langkah 5
5. Set $C_{k(x)}$ di mesin yang akan digunakan oleh *job*/operasi sekarang. Lanjut ke langkah 6.
6. Lakukan identifikasi terhadap operasi/*job* pendahulu dari operasi/*job* sekarang (cek *precedence*), catat dan hitung jumlahnya.
7. Cek apakah $precedence \leq 1$?
 - Jika ya maka lanjut ke langkah 10.
 - Jika tidak, maka identifikasi *precedence* operasi sekarang, set sebagai j_γ dimana notasi ini dibaca sebagai operasi pendahulu dari operasi sekarang. Nilai dari γ bergantung pada operasi yang mendahului operasi sekarang. Lanjutkan ke langkah 8.
8. Set nilai *completion time* dari masing-masing operasi *precedence* (j_γ) ke dalam notasi $C_{i,j_\gamma,k(x)}$ berdasarkan data perhitungan sebelumnya.
9. Menentukan nilai $r_{i,j,k(x)}$ dari operasi sekarang.
 $r_{i,j,k(x)} = \max\{C_{i,j_\gamma,k(x)}\}$, nilai *ready time* operasi sekarang berdasarkan pemilihan nilai $C_{i,j_\gamma,k(x)}$ yang terbesar. Lanjut ke langkah 12.
10. Cek apakah *precedence* adalah tidak ada atau sama dengan 0?
 - Jika ya, maka nilai $r_{i,j,k(x)}$ tetap sesuai langkah 3. Lanjut ke langkah 12.
 - Jika tidak, maka set *precedence* operasi sekarang sebagai j_γ . Lanjut ke langkah 11.
11. Menentukan nilai $r_{i,j,k(x)}$ operasi sekarang dimana $r_{i,j,k(x)} = C_{i,j_\gamma,k(x)}$ (tidak perlu dibandingkan dengan beberapa *precedence*). Lanjut ke langkah 12

12. Lakukan cek kondisi mesin dengan syarat berikut,
 - $C_{k(x)} \leq r_{i,j,k(x)}$, maka $S_{i,j,k(x)} = r_{i,j,k(x)}$, $\rightarrow$ mesin menganggur
 - $C_{k(x)} > r_{i,j,k(x)}$, maka $S_{i,j,k(x)} = C_{k(x)}$, $\rightarrow$ mesin masih digunakan
 Lanjutkan ke langkah 13.
13. Tugaskan operasi pada mesin $k(x)$ yang *eligible* $k(x)$ sesuai dengan operasinya. Untuk menugaskan operasi ini gunakan waktu $t_{i,j,k(x)}$. Lanjutkan ke langkah 14.
14. Hitung nilai *completion time* untuk operasi j di mesin k yang digunakan menggunakan rumus $C_{i,j,k(x)} = S_{i,j,k(x)} + t_{i,j,k(x)}$.
15. Set nilai $C_{i,j,k(x)}$ pada langkah 14 sebagai $C_{k(x)}$
 $C_{k(x)}$ merupakan waktu acuan bagi operasi/*job* selanjutnya bahwa waktu selesai paling akhir di mesin k adalah pada saat t /waktu saat mesin k *eligible*.
16. Lanjutkan ke operasi selanjutnya.
 Set $i=i$; $j=j+1$; $k(x)$, notasi ini dibaca *job* yang sama di operasi selanjutnya yang akan dikerjakan di mesin $k(x)$. Notasi j semakin lama semakin besar hingga seluruh operasi yang terdapat pada *job* urutan pertama selesai dikerjakan atau ketika $j = p$.
 Cek apakah $j = p + 1$?
 - Jika ya, maka set $i = i+1$; $j=1$, lanjut ke langkah 17.
 - Jika tidak, ulangi langkah 5.
17. Cek apakah $i = o+1$?
 - Jika ya maka semua *job* telah terjadwalkan dan lanjut ke langkah 18.
 - Jika tidak maka kembali ke langkah 3.
18. Hitung lama waktu penyelesaian masing-masing *job* i dengan menggunakan rumus:
 $C_i = C_{i,j=p,k(x)}$; notasi ini dibaca adalah waktu penyelesaian operasi paling akhir di tiap *job*.
19. Hitung nilai *Lateness* : $L_i = C_i - D_i$
20. Hitung nilai *Tardiness* : $T_i = \max.\{0; L_i\}$.
21. Cek apakah $T_i \leq 30$?
 - Jika ya, maka $P_i = 5\% \times$ nilai kontrak *job* $i \times T_i$
 - Jika tidak, maka $P_i = 5\% \times$ nilai kontrak *job* i

Untuk lebih jelasnya akan dijelaskan dalam *flowchart* pada Gambar 4.15.



Gambar 4.16 Algoritma Sequencing dalam Lingkungan Flexible Job Shop



Gambar 4.16 Algoritma *Sequencing* dalam Lingkungan Fleksibel *Job Shop* (Lanjutan)

5. Algoritma *Tabu search*

Algoritma ini merupakan algoritma yang akan mencari solusi optimal lain diantara berapa solusi tetangga, sehingga nanti hasilnya akan didapatkan satu solusi yang terbaik di iterasi tertentu. Setiap iterasi belum tentu selalu menghasilkan satu solusi yang paling optimal atau bisa saja hasilnya lebih jelek dibandingkan dengan solusi optimal sebelumnya, bergantung pada fungsi tujuan dan fungsi kendalanya. Berikut merupakan langkah-langkah algoritma *Tabu search* (Pinedo, 2009) yang telah diterjemahkan dalam bahasa Indonesia, dan digambarkan pada Gambar 4.16.

Langkah 1

Atur nilai $k=1$

Pilih satu urutan inisiasi S_1 menggunakan metode heuristik tertentu

Atur $S_0 = S_1$

Langkah 2

Pilih satu jadwal kandidat S_c dari solusi tetangga S_k

Jika move dari $S_k \rightarrow S_c$ terlarang oleh mutasi dalam tabu-list

Maka atur $S_{k+1} = S_k$ dan lanjut ke langkah 3

Jika move dari $S_k \rightarrow S_c$ tidak terlarang oleh mutasi manapun dalam tabu-list

Maka atur $S_{k+1} = S_c$ dan

masukkan mutasi kebalikannya dalam tabu-list yang paling atas

Pindah semua masukan dalam tabu-list satu posisi ke bawah dan hapus masukan paling akhir di dalam tabu-list

Jika $G(S_c) < G(S_0)$, atur $S_0 = S_c$;

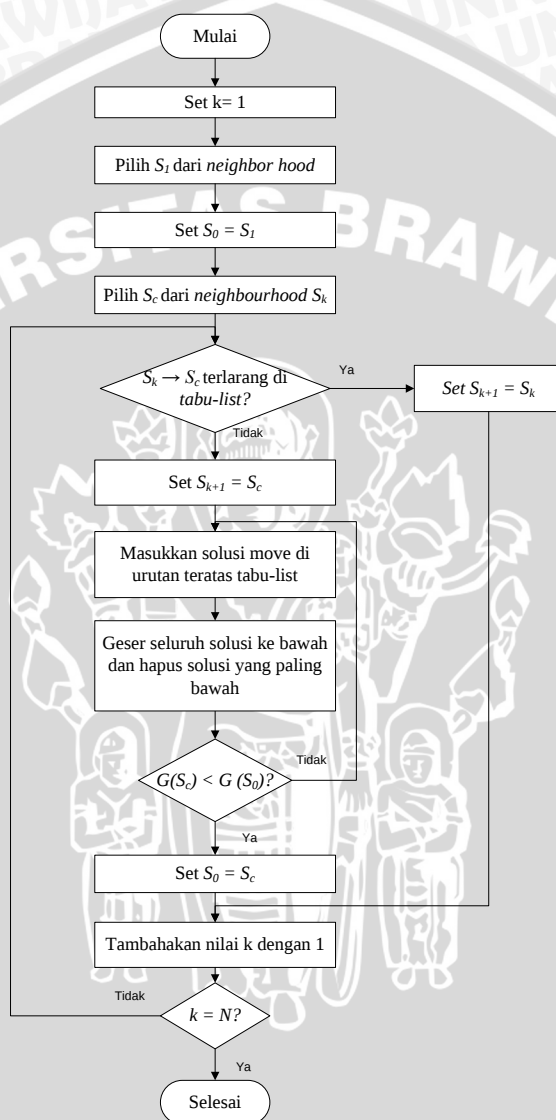
Lanjut ke langkah 3

Langkah 3

Tambahkan nilai k dengan 1

Jika nilai k = N maka BERHENTI

Jika tidak, kembali ke langkah 2



Gambar 4.17 Diagram alir algoritma Tabu Search

4.2.1.4 Fungsi Tujuan dan Fungsi Kendala Model

Berikut merupakan beberapa fungsi tujuan dan kendala yang harus dipenuhi selama proses pencarian solusi dalam penjadwalan ini :

1. Fungsi tujuan

a. Minimasi *tardiness* (T_i), dengan menggunakan rumus pada persamaan (2-8).

$$T_i = \max \{0; L_i\}, \quad (4-1)$$

Persamaan (4-1) menjelaskan bahwa fungsi tujuan tersebut adalah untuk menghitung nilai keterlambatan positif dari masing-masing *job i* (dalam hari).

- b. Minimasi total *tardiness* ($\sum T_i$), dengan menggunakan penyesuaian dari persamaan (2-8), maka diperoleh notasi fungsi tujuan ini sebagai berikut:

$$\sum T_i = \sum_{i=1}^o \max \{0; L_i\} \quad (4-2)$$

Persamaan (4-2) menjelaskan bahwa fungsi tujuan meminimasi total *tardiness* dari seluruh *job* yang ada pada sistem. Dimana *job* yang ada adalah $i = 1, 2, 3, \dots, o$. Dan fungsi dari L_i adalah lama keterlambatan dari masing-masing *job i* dan dijabarkan sebagai $L_i = C_i - D_i$.

- c. Minimasi biaya penalti (P_i)

Penalti yang akan dikenakan bergantung pada seberapa lama *tardiness* yang terjadi. Ada dua kondisi yang digunakan sebagai syarat dalam menentukan banyaknya penalti yang harus dibayar perusahaan kepada pihak kesatu, dijelaskan sebagai pada rumus (4-3) dan (4-4)

$$\rightarrow T_i \leq 30, \text{ maka } P_i = 5\%_0 \times \text{nilai kontrak } job\ i \times T_i \quad (4-3)$$

$$\rightarrow T_i > 30, \text{ maka } P_i = 5\% \times \text{nilai kontrak } job\ i \quad (4-4)$$

2. Fungsi kendala

- a. Fungsi kendala dalam menentukan waktu siap dari *job i*

$$r_i = \Sigma \text{hari aktif tanggal order } job\ i - \text{tanggal } job\ \text{pertama dikerjakan} \times 8 \text{ jam} \quad (4-5)$$

- b. Fungsi kendala dalam menentukan waktu siap operasi *j job i* dalam sistem.

$$r_{i,j,k(x)} = 0, \text{ untuk } job\ \text{urutan pertama operasi pertama} \quad (4-6)$$

$$r_{i,j,k(x)} = C_{k(x)}, \text{ untuk operasi ke-} j\ job\ i \quad (4-7)$$

Persamaan (4-6) dan (4-7) merupakan syarat yang harus dipenuhi sebelum menentukan waktu siap dari operasi *j job i* dalam sistem.

- c. Fungsi kendala dalam menentukan waktu mulai dari *job i* proses *j*

$$S_{i,j,k(x)} = r_{i,j,k(x)}, \text{ ketika } C_{k(x)} \leq r_{i,j,k(x)} \quad (4-8)$$

$$S_{i,j,k(x)} = C_{k(x)}, \text{ ketika } C_{k(x)} > r_{i,j,k(x)} \quad (4-9)$$

Persamaan (4-8) dan (4-9) harus dipenuhi sebelum menentukan waktu mulai dari operasi *j job i* dalam sistem dimana sebelum masuk ke syarat ini nilai dari $C_{k(x)}$ harus diketahui terlebih dahulu. $C_{k(x)}$ merupakan waktu penyelesaian (*completion time*) dari operasi sebelumnya di mesin *k* yang akan digunakan sekarang.

- d. Fungsi kendala waktu penyelesaian proses *j job i* di mesin *k* terhadap waktu penyelesaian akhir *job i*.

$$C_{i,j,k(x)} = S_{i,j,k(x)} + t_{i,j,k(x)} \quad (4-10)$$

$$C_{k(x)} = C_{i,j,k(x)} \quad (4-11)$$

$$C_i = (C_{i,j=p,k(x)}) - r_i \quad (4-12)$$

Persamaan (4-10) merupakan waktu penyelesaian dari proses j *job* i di mesin k pada satu titik waktu tertentu dan nilai ini selanjutnya akan dinyatakan sebagai $C_{k(x)}$ seperti yang dituangkan dalam persamaan (4-11). $C_{k(x)}$ berfungsi sebagai acuan waktu mulai bagi *job* atau operasi selanjutnya yang akan dijadwalkan di mesin k pada satu waktu tersebut. Sedangkan persamaan (4-12) menyatakan nilai dari waktu penyelesaian akhir *job* i yang nilainya didapat dari selisih antara nilai *completion time* operasi terakhir dari satu *job* i dan waktu siap *job* i (r_i).

- e. Fungsi kendala waktu operasi j di mesin k yang paralel
 n = kuantitas *order* (*job*)

$$t_{i,j,k(x)} = \frac{n}{x} \times t_{i,j} \quad (4-13)$$

persamaan (4-13) merupakan durasi waktu operasi j *job* i di mesin k yang dipengaruhi oleh banyaknya mesin yang tersedia. Apabila mesin yang digunakan bersifat paralel, maka durasi waktu dibagi merata berdasarkan kuantitas *order*-nya.

Permasalahan pada model ini merupakan model yang terbilang kompleks namun karena solusi penyelesaiannya dilakukan pendekatan yang mengacu pada sistem aliran produksi *pure job shop*, maka variabel-variabel yang digunakan juga terbilang lebih mudah dipahami. Oleh karena itu, penggunaan metode heuristik diharapkan dapat digunakan sebagai langkah acuan dalam menyelesaikan permasalahan ini dengan mempertimbangkan fungsi tujuan dan fungsi kendala yang ada.

4.3 Pembahasan Hasil

Dalam subbab 4.3 akan dibahas mengenai hasil dari data-data yang telah diolah pada subbab 4.2.

4.3.1 Pengurutan *Job* Menggunakan Metode EDD

Berdasarkan aturan metode EDD yang mengurutkan penjadwalan *job* menurut waktu *due date*-nya ($d_1 < d_2 < d_3 < \dots < d_n$) dimana *job* dengan *due date* terawal adalah *job* yang memiliki urutan pertama untuk diproses di rantai produksi. Pada tabel

4.4 dapat dilihat bahwa *job* 1 memiliki waktu *due date* terawal sehingga di tempatkan pada urutan pertama dengan notasi d_1 , *job* 2 memiliki *due date* diantara *job* 1 dan *job* 3 sehingga ditempatkan pada urutan kedua dan dinotasikan sebagai d_2 , begitu juga *job* 3 yang memiliki waktu *due date* paling akhir maka *job* 3 diurutkan paling akhir dan dinotasikan sebagai d_3 . Dari metode EDD, urutan *job* yang harus dikerjakan pada lantai produksi adalah 1-2-3.

Tabel 4.4 merupakan keterangan waktu *due date* dari masing-masing *job*.

Tabel 4.4 Pengurutan dengan Menggunakan Metode EDD

	Due date	Notasi	Urutan
Job 1 (<i>laddle</i>)	22-Jan-14	d_1	1
Job 2 (<i>kuali timah</i>)	20-Feb-14	d_2	2
Job 3 (<i>laddle</i>)	17-Mar-14	d_3	3

4.3.2 Pengurutan Job Menggunakan Metode FCFS

Berdasarkan aturan metode FCFS yang mengurutkan penjadwalan *job* menurut waktu kedatangannya ($r_1 < r_2 < r_3 < \dots < r_n$) dimana *job* yang memiliki waktu datang terawal diurutkan sebagai urutan pertama dalam proses produksinya. Pada tabel 4.5 dapat dilihat bahwa *job* 1 memiliki waktu kedatangan terawal sehingga di tempatkan pada urutan pertama dengan notasi r_1 , *job* 2 memiliki waktu kedatangan diantara *job* 1 dan *job* 3 sehingga ditempatkan pada urutan kedua dan dinotasikan sebagai r_2 , begitu juga *job* 3 yang memiliki waktu kedatangan paling akhir maka *job* 3 diurutkan paling akhir dan dinotasikan sebagai r_3 . Dari metode FCFS, urutan *job* yang harus dikerjakan pada lantai produksi adalah 1-2-3, atau sama dengan menggunakan metode EDD.

Tabel 4.5 merupakan keterangan waktu kedatangan dari masing-masing *job*.

Tabel 4.5 Pengurutan dengan Menggunakan Metode FCFS

	Waktu Kedatangan	Notasi	Urutan
Job 1 (<i>laddle</i>)	22-Des-13	r_1	1
Job 2 (<i>kuali timah</i>)	06-Jan-14	r_2	2
Job 3 (<i>laddle</i>)	16-Jan-14	r_3	3

4.3.3 Perhitungan Tardiness Menggunakan Algoritma Sequencing Usulan

Pada bab ini akan dijelaskan mengenai perhitungan *sequencing* dengan menggunakan algoritma yang diusulkan. Hal pertama yang dilakukan adalah dengan melakukan identifikasi data-data yang dibutuhkan dalam penjadwalan yang meliputi waktu siap dari masing-masing *job* untuk mulai diproduksi, *precedence* masing-masing operasi, waktu proses masing-masing operasi, serta mesin-mesin yang digunakan.

- Waktu siap masing-masing *job* (waktu kontrak disepakati):

Job 1 $\rightarrow r_{1,1,1(1)} = 0$ (23 Desember 2014);

Job 2 $\rightarrow r_{2,1,1(1)} = 6$ Januari 2014 = (10 hari x 8 jam/hari) + 1 jam = 81;

Job 3 $\rightarrow r_{3,1,1(1)} = 16$ Januari 2014 = (10 hari x 8 jam/hari) + 1 jam = 161

Selanjutnya pada Tabel 4.6 akan dijelaskan predesesor dan suksesor dari masing-masing proses/operasi yang dapat mempengaruhi *ready time* masing-masing operasinya. Sedangkan, pada Tabel 4.7 dijabarkan perhitungan waktu operasi di tiap-tiap mesin yang *eligible* menggunakan rumus (4-13):

Tabel 4.6 Data Predesesor dan Suksesor Masing-masing Operasi

Operasi Laddle	Prece dence	Sukse sor	$r_{i,j,k(x)}$	Operasi Kuali	Preced ence	Suk sesor	$r_{i,j,k(x)}$
1	-	3	$r_{1,1,k(x)}$	1	-	3,4	$r_{1,1,k(x)}$
2	-	4,5,6	$r_{1,2,k(x)}$	2	-	5	$r_{1,2,k(x)}$
3	1	12	$r_{1,3,k(x)}$	3	1	6	$r_{1,3,k(x)}$
4	2	7	$r_{1,4,k(x)}$	4	1	7	$r_{1,4,k(x)}$
5	2	18	$r_{1,5,k(x)}$	5	2	8	$r_{1,5,k(x)}$
6	2	20	$r_{1,6,k(x)}$	6	3	9	$r_{1,6,k(x)}$
7	4	8	$r_{1,7,k(x)}$	7	4	16	$r_{1,7,k(x)}$
8	7	9	$r_{1,8,k(x)}$	8	5	24	$r_{1,8,k(x)}$
9	8	10	$r_{1,9,k(x)}$	9	6	10	$r_{1,9,k(x)}$
10	9	11	$r_{1,10,k(x)}$	10	9	11	$r_{1,10,k(x)}$
11	10	13	$r_{1,11,k(x)}$	11	10	12	$r_{1,11,k(x)}$
12	3	13	$r_{1,12,k(x)}$	12	11	13	$r_{1,12,k(x)}$
13	11,12	14	$r_{1,13,k(x)}, r_{2,13,k(x)}$	13	12	14	$r_{1,13,k(x)}$
14	13	15	$r_{1,14,k(x)}$	14	13	15	$r_{1,14,k(x)}$
15	14	16	$r_{1,15,k(x)}$	15	14	24	$r_{1,15,k(x)}$
16	15	17	$r_{1,16,k(x)}$	16	7	17	$r_{1,16,k(x)}$
17	16	18	$r_{1,17,k(x)}$	17	16	18	$r_{1,17,k(x)}$
18	5,17	19	$r_{1,18,k(x)}, r_{2,18,k(x)}$	18	17	19	$r_{1,18,k(x)}$
19	18	20	$r_{1,19,k(x)}$	19	18	20	$r_{1,19,k(x)}$
20	6,19	21	$r_{1,20,k(x)}, r_{2,20,k(x)}$	20	19	21	$r_{1,20,k(x)}$
21	20	22	$r_{1,21,k(x)}$	21	20	22	$r_{1,21,k(x)}$
22	21	23	$r_{1,22,k(x)}$	22	21	23	$r_{1,22,k(x)}$
23	22	-	$r_{1,23,k(x)}$	23	22	29	$r_{1,23,k(x)}$
				24	8,15	25	$r_{1,24,k(x)}, r_{2,24,k(x)}$
				25	24	26	$r_{1,25,k(x)}$
				26	25	27	$r_{1,26,k(x)}$
				27	26	28	$r_{1,27,k(x)}$
				28	27	29	$r_{1,28,k(x)}$
				29	23,28	30	$r_{1,29,k(x)}, r_{2,29,k(x)}$
				30	29	31	$r_{1,30,k(x)}$
				31	30	32	$r_{1,31,k(x)}$
				32	31	33	$r_{1,32,k(x)}$
				33	32	34	$r_{1,33,k(x)}$
				34	33	-	$r_{1,34,k(x)}$

Rumus (4-13) : $t_{i,j,k(x)} = \frac{n}{x} \times t_{i,j}$; n = kuantitas order.

Tabel 4.7 Data Waktu Proses Masing-masing Operasi

Operasi	Job 1	Job 2	Job 3
1	$t_{2,1,1(1)} = \frac{16 \times 10}{60} = 2,7 \text{ jam}$	$t_{1,1,1(1)} = \frac{60 \times 10}{60} = 10 \text{ jam}$	$t_{2,1,1(1)} = \frac{16 \times 25}{60} = 6,7 \text{ jam}$
2	$t_{2,2,1(1)} = \frac{20 \times 10}{60} = 3,3 \text{ jam}$	$t_{1,2,1(1)} = \frac{16 \times 10}{60} = 2,7 \text{ jam}$	$t_{2,2,1(1)} = \frac{20 \times 25}{60} = 8,3 \text{ jam}$
3	$t_{2,3,9(5)} = \frac{10}{5} \times \frac{12}{60} = 0,4 \text{ jam}$	$t_{1,3,9(5)} = \frac{10}{5} \times \frac{10}{60} = 0,3 \text{ jam}$	$t_{2,3,9(5)} = \frac{25}{5} \times \frac{12}{60} = 1 \text{ jam}$
4	$t_{2,4,9(5)} = \frac{10}{5} \times \frac{10}{60} = 0,3 \text{ jam}$	$t_{1,4,9(5)} = \frac{10}{5} \times \frac{12}{60} = 0,4 \text{ jam}$	$t_{2,4,9(5)} = \frac{25}{5} \times \frac{10}{60} = 0,8 \text{ jam}$
5	$t_{2,5,9(5)} = \frac{10}{5} \times \frac{2}{60} = 0,1 \text{ jam}$	$t_{1,5,9(5)} = \frac{10}{5} \times \frac{13}{60} = 0,4 \text{ jam}$	$t_{2,5,9(5)} = \frac{25}{5} \times \frac{2}{60} = 0,2 \text{ jam}$
6	$t_{2,6,9(5)} = \frac{10}{5} \times \frac{2}{60} = 0,1 \text{ jam}$	$t_{1,6,2(1)} = \frac{10 \times 30}{60} = 5 \text{ jam}$	$t_{2,6,9(5)} = \frac{25}{5} \times \frac{2}{60} = 0,2 \text{ jam}$
7	$t_{2,7,2(1)} = \frac{10 \times 35}{60} = 5,8 \text{ jam}$	$t_{1,7,2(1)} = \frac{10 \times 40}{60} = 6,7 \text{ jam}$	$t_{2,7,2(1)} = \frac{25 \times 40}{60} = 14,6 \text{ jam}$
8	$t_{2,8,6(1)} = \frac{10 \times 120}{60} = 20 \text{ jam}$	$t_{1,8,2(1)} = \frac{10 \times 35}{60} = 5,8 \text{ jam}$	$t_{2,8,6(1)} = \frac{25 \times 120}{60} = 50 \text{ jam}$
9	$t_{2,9,8(5)} = \frac{10}{5} \times \frac{150}{60} = 5 \text{ jam}$	$t_{1,9,5(1)} = \frac{10 \times 240}{60} = 40 \text{ jam}$	$t_{2,9,8(5)} = \frac{25}{5} \times \frac{150}{60} = 12,5 \text{ jam}$
10	$t_{2,10,8(5)} = \frac{10}{5} \times \frac{270}{60} = 9 \text{ jam}$	$t_{1,10,8(5)} = \frac{10}{5} \times \frac{90}{60} = 3 \text{ jam}$	$t_{2,10,8(5)} = \frac{25}{5} \times \frac{270}{60} = 22,5 \text{ jam}$
11	$t_{2,11,9(5)} = 0,6 \text{ jam}$	$t_{1,11,8(5)} = 3,3 \text{ jam}$	$t_{2,11,9(5)} = 1,5 \text{ jam}$
12	$t_{2,12,3(1)} = 40 \text{ jam}$	$t_{1,12,9(5)} = 1,3 \text{ jam}$	$t_{2,12,3(1)} = 100 \text{ jam}$
13	$t_{2,13,8(5)} = 3 \text{ jam}$	$t_{1,13,8(5)} = 3,3 \text{ jam}$	$t_{2,13,8(5)} = 7,5 \text{ jam}$
14	$t_{2,14,8(5)} = 4,5 \text{ jam}$	$t_{1,14,9(5)} = 1,5 \text{ jam}$	$t_{2,14,8(5)} = 11,3 \text{ jam}$
15	$t_{2,15,9(5)} = 1 \text{ jam}$	$t_{1,15,5(1)} = 22,5 \text{ jam}$	$t_{2,15,9(5)} = 2,5 \text{ jam}$
16	$t_{2,16,8(5)} = 4,5 \text{ jam}$	$t_{1,16,8(5)} = 1,5 \text{ jam}$	$t_{2,16,8(5)} = 11,3 \text{ jam}$
17	$t_{2,17,9(5)} = 0,6 \text{ jam}$	$t_{1,17,8(5)} = 3,3 \text{ jam}$	$t_{2,17,9(5)} = 1,5 \text{ jam}$
18	$t_{2,18,8(5)} = 5 \text{ jam}$	$t_{1,18,9(5)} = 1,7 \text{ jam}$	$t_{2,18,8(5)} = 12,5 \text{ jam}$
19	$t_{2,19,8(5)} = 5 \text{ jam}$	$t_{1,19,8(5)} = 4 \text{ jam}$	$t_{2,19,8(5)} = 12,5 \text{ jam}$
20	$t_{2,20,8(5)} = 4 \text{ jam}$	$t_{1,20,9(5)} = 1,5$	$t_{2,20,8(5)} = 10 \text{ jam}$
21	$t_{2,21,8(5)} = 3,3 \text{ jam}$	$t_{1,21,3(1)} = 40 \text{ jam}$	$t_{2,21,8(5)} = 8,3 \text{ jam}$
22	$t_{2,22,9(5)} = 1,4 \text{ jam}$	$t_{1,22,4(1)} = 40 \text{ jam}$	$t_{2,22,9(5)} = 3,5 \text{ jam}$
23	$t_{2,23,0(0)} = 8,3 \text{ jam}$	$t_{1,23,7(1)} = 10 \text{ jam}$	$t_{2,23,0(0)} = 20,8 \text{ jam}$
24		$t_{1,24,8(5)} = 3 \text{ jam}$	
25		$t_{1,25,8(5)} = 5 \text{ jam}$	
26		$t_{1,26,9(5)} = 1,3 \text{ jam}$	
27		$t_{1,27,8(5)} = 5,3 \text{ jam}$	
28		$t_{1,28,9(5)} = 0,7 \text{ jam}$	
29		$t_{1,29,8(5)} = 5 \text{ jam}$	
30		$t_{1,30,8(5)} = 7 \text{ jam}$	
31		$t_{1,31,9(5)} = 1,7 \text{ jam}$	
32		$t_{1,32,8(5)} = 7 \text{ jam}$	
33		$t_{1,33,9(5)} = 3 \text{ jam}$	
34		$t_{1,34,0(0)} = 8,3 \text{ jam}$	

- Urutan *job* berdasarkan aturan EDD adalah *job* 1, *job* 2, dan *job* 3. Maka *job* yang pertama dijadwalkan dalam mesin yang *eligible* adalah *job* 1 sehingga $i = 1$.
- Set $i = 1$ dan $j = 1$; $k(x) = 1(1)$ karena mesin yang digunakan adalah mesin *coppier* (mesin nomor 1) dan banyaknya hanya satu (*single*).
- Set $i = \text{job } 1$ sehingga $i = 1$, $j = 1$. $k = 1$ dan $x = 1$, karena mesin 1 adalah mesin *coppier* (mesin nomor 1) dan banyaknya hanya satu (*single*).
- Set nilai $r_{1,1,1(1)} = \text{waktu siap order/job } 1 = 0$ (23 Desember 2013).
- Apakah $i = 1$ dan $j = 1$? Ya, maka set $S_{1,1,1(1)} = r_{1,1,1(1)} = 0$.
 $C_{1,1,1(1)} = S_{1,1,1(1)} + t_{1,1,1(1)} = 0 + 2,7 \text{ jam} = 2,7 \text{ jam}$
Set $C_{1,1,1(1)} = C_{1(1)} = 2,7 \text{ jam}$
- Set $i = 1$; $j = 1+1 = 2$; $k(x) = 1(1)$
 $j = 24$? Tidak, maka set nilai $C_{1(1)} = 2,7 \text{ jam}$.
Cek *precedence job* sekarang,
Apakah $\text{precedence} \leq 1$? Ya, maka lakukan pengecekan lanjut.
Apakah precedence tidak ada atau sama dengan 0? Ya, maka $r_{1,2,1(1)} = 0$.
Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{1(1)} \leq r_{1,2,1(1)}$? Tidak $\rightarrow 2,7 > 0$,
maka $S_{1,2,1(1)} = C_{1(1)} = 2,7 \text{ jam}$
 $C_{1,2,1(1)} = S_{1,2,1(1)} + t_{1,2,1(1)} = 2,7 \text{ jam} + 3,3 \text{ jam} = 6 \text{ jam}$
Set $C_{1,2,1(1)} = C_{1(1)} = 6 \text{ jam}$
- Set $i = 1$; $j = 2+1 = 3$; $k(x) = 9(5)$
 $j = 24$? Tidak, maka set nilai $C_{9(5)} = 0 \text{ jam}$.
Cek *precedence job* sekarang,
Apakah $\text{precedence} \leq 1$? Ya, maka lakukan pengecekan lanjut.
Apakah precedence tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_y = 1$.
 $r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,3,9(5)} = C_{1,1,1(1)} = 2,7 \text{ jam}$
Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{1,3,9(5)}$? Ya $\rightarrow 0 \leq 2,7$;
maka $S_{1,3,9(5)} = r_{1,3,9(5)} = 2,7 \text{ jam}$
 $C_{1,3,9(5)} = S_{1,3,9(5)} + t_{1,3,9(5)} = 2,7 \text{ jam} + 0,4 \text{ jam} = 3,1 \text{ jam}$
Set $C_{1,3,9(5)} = C_{9(5)} = 3,1 \text{ jam}$.
- Set $i = 1$; $j = 3+1 = 4$; $k(x) = 9(5)$
 $j = 24$? Tidak, maka $C_{9(5)} = 3,1 \text{ jam}$
Cek *precedence job* sekarang,

Apakah $precedence \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah $precedence$ tidak ada atau sama dengan 0? Tidak, maka set $precedence$ operasi sekarang, $j_\gamma = 2$.

$$r_{i,j,k(x)} = C_{i,j^\gamma,k(x)} \rightarrow r_{1,4,9(5)} = C_{1,2,1(1)} = 6 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{1,4,9(5)}$? Ya $\rightarrow 3,1 \leq 6$;

$$\text{maka } S_{1,4,9(5)} = r_{1,4,9(5)} = 6 \text{ jam}$$

$$C_{1,4,9(5)} = S_{1,4,9(5)} + t_{1,4,9(5)} = 6 \text{ jam} + 0,3 \text{ jam} = 6,3 \text{ jam}$$

$$\text{Set } C_{1,4,9(5)} = C_{9(5)} = 6,3 \text{ jam.}$$

- Set $i = 1$; $j = 4+1 = 5$; $k(x) = 9(5)$

$j = 24$? Tidak, maka $C_{9(5)} = 6,3 \text{ jam}$

Cek $precedence$ job sekarang,

Apakah $precedence \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah $precedence$ tidak ada atau sama dengan 0? Tidak, maka set $precedence$ operasi sekarang, $j_\gamma = 2$.

$$r_{i,j,k(x)} = C_{i,j^\gamma,k(x)} \rightarrow r_{1,5,9(5)} = C_{1,2,1(1)} = 6 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{1,5,9(5)}$? Tidak $\rightarrow 6,3 > 6$;

$$\text{maka } S_{1,5,9(5)} = C_{9(5)} = 6,3 \text{ jam}$$

$$C_{1,5,9(5)} = S_{1,5,9(5)} + t_{1,5,9(5)} = 6,3 \text{ jam} + 0,1 \text{ jam} = 6,4 \text{ jam}$$

$$\text{Set } C_{1,5,9(5)} = C_{9(5)} = 6,4 \text{ jam.}$$

- Set $i = 1$; $j = 5+1 = 6$; $k(x) = 9(5)$

$j = 24$? Tidak, maka $C_{9(5)} = 6,4 \text{ jam}$

Cek $precedence$ job sekarang,

Apakah $precedence \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah $precedence$ tidak ada atau sama dengan 0? Tidak, maka set $precedence$ operasi sekarang, $j_\gamma = 2$.

$$r_{i,j,k(x)} = C_{i,j^\gamma,k(x)} \rightarrow r_{1,6,9(5)} = C_{1,2,1(1)} = 6 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{1,6,9(5)}$? Tidak $\rightarrow 6,4 > 6$;

$$\text{maka } S_{1,6,9(5)} = C_{9(5)} = 6,4 \text{ jam}$$

$$C_{1,6,9(5)} = S_{1,6,9(5)} + t_{1,6,9(5)} = 6,4 \text{ jam} + 0,1 \text{ jam} = 6,5 \text{ jam}$$

$$\text{Set } C_{1,6,9(5)} = C_{9(5)} = 6,5 \text{ jam.}$$

- Set $i = 1$; $j = 6+1 = 7$; $k(x) = 2(1)$

$j = 24$? Tidak, maka $C_{2(1)} = 0 \text{ jam}$

Cek $precedence$ job sekarang,

Apakah $precedence \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 4$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,7,2(1)} = C_{1,4,9(5)} = 6,3 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{2(1)} \leq r_{1,7,2(1)}$? Ya $\rightarrow 0 < 6,3$;

$$\text{maka } S_{1,7,2(1)} = r_{1,7,2(1)} = 6,3 \text{ jam}$$

$$C_{1,7,2(1)} = S_{1,7,2(1)} + t_{1,7,2(1)} = 6,3 \text{ jam} + 5,8 \text{ jam} = 12,1 \text{ jam}$$

$$\text{Set } C_{1,7,2(1)} = C_{2(1)} = 12,1 \text{ jam.}$$

- Set $i = 1$; $j = 7+1 = 8$; $k(x) = 6(1)$

$$j = 24? \text{ Tidak, maka } C_{6(1)} = 0 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 7$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,8,6(1)} = C_{1,7,2(1)} = 12,1 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{6(1)} \leq r_{1,8,6(1)}$? Ya $\rightarrow 0 < 12,1$;

$$\text{maka } S_{1,8,6(1)} = r_{1,8,6(1)} = 12,1 \text{ jam}$$

$$C_{1,8,6(1)} = S_{1,8,6(1)} + t_{1,8,6(1)} = 12,1 \text{ jam} + 20 \text{ jam} = 32,1 \text{ jam}$$

$$\text{Set } C_{1,8,6(1)} = C_{6(1)} = 32,1 \text{ jam.}$$

- Set $i = 1$; $j = 8+1 = 9$; $k(x) = 8(5)$

$$j = 24? \text{ Tidak, maka } C_{8(5)} = 0 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 8$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,9,8(5)} = C_{1,8,6(1)} = 32,1 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{1,9,8(5)}$? Ya $\rightarrow 0 < 32,1$;

$$\text{maka } S_{1,9,8(5)} = r_{1,9,8(5)} = 32,1 \text{ jam}$$

$$C_{1,9,8(5)} = S_{1,9,8(5)} + t_{1,9,8(5)} = 32,1 \text{ jam} + 5 \text{ jam} = 37,1 \text{ jam}$$

$$\text{Set } C_{1,9,8(5)} = C_{8(5)} = 37,1 \text{ jam.}$$

- Set $i = 1$; $j = 9+1 = 10$; $k(x) = 8(5)$

$$j = 24? \text{ Tidak, maka } C_{8(5)} = 37,1 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 9$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,10,8(5)} = C_{1,9,8(5)} = 37,1 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{1,10,8(5)}$? Ya $\rightarrow 37,1 = 37,1$;

maka $S_{1,10,8(5)} = r_{1,10,8(5)} = 37,1 \text{ jam}$

$$C_{1,10,8(5)} = S_{1,10,8(5)} + t_{1,10,8(5)} = 37,1 \text{ jam} + 9 \text{ jam} = 46,1 \text{ jam}$$

Set $C_{1,10,8(5)} = C_{8(5)} = 46,1 \text{ jam}$.

- Set $i = 1$; $j = 10+1 = 11$; $k(x) = 9(5)$

$j = 24$? Tidak, maka $C_{9(5)} = 6,5 \text{ jam}$.

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 10$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,11,9(5)} = C_{1,10,8(5)} = 46,1 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{1,11,9(5)}$? Ya $\rightarrow 6,5 < 46,1$;

maka $S_{1,11,9(5)} = r_{1,11,9(5)} = 46,1 \text{ jam}$

$$C_{1,11,9(5)} = S_{1,11,9(5)} + t_{1,11,9(5)} = 46,1 \text{ jam} + 0,6 \text{ jam} = 46,7 \text{ jam}$$

Set $C_{1,11,9(5)} = C_{9(5)} = 46,7 \text{ jam}$.

- Set $i = 1$; $j = 11+1 = 12$; $k(x) = 3(1)$

$j = 24$? Tidak, maka $C_{3(1)} = 0 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 3$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,12,3(1)} = C_{1,3,9(5)} = 3,1 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{3(1)} \leq r_{1,12,3(1)}$? Ya $\rightarrow 0 < 3,1$;

maka $S_{1,12,3(1)} = r_{1,12,3(1)} = 3,1 \text{ jam}$

$$C_{1,12,3(1)} = S_{1,12,3(1)} + t_{1,12,3(1)} = 3,1 \text{ jam} + 40 \text{ jam} = 43,1 \text{ jam}$$

Set $C_{1,12,3(1)} = C_{3(1)} = 43,1 \text{ jam}$.

- Set $i = 1$; $j = 12+1 = 13$; $k(x) = 8(5)$

$j = 24$? Tidak, maka $C_{8(5)} = 46,1 \text{ jam}$

Cek *precedence job* sekarang, apakah *precedence* ≤ 1 ? Tidak, maka

Set *precedence* operasi sekarang, $j_7 = 11;12$.

Bandingkan nilai *completion* dari *precedence job*nya.

$$C_{1,11,9(5)} = 46,7 \text{ jam. } C_{1,12,3(1)} = 43,1 \text{ jam.}$$

$$r_{1,13,8(5)} = \text{maks} \{C_{1,11,9(5)} ; C_{1,12,3(1)}\} = \text{maks.} \{46,7 ; 43,1\} = 46,7 \text{ jam}$$

$$\text{Cek apakah } C_{k(x)} \leq r_{i,j,k(x)}? ; C_{8(5)} \leq r_{1,13,8(5)}? \text{ Ya } \rightarrow 46,1 < 46,7;$$

$$\text{maka } S_{1,13,8(5)} = r_{1,13,8(5)} = 46,7 \text{ jam}$$

$$C_{1,13,8(5)} = S_{1,13,8(5)} + t_{1,13,8(5)} = 46,7 \text{ jam} + 3 \text{ jam} = 49,7 \text{ jam}$$

$$\text{Set } C_{1,13,8(5)} = C_{8(5)} = 49,7 \text{ jam.}$$

- Set $i = 1; j = 13+1 = 14; k(x) = 8(5)$

$$j = 24? \text{ Tidak, maka } C_{8(5)} = 49,7 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah $\text{precedence} \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 13$.

$$r_{i,j,k(x)} = C_{i,j^*,k(x)} \rightarrow r_{1,14,8(5)} = C_{1,13,8(5)} = 49,7 \text{ jam}$$

$$\text{Cek apakah } C_{k(x)} \leq r_{i,j,k(x)}? ; C_{8(5)} \leq r_{1,14,8(5)}? \text{ Ya } \rightarrow 49,7 = 49,7;$$

$$\text{maka } S_{1,14,8(5)} = C_{1,14,8(5)} = 49,7 \text{ jam}$$

$$C_{1,14,8(5)} = S_{1,14,8(5)} + t_{1,14,8(5)} = 49,7 \text{ jam} + 4,5 = 54,2 \text{ jam.}$$

$$\text{Set } C_{1,14,8(5)} = C_{8(5)} = 54,2 \text{ jam.}$$

- Set $i = 1; j = 14+1 = 15; k(x) = 9(5)$

$$j = 24? \text{ Tidak, maka } C_{9(5)} = 46,7 \text{ jam.}$$

Cek *precedence job* sekarang,

Apakah $\text{precedence} \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 14$.

$$r_{i,j,k(x)} = C_{i,j^*,k(x)} \rightarrow r_{1,15,9(5)} = C_{1,14,8(5)} = 54,2 \text{ jam}$$

$$\text{Cek apakah } C_{k(x)} \leq r_{i,j,k(x)}? ; C_{9(5)} \leq r_{1,15,9(5)}? \text{ Ya } \rightarrow 46,7 < 54,2;$$

$$\text{maka } S_{1,15,9(5)} = r_{1,15,9(5)} = 54,2 \text{ jam}$$

$$C_{1,15,9(5)} = S_{1,15,9(5)} + t_{1,15,9(5)} = 54,2 \text{ jam} + 1 \text{ jam} = 55,2 \text{ jam}$$

$$\text{Set } C_{1,15,9(5)} = C_{9(5)} = 55,2 \text{ jam.}$$

- Set $i = 1; j = 15+1 = 16; k(x) = 8(5)$

$$j = 24? \text{ Tidak, maka } C_{8(5)} = 54,2 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah $\text{precedence} \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 15$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,16,8(5)} = C_{1,15,9(5)} = 55,2 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{1,16,8(5)}$? Ya $\rightarrow 54,2 < 55,2$;

maka $S_{1,16,8(5)} = r_{1,16,8(5)} = 55,2 \text{ jam}$

$$C_{1,16,8(5)} = S_{1,16,8(5)} + t_{1,16,8(5)} = 55,2 \text{ jam} + 4,5 \text{ jam} = 59,7 \text{ jam}$$

$$\text{Set } C_{1,16,8(5)} = C_{8(5)} = 59,7 \text{ jam.}$$

- Set $i = 1$; $j = 16+1 = 17$; $k(x) = 9(5)$

$j = 24$? Tidak, maka $C_{9(5)} = 55,2 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_y = 16$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,17,9(5)} = C_{1,16,8(5)} = 59,7 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{1,17,9(5)}$? Ya $\rightarrow 55,2 < 59,7$;

maka $S_{1,17,9(5)} = r_{1,17,9(5)} = 59,7 \text{ jam}$

$$C_{1,17,9(5)} = S_{1,17,9(5)} + t_{1,17,9(5)} = 59,7 \text{ jam} + 0,6 \text{ jam} = 60,3 \text{ jam}$$

$$\text{Set } C_{1,17,9(5)} = C_{9(5)} = 60,3 \text{ jam.}$$

- Set $i = 1$; $j = 17+1 = 18$; $k(x) = 8(5)$

$j = 24$? Tidak, maka $C_{8(5)} = 59,7 \text{ jam}$

Cek *precedence job* sekarang, apakah *precedence* ≤ 1 ? Tidak, maka

Set *precedence* operasi sekarang, $j_y = 5; 17$.

Bandingkan nilai *completion* dari *precedence job*nya.

$$C_{1,5,9(5)} = 6,4 \text{ jam. } C_{1,17,9(5)} = 60,3 \text{ jam.}$$

$$r_{1,18,8(5)} = \text{maks} \{C_{1,5,9(5)} ; C_{1,17,9(5)}\} = \text{maks.} \{6,4 ; 60,3\} = 60,3 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{1,18,8(5)}$? Ya $\rightarrow 59,7 < 60,3$;

maka $S_{1,18,8(5)} = r_{1,18,8(5)} = 60,3 \text{ jam}$

$$C_{1,18,8(5)} = S_{1,18,8(5)} + t_{1,18,8(5)} = 60,3 \text{ jam} + 5 \text{ jam} = 65,3 \text{ jam}$$

$$\text{Set } C_{1,18,8(5)} = C_{8(5)} = 65,3 \text{ jam.}$$

- Set $i = 1$; $j = 18+1 = 19$; $k(x) = 8(5)$

$j = 24$? Tidak, maka $C_{8(5)} = 65,3 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_y = 18$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,19,8(5)} = C_{1,18,8(5)} = 65,3 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{1,19,8(5)}$? Ya $\rightarrow 65,3 = 65,3$;

maka $S_{1,19,8(5)} = r_{1,19,8(5)} = 65,3$ jam

$C_{1,19,8(5)} = S_{1,19,8(5)} + t_{1,19,8(5)} = 65,3 \text{ jam} + 5 \text{ jam} = 70,3 \text{ jam}$

Set $C_{1,19,8(5)} = C_{8(5)} = 70,3 \text{ jam}$

- Set $i = 1$; $j = 19+1 = 20$; $k(x) = 8(5)$

$j = 24$? Tidak, maka $C_{8(5)} = 70,3 \text{ jam}$

Cek *precedence job* sekarang, apakah *precedence* ≤ 1 ? Tidak, maka

Set *precedence* operasi sekarang, $j_\gamma = 6;19$.

Bandingkan nilai *completion* dari *precedence job*nya.

$C_{1,6,9(5)} = 6,5 \text{ jam}$. $C_{1,19,8(5)} = 70,3 \text{ jam}$.

$r_{1,20,8(5)} = \text{maks} \{C_{1,6,9(5)} ; C_{1,19,8(5)}\} = \text{maks} \{6,5 ; 70,3\} = 70,3 \text{ jam}$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{1,20,8(5)}$? Ya $\rightarrow 70,3 = 70,3$;

maka $S_{1,20,8(5)} = r_{1,20,8(5)} = 70,3 \text{ jam}$

$C_{1,20,8(5)} = S_{1,20,8(5)} + t_{1,20,8(5)} = 70,3 \text{ jam} + 4 \text{ jam} = 74,3 \text{ jam}$

Set $C_{1,20,8(5)} = C_{8(5)} = 74,3 \text{ jam}$.

- Set $i = 1$; $j = 20+1 = 21$; $k(x) = 8(5)$

$j = 24$? Tidak, maka $C_{8(5)} = 74,3 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 20$.

$r_{i,j,k(x)} = C_{i,j_\gamma,k(x)} \rightarrow r_{1,21,8(5)} = C_{1,20,8(5)} = 74,3 \text{ jam}$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{1,21,8(5)}$? Ya $\rightarrow 74,3 = 74,3$;

maka $S_{1,21,8(5)} = r_{1,21,8(5)} = 74,3 \text{ jam}$

$C_{1,21,8(5)} = S_{1,21,8(5)} + t_{1,21,8(5)} = 74,3 \text{ jam} + 3,3 \text{ jam} = 77,6 \text{ jam}$

Set $C_{1,21,8(5)} = C_{8(5)} = 77,6 \text{ jam}$.

- Set $i = 1$; $j = 21+1 = 22$; $k(x) = 9(5)$

$j = 24$? Tidak, maka $C_{9(5)} = 60,3 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 21$.

$r_{i,j,k(x)} = C_{i,j_\gamma,k(x)} \rightarrow r_{1,22,9(5)} = C_{1,21,8(5)} = 77,6 \text{ jam}$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{1,22,9(5)}$? Ya $\rightarrow 60,3 < 77,6$;

maka $S_{1,22,9(5)} = r_{1,22,9(5)} = 77,6$ jam

$C_{1,22,9(5)} = S_{1,22,9(5)} + t_{1,22,9(5)} = 77,6 \text{ jam} + 1,4 \text{ jam} = 79 \text{ jam}$

Set $C_{1,22,9(5)} = C_{9(5)} = 79 \text{ jam}$

- Set $i = 1$; $j = 22+1 = 23$; $k(x) = -$

$j = 24$? Tidak, maka $C_{k(x)} = 0 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_y = 22$.

$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{1,23,k(x)} = C_{1,22,9(5)} = 79 \text{ jam}$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{k(x)} \leq r_{1,23,k(x)}$? Ya $\rightarrow 0 < 79$;

maka $S_{1,23,k(x)} = r_{1,23,k(x)} = 79 \text{ jam}$

$C_{1,23,k(x)} = S_{1,23,k(x)} + t_{1,23,k(x)} = 79 \text{ jam} + 8,3 \text{ jam} = 87,3 \text{ jam}$

Set $C_{1,23,k(x)} = C_{k(x)} = \mathbf{87,3 \text{ jam}}$

- Set $i = 1$; $j = 23+1 = 24$; $k(x) = -$

$j = 24$? Ya, maka set $i = i+1$; $j=1$, $k= 1(1)$

Nilai dari j telah sama dengan nilai $p+1$ sehingga seluruh operasi *job* pertama telah dijadwalkan ke dalam mesin yang *eligible*. Sesuai dengan algoritma *sequencing* langkah ke-16 poin pertama, ketika seluruh operasi satu *job* telah dijadwalkan maka *job* dengan urutan prioritas selanjutnya yang akan dijadwalkan, yaitu *job* dengan notasi $i = i+1$. Pada langkah sebelumnya nilai i adalah 1, maka dalam iterasi selanjutnya *job* urutan kedua yang akan dijadwalkan.

- $i = i+1 = 1+1 = 2$; $j=1$; $k(x) = 1(1)$

Apakah $i = 4$? Tidak, maka set nilai r sesuai waktu *job* siap untuk diproses.

$r_{2,1,1(1)} = \text{jam ke-81 (06 Januari 2014)}$.

- Apakah $i = 1$ dan $j = 1$? Tidak, maka set nilai $C_{k(x)}$. $C_{k(x)} \rightarrow C_{1(1)} = 6 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Ya, maka $r_{2,1,1(1)} = 81 \text{ jam}$.

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{1(1)} \leq r_{2,1,1(1)}$? Ya $\rightarrow 6 < 81$;

maka $S_{2,1,1(1)} = r_{2,1,1(1)} = 81 \text{ jam}$

$C_{2,1,1(1)} = S_{2,1,1(1)} + t_{2,1,1(1)} = 81 \text{ jam} + 10 \text{ jam} = 91 \text{ jam}$

Set $C_{2,1,1(1)} = C_{1(1)} = 91 \text{ jam}$

- Set $i = 2$; $j = 1+1 = 2$; $k(x) = 1(1)$
 $j = 35$? Tidak, maka $C_{1(1)} = 91$ jam
 Cek *precedence job* sekarang,
 Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.
 Apakah *precedence* tidak ada atau sama dengan 0? Ya, maka $r_{2,2,1(1)} = 81$ jam.
 Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{1(1)} \leq r_{2,2,1(1)}$? Tidak $\rightarrow 91 > 81$;
 maka $S_{2,2,1(1)} = C_{1(1)} = 91$ jam
 $C_{2,2,1(1)} = S_{2,2,1(1)} + t_{2,2,1(1)} = 91 \text{ jam} + 2,7 \text{ jam} = 93,7 \text{ jam}$
 Set $C_{2,2,1(1)} = C_{1(1)} = 93,7 \text{ jam}$
- Set $i = 2$; $j = 2+1 = 3$; $k(x) = 9(5)$
 $j = 35$? Tidak, maka $C_{9(5)} = 79$ jam
 Cek *precedence job* sekarang,
 Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.
 Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 1$.
 $r_{i,j,k(x)} = C_{i,j^*,k(x)} \rightarrow r_{2,3,9(5)} = C_{2,1,1(1)} = 91 \text{ jam}$
 Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{2,3,9(5)}$? Ya $\rightarrow 79 < 91$;
 maka $S_{2,3,9(5)} = r_{2,3,9(5)} = 91 \text{ jam}$
 $C_{2,3,9(5)} = S_{2,3,9(5)} + t_{2,3,9(5)} = 91 \text{ jam} + 0,3 \text{ jam} = 91,3 \text{ jam}$
 Set $C_{2,3,9(5)} = C_{9(5)} = 91,3 \text{ jam}$
- Set $i = 2$; $j = 3+1 = 4$; $k(x) = 9(5)$
 $j = 35$? Tidak, maka $C_{9(5)} = 91,3 \text{ jam}$
 Cek *precedence job* sekarang,
 Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.
 Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 1$.
 $r_{i,j,k(x)} = C_{i,j^*,k(x)} \rightarrow r_{2,4,9(5)} = C_{2,1,1(1)} = 91 \text{ jam}$
 Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{2,4,9(5)}$? Tidak $\rightarrow 91,3 > 91$;
 maka $S_{2,4,9(5)} = C_{9(5)} = 91,3 \text{ jam}$
 $C_{2,4,9(5)} = S_{2,4,9(5)} + t_{2,4,9(5)} = 91,3 \text{ jam} + 0,4 \text{ jam} = 91,7 \text{ jam}$
 Set $C_{2,4,9(5)} = C_{9(5)} = 91,7 \text{ jam}$
- Set $i = 2$; $j = 4+1 = 5$; $k(x) = 9(5)$
 $j = 35$? Tidak, maka $C_{9(5)} = 91,7 \text{ jam}$

Cek *precedence* job sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 2$.

$$r_{i,j,k(x)} = C_{i,j^7,k(x)} \rightarrow r_{2,5,9(5)} = C_{2,2,1(1)} = 93,7 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{2,5,9(5)}$? Ya $\rightarrow 91,5 < 93,7$;

maka $S_{2,5,9(5)} = r_{2,5,9(5)} = 93,7 \text{ jam}$

$$C_{2,5,9(5)} = S_{2,5,9(5)} + t_{2,5,9(5)} = 93,7 \text{ jam} + 0,4 \text{ jam} = 94,1 \text{ jam}$$

Set $C_{2,5,9(5)} = C_{9(5)} = 94,1 \text{ jam}$

- Set $i = 2$; $j = 5+1 = 6$; $k(x) = 2(1)$
 $j = 35$? Tidak, maka $C_{2(1)} = 7,5 \text{ jam}$

Cek *precedence* job sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 3$.

$$r_{i,j,k(x)} = C_{i,j^7,k(x)} \rightarrow r_{2,6,2(1)} = C_{2,3,9(5)} = 91,3 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{2(1)} \leq r_{2,6,2(1)}$? Ya $\rightarrow 7,5 < 91,3$;

maka $S_{2,6,2(1)} = r_{2,6,2(1)} = 91,3 \text{ jam}$

$$C_{2,6,2(1)} = S_{2,6,2(1)} + t_{2,6,2(1)} = 91,3 \text{ jam} + 5 \text{ jam} = 96,3 \text{ jam}$$

Set $C_{2,6,2(1)} = C_{2(1)} = 96,3 \text{ jam}$

- Set $2 = 1$; $j = 6+1 = 7$; $k(x) = 2(1)$
 $j = 35$? Tidak, maka $C_{2(1)} = 96,3 \text{ jam}$

Cek *precedence* job sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 4$.

$$r_{i,j,k(x)} = C_{i,j^7,k(x)} \rightarrow r_{2,7,2(1)} = C_{2,4,9(5)} = 91,7 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{2(1)} \leq r_{2,7,2(1)}$? Tidak $\rightarrow 96,3 > 91,7$;

maka $S_{2,7,2(1)} = C_{(1)} = 96,3 \text{ jam}$

$$C_{2,7,2(1)} = S_{2,7,2(1)} + t_{2,7,2(1)} = 96,3 \text{ jam} + 6,7 \text{ jam} = 103 \text{ jam}$$

Set $C_{2,7,2(1)} = C_{2(1)} = 103 \text{ jam}$

- Set $i = 2$; $j = 7+1 = 8$; $k(x) = 2(1)$
 $j = 35$? Tidak, maka $C_{2(1)} = 103 \text{ jam}$

Cek *precedence* job sekarang,

Apakah $precedence \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah $precedence$ tidak ada atau sama dengan 0? Tidak, maka set $precedence$ operasi sekarang, $j_7 = 5$.

$$r_{i,j,k(x)} = C_{i,j^7,k(x)} \rightarrow r_{2,8,2(1)} = C_{2,5,9(5)} = 94,1 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{2(1)} \leq r_{2,7,2(1)}$? Tidak $\rightarrow 103 > 94,1$;

$$\text{maka } S_{2,8,2(1)} = C_{2(1)} = 103 \text{ jam}$$

$$C_{2,8,2(1)} = S_{2,8,2(1)} + t_{2,8,2(1)} = 103 \text{ jam} + 5,8 \text{ jam} = 108,8 \text{ jam}$$

$$\text{Set } C_{2,8,2(1)} = C_{2(1)} = 108,8 \text{ jam.}$$

- Set $i = 2$; $j = 8+1 = 9$; $k(x) = 5(1)$

$j = 35$? Tidak, maka $C_{5(1)} = 0 \text{ jam}$

Cek $precedence$ job sekarang,

Apakah $precedence \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah $precedence$ tidak ada atau sama dengan 0? Tidak, maka set $precedence$ operasi sekarang, $j_7 = 6$.

$$r_{i,j,k(x)} = C_{i,j^6,k(x)} \rightarrow r_{2,9,5(1)} = C_{2,6,2(1)} = 96,3 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{2(1)} \leq r_{2,7,2(1)}$? Ya $\rightarrow 0 < 96,3$;

$$\text{maka } S_{2,9,5(1)} = r_{2,9,5(1)} = 96,3 \text{ jam}$$

$$C_{2,9,5(1)} = S_{2,9,5(1)} + t_{2,9,5(1)} = 96,3 \text{ jam} + 40 \text{ jam} = 136,3 \text{ jam}$$

$$\text{Set } C_{2,9,5(1)} = C_{5(1)} = 136,3 \text{ jam}$$

- Set $i = 2$; $j = 9+1 = 10$; $k(x) = 8(5)$

$j = 35$? Tidak, maka $C_{8(5)} = 77,6 \text{ jam}$

Cek $precedence$ job sekarang,

Apakah $precedence \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah $precedence$ tidak ada atau sama dengan 0? Tidak, maka set $precedence$ operasi sekarang, $j_7 = 9$.

$$r_{i,j,k(x)} = C_{i,j^9,k(x)} \rightarrow r_{2,10,8(5)} = C_{2,9,5(1)} = 136,3 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{2(1)} \leq r_{2,7,2(1)}$? Ya $\rightarrow 77,6 < 136,3$;

$$\text{maka } S_{2,10,8(5)} = r_{2,10,8(5)} = 136,3 \text{ jam}$$

$$C_{2,10,8(5)} = S_{2,10,8(5)} + t_{2,10,8(5)} = 136,3 \text{ jam} + 3 \text{ jam} = 139,3 \text{ jam}$$

$$\text{Set } C_{2,10,8(5)} = C_{8(5)} = 139,3 \text{ jam.}$$

- Set $i = 2$; $j = 10+1 = 11$; $k(x) = 8(5)$

$j = 35$? Tidak, maka $C_{8(5)} = 139,3 \text{ jam}$

Cek $precedence$ job sekarang,

Apakah $precedence \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 10$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,7,2(1)} = C_{2,4,9(5)} = 139,3 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{2(1)} \leq r_{2,7,2(1)}$? Ya $\rightarrow 139,3 = 139,3$;

$$\text{maka } S_{2,11,8(5)} = r_{2,11,8(5)} = 139,3 \text{ jam}$$

$$C_{2,11,8(5)} = S_{2,11,8(5)} + t_{2,11,8(5)} = 139,3 \text{ jam} + 3,3 \text{ jam} = 142,6 \text{ jam}$$

$$\text{Set } C_{2,11,8(5)} = C_{8(5)} = 142,6 \text{ jam.}$$

- Set $i = 2$; $j = 11+1 = 12$; $k(x) = 9(5)$

$$j = 35? \text{ Tidak, maka } C_{9(5)} = 94,1 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 11$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,12,9(5)} = C_{2,11,8(5)} = 142,6 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{2,12,9(5)}$? Ya $\rightarrow 94,1 < 142,6$;

$$\text{maka } S_{2,12,9(5)} = r_{2,12,9(5)} = 142,6 \text{ jam}$$

$$C_{2,12,9(5)} = S_{2,12,9(5)} + t_{2,12,9(5)} = 142,6 \text{ jam} + 1,3 \text{ jam} = 143,9 \text{ jam}$$

$$\text{Set } C_{2,12,9(5)} = C_{9(5)} = 143,9 \text{ jam.}$$

- Set $i = 2$; $j = 12+1 = 13$; $k(x) = 8(5)$

$$j = 35? \text{ Tidak, maka } C_{8(5)} = 142,6 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 12$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,13,8(5)} = C_{2,12,9(5)} = 143,9 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{2,13,8(5)}$? Ya $\rightarrow 142,6 < 143,9$;

$$\text{maka } S_{2,13,8(5)} = r_{2,13,8(5)} = 143,9 \text{ jam}$$

$$C_{2,13,8(5)} = S_{2,13,8(5)} + t_{2,13,8(5)} = 143,9 \text{ jam} + 3,3 \text{ jam} = 147,2 \text{ jam}$$

$$\text{Set } C_{2,13,8(5)} = C_{8(5)} = 147,2 \text{ jam.}$$

- Set $i = 2$; $j = 13+1 = 14$; $k(x) = 9(5)$

$$j = 35? \text{ Tidak, maka } C_{9(5)} = 143,9 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 13$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,14,9(5)} = C_{2,13,8(5)} = 147,2 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{2,14,9(5)}$? Ya $\rightarrow 143,9 < 147,2$;

$$\text{maka } S_{2,14,9(5)} = r_{2,14,9(5)} = 147,2 \text{ jam}$$

$$C_{2,14,9(5)} = S_{2,14,9(5)} + t_{2,14,9(5)} = 147,2 \text{ jam} + 1,5 \text{ jam} = 148,7 \text{ jam}$$

$$\text{Set } C_{2,14,9(5)} = C_{9(5)} = 148,7 \text{ jam.}$$

- Set $i = 2$; $j = 14+1 = 15$; $k(x) = 5(1)$
 $j = 35$? Tidak, maka $C_{5(1)} = 136,3 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 14$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,15,5(1)} = C_{2,14,9(5)} = 148,7 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{5(1)} \leq r_{2,15,5(1)}$? Ya $\rightarrow 136,3 < 148,7$;

$$\text{maka } S_{2,15,5(1)} = r_{2,15,5(1)} = 148,7 \text{ jam}$$

$$C_{2,15,5(1)} = S_{2,15,5(1)} + t_{2,15,5(1)} = 148,7 \text{ jam} + 22,5 \text{ jam} = 171,2 \text{ jam}$$

$$\text{Set } C_{2,15,5(1)} = C_{5(1)} = 171,2 \text{ jam.}$$

- Set $i = 2$; $j = 15+1 = 16$; $k(x) = 8(5)$
 $j = 35$? Tidak, maka $C_{8(5)} = 77,7 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 7$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,16,8(5)} = C_{2,7,2(1)} = 103 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{2,16,8(5)}$? Tidak $\rightarrow 77,7 < 103$;

$$\text{maka } S_{2,16,8(5)} = r_{2,16,8(5)} = 103 \text{ jam}$$

$$C_{2,16,8(5)} = S_{2,16,8(5)} + t_{2,16,8(5)} = 103 \text{ jam} + 1,5 \text{ jam} = 104,5 \text{ jam}$$

$$\text{Set } C_{2,16,8(5)} = C_{8(5)} = 104,5 \text{ jam.}$$

- Set $i = 2$; $j = 16+1 = 17$; $k(x) = 8(5)$
 $j = 35$? Tidak, maka $C_{8(5)} = 104,5 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 16$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,17,8(5)} = C_{2,16,8(5)} = 104,5 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{2,17,8(5)}$? Ya $\rightarrow 104,5 = 104,5$;

maka $S_{2,17,8(5)} = r_{2,17,8(5)} = 104,5 \text{ jam}$

$$C_{2,17,8(5)} = S_{2,17,8(5)} + t_{2,17,8(5)} = 104,5 \text{ jam} + 3,3 \text{ jam} = 107,8 \text{ jam}$$

Set $C_{2,17,8(5)} = C_{8(5)} = 107,8 \text{ jam}$.

- Set $i = 2$; $j = 17+1 = 18$; $k(x) = 9(5)$

$j = 35$? Tidak, maka $C_{9(5)} = 94,1 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 17$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,18,9(5)} = C_{2,17,8(5)} = 107,8 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{2,18,9(5)}$? Ya $\rightarrow 94,1 < 107,8$;

maka $S_{2,18,9(5)} = r_{2,18,9(5)} = 107,8 \text{ jam}$

$$C_{2,18,9(5)} = S_{2,18,9(5)} + t_{2,18,9(5)} = 107,8 \text{ jam} + 1,7 \text{ jam} = 109,5 \text{ jam}$$

Set $C_{2,18,9(5)} = C_{9(5)} = 109,5 \text{ jam}$.

- Set $i = 2$; $j = 18+1 = 19$; $k(x) = 8(5)$

$j = 35$? Tidak, maka $C_{8(5)} = 107,8 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_7 = 18$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,19,8(5)} = C_{2,18,9(5)} = 109,5 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{2,19,8(5)}$? Ya $\rightarrow 107,8 < 109,5$;

maka $S_{2,19,8(5)} = r_{2,19,8(5)} = 109,5 \text{ jam}$

$$C_{2,19,8(5)} = S_{2,19,8(5)} + t_{2,19,8(5)} = 109,5 \text{ jam} + 4 \text{ jam} = 113,5 \text{ jam}$$

Set $C_{2,19,8(5)} = C_{8(5)} = 113,5 \text{ jam}$.

- Set $i = 2$; $j = 19+1 = 20$; $k(x) = 9(5)$

$j = 35$? Tidak, maka $C_{9(5)} = 109,5 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 19$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,20,9(5)} = C_{2,19,8(5)} = 113,5 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{2,20,9(5)}$? Ya $\rightarrow 109,5 < 113,5$;

$$\text{maka } S_{2,20,9(5)} = r_{2,20,9(5)} = 113,5 \text{ jam}$$

$$C_{2,20,9(5)} = S_{2,20,9(5)} + t_{2,20,9(5)} = 113,5 \text{ jam} + 1,5 \text{ jam} = 115 \text{ jam}$$

$$\text{Set } C_{2,20,9(5)} = C_{9(5)} = 115 \text{ jam.}$$

- Set $i = 2$; $j = 20+1 = 21$; $k(x) = 3(1)$

$$j = 35? \text{ Tidak, maka } C_{3(1)} = 43,1 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 20$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,21,3(1)} = C_{2,20,9(5)} = 115 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{3(1)} \leq r_{2,21,3(1)}$? Ya $\rightarrow 43,1 < 115$;

$$\text{maka } S_{2,21,3(1)} = r_{2,21,3(1)} = 115 \text{ jam}$$

$$C_{2,21,3(1)} = S_{2,21,3(1)} + t_{2,21,3(1)} = 115 \text{ jam} + 40 \text{ jam} = 155 \text{ jam}$$

$$\text{Set } C_{2,21,3(1)} = C_{3(1)} = 155 \text{ jam.}$$

- Set $i = 2$; $j = 21+1 = 22$; $k(x) = 4(1)$

$$j = 35? \text{ Tidak, maka } C_{4(1)} = 0 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 21$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,22,4(1)} = C_{2,21,3(1)} = 155 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{4(1)} \leq r_{2,22,4(1)}$? Ya $\rightarrow 0 < 155$;

$$\text{maka } S_{2,22,4(1)} = r_{2,22,4(1)} = 155 \text{ jam}$$

$$C_{2,22,4(1)} = S_{2,22,4(1)} + t_{2,22,4(1)} = 155 \text{ jam} + 40 \text{ jam} = 195 \text{ jam}$$

$$\text{Set } C_{2,22,4(1)} = C_{4(1)} = 195 \text{ jam.}$$

- Set $i = 2$; $j = 22+1 = 23$; $k(x) = 7(1)$

$$j = 35? \text{ Tidak, maka } C_{7(1)} = 0 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_y = 22$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,23,7(1)} = C_{2,22,4(1)} = 195 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{7(1)} \leq r_{2,23,7(1)}$? Ya $\rightarrow 0 < 195$;

$$\text{maka } S_{2,23,7(1)} = r_{2,23,7(1)} = 195 \text{ jam}$$

$$C_{2,23,7(1)} = S_{2,23,7(1)} + t_{2,23,7(1)} = 195 \text{ jam} + 10 \text{ jam} = 205 \text{ jam}$$

$$\text{Set } C_{2,23,7(1)} = C_{7(1)} = 205 \text{ jam.}$$

- Set $i = 2$; $j = 23+1 = 24$; $k(x) = 8(5)$

$$j = 35? \text{ Tidak, maka } C_{8(5)} = 147,3 \text{ jam}$$

Cek *precedence job* sekarang, apakah *precedence* ≤ 1 ? Tidak, maka

Set *precedence* operasi sekarang, $j_y = 8;15$.

Bandingkan nilai *completion* dari *precedence job*nya.

$$C_{2,8,2(1)} = 108,8 \text{ jam. } C_{2,15,5(1)} = 171,2 \text{ jam.}$$

$$r_{2,24,8(5)} = \text{maks} \{C_{2,8,2(1)} ; C_{2,15,5(1)}\} = \text{maks.} \{108,8 ; 171,2\} = 171,2 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{2,24,8(5)}$? Ya $\rightarrow 147,3 < 171,2$;

$$\text{maka } S_{2,24,8(5)} = r_{2,24,8(5)} = 171,2 \text{ jam}$$

$$C_{2,24,8(5)} = S_{2,24,8(5)} + t_{2,24,8(5)} = 171,2 \text{ jam} + 3 \text{ jam} = 174,2 \text{ jam}$$

$$\text{Set } C_{2,24,8(5)} = C_{8(5)} = 174,2 \text{ jam.}$$

- Set $i = 2$; $j = 24+1 = 25$; $k(x) = 8(5)$

$$j = 35? \text{ Tidak, maka } C_{8(5)} = 174,2 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_y = 24$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,25,8(5)} = C_{2,24,8(5)} = 174,2 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{2,25,8(5)}$? Ya $\rightarrow 174,2 = 174,2$;

$$\text{maka } S_{2,25,8(5)} = r_{2,25,8(5)} = 174,2 \text{ jam}$$

$$C_{2,25,8(5)} = S_{2,25,8(5)} + t_{2,25,8(5)} = 174,2 \text{ jam} + 5 \text{ jam} = 179,2 \text{ jam}$$

$$\text{Set } C_{2,25,8(5)} = C_{8(5)} = 179,2 \text{ jam.}$$

- Set $i = 2$; $j = 25+1 = 26$; $k(x) = 9(5)$

$$j = 35? \text{ Tidak, maka } C_{9(5)} = 115 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_r = 25$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,26,9(5)} = C_{2,25,8(5)} = 179,2 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{2,26,9(5)}$? Ya $\rightarrow 115 < 179,2$;

$$\text{maka } S_{2,26,9(5)} = r_{2,26,9(5)} = 179,2 \text{ jam}$$

$$C_{2,26,9(5)} = S_{2,26,9(5)} + t_{2,26,9(5)} = 179,2 \text{ jam} + 1,3 \text{ jam} = 180,5 \text{ jam}$$

$$\text{Set } C_{2,26,9(5)} = C_{9(5)} = 180,5 \text{ jam.}$$

- Set $i = 2$; $j = 26+1 = 27$; $k(x) = 8(5)$
 $j = 35$? Tidak, maka $C_{8(5)} = 179,2 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_r = 26$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,27,8(5)} = C_{2,26,9(5)} = 180,5 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{2,27,8(5)}$? Ya $\rightarrow 179,2 < 180,5$;

$$\text{maka } S_{2,27,8(5)} = r_{2,27,8(5)} = 180,5 \text{ jam}$$

$$C_{2,27,8(5)} = S_{2,27,8(5)} + t_{2,27,8(5)} = 180,5 \text{ jam} + 5,3 \text{ jam} = 185,8 \text{ jam}$$

$$\text{Set } C_{2,27,8(5)} = C_{8(5)} = 185,8 \text{ jam.}$$

- Set $i = 2$; $j = 27+1 = 28$; $k(x) = 9(5)$
 $j = 35$? Tidak, maka $C_{9(5)} = 180,5 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_r = 27$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,28,9(5)} = C_{2,27,8(5)} = 185,8 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{2,28,9(5)}$? Ya $\rightarrow 180,5 < 185,8$;

$$\text{maka } S_{2,28,9(5)} = r_{2,28,9(5)} = 185,8 \text{ jam}$$

$$C_{2,28,9(5)} = S_{2,28,9(5)} + t_{2,28,9(5)} = 185,8 \text{ jam} + 0,7 \text{ jam} = 186,5 \text{ jam}$$

$$\text{Set } C_{2,28,9(5)} = C_{9(5)} = 186,5 \text{ jam.}$$

- Set $i = 2$; $j = 28+1 = 29$; $k(x) = 8(5)$
 $j = 35$? Tidak, maka $C_{8(5)} = 185,8 \text{ jam}$

Cek *precedence job* sekarang, apakah *precedence* ≤ 1 ? Tidak, maka

Set *precedence* operasi sekarang, $j_r = 23; 28$.

Bandingkan nilai *completion* dari *precedence job*nya.

$$C_{2,23,7(1)} = 205 \text{ jam. } C_{2,28,9(5)} = 186,5 \text{ jam.}$$

$$r_{2,29,8(5)} = \text{maks} \{C_{2,23,7(1)} ; C_{2,28,8(5)}\} = \text{maks.} \{205 ; 186,5\} = 205 \text{ jam}$$

$$\text{Cek apakah } C_{k(x)} \leq r_{i,j,k(x)}? ; C_{8(5)} \leq r_{2,29,8(5)}? \text{ Ya } \rightarrow 185,8 < 205;$$

$$\text{Maka } S_{2,29,8(5)} = r_{2,29,8(5)} = 205 \text{ jam}$$

$$C_{2,29,8(5)} = S_{2,29,8(5)} + t_{2,29,8(5)} = 205 \text{ jam} + 5 \text{ jam} = 210 \text{ jam}$$

$$\text{Set } C_{2,29,8(5)} = C_{8(5)} = 210 \text{ jam.}$$

- Set $i = 2$; $j = 29+1 = 30$; $k(x) = 8(5)$

$$j = 35? \text{ Tidak, maka } C_{8(5)} = 210 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah $\text{precedence} \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 29$.

$$r_{i,j,k(x)} = C_{i,j^*,k(x)} \rightarrow r_{2,30,8(5)} = C_{2,29,8(5)} = 210 \text{ jam}$$

$$\text{Cek apakah } C_{k(x)} \leq r_{i,j,k(x)}? ; C_{8(5)} \leq r_{2,30,8(5)}? \text{ Ya } \rightarrow 210 = 210;$$

$$\text{maka } S_{2,30,8(5)} = r_{2,30,8(5)} = 210 \text{ jam}$$

$$C_{2,30,8(5)} = S_{2,30,8(5)} + t_{2,30,8(5)} = 210 \text{ jam} + 7 \text{ jam} = 217 \text{ jam}$$

$$\text{Set } C_{2,30,8(5)} = C_{8(5)} = 217 \text{ jam.}$$

- Set $i = 2$; $j = 30+1 = 31$; $k(x) = 9(5)$

$$j = 35? \text{ Tidak, maka } C_{9(5)} = 186,5 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah $\text{precedence} \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 30$.

$$r_{i,j,k(x)} = C_{i,j^*,k(x)} \rightarrow r_{2,31,9(5)} = C_{2,30,8(5)} = 217 \text{ jam}$$

$$\text{Cek apakah } C_{k(x)} \leq r_{i,j,k(x)}? ; C_{9(5)} \leq r_{2,31,9(5)}? \text{ Ya } \rightarrow 186,5 < 217;$$

$$\text{maka } S_{2,31,9(5)} = r_{2,31,9(5)} = 217 \text{ jam}$$

$$C_{2,31,9(5)} = S_{2,31,9(5)} + t_{2,31,9(5)} = 217 \text{ jam} + 1,7 \text{ jam} = 218,7 \text{ jam}$$

$$\text{Set } C_{2,31,9(5)} = C_{9(5)} = 218,7 \text{ jam.}$$

- Set $i = 2$; $j = 31+1 = 32$; $k(x) = 8(5)$

$$j = 35? \text{ Tidak, maka } C_{8(5)} = 217 \text{ jam}$$

Cek *precedence job* sekarang,

Apakah $\text{precedence} \leq 1$? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 31$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,32,8(5)} = C_{2,31,9(5)} = 218,7 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{8(5)} \leq r_{2,32,8(5)}$? Ya $\rightarrow 217 < 218,7$;

maka $S_{2,32,8(5)} = r_{2,32,8(5)} = 218,7 \text{ jam}$

$$C_{2,32,8(5)} = S_{2,32,8(5)} + t_{2,32,8(5)} = 218,7 \text{ jam} + 7 \text{ jam} = 225,7 \text{ jam}$$

Set $C_{2,32,8(5)} = C_{8(5)} = 225,7 \text{ jam}$.

- Set $i = 2$; $j = 32+1 = 33$; $k(x) = 9(5)$
 $j = 35$? Tidak, maka $C_{9(5)} = 218,7 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 32$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,32,9(5)} = C_{2,32,8(5)} = 225,7 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{9(5)} \leq r_{2,32,9(5)}$? Ya $\rightarrow 218,7 < 225,7$;

maka $S_{2,32,9(5)} = r_{2,32,9(5)} = 225,7 \text{ jam}$

$$C_{2,32,9(5)} = S_{2,32,9(5)} + t_{2,32,9(5)} = 225,7 \text{ jam} + 3 \text{ jam} = 228,7 \text{ jam}$$

Set $C_{2,32,9(5)} = C_{9(5)} = 228,7 \text{ jam}$.

- Set $i = 2$; $j = 33+1 = 34$; $k(x) = -$
 $j = 35$? Tidak, maka $C_{k(x)} = 87,3 \text{ jam}$

Cek *precedence job* sekarang,

Apakah *precedence* ≤ 1 ? Ya, maka lakukan pengecekan lanjut.

Apakah *precedence* tidak ada atau sama dengan 0? Tidak, maka set *precedence* operasi sekarang, $j_\gamma = 33$.

$$r_{i,j,k(x)} = C_{i,j,k(x)} \rightarrow r_{2,34,k(x)} = C_{2,33,9(5)} = 228,7 \text{ jam}$$

Cek apakah $C_{k(x)} \leq r_{i,j,k(x)}$? ; $C_{k(x)} \leq r_{2,34,k(x)}$? Ya $\rightarrow 87,5 < 228,7$;

maka $S_{2,34,k(x)} = r_{2,34,k(x)} = 228,7 \text{ jam}$

$$C_{2,34,k(x)} = S_{2,34,k(x)} + t_{2,34,k(x)} = 228,7 \text{ jam} + 8,3 \text{ jam} = 237 \text{ jam}$$

Set $C_{2,34,k(x)} = C_{k(x)} = 237 \text{ jam}$.

- Set $i = 2$; $j = 34+1 = 35$; $k(x) = -$
 $j = 35$? Ya, maka set $i = i+1$; $j=1$, $k= 1(1)$

Nilai dari j telah sama dengan nilai $p+1$ sehingga seluruh operasi *job* kedua telah dijadwalkan ke dalam mesin yang *eligible*. Sesuai dengan algoritma *sequencing* langkah ke-16 poin pertama, ketika seluruh operasi satu *job* telah dijadwalkan maka *job* dengan urutan prioritas selanjutnya yang akan dijadwalkan, yaitu *job* dengan notasi $i =$

$i+1$. Pada langkah sebelumnya nilai i adalah 2, maka dalam iterasi selanjutnya *job* urutan ketiga (*laddle*) yang akan dijadwalkan.

Tabel 4.8 Rekap Perhitungan Penjadwalan Tiap Operasi (dalam Jam)

Ope rasi	Job 1			Job 2			Job 3		
	$S_{i,j,k(x)}$	$t_{i,j,k(x)}$ (jam)	$C_{i,j,k(x)}$	$S_{i,j,k(x)}$	$t_{i,j,k(x)}$ (jam)	$C_{i,j,k(x)}$	$S_{i,j,k(x)}$	$t_{i,j,k(x)}$ (jam)	$C_{i,j,k(x)}$
1	0,0	2,7	2,7	81,0	10,0	91,0	161,0	6,7	167,7
2	2,7	3,3	6,0	91,0	2,7	93,7	167,7	8,3	176,0
3	2,7	0,4	3,1	91,0	0,3	91,3	167,7	1,0	168,7
4	6,0	0,3	6,3	91,3	0,4	91,7	176,0	0,8	176,8
5	6,3	0,1	6,4	93,7	0,4	94,1	176,8	0,2	177,0
6	6,4	0,1	6,5	91,3	5,0	96,3	177,0	0,2	177,2
7	6,3	5,8	12,2	96,3	6,7	103,0	176,8	14,6	191,4
8	12,2	20,0	32,2	103,0	5,8	108,8	191,4	50,0	241,4
9	32,2	5,0	37,2	96,3	40,0	136,3	241,4	12,5	253,9
10	37,2	9,0	46,2	136,3	3,0	139,3	253,9	22,5	276,4
11	46,2	0,6	46,8	139,3	3,3	142,6	276,4	1,5	277,9
12	3,1	40,0	43,1	142,6	1,3	143,9	168,7	100,0	268,7
13	46,8	3,0	49,8	143,9	3,3	147,2	277,9	7,5	285,4
14	49,8	4,5	54,3	147,2	1,5	148,7	285,4	11,3	296,7
15	54,3	1,0	55,3	148,7	22,5	171,2	296,7	2,5	299,2
16	55,3	4,5	59,8	103,0	1,5	104,5	299,2	11,3	310,4
17	59,8	0,6	60,4	104,5	3,3	107,8	310,4	1,5	311,9
18	60,4	5,0	65,4	107,8	1,7	109,5	311,9	12,5	324,4
19	65,4	5,0	70,4	109,5	4,0	113,5	324,4	12,5	336,9
20	70,4	4,0	74,4	113,5	1,5	115,0	336,9	10,0	346,9
21	74,4	3,3	77,7	115,0	40,0	155,0	346,9	8,3	355,3
22	77,7	1,4	79,1	155,0	40,0	195,0	355,3	3,5	358,8
23	79,1	8,3	87,4	195,0	10,0	205,00	358,8	20,8	379,6
24				171,2	3,0	174,2			
25				174,2	5,3	179,2			
26				179,2	1,3	180,5			
27				180,5	5,3	185,8			
28				185,8	0,7	186,5			
29				205,0	5,0	210,0			
30				210,0	7,0	217,0			
31				217,0	1,7	218,7			
32				218,7	7,0	225,7			
33				225,7	3,0	228,7			
34				228,7	8,3	237,0			

Dengan menggunakan langkah pengerjaan yang sama untuk menjadwalkan *job* yang terakhir yaitu *job* 3, maka didapatkan hasil akhir bahwa waktu penyelesaian dari *job* 1 adalah selama 87,3 jam, *job* 2 selama 237 jam dan *job* 3 selama 379,6 jam. Pada

perhitungan menggunakan algoritma *sequencing* tersebut terlihat beberapa langkah yang diulang (*looping*), yaitu ketika kondisi seluruh *job* dalam satu operasi belum dijadwalkan dalam mesin dan ketika seluruh *job* belum dijadwalkan. Ketika kondisi ini belum dipenuhi maka perhitungan algoritma akan terus dilakukan dengan mengulang langkah 5 dan 3 secara terus menerus. Namun apabila kondisi ini telah dipenuhi maka langkah pengerjaan akan lanjut ke langkah 18, yaitu menghitung lama waktu penyelesaian masing-masing *order* atau *job*. Perhitungan lebih lengkap dari *sequencing* ini dapat dilihat pada Lampiran 4 Perhitungan Algoritma *Sequencing Job* 1-2-3 Urutan ketiga (*Job* 3). Selanjutnya, rekap data hasil penjadwalan secara rinci ditampilkan pada Tabel 4.8.

Completion time pada Tabel 4.8 masih merupakan data mentah dari penjadwalan ini karena waktu ini merupakan waktu penyelesaian bukan waktu produksi yang dibutuhkan oleh masing-masing *job*. Untuk mengetahui lamanya waktu produksi yang dibutuhkan masing-masing produk dapat dihitung sesuai dengan algoritma pada langkah ke-18 yang menggunakan rumus (4-12) dan akan ditampilkan sebagai berikut.

Rumus (4-12) : $C_i = (C_{i,j=p,k(x)}) - r_i$

- Job 1 $\rightarrow C_1 = (C_{1,23,k(x)}) - r_1 = 87,3 - 0 = 87,3$ jam
- Job 2 $\rightarrow C_2 = (C_{2,34,k(x)}) - r_2 = 237 - 81 = 156$ jam
- Job 3 $\rightarrow C_3 = (C_{3,23,k(x)}) - r_3 = 379,6 - 161 = 218,6$ jam

Dari perhitungan tersebut *job* 1 membutuhkan waktu produksi selama 87,3 jam, *job* 2 selama 156 jam, dan *job* 3 selama 218,6 jam. Nilai ini didapatkan dari selisih antara waktu *completion* dengan waktu siap dari masing-masing *order*-nya dimana waktu siap dari tiap *order* adalah berbeda sehingga akan mempengaruhi nilai dari *completion time* dan lama waktu produksinya. Selanjutnya keterangan tanggal dimana setiap *job* dapat selesai ditampilkan pada Tabel 4.9

Tabel 4.9 Rekap Perhitungan Lama Penjadwalan dan Tanggal Penyelesaiannya

<i>Job</i>	Tanggal Order	C_i	Jam Kerja	Lama hari	Tanggal Selesai
<i>Job</i> 1	22-Des-2013	87,3 jam	8 jam	11 hari	8-Jan-2014
<i>Job</i> 2	06-Jan-2014	156 jam	8 jam	20 hari	04-Feb-2014
<i>Job</i> 3	16-Jan-2014	218,6 jam	8 jam	28 hari	25-Feb-2014

Berdasarkan Tabel 4.9 didapatkan keterangan mengenai lama hari penyelesaian dan tanggal selesai dari masing-masing *job*. Nilai lama hari penyelesaian didapatkan dari perhitungan waktu produksi yang dibutuhkan oleh masing-masing *job* sesuai dengan rumus (4-12). Nilai ini kemudian dikonversikan ke dalam satuan hari dan diperkirakan tanggal selesainya tanpa memperhatikan hari sabtu dan minggu, dengan

kata lain perusahaan tidak melakukan lembur untuk menyelesaikan tiga *job* tersebut. Selanjutnya, dapat dihitung nilai dari *lateness*, *tardiness* hingga besarnya penalti dari *job* tersebut. Berikut tahapan dalam menghitung *lateness*, *tardiness*, dan penalti dari penjadwalan dengan berdasarkan urutan EDD.

- Perhitungan *lateness*.

$$\text{Job 1} \rightarrow L_1 = C_1 - D_1 = 8 \text{ Januari 2014} - 22 \text{ Januari 2013} = -9 \text{ hari}$$

$$\text{Job 2} \rightarrow L_2 = C_2 - D_2 = 04 \text{ Februari 2014} - 20 \text{ Februari 2014} = -12 \text{ hari}$$

$$\text{Job 3} \rightarrow L_3 = C_3 - D_3 = 25 \text{ Februari 2014} - 17 \text{ Maret 2014} = -14 \text{ hari}$$

- Perhitungan *tardiness*

$$\text{Job 1} \rightarrow T_1 = \text{Max}\{0, L_1\} = \text{Max}\{0, -9\} = 0 \text{ hari}$$

$$\text{Job 2} \rightarrow T_2 = \text{Max}\{0, L_2\} = \text{Max}\{0, -12\} = 0 \text{ hari}$$

$$\text{Job 3} \rightarrow T_3 = \text{Max}\{0, L_3\} = \text{Max}\{0, -14\} = 0 \text{ hari}$$

- Perhitungan penalti

$\text{Job 1} \rightarrow P_1 = 5\% \times T_1 \times \text{nilai kontrak} = 0$, karena tidak terjadi keterlambatan, maka penjadwalan untuk *job 1* tidak menimbulkan biaya penalti.

$\text{Job 2} \rightarrow P_2 = 5\% \times T_2 \times \text{nilai kontrak} = 0$, karena tidak terjadi keterlambatan, maka penjadwalan untuk *job 2* tidak menimbulkan biaya penalti.

$\text{Job 3} \rightarrow P_3 = 5\% \times T_1 \times \text{nilai kontrak} = 0$, karena tidak terjadi keterlambatan, maka penjadwalan untuk *job 3* tidak menimbulkan biaya penalti.

Berdasarkan perhitungan poin diatas dapat diketahui hasil dari perhitungan masing-masing *job* untuk *completion time*, *lateness*, dan *tardiness*-nya. Lama waktu penyelesaian (*completion time*) *job 1* adalah 87,3 jam; *job 2* adalah 156 jam; dan *job 3* adalah 218,6 jam. Lama waktu keterlambatan (*lateness*) untuk *job 1* adalah -9 hari, *job 2* adalah -12 hari, dan *job 3* adalah -14 hari. Nilai negatif (-) dari hasil perhitungan *lateness* menandakan bahwa *job* yang bersangkutan dapat selesai lebih awal dari *due date* yang telah ditentukan, untuk ketiga *job* ini semuanya memiliki nilai yang negatif sehingga dapat dikatakan bahwa seluruh *job* dapat selesai sebelum tanggal *due date*-nya dan tidak menghasilkan keterlambatan. Total waktu keterlambatan (*tardiness*) untuk semua *job* adalah nol atau tidak terjadi sehingga prioritas urutan penjadwalan ini tidak menimbulkan biaya penalti juga. Untuk lebih ringkasnya, hasil yang didapat pada poin diatas akan ditampilkan pada Tabel 4.10 agar memberikan informasi dari penjadwalan ini secara keseluruhan.

Tabel 4.10 Rekap Perhitungan Waktu Penyelesaian dan Penalti

Job	Due date (D_i)	C_i	Lateness (L_i)	Tardiness (T_i)	Penalti (P_i)
1	22-Jan-2014	08-Jan-2014	-9 hari	0	0
2	20-Feb-2014	04-Feb-2014	-12 hari	0	0
3	17-Mar-2014	25-Feb-2014	-14 hari	0	0

Metode selanjutnya yang dipertimbangkan untuk dijadwalkan adalah dengan menggunakan inisialisasi metode FCFS dimana *job* yang datang terlebih dahulu akan langsung diproses pada mesin yang *eligible*. Sesuai dengan Tabel 4.5, pengurutan dengan menggunakan metode ini menghasilkan urutan penjadwalan *job* yang sama dengan metode EDD, yaitu *job* 1-*job* 2-*job* 3, sehingga apabila dilakukan perhitungan untuk penjadwalan maka akan memberikan hasil yang sama. Hal ini dikarenakan waktu kedatangan *job* dan waktu *due date* dari masing-masing *job* berbanding lurus, dengan kata lain *job* yang datang lebih awal juga memiliki waktu *due date* yang lebih awal juga. Mempertimbangkan hal ini, maka perhitungan menggunakan metode FCFS tidak dilakukan karena akan memberikan hasil yang sama. Namun untuk kasus yang berbeda, perhitungan dengan menggunakan metode FCFS bisa saja tetap diperhitungkan apabila waktu *order* datang tidak berbanding lurus dengan waktu *due date*-nya.

4.3.4 Perhitungan Menggunakan Algoritma *Tabu search*

Algoritma *Tabu search* dimulai dengan merepresentasikan fungsi tujuan dan fungsi kendala pada penjadwalan ini. Fungsi tujuan yang digunakan sebagai acuan dalam penelitian ini didefinisikan sebagai *aspiration criterion* dari *tabu search* selama masa iterasinya, yaitu memilih jadwal yang memiliki nilai *tardiness* dan penalti yang terkecil. Sehingga nilai $G(S_c)$ akan dihitung berdasarkan nilai *tardiness* dan penaltinya. Sedangkan fungsi kendala yang dimiliki adalah jumlah *job* yang akan dijadwalkan terbilang sedikit sehingga akan mempengaruhi penentuan nilai dari *move*. *Move* yang ideal adalah berkisar antara 5-9 *move*. Namun karena keterbatasan dari jumlah *job* yang ada yaitu hanya sebanyak 3 *job*, maka *move* yang digunakan adalah sebanyak 2 untuk menghindari ruang pencarian yang sempit. Selain itu, batas iterasi (k) adalah peluang dengan menggunakan aturan permutasi tanpa pengulangan yang nilainya didapatkan dari rumus $\frac{n!}{(n-r)!}$. Berdasarkan rumus tersebut maka didapatkan hasil nilai N adalah sebanyak 6 iterasi, hal ini dipertimbangkan untuk memastikan bahwa solusi yang dipilih adalah benar-benar yang terbaik dimana dengan jumlah *job* sangat sedikit.

Berikut penjelasan masing-masing notasi yang digunakan selama perhitungan menggunakan algoritma *Tabu Search*:

- a. Vektor S_k = Kumpulan dari seluruh urutan jadwal dari *job* yang ada
- b. Vektor $S_{best} (S_o)$ = Jadwal inisial yang optimum
- c. Vektor S_c = Jadwal yang dipilih sebagai kandidat untuk dibandingkan dengan S_{best}
- d. $G(S_c)$ = Nilai yang dimiliki oleh jadwal kandidat berdasarkan fungsi tujuannya, dalam hal ini yaitu nilai dari *tardiness*.
- e. $G(S_{best})$ = Nilai yang dimiliki oleh jadwal acuan (jadwal yang optimum saat ini).
- f. *Move* = Pergerakan dari urutan *job* yang ditukar.
- g. *Tabu list* = Daftar yang menyimpan pergerakan dari atribut *move*. *Job* yang ditukar berpasangan (j,k) yang telah ditukar dalam dua *move* terakhir.
- h. N = Banyak iterasi dimana suatu kondisi untuk menyatakan bahwa pencarian solusi telah berhenti.

Setelah masing-masing notasi terdefinisi maka perhitungan awalnya akan dijelaskan sebagai berikut:

$k = 1$; $Move = 2$

$N = 6$

$G(S_o) = 0$

1. Iterasi pertama

- Set $k = 1$
 S_1 menggunakan metode EDD = 1-2-3
 $S_o = S_1$
- $S_c = 2-1-3$ dan $S_c = 1-3-2$
Move terlarang? Tidak.
Maka $S_{c1} = 2-1-3 \rightarrow Tabu\ list = \{(1,2)\}$
 $S_{c2} = 1-3-2 \rightarrow Tabu\ list = \{(2,3)\}$
Cek $G(S_c) < G(S_o)$? $G(S_o) = 0$ hari

$$G(S_{c1}) = 0, \rightarrow G(S_{best})$$

Tabel 4.11 Hasil Penjadwalan Algoritma *Tabu Search* Iterasi Pertama Solusi 1

	r_i	<i>Due date</i>	C_i	<i>Lateness</i>	<i>Tardiness</i>	Penalti
<i>Job 1</i>	22-Des-13	22-Jan-13	08-Jan-14	-9	0	0
<i>Job 2</i>	06-Jan-14	20-Feb-14	04-Feb-14	-12	0	0
<i>Job 3</i>	16-Jan-14	17-Mar-14	25-Feb-14	-14	0	0

$$G(S_{c2}) = 0,$$

Tabel 4.12 Hasil Penjadwalan Algoritma *Tabu Search* Iterasi Pertama Solusi 2

	r_i	<i>Due date</i>	C_i	<i>Lateness</i>	<i>Tardiness</i>	Penalti
<i>Job 1</i>	22-Des-13	22-Jan-13	08-Jan-14	-9	0	0
<i>Job 2</i>	06-Jan-14	20-Feb-14	04-Feb-14	-12	0	0
<i>Job 3</i>	16-Jan-14	17-Mar-14	25-Feb-14	-14	0	0

Tabu-list:{(1,2)}

- $k = 1+1 = 2$,
 $k = 6$? Tidak, maka lanjut ke iterasi kedua.

2. Iterasi kedua

- $S_{c1} = 2-3-1$ dan $S_{c2} = 1-2-3$ (*tabu!*)
Move terlarang? Tidak.
Maka $S_{c1} = 2-3-1 \rightarrow$ *Tabu list* = {(1,3)}
Cek $G(S_c) < G(S_o)$? $G(S_o) = 0$ hari
 $G(S_{c1}) = 0$ hari, $\rightarrow G(S_{best}) = 0$ hari

Tabel 4.13 Hasil Penjadwalan Algoritma *Tabu Search* Iterasi Kedua Solusi 1

	r_i	<i>Due date</i>	C_i	<i>Lateness</i>	<i>Tardiness</i>	Penalti
<i>Job 1</i>	22-Des-13	22-Jan-13	28-Jan-14	-9	0	0
<i>Job 2</i>	06-Jan-14	20-Feb-14	12-Feb-14	-6	0	0
<i>Job 3</i>	16-Jan-14	17-Mar-14	28-Feb-14	-11	0	0

Tabu-list:{(1,3) ; (1,2)}

- $k = 2+1 = 3$
 $k = 6$? Tidak, maka lanjut ke iterasi ketiga.

3. Iterasi ketiga

- $S_{c1} = 3-2-1$ dan $S_{c2} = 2-1-3$ (*tabu!*)
Move terlarang? Tidak.
Maka $S_{c1} = 3-2-1 \rightarrow$ *Tabu list* = {(1,3)}
Cek $G(S_c) < G(S_o)$? $G(S_o) = 0$ hari

$$G(S_{c1}) = 0 \text{ hari}, \rightarrow G(S_{\text{best}}) = 0 \text{ hari}$$

Tabel 4.14 Hasil Penjadwalan Algoritma *Tabu Search* Iterasi Ketiga Solusi 1

	r_i	<i>Due date</i>	C_i	<i>Lateness</i>	<i>Tardiness</i>	Penalti
<i>Job 1</i>	22-Des-13	22-Jan-13	08-Jan-14	-9	0	0
<i>Job 2</i>	06-Jan-14	20-Feb-14	04-Feb-14	-12	0	0
<i>Job 3</i>	16-Jan-14	17-Mar-14	25-Feb-14	-14	0	0

Tabu-list:{(2,3) ; (1,3)}

- $k = 3+1 = 4$
 $k = 6$? Tidak, maka lanjut ke iterasi keempat.

4. Iterasi keempat

- $S_c = 3-1-2$ dan $S_c = 2-3-1$ (*tabu!*)
Move terlarang? Tidak.
Maka $S_{c1} = 3-1-2 \rightarrow \text{Tabu list} = \{(2,1)\}$
Cek $G(S_c) < G(S_o)$? $G(S_o) = 0$ hari
 $G(S_{c1}) = 0$ hari, $\rightarrow G(S_{\text{best}}) = 0$ hari

Tabel 4.15 Hasil Penjadwalan Algoritma *Tabu Search* Iterasi Keempat Solusi 1

	r_i	<i>Due date</i>	C_i	<i>Lateness</i>	<i>Tardiness</i>	Penalti
<i>Job 1</i>	22-Des-13	22-Jan-13	08-Jan-14	-9	0	0
<i>Job 2</i>	06-Jan-14	20-Feb-14	06-Feb-14	-10	0	0
<i>Job 3</i>	16-Jan-14	17-Mar-14	25-Feb-14	-14	0	0

Tabu-list:{(2,1) ; (2,3)}

- $k = 4+1 = 5$
 $k = 6$? Tidak, maka lanjut ke iterasi kelima.

5. Iterasi kelima

- $S_c = 1-3-2$ dan $S_c = 3-2-1$ (*tabu!*)
Move terlarang? Tidak.
Maka $S_{c1} = 3-1-2 \rightarrow \text{Tabu list} = \{(3,1)\}$
Cek $G(S_c) < G(S_o)$? $G(S_o) = 0$ hari
 $G(S_{c1}) = 0$ hari

Tabel 4.16 Hasil Penjadwalan Algoritma *Tabu Search* Iterasi Kelima Solusi 1

	r_i	<i>Due date</i>	C_i	<i>Lateness</i>	<i>Tardiness</i>	Penalti
<i>Job 1</i>	22-Des-14	22-Jan-13	08-Jan-14	-9	0	0
<i>Job 2</i>	06-Jan-14	20-Feb-14	03-Mar-14	-12	0	0
<i>Job 3</i>	16-Jan-14	17-Mar-14	28-Feb-14	-14	0	0

Tabu-list:{((3,2) ; (2,3)}

- $k = 5+1 = 6$

$k = 6$? Ya, maka STOP.

Perhitungan dengan menggunakan algoritma *tabu search* mulai dari iterasi pertama hingga iterasi kelima dapat dijelaskan sebagai berikut :

1. Pada iterasi pertama ada dua alternatif solusi penyelesaian yang dapat dipertimbangkan, yakni solusi penjadwalan dengan menggunakan urutan 2-1-3 dan urutan 1-3-2. Kedua solusi ini dibandingkan nilai *tardiness*-nya terhadap nilai *tardiness* yang dimiliki oleh jadwal dengan urutan 1-2-3. Nilai acuan ini dinotasikan sebagai $G(S_0)$. Dua alternatif solusi pada iterasi pertama memiliki nilai $G(S_0)$ yang sama dengan solusi inisial awalnya. Namun, dalam iterasi ini hanya akan dipilih satu alternatif saja, yakni jadwal dengan urutan 2-3-1. Kemudian, definisikan nilai k , lanjut ke iterasi selanjutnya.
2. Pada iterasi kedua ada dua alternatif solusi penyelesaian yang dapat dipertimbangkan, yakni jadwal dengan urutan 2-3-1 dan 1-2-3. Sesuai dengan konsep *tabu search* yang tidak akan mengulang alternatif solusi yang telah terpilih sebelumnya (*tabu!*) maka jadwal dengan urutan 1-2-3 telah dianggap tabu dan tidak diperhitungkan lagi keberadaannya, sehingga pada iterasi ini hanya ada satu solusi yakni jadwal dengan urutan 2-3-1. Jadwal ini menghasilkan nilai yang tidak lebih baik dari jadwal sebelumnya tapi masih memenuhi syarat fungsi tujuan, yakni meminimasi *tardiness* atau nilai $G(S_0) = 0$, sehingga alternatif solusi ini masih dipertimbangkan. Definisikan nilai k yang baru, lanjutkan ke iterasi berikutnya.
3. Pada iterasi ketiga hanya ada satu alternatif solusi penyelesaian yang dapat dipertimbangkan, yakni jadwal dengan urutan 3-2-1. Jadwal ini menghasilkan nilai yang sama dengan dua solusi terdahulunya, yaitu jadwal dengan urutan 1-2-3 dan 2-1-3. Meskipun begitu, untuk mengantisipasi kemungkinan adanya solusi yang lebih baik dari tiga alternatif urutan penjadwalan sebelumnya, maka iterasi masih dilanjutkan. Definisikan nilai k yang baru, lanjutkan ke iterasi berikutnya untuk melakukan pengecekan lebih lanjut.
4. Pada iterasi keempat ada dua alternatif solusi penyelesaian yang dapat dipertimbangkan, yakni jadwal dengan urutan 3-1-2 dan 2-3-1. Jadwal dengan urutan 2-3-1 sudah dianggap tabu sehingga jadwal ini tidak diperhitungkan lagi dan pada iterasi keempat hanya ada satu alternatif solusi, yaitu jadwal dengan urutan 3-1-2. Nilai $G(S_0)$ dari jadwal ini terpenuhi tapi tidak menghasilkan nilai yang lebih baik dari jadwal di iterasi ketiga, sehingga alternatif solusi ini masih bisa

dipertimbangkan untuk mengetahui apakah solusi pada iterasi ketiga sudah yang terbaik ataukah ada yang lebih baik di iterasi selanjutnya. Oleh karena itu, iterasi dilanjutkan dengan mendefinisikan nilai k yang baru.

5. Pada iterasi kelima ada dua alternatif solusi penyelesaian yang dapat dipertimbangkan, yakni jadwal dengan urutan 3-2-1 dan 1-3-2. Jadwal dengan urutan 3-2-1 sudah dianggap tabu sehingga jadwal ini tidak diperhitungkan lagi dan pada iterasi kelima hanya ada satu alternatif solusi, yaitu jadwal dengan urutan 1-3-2. Jadwal ini menghasilkan nilai yang tidak lebih baik dari jadwal pada iterasi pertama dan ketiga karena meskipun tidak menghasilkan *tardiness* tapi urutan ini menghasilkan waktu penyelesaian *job* yang lebih lama dibandingkan penjadwalan yang terpilih pada iterasi pertama dan ketiga. Oleh karena itu, alternatif solusi ini tidak bisa dipertimbangkan untuk dipilih. Selain itu, karena k telah sama dengan nilai N maka iterasi berhenti sampai pada iterasi kelima.

Berdasarkan penjadwalan lebih lanjut dengan menggunakan algoritma *Tabu Search*, didapatkan hasil bahwa penjadwalan produksi dengan mengikuti 3 alternatif urutan yaitu 2-3-1, 3-2-1 dan 1-3-2 karena dengan menggunakan urutan ini seluruh *job* tidak menghasilkan nilai *tardiness* dan penalti. Hasil dengan menggunakan algoritma *Tabu Search* sama dengan alternatif inisialnya yang tidak menghasilkan nilai *tardiness* dan penalti, yaitu jadwal dengan urutan 1-2-3. Hal ini dapat terjadi karena seluruh *job* datang satu persatu dan tidak ada beberapa *job* yang dikerjakan secara bersamaan dalam satu jangkauan waktu di rantai produksi.

Namun secara keseluruhan, penjadwalan dengan menggunakan metode inisialisasi EDD dan FCFS yang kemudian diintegrasikan dengan Algoritma *Tabu Search* dapat diterapkan untuk mengurangi *tardiness*, total *tardiness*, dan penalti dalam lingkungan *Flexible Job Shop*, bahkan jadwal proses produksi yang direncanakan berdasarkan hasil perhitungannya dapat selesai jauh sebelum *due date*-nya. Selisih hari yang dihasilkan dengan urutan ini terbilang cukup besar, yakni mulai dari 9 hingga 20 hari aktif kerja. Harapannya dengan selisih hari yang terbilang besar tersebut perusahaan dapat menerima order baru atau melanjutkan order lain yang belum terselesaikan yang tentunya akan memberikan dampak yang sangat positif bagi perusahaan.

4.3.4 Verifikasi dan Validasi Model Penjadwalan

Model penjadwalan dengan menggunakan pendekatan dari metode heuristik yaitu metode EDD dan FCFS yang kemudian diintegrasikan dengan metode *tabu search* untuk meminimasi *tardiness* dan total penalti, maka diperoleh hasil jadwal akhir yang diusulkan untuk diterapkan dalam lingkungan *flexible jobshop* seperti yang telah ditampilkan pada Tabel 4.8, Tabel 4.11, Tabel 4.11, dan Tabel 4.16.

Pada Tabel 4.8 telah dijabarkan mengenai waktu mulai, lama waktu pemrosesan dan waktu selesai dari masing-masing operasi. Berdasarkan proses perhitungan yang telah dilakukan, model penjadwalan ini telah memenuhi kondisi sistem dan tujuan akhir. Perhitungan pertama yaitu dengan menggunakan metode EDD, menghasilkan jadwal produksi yang dapat memenuhi *due date*-nya, sedangkan penjadwalan yang kedua berdasarkan waktu kedatangan *order*-nya juga dapat memenuhi tujuan dari penjadwalan ini, yaitu untuk meminimasi *tardiness* dan nilai dari penalti. Selain itu, model algoritma *sequencing* yang menghasilkan waktu selesai masing-masing operasi sesuai dengan logika model dan terjadwal secara sistematis. Maka dari itu, model penjadwalan ini telah terverifikasi.

Sedangkan, validasi merupakan proses mengevaluasi suatu model apakah telah sesuai mewakili sistem yang dimodelkan baik dalam memberikan keluaran yang sama maupun dalam mempresentasikan perilaku sistem tersebut. Proses validasi untuk algoritma *sequencing* dilakukan dengan membandingkan hasil perhitungan secara manual dengan *output* yang dihasilkan dengan menggunakan *gantt chart*. Apabila kedua hasil ini cocok, maka dapat dikatakan algoritma *sequencing* telah tervalidasi.

Selain itu, proses validasi diperlukan untuk memeriksa apakah algoritma sudah sesuai dengan asumsi, fungsi tujuan, dan fungsi kendala yang dirumuskan pada awal pemodelan. Berikut ini merupakan analisis singkat untuk menunjukkan bahwa algoritma sesuai atau telah memenuhi asumsi, fungsi tujuan dan fungsi kendala.

1. Asumsi

Berdasarkan *sequencing* proses produksi dan perhitungan waktu produksi dengan menggunakan algoritma usulan, 10 asumsi yang telah ditetapkan di awal penjadwalan dapat terpenuhi dengan menggunakan algoritma *sequencing* ini, sehingga dapat diambil kesimpulan bahwa algoritma *sequencing* ini telah tervalidasi atau mendekati sistem nyatanya. Untuk lebih jelas, hasil penjadwalan dengan menggunakan *gantt chart* dapat dilihat pada Lampiran 7 Gambar *Gantt Chart* Validasi Algoritma *Sequencing*.

2. Fungsi Tujuan

a. Minimasi Total *Tardiness*

Minimasi total *tardiness* dapat dihitung dengan menggunakan rumus (4-2). Rangkuman *tardiness* dari masing-masing *job* dapat dilihat pada Tabel 4.17. Dari tabel 4.17 dapat disimpulkan bahwa model algoritma usulan dapat memenuhi fungsi tujuan untuk meminimasi total *tardiness* *job* yang ada. Selain itu, dapat dilihat juga bahwa dengan menggunakan algoritma *sequencing* ini seluruh *job* dapat selesai jauh lebih cepat dan tidak terjadi keterlambatan sama sekali.

Tabel 4.17 Perbandingan *Tardiness* antara Sistem Penjadwalan *Existing* dengan Penjadwalan Usulan

Order atau Job (i)	Realisasi penyelesaian job dengan sistem penjadwalan <i>existing</i> , C_i (hari).	Realisasi penyelesaian job dengan sistem penjadwalan usulan, C_i (hari).	<i>Tardiness</i> sistem penjadwalan <i>existing</i> $T_1 = \text{Max}\{0, L_1\}$ (hari)	<i>Tardiness</i> sistem penjadwalan <i>existing</i> $T_1 = \text{Max}\{0, L_1\}$ (hari)
1	31	11	12	0
2	41	20	13	0
3	63	28	20	0
Total <i>Tardiness</i>				0

b. Minimasi Nilai Penalti

Perhitungan penalti diperoleh dengan menggunakan rumus (4-3) atau (4-4) bergantung dari lama hari keterlambatannya. Jumlah keterlambatan dan penalti yang harus dibayar masing-masing *job* ditampilkan pada Tabel 4.18.

Tabel 4.18 Perbandingan Penalti antara Sistem Penjadwalan *existing* dengan Penjadwalan Usulan

Order atau Job (i)	<i>Tardiness</i> sistem penjadwalan <i>existing</i> $T_1 = \text{Max}\{0, L_1\}$ (hari)	<i>Tardiness</i> sistem penjadwalan <i>existing</i> $T_1 = \text{Max}\{0, L_1\}$ (hari)	Prosentase Penalti penjadwalan <i>existing</i> (P_i)	Prosentase Penalti penjadwalan <i>existing</i> (P_i)
1	12	0	60%	0
2	13	0	65%	0
3	20	0	100%	0
Total Penalti				0

Dari tabel 4.18 dapat disimpulkan bahwa model algoritma usulan dapat memenuhi fungsi tujuan untuk meminimasi penalti dari masing-masing *job*. Selain itu, dapat dilihat bahwa dengan menggunakan algoritma *sequencing* ini

seluruh *job* tidak terjadi keterlambatan dan tidak menghasilkan penalti sama sekali.

3. Fungsi Kendala

a. Fungsi kendala dalam menentukan waktu siap dari *job* *i*.

Waktu siap dari *job* *i* berbeda bergantung waktu persetujuan kontraknya. *Job* yang memiliki tanggal paling awal memiliki waktu siap nol ($t_i = 0$). Selanjutnya, untuk *job* yang datang setelah *job* pertama tersebut dihitung berdasarkan selisih hari aktif antara tanggal pada saat $t = 0$ hingga tanggal persetujuan dari *job* itu sendiri. Hasil dari selisih tersebut kemudian dikali dengan jumlah jam kerja selama satu hari, yakni 8 jam. Total dari jam tersebut kemudian menjadi waktu siap untuk *job* *i*. Penjelasan tersebut dapat dinyatakan seperti pada rumus (4-5).

$$r_i = \Sigma \text{hari aktif}_{\text{tanggal order job } i - \text{tanggal job pertama dikerjakan}} \times 8 \text{ jam}$$

Dengan mengacu pada persamaan tersebut maka waktu ready dari masing masing *job* dapat ditentukan dengan akurat dan disesuaikan dengan kondisi sistem yang sebenarnya, sehingga dapat dikatakan bahwa algoritma *sequencing* dapat memenuhi fungsi kendala dalam menentukan waktu siap dari *job* *i*. Perhitungan dari fungsi kendala ini akan ditampilkan pada Tabel 4.19.

Tabel 4.19 Daftar Waktu Siap *Job* *i*

<i>Job</i> (<i>i</i>)	Tanggal datang kontrak	Selisih hari	Waktu siap (jam)
1	23-Des-2014	0	$r_1 = (0 \times 8) = 0$
2	06-Jan-2014	10	$r_2 = (10 \times 8) + 1 = 81$
3	16-Jan-2014	20	$r_3 = (20 \times 8) + 1 = 161$

b. Fungsi kendala dalam menentukan waktu siap dari operasi *j* *job* *i* dalam sistem.

Waktu siap dari operasi *j* *job* *i* dibagi menjadi dua kriteria seperti pada rumus (4-6) dan (4-7). Waktu siap dari operasi pertama *job* urutan pertama adalah sama dengan nol, sedangkan untuk operasi lainnya berdasarkan waktu *completion time* dari *precedence* operasinya.

$$r_{i,j,k(x)} = 0, \text{ untuk } \textit{job} \text{ urutan pertama operasi pertama} \quad (4-6)$$

$$r_{i,j,k(x)} = C_{k(x)}, \text{ untuk operasi ke-} j \text{ } \textit{job} \text{ } i \quad (4-7)$$

Pada Tabel 4.20 dijelaskan tentang *precedence* yang merupakan indikator bahwa operasi *j* dapat diproses setelah operasi pendahulunya selesai dikerjakan pada mesin *k* (waktu siap masing-masing operasi).

Tabel 4.20 Waktu Siap Masing-masing Operasi j Job i Urutan 1-2-3

Ope rasi	Job 1			Job 2			Job 3		
	Prece dence	$r_{i,j,k(x)}$	$C_{i,j,k(x)}$	Prece dence	$r_{i,j,k(x)}$	$C_{i,j,k(x)}$	Prece dence	$r_{i,j,k(x)}$	$C_{i,j,k(x)}$
1	-	0	2,7	-	0	91	-	0	167,7
2	-	0	6	-	0	93,7	-	0	176
3	1	2,7	3,1	1	91	91,3	1	167,7	168,7
4	2	6	6,3	1	91	91,7	2	176	176,8
5	2	6	6,4	2	93,7	94,1	2	176	177
6	2	6	6,5	3	91,3	96,3	2	176	177,2
7	4	6,3	12,2	4	91,7	103	4	176,8	191,4
8	7	12,2	32,2	5	94,1	108,8	7	191,4	241,4
9	8	32,2	37,2	6	96,3	136,3	8	241,4	253,9
10	9	37,2	46,2	9	136,3	139,3	9	253,9	276,4
11	10	46,2	46,8	10	139,3	142,7	10	276,4	277,9
12	3	3,1	43,1	11	142,7	144	3	168,7	268,7
13	11,12	46,8	49,8	12	144,0	147,3	11,12	277,9	285,4
14	13	49,8	54,3	13	147,3	148,8	13	285,4	296,7
15	14	54,3	55,3	14	148,8	171,3	14	296,7	299,2
16	15	55,3	59,8	7	103	104,5	15	299,2	310,4
17	16	59,8	60,4	16	104,5	107,8	16	310,4	311,9
18	5,17	60,4	65,4	17	107,8	109,5	5,17	311,9	324,4
19	18	65,4	70,4	18	109,5	113,5	18	324,4	336,9
20	6,19	70,4	74,4	19	113,5	115	6,19	336,9	346,9
21	20	74,4	77,7	20	115,0	155	20	346,9	355,3
22	21	77,7	79,1	21	155,0	195	21	355,3	358,8
23	22	79,1	87,4	22	195,0	205	22	358,8	379,6
24				8,15	171,3	174,3			
25				24	174,3	179,3			
26				25	179,3	180,7			
27				26	180,7	186			
28				27	186,0	186,7			
29				23,28	205	210			
30				29	210	217			
31				30	217	218,7			
32				31	218,7	225,7			
33				32	225,7	228,7			
34				33	228,7	237			

- c. Fungsi kendala dalam menentukan waktu mulai dari operasi j job i dalam sistem.

Dalam menentukan waktu mulai dari operasi j job i harus memperhatikan dua hal, yakni waktu saat mesin yang digunakan telah berstatus *eligible* dan waktu

siap dari operasi itu sendiri. Kedua kondisi tersebut sesuai dengan rumus (4-8) dan (4-9).

$$S_{i,j,k(x)} = r_{i,j,k(x)}, \text{ ketika } C_{k(x)} \leq r_{i,j,k(x)} \quad (4-8)$$

$$S_{i,j,k(x)} = C_{k(x)}, \text{ ketika } C_{k(x)} > r_{i,j,k(x)} \quad (4-9)$$

Dari kedua hal tersebut nantinya akan menentukan kapan suatu operasi akan mulai diproses dan ditugaskan pada mesin yang sesuai.

- d. Fungsi kendala waktu penyelesaian proses j job i di mesin k terhadap waktu penyelesaian akhir job i .

Pada rumus (4-10) dan (4-11) dijelaskan mengenai langkah-langkah dalam menentukan waktu penyelesaian (*completion time*) dari masing-masing job. Pada rumus (4-11) dimaksudkan bahwa untuk menghitung waktu *completion* masing-masing job adalah dengan mengambil waktu penyelesaian operasi terakhir dari masing-masing job. Sesuai dengan Tabel 4.8 dapat dilihat waktu penyelesaian pada saat operasi terakhir dari masing-masing job adalah waktu penyelesaian dari job itu sendiri.

- e. Fungsi kendala waktu operasi j di mesin k yang paralel.

Validasi untuk fungsi kendala ini dapat dilihat pada Tabel 4.7. Pada tabel tersebut telah dijelaskan waktu proses masing-masing operasi di mesin yang sesuai. Sesuai dengan asumsi yang menyatakan pemerataan waktu operasi untuk di tiap-tiap mesin yang paralel, maka Tabel 4.7 sudah sesuai dengan asumsi ini.

Berdasarkan hasil verifikasi dan validasi dapat disimpulkan bahwa metode yang diusulkan dalam penjadwalan di lingkungan *Flexible Job Shop* terbukti dapat mengurangi nilai *tardiness* dan penalti yang sebelumnya sering terjadi di lantai produksi. Dengan mempertimbangkan waktu kedatangan setiap *order* dan *due date*-nya, metode ini dapat diterapkan untuk memenuhi fungsi tujuan tersebut.

Dalam upaya meminimasi *tardiness* dan penalti yang sering terjadi di perusahaan dengan alur produksi berupa *Flexible Job Shop*, maka diberikan dua alternatif inisialisasi jadwal awal yang disesuaikan dengan kondisi kedatangan *order*-nya. Mempertimbangkan waktu kedatangan *order* yang sifatnya dinamis deterministik, maka metode yang diusulkan adalah metode pengurutan job berdasarkan waktu kedatangannya dan berdasarkan waktu pemenuhan *due date*-nya. Oleh karena itu, metode FCFS dan EDD dipertimbangkan dapat memenuhi dua kondisi tujuan tersebut.

Selain itu, untuk mendapatkan hasil penjadwalan yang lebih baik lagi, hasil inisialisasi perbandingan dua metode di awal tersebut akan diolah lebih lanjut dengan menggunakan algoritma *Tabu search* dimana di setiap iterasinya diintegrasikan dengan aturan *Pairwise Interchange*. Proses pengolahan data dalam penelitian ini mempertimbangkan beberapa aspek penting lainnya yang berupa waktu operasi untuk mesin paralel yang mengadopsi alur produksi berupa *Pure Job Shop*, algoritma *sequencing* dengan beberapa penyesuaian, penentuan waktu mulai *job*, penentuan waktu mulai setiap operasi dalam satu *job*, dan lain-lain.

Inisialisasi awal dengan menggunakan EDD dan FCFS akan menghasilkan urutan prioritas *job* yang sama apabila waktu datang *order* berbanding lurus dengan waktu *due date*-nya. Sehingga apabila *job* memiliki waktu datang awal dan *due date* yang lebih awal dibandingkan dengan *job* yang lain maka hasil urutan jadwal produksi untuk semua *job* di triwulan awal tahun akan sama. Berdasarkan hal tersebut, maka pengurutan prioritas *job* dengan menggunakan metode EDD menghasilkan penjadwalan sama dengan metode FCFS yaitu urutan 1-2-3. Urutan ini tidak menyebabkan *tardiness* karena *job* 1 dapat dikerjakan 9 hari lebih cepat dari *due date*-nya, *job* 2 dapat dikerjakan 12 hari lebih cepat dari *due date*-nya, dan *job* 3 dapat diselesaikan 14 hari lebih cepat dari *due date*-nya. Waktu yang digunakan untuk menyelesaikan masing-masing *job* dari *job* 1 hingga 3 secara berurutan adalah 87,3 jam; 156 jam ; 218,6 jam.

Pengurutan jadwal lebih lanjut dengan menggunakan algoritma *Tabu search* menghasilkan alternatif yang tidak jauh berbeda dari alternatif inisial awalnya, yaitu ketika dijadwalkan dengan aturan menggunakan metode EDD dan FCFS. Pada algoritma *Tabu Search* penentuan nilai N ditentukan berdasarkan pertimbangan menurut teori permutasi tanpa pengulangan. Teori ini memperhatikan urutan pengambilan *job* dan *job* yang akan dijadwalkan hanya boleh dipilih sekali sehingga *job* yang diurutkan tidak boleh sama dalam satu kandidat. Hasil perhitungan peluang menggunakan teori permutasi tanpa pengulangan dari 3 *job* yang ada adalah 6, sehingga ketika nilai $k=N$ atau ketika $k=6$ maka iterasi dihentikan. Hasilnya, ada 3 alternatif solusi yang diusulkan yaitu urutan *job* 2-1-3, 3-2-1, dan 1-3-2 yang memiliki nilai yang sama dengan alternatif inisial awalnya, yaitu jadwal dengan urutan 1-2-3. Hal ini dikarenakan setiap *job* datang satu-persatu dan tidak ada beberapa *job* yang dikerjakan secara bersamaan dalam satu jangkauan waktu yang sama. Selain itu, konsep pengurutannya adalah untuk prioritas *job* dijadwalkan di *resource* yang sesuai, bukan urutan mulai dikerjakannya suatu *job*.

Jadi, meskipun urutannya diubah-ubah tetap akan menghasilkan urutan jadwal yang dapat memenuhi fungsi tujuannya, yaitu meminimasi *tardiness* dan penaltinya.

Perbandingan nilai *tardiness* dan penalti pada Tabel 4.18 dan 4.19 menunjukkan bahwa metode yang diusulkan dapat membantu perusahaan untuk menyelesaikan produksi lebih awal dari *due date*-nya. Hal ini karena jadwal dengan menggunakan metode usulan menghasilkan waktu produksi yang lebih singkat dan tidak menyebabkan *tardiness* dan penalti sama sekali. Berdasarkan Tabel 4.18, penjadwalan *existing* untuk job 1 hingga 3 menghasilkan *tardiness* selama 12 hari, 13 hari dan 20 hari, sedangkan penjadwalan baru dengan menggunakan metode usulan tidak menghasilkan *tardiness* atau keterlambatan pengerjaan. Selanjutnya pada tabel 4.19, jadwal *existing* juga mengakibatkan penalti untuk job 1 hingga 3 sebesar 60%; 65%; dan 100% dari besarnya nilai kontrak masing-masing. Sedangkan dengan menggunakan metode usulan, jadwal tidak menghasilkan nilai penalti sama sekali karena nilai penalti dipengaruhi oleh lamanya keterlambatan atau *tardiness* yang terjadi sehingga ketika tidak ada *tardy job* maka penalti juga tidak terjadi.

Secara keseluruhan, proses verifikasi dan validasi dapat dilakukan dengan beberapa proses, yakni *review* hasil, analisis data dan informasi nyata di lapangan, hingga konfirmasi ke pihak terkait, sehingga hasil penjadwalan dengan menggunakan algoritma *sequencing* ini telah terverifikasi dan tervalidasi serta dapat diaplikasikan langsung di rantai produksi. Berdasarkan beberapa penjelasan-penjelasan tersebut, maka hasil ini menunjukkan metode usulan telah sesuai dengan kondisi sistem secara nyata dan dapat diimplementasikan di PT Barata Indonesia (Persero) kedepannya.