

PEMODELAN SIMULASI DALAM BAHASA PEMROGRAMAN JULIA

LELY ROMANDHANI



**PROGRAM STUDI MATEMATIKA
SEKOLAH SAINS DATA, MATEMATIKA, DAN INFORMATIKA
INSTITUT PERTANIAN BOGOR
2025**

PERNYATAAN MENGENAI SKRIPSI DAN SUMBER INFORMASI SERTA PELIMPAHAN HAK CIPTA

Dengan ini saya menyatakan bahwa skripsi dengan judul “Pemodelan Simulasi dalam Bahasa Pemrograman Julia” adalah karya saya dengan arahan dari dosen pembimbing dan belum diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka di bagian akhir skripsi ini.

Dengan ini saya melimpahkan hak cipta dari karya tulis saya kepada Institut Pertanian Bogor.

Bogor, Mei 2025

Lely Romandhani
G54180039

ABSTRAK

LELY ROMANDHANI. Pemodelan Simulasi dalam Bahasa Pemrograman Julia. Dibimbing oleh FARIDA HANUM dan FAHREN BUKHARI.

Pemodelan simulasi adalah teknik yang penting untuk mempelajari perilaku sistem dinamis secara virtual. Penelitian ini mengeksplorasi kemampuan bahasa pemrograman Julia dalam menyelesaikan permasalahan pemodelan simulasi menggunakan paket-paket seperti *Distributions.jl* dan *ConcurrentSim.jl*. Contoh studi kasus mencakup manajemen antrean truk di tambang dan inspeksi pesawat di fasilitas layanan. Hasil simulasi menunjukkan efisiensi pengelolaan sistem dengan pendekatan berbasis peristiwa diskret serta pentingnya pemilihan parameter simulasi untuk meminimalkan waktu tunggu dan biaya operasional. Penelitian ini memberikan kontribusi dalam mengaplikasikan Julia sebagai alat pemodelan untuk meningkatkan efisiensi sistem di berbagai domain.

Kata kunci: ConcurrentSim.jl, Distributions.jl, Julia, pemodelan simulasi, sistem dinamis, simulasi antrean

ABSTRACT

LELY ROMANDHANI. Simulation Modeling in the Julia Programming Language. Supervised by FARIDA HANUM and FAHREN BUKHARI.

Simulation modeling is an essential technique for studying the behavior of dynamic systems virtually. This research explores the capabilities of the Julia programming language in addressing simulation modeling problems using packages such as *Distributions.jl* and *ConcurrentSim.jl*. Case studies include truck queue management in mining operations and aircraft engine inspection in service facilities. The simulation results demonstrate the efficiency of system management using a discrete-event-based approach and highlight the importance of selecting appropriate simulation parameters to minimize waiting times and operational costs. This study contributes to the application of Julia as a modeling tool to enhance system efficiency across various domains.

Keywords: ConcurrentSim.jl, Distributions.jl, dynamic systems, Julia, simulation modeling, queue simulation



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apa pun tanpa izin IPB University.

© Hak Cipta milik IPB, tahun 2025
Hak Cipta dilindungi Undang-Undang

Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan atau menyebutkan sumbernya. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik, atau tinjauan masalah, dan pengutipan tersebut tidak merugikan kepentingan IPB.

Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apa pun tanpa izin IPB.



PEMODELAN SIMULASI DALAM BAHASA PEMROGRAMAN JULIA

@Hak cipta milik IPB University

LELY ROMANDHANI

Skripsi
sebagai salah satu syarat untuk memperoleh gelar
Sarjana Matematika
pada
Program Studi Matematika

**PROGRAM STUDI MATEMATIKA
SEKOLAH SAINS DATA, MATEMATIKA, DAN INFORMATIKA
INSTITUT PERTANIAN BOGOR
BOGOR
2025**



@Hak cipta milik IPB University

IPB University

Penguji pada Ujian Skripsi:
Elis Khatizah S.Si., M.Si., Ph.D.



IPB University
— Bogor Indonesia —

Hak Cipta Dilindungi Undang-undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

Perpustakaan IPB University



Judul Skripsi : Pemodelan Simulasi dalam Bahasa Pemrograman Julia
Nama : Lely Romandhani
NIM : G54180039

Disetujui oleh

Pembimbing I:
Dra. Farida Hanum, M.Si.

Pembimbing II:
Dr. Ir. Fahren Bukhari, M.Sc.

Diketahui oleh

Ketua Program Studi Matematika
Dr. Donny Citra Lesmana, S.Si., M.Fin.Math.
NIP : 19790227 200501 1 001

Tanggal Ujian: 28 Februari 2025

Tanggal Lulus:

PRAKATA

Puji dan syukur penulis panjatkan kepada Allah *subhanaahu wa ta'ala* atas segala karunia-Nya sehingga karya ilmiah ini berhasil diselesaikan dengan judul “Pemodelan Simulasi dalam Bahasa Pemrograman Julia”. Penyusunan karya ilmiah ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada :

1. Allah SWT sebagai Tuhan yang telah memberikan segalanya,
2. Nabi Muhammad SAW sebagai utusan Allah SWT yang telah menyebarkan agama kebaikan dan kedamaian sehingga penulis dapat merasakan nikmat Islam dengan cara semudah ini,
3. Keluarga yang telah memberikan dukungan, doa dan kasih sayangnnya bagi penulis selama penyusunan karya ilmiah ini,
4. Dr. Ir. Fahren Bukhari, M.Sc. dan Dra. Farida Hanum, M.Si selaku dosen pembimbing atas segala ilmu, bimbingan, dan motivasinya selama penyusunan karya ilmiah ini,
5. Elis Khatizah, S.Si, M.Si, Ph.D. selaku dosen penguji atas motivasi, kritik, dan saran yang telah diberikan kepada penulis,
6. rekan-rekan mahasiswa matematika angkatan 55 yang telah memberikan dukungan, saran dan motivasi kepada penulis selama penyusunan karya ilmiah ini,
7. rekan-rekan lainnya yang tidak dapat dituliskan satu per satu yang telah memberikan dukungan, doa, dan saran kepada penulis.

Semoga karya ilmiah ini bermanfaat bagi pihak yang membutuhkan dan bagi kemajuan ilmu pengetahuan.

Bogor, Mei 2025

Lely Romandhani

DAFTAR ISI

DAFTAR TABEL	x
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN	xi
I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan Penelitian	2
II TINJAUAN PUSTAKA	3
2.1 Pemodelan Simulasi	3
2.2 <i>Probability Programming Language</i>	5
2.3 Distribusi Eksponensial	5
2.4 Distribusi Uniform	6
2.5 Distribusi 2-Erlang	7
2.6 Paket <i>Distributions.jl</i>	7
2.7 Paket <i>ConcurrentSim.jl</i>	9
III HASIL DAN PEMBAHASAN	11
3.1 Masalah Antrean Truk di Pertambangan	11
3.2 Masalah Antrean Pesawat pada Fasilitas Inspeksi/Perbaikan	20
IV SIMPULAN DAN SARAN	36
4.1 Simpulan	36
4.2 Saran	36
DAFTAR PUSTAKA	37
LAMPIRAN	38
RIWAYAT HIDUP	61

DAFTAR TABEL

1	Perlakuan dari setiap kapasitas truk	11
2	Hasil yang didapatkan setelah dilakukan simulasi selama waktu 8 jam	20
3	Data untuk setiap jenis pesawat	20
4	Perbandingan total biaya <i>downtime</i> harian	31
5	Uji statistika (Uji-T)	33
6	Data total biaya <i>downtime</i> harian	34

DAFTAR GAMBAR

1	<i>Script</i> untuk pemanggilan pustaka untuk simulasi antrean di Julia	13
2	<i>Script</i> untuk penetapan nilai <i>seed</i>	14
3	<i>Script</i> untuk inisialisasi struktur data ShovelStruct dan CrusherStruct dalam Julia	14
4	<i>Script</i> untuk inisialisasi generator angka acak dengan Mersenne Twister	15
5	<i>Script</i> untuk fungsi penentuan waktu <i>loading</i> dan <i>dumping</i> dengan distribusi eksponensial	15
6	<i>Script</i> untuk simulasi proses operasional truk 50 ton pada sistem <i>Shovel-Crusher</i> menggunakan Julia	16
7	<i>Script</i> untuk simulasi proses operasional truk 20 ton pada sistem <i>Shovel-Crusher</i> menggunakan Julia	17
8	<i>Script</i> untuk proses simulasi penggunaan <i>shovel</i> dan <i>crusher</i> dalam sistem transportasi material	18
9	<i>Script</i> untuk simulasi proses pengangkutan material di tambang menggunakan Julia	19
10	<i>Script</i> untuk struktur data untuk statistik simulasi antrean pesawat dalam Julia	23
11	<i>Script</i> untuk inisialisasi parameter dan generator angka acak dalam simulasi Julia	24
12	<i>Script</i> untuk fungsi pembangkitan variabel acak untuk simulasi pergerakan pesawat	25
13	<i>Script</i> untuk implementasi proses inspeksi dan perbaikan pesawat dalam simulasi	26
14	<i>Script</i> untuk fungsi generator pesawat dalam simulasi	27
15	<i>Script</i> untuk fungsi simulasi proses perawatan pesawat	27
16	<i>Script</i> untuk perhitungan statistik dan biaya <i>downtime</i> dalam simulasi perawatan pesawat	27



Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

17	<i>Script</i> untuk simulasi pemeliharaan pesawat pada <i>regular plane</i> dalam Julia	28
18	<i>Script</i> untuk simulasi pemeliharaan pesawat pada berbadan lebar (<i>widebody</i>) <i>plane</i> dalam Julia	29
19	<i>Script</i> untuk simulasi pemeliharaan pesawat (model perawatan dan analisis kinerja stasiun servis)	31
20	Grafik perbandingan total biaya <i>downtime</i> harian	32
21	Tata letak alternatif untuk fasilitas perbaikan pesawat	33

DAFTAR LAMPIRAN

1	<i>Output Software Julia</i> untuk Kasus 1	38
2	Sintaks untuk Kasus 2 menggunakan cara alternatif	57

I PENDAHULUAN

1.1 Latar Belakang

Pada abad ke-20, revolusi digital mulai berkembang, terutama dengan hadirnya komputer yang menjadi alat penting dalam kemajuan teknologi dan peradaban manusia. Kemajuan teknologi komputer yang semakin canggih mendorong peningkatan penggunaan komputasi ilmiah modern. Bahasa pemrograman seperti Python, R, *Mathematica*, Octave, Matlab, dan Scilab menjadi semakin populer. Bahasa-bahasa ini dikategorikan sebagai bahasa pemrograman dinamis yang dapat mengubah tipe datanya selama eksekusi.

Meski begitu, bahasa seperti C dan Fortran masih menjadi pilihan utama untuk komputasi berat, karena bahasa dinamis tingkat tinggi belum mampu memberikan performa sebaik C dan Fortran. Oleh karena itu, bidang komputasi teknis yang memerlukan banyak daya hitung masih sulit untuk memanfaatkan kemudahan penggunaan yang ditawarkan oleh bahasa pemrograman tingkat tinggi. (Bezanson *et al.* 2012)

Pada Februari 2012, bahasa pemrograman Julia pertama kali diluncurkan sebagai proyek *open source* (Bezanson *et al.* 2017). Julia adalah bahasa pemrograman yang lebih baru dibandingkan dengan bahasa pemrograman yang lain dalam konteks sejarah perkembangan bahasa pemrograman dan dirancang khusus untuk komputasi ilmiah dengan fitur-fitur modern yang menyesuaikan kebutuhan pengguna saat ini. Bahasa ini telah menarik banyak minat, terutama dalam komunitas kontrol, karena banyaknya paket perangkat lunak yang tersedia. Keunggulan Julia terletak pada kombinasi antara produktivitas yang tinggi dan kinerja yang cepat. Pengguna dapat dengan mudah mempelajari, memodifikasi, atau memperluas kode pustaka standarnya sesuai dengan kebutuhan masing-masing.

Bahasa pemrograman Julia sangat cepat karena desain bahasanya yang cermat dan pemilihan teknologi yang optimal. Menurut Bezanson *et al.* (2012), Julia menawarkan kinerja setara dengan bahasa pemrograman yang dikompilasi secara statis, sambil tetap memberikan fleksibilitas dinamis dan kemudahan interaktif seperti Python, LISP, atau Ruby.

Di lingkungan komputasi modern, efisiensi dalam manipulasi distribusi peluang menjadi sangat penting untuk memastikan analisis data berjalan cepat dan tepat. Oleh karena itu, perangkat lunak yang dapat menangani berbagai jenis distribusi dengan cepat menjadi kunci dalam penerapan komputasi ilmiah. Salah satu solusi yang diciptakan untuk kebutuhan ini adalah *Distributions.jl*, sebuah paket yang dikembangkan dalam bahasa pemrograman Julia. Paket ini menyediakan berbagai macam distribusi peluang, mulai dari distribusi diskret hingga distribusi kontinu, baik untuk data univariat maupun multivariat, serta distribusi matriks.

Dalam simulasi sistem dinamis yang kompleks, seperti jaringan komputer, manufaktur, atau sistem transportasi, komponen-komponen sistem sering beroperasi secara bersamaan dan saling berinteraksi. Ini menciptakan tantangan besar dalam menjaga sinkronisasi waktu antarproses agar kejadian-kejadian di dalam sistem tetap sesuai dengan urutan yang seharusnya. *ConcurrentSim.jl*,

sebuah kerangka kerja yang dikembangkan untuk Julia, hadir untuk mengatasi tantangan ini. Paket ini memanfaatkan pendekatan pemodelan berbasis simulasi peristiwa diskret yang memungkinkan simulasi komponen yang berjalan secara paralel dengan mengandalkan penjadwalan peristiwa.

Karya ilmiah ini mengacu pada dokumentasi resmi JuliaDynamics tentang *ConcurrentSim.jl*, yaitu *ConcurrentSim.jl: A Julia package for concurrent simulations*, sebagai rujukan utama. Inti dari dokumentasi ini ialah menjelaskan cara kerangka kerja *ConcurrentSim.jl* dapat digunakan untuk memodelkan sistem simulasi berbasis kejadian diskret yang berjalan secara paralel. Dokumentasi ini memaparkan fitur-fitur utama paket *ConcurrentSim.jl*, seperti penjadwalan peristiwa, sinkronisasi waktu, dan pemrosesan paralel. Hal ini memungkinkan pengembangan simulasi sistem dinamis yang efisien dengan sintaks yang sederhana dan konsisten di dalam ekosistem Julia. Dengan demikian, pengguna dapat lebih mudah memahami dan mengimplementasikan simulasi tersebut.

1.2 Tujuan Penelitian

Penelitian ini bertujuan memperkenalkan beberapa paket di dalam bahasa pemrograman Julia yang dapat digunakan untuk menyelesaikan beberapa permasalahan mengenai pemodelan simulasi.



II TINJAUAN PUSTAKA

2.1 Pemodelan Simulasi

Menurut Harrell (2004), simulasi sebagai imitasi dari suatu sistem dinamis menggunakan model komputer dengan tujuan mengevaluasi dan memperbaiki kinerja sistem. Dari perspektif simulasi, sistem terdiri dari entitas, aktivitas, sumber daya, dan kontrol. Elemen-elemen ini mendefinisikan siapa, apa, di mana, kapan, dan proses dari entitas tersebut berlangsung.

Pemodelan simulasi adalah teknik yang digunakan untuk merepresentasikan sistem nyata melalui model matematis dan menjalankan eksperimen secara virtual untuk mempelajari perilaku atau kinerja sistem tersebut. Teknik ini sering digunakan di berbagai bidang, mulai dari teknik, ekonomi, hingga sains karena dapat membantu memprediksi dan menganalisis dinamika kompleks suatu sistem tanpa harus langsung melakukan percobaan pada dunia nyata (Law dan Kelton 2014).

Pemodelan simulasi dapat dibedakan berdasarkan jenis sistem yang dimodelkan dan tujuan dari simulasi itu sendiri. Dua pendekatan umum dalam pemodelan simulasi adalah:

1. Pemodelan deterministik, yaitu model yang semua variabel dan parameter diketahui dengan pasti. Sistem berjalan sesuai dengan aturan yang telah ditentukan tanpa melibatkan elemen acak. Misalnya, model fisika Newtonian untuk memprediksi gerak benda dalam mekanika klasik.
2. Pemodelan stokastik, yaitu model yang melibatkan elemen ketidakpastian atau variabel acak. Simulasi sering kali memberikan hasil yang berbeda untuk setiap iterasi. Contoh umum termasuk simulasi Monte Carlo yang digunakan untuk memperkirakan probabilitas hasil dalam sistem yang melibatkan ketidakpastian.

Ada beberapa metode utama yang digunakan dalam pemodelan simulasi, bergantung pada karakteristik sistem yang dianalisis dan tujuan simulasi:

1. Simulasi Berbasis Agen (*Agent-Based Modeling*)
Metode ini memodelkan individu (agen) dalam sistem yang memiliki perilaku dan aturan sendiri. Interaksi antara agen dapat menghasilkan perilaku makro yang kompleks. Pendekatan ini sering digunakan dalam ekonomi, ekologi, dan studi perilaku sosial (Railsback dan Grimm 2019).
2. Simulasi Dinamika Sistem (*System Dynamics*)
Metode ini digunakan untuk menganalisis perubahan dinamis dalam sistem dengan memodelkan hubungan antara variabel-variabel penting melalui persamaan diferensial. Biasanya digunakan untuk memodelkan sistem makro seperti populasi, ekosistem, dan ekonomi (Sterman 2000).
3. Simulasi Diskret (*Discrete Event Simulation*)
Metode ini digunakan untuk memodelkan sistem yang berubah pada titik waktu tertentu karena peristiwa diskret. Contohnya termasuk simulasi antrean atau pemodelan proses manufaktur (Law dan Kelton 2014).

Pemodelan simulasi terdiri dari beberapa langkah kunci yang harus diikuti untuk menghasilkan model yang valid dan dapat diandalkan (Law dan Kelton 2014; Banks *et al.* 2005), yaitu

1. **Formulasi Masalah**
Langkah ini mencakup pendefinisian tujuan simulasi, sistem yang dimodelkan, serta parameter dan variabel kunci.
2. **Pengembangan Model**
Pada tahap ini, model matematis atau komputasional berdasarkan asumsi yang relevan, hubungan antara variabel, dan data yang tersedia.
3. **Verifikasi dan Validasi**
Verifikasi memastikan bahwa model telah dibangun dengan benar, sedangkan validasi memastikan bahwa model cukup akurat untuk merepresentasikan sistem nyata.
4. **Eksperimen Simulasi**
Eksperimen dilakukan dengan menjalankan simulasi untuk berbagai skenario, kemudian hasilnya diamati untuk memahami perilaku sistem.
5. **Analisis Hasil**
Hasil simulasi dianalisis untuk mendapatkan wawasan tentang perilaku sistem dan membuat keputusan yang berdasarkan data.

Pemodelan simulasi memiliki aplikasi yang luas di berbagai bidang, termasuk industri dan manufaktur, ekonomi dan keuangan, sistem kesehatan, serta transportasi dan logistik. Dalam industri dan manufaktur, simulasi digunakan untuk mengoptimalkan efisiensi proses produksi, mengurangi waktu produksi, serta meningkatkan kualitas produk. Dengan model simulasi, perusahaan dapat mengidentifikasi hambatan produksi dan menguji berbagai skenario sebelum menerapkannya di dunia nyata. Di bidang ekonomi dan keuangan, pemodelan simulasi membantu dalam memprediksi perilaku pasar, menganalisis risiko investasi, dan mengembangkan strategi portofolio yang optimal. Simulasi Monte Carlo, misalnya, sering digunakan dalam perhitungan risiko dan perencanaan keuangan untuk mengantisipasi ketidakpastian dalam pasar global.

Dalam sistem kesehatan, simulasi memainkan peran penting dalam memodelkan alur pasien di rumah sakit, perencanaan sumber daya medis, serta evaluasi dampak kebijakan kesehatan baru. Misalnya, simulasi dapat digunakan untuk mengoptimalkan penjadwalan dokter dan ketersediaan ruang perawatan guna mengurangi waktu tunggu pasien. Sementara itu, dalam transportasi dan logistik, simulasi digunakan untuk merancang jaringan transportasi, mengoptimalkan rute pengiriman, dan memodelkan lalu lintas. Dengan menggunakan simulasi, operator transportasi dapat mengidentifikasi titik kemacetan, menguji kebijakan lalu lintas, serta meningkatkan efisiensi distribusi barang dalam rantai pasok global.

Pemodelan simulasi adalah alat yang kuat untuk memprediksi, menganalisis, dan memahami sistem yang kompleks. Dengan pendekatan dan metode yang tepat, simulasi memungkinkan pengambilan keputusan yang lebih baik dalam menghadapi ketidakpastian dan variasi di dunia nyata. Sebagai bagian dari perkembangan teknologi, simulasi terus berkembang menggunakan algoritma canggih dan kekuatan komputasi yang semakin meningkat, menjadikannya semakin relevan dalam banyak aspek kehidupan.

2.2 Probability Programming Language

Probabilistik mengacu pada pendekatan yang melibatkan teori peluang dalam menganalisis ketidakpastian suatu peristiwa atau sistem (Grinstead & Snell 1997). Menurut Jaynes (2003), probabilistik adalah cara untuk mengukur dan mengelola ketidakpastian menggunakan teori probabilitas sebagai alat formal untuk inferensi dan pengambilan keputusan. Menurut Gordon *et al.* (2014), program probabilistik adalah program komputer dengan kemampuan tambahan untuk menarik nilai suatu variabel dari distribusi probabilitas dan mengkondisikan nilai variabel pada pengamatan. Mereka memungkinkan spesifikasi hubungan kompleks antara variabel acak menggunakan satu set konstruksi yang melampaui model grafis khas untuk memasukkan kontrol aliran (misalnya, *loop*, kondisi), rekursi, tipe yang ditentukan pengguna dan blok bangunan algoritmik lainnya, berbeda dari grafik eksplisit konstruksi oleh pengguna, seperti yang dilakukan misalnya di Smith (2018).

Bahasa pemrograman probabilistik adalah bahasa pemrograman atau perpustakaan yang memungkinkan pengembang untuk menulis program probabilistik tersebut. Artinya, mereka adalah jenis khusus domain (Fowler 2010). Bahasa pemrograman probabilistik adalah bahasa pemrograman yang dirancang untuk menangani model probabilistik, statistik, dan inferensi. Bahasa ini sering digunakan dalam analisis data, pemodelan matematis, dan simulasi untuk mengoptimalkan pemrosesan informasi berbasis probabilitas.

Salah satu keunggulan bahasa pemrograman probabilistik terletak pada kemampuan modelisasi, yang memudahkan pengguna dalam mendefinisikan dan mengimplementasikan model probabilistik yang kompleks. Hal ini membuat proses modelisasi menjadi lebih efisien dan intuitif, terutama dalam analisis data besar dan simulasi probabilistik. Selain itu, bahasa ini menyediakan alat inferensi statistik yang memungkinkan pengguna melakukan estimasi parameter, analisis data, serta pengambilan keputusan berbasis probabilitas secara efektif. Bahasa pemrograman probabilistik juga menawarkan fleksibilitas tinggi, karena mendukung berbagai metode inferensi, baik pendekatan *Bayesian* maupun *Frequentist*. Dengan demikian, bahasa ini dapat diterapkan dalam berbagai konteks, seperti pembelajaran mesin, pemodelan risiko, serta analisis data eksperimental.

2.3 Distribusi Eksponensial

Distribusi eksponensial adalah distribusi probabilitas kontinu yang umum digunakan untuk memodelkan waktu antar kejadian dalam proses acak, seperti kedatangan pelanggan atau waktu hingga kegagalan alat. Distribusi ini banyak diterapkan dalam manajemen operasi dan teori antrean, terutama untuk memprediksi waktu tunggu dan durasi layanan dalam sistem antrean, seperti di perbankan dan layanan publik. Selain itu, distribusi eksponensial juga berperan penting dalam analisis reliabilitas dan kegagalan sistem. Model ini digunakan untuk memprediksi waktu hingga kegagalan suatu perangkat jika tingkat kegagalan dianggap konstan. Hal ini sangat berguna dalam industri manufaktur dan pemeliharaan prediktif untuk memperkirakan umur pakai suatu produk atau komponen elektronik.

Dalam bidang telekomunikasi, distribusi eksponensial sering digunakan untuk memodelkan durasi panggilan telepon serta waktu antar kedatangan paket data di jaringan komputer. Model ini membantu dalam analisis lalu lintas jaringan dan optimalisasi alokasi sumber daya dalam sistem komunikasi.

Selain itu, distribusi eksponensial juga memiliki peran penting dalam simulasi Monte Carlo, yang digunakan untuk menghasilkan waktu antar kejadian secara acak dalam berbagai eksperimen simulasi. Aplikasi ini sangat bermanfaat dalam berbagai bidang, termasuk ekonomi, rekayasa sistem, dan optimasi proses industri.

Distribusi eksponensial merupakan alat yang sangat berguna dalam analisis statistik dan probabilitas, terutama untuk memodelkan waktu antar kejadian dalam berbagai proses acak. Sifat *memory-less* dan hubungan erat dengan distribusi Poisson menjadikannya relevan dalam berbagai bidang, termasuk manajemen operasi, telekomunikasi, dan analisis reliabilitas. Dengan menggunakan perangkat lunak seperti Julia, Python, atau R, distribusi eksponensial dapat dengan mudah diimplementasikan untuk analisis data dan simulasi.

2.4 Distribusi Uniform

Distribusi uniform adalah salah satu distribusi probabilitas yang paling sederhana dalam statistik. Distribusi ini mencerminkan probabilitas yang seragam, dalam hal ini semua nilai dalam rentang tertentu memiliki peluang yang sama untuk terjadi. Dalam probabilitas, distribusi uniform dapat dibagi menjadi dua jenis utama diskret dan kontinu (Ross 2014).

Distribusi uniform adalah salah satu distribusi probabilitas paling sederhana dan sering digunakan dalam analisis statistik (Law & Kelton 2014; Ross 2014). Sifat utamanya adalah probabilitas seragam, di mana semua hasil dalam rentang tertentu memiliki peluang yang sama, menjadikannya dasar bagi banyak model probabilistik. Distribusi uniform kontinu juga memiliki hubungan erat dengan distribusi lain, terutama dalam simulasi Monte Carlo, karena dapat digunakan untuk menghasilkan angka acak yang dikonversi ke distribusi lain. Selain itu, kesederhanaan model membuatnya sering dijadikan asumsi awal dalam pemodelan statistik dan simulasi stokastik.

Distribusi uniform banyak digunakan dalam simulasi Monte Carlo, pemodelan sistem, eksperimen statistik, dan pemrograman komputasi (Law & Kelton 2014; Montgomery & Runger 2014). Dalam simulasi Monte Carlo, distribusi ini menghasilkan angka acak untuk analisis risiko dan perencanaan stokastik. Dalam pemodelan sistem, distribusi uniform memodelkan waktu antar kejadian dalam sistem antrean atau manufaktur saat data tidak tersedia. Selain itu, dalam eksperimen statistik, distribusi uniform digunakan untuk pemilihan sampel acak, sedangkan dalam pemrograman komputasi, banyak generator angka acak dasar menggunakan distribusi ini untuk simulasi dan pemodelan.

Distribusi uniform adalah alat yang sederhana namun kuat dalam probabilitas dan statistik. Dengan aplikasi luas dalam simulasi dan pemodelan, distribusi ini menawarkan pendekatan awal yang fleksibel untuk menganalisis sistem dinamis atau menghasilkan angka acak. Berkat kemudahan implementasinya, distribusi uniform telah menjadi bagian penting dari perangkat alat ilmuwan data dan peneliti di berbagai bidang.

2.5 Distribusi 2-Erlang

Distribusi Erlang adalah distribusi probabilitas kontinu yang sering digunakan untuk memodelkan waktu antar kejadian dalam proses Poisson. Distribusi ini merupakan kasus khusus dari distribusi gamma dengan parameter bentuknya (k) adalah bilangan bulat positif. Distribusi 2-Erlang adalah distribusi Erlang dengan parameter $k = 2$, yang menunjukkan bahwa distribusi ini adalah hasil penjumlahan dua variabel acak eksponensial identik dan independen (Ross 2014).

Distribusi 2-Erlang digunakan secara luas dalam berbagai aplikasi, termasuk teori antrean, reliabilitas sistem, dan telekomunikasi, dengan waktu tunggu atau waktu pemrosesan sering kali memiliki pola yang menyerupai distribusi ini. Aplikasi distribusi Erlang antara lain:

1. Teori Antrean: Distribusi ini digunakan untuk memodelkan waktu layanan dalam sistem antrean di mana proses layanan memiliki dua tahapan independen.
2. Reliabilitas Sistem: Dalam analisis reliabilitas, distribusi 2-Erlang digunakan untuk memodelkan waktu hingga kegagalan sistem dengan dua komponen yang bekerja secara seri.
3. Telekomunikasi: Distribusi ini digunakan dalam pemodelan waktu antar kedatangan paket data dalam jaringan komunikasi atau durasi panggilan telepon.
4. Simulasi Monte Carlo: Dalam simulasi stokastik, distribusi 2-Erlang sering digunakan untuk menghasilkan data waktu antar kejadian dengan pola tertentu.

Distribusi Erlang memiliki beberapa keunggulan dan keterbatasan dalam pemodelan probabilistik. Salah satu keunggulannya adalah fleksibilitas yang lebih tinggi dibandingkan dengan distribusi eksponensial, karena mampu memodelkan waktu antar kejadian dengan variabilitas yang lebih besar. Selain itu, distribusi ini mudah diimplementasikan dalam perangkat lunak statistik seperti Julia, Python, dan R, sehingga banyak digunakan dalam berbagai analisis stokastik. Namun, distribusi Erlang juga memiliki keterbatasan, terutama dalam pemodelan waktu antar kejadian yang pola distribusinya jauh dari eksponensial atau melibatkan lebih dari dua tahap pemrosesan. Dalam kasus seperti itu, distribusi lain, seperti gamma atau log-normal, mungkin lebih sesuai untuk digunakan.

Distribusi 2-Erlang adalah alat penting dalam probabilitas dan statistik untuk memodelkan waktu antar kejadian dalam sistem yang melibatkan dua tahap proses. Dengan sifat matematisnya yang sederhana dan aplikasi luasnya, distribusi ini terus menjadi pilihan utama dalam berbagai bidang seperti teori antrean, reliabilitas, dan telekomunikasi.

2.6 Paket *Distributions.jl*

Paket *Distributions.jl* adalah pustaka yang penting dalam ekosistem Julia untuk pemodelan distribusi probabilitas, baik yang diskret maupun kontinu. Paket ini menawarkan fungsionalitas untuk mengakses, memanipulasi, dan melakukan analisis distribusi probabilitas yang beragam dengan efisiensi tinggi. Tulisan ini membahas arsitektur dasar *Distributions.jl*, fitur utamanya, implementasi

matematis, dan aplikasi dalam berbagai bidang seperti statistika, analisis risiko, dan pemodelan stokastik.

Distributions.jl telah menjadi dasar untuk pemrograman statistik di Julia, dan banyak digunakan dalam pemodelan ekonomi (FRBONY 2019; Lyon *et al.* 2017), simulasi rantai Markov Monte Carlo (Smith 2018) atau optimasi *black-box* acak Feldt (2017).

Distribusi dalam *Distributions.jl* diimplementasikan menggunakan prinsip-prinsip probabilitas yang kuat dan algoritma efisien untuk kalkulasi fungsi PDF (*Probability Density Function*) dan CDF (*Cumulative Distribution Function*). Misalnya:

1. Distribusi Diskret: Untuk distribusi seperti binomial atau Poisson, PDF (*Probability Density Function*) dihitung berdasarkan fungsi faktorial atau fungsi gamma yang dioptimalkan. Algoritma sampling diskret yang cepat diintegrasikan untuk pengambilan sampel secara acak.
2. Distribusi Kontinu: Distribusi kontinu seperti distribusi normal atau Gamma memanfaatkan algoritma pengambilan sampel Gaussian Box-Muller atau Metropolis-Hastings untuk menghasilkan angka acak yang mengikuti distribusi target.
3. Distribusi Multivariat: Paket ini juga mendukung distribusi multivariat, seperti distribusi normal multivariat, di mana matriks kovarians digunakan untuk menangani korelasi antar variabel.

Setiap distribusi diimplementasikan sebagai objek, dan distribusi gabungan (*mixture*) atau transformasi distribusi dapat dilakukan dengan mudah.

Paket *Distributions.jl* digunakan dalam berbagai aplikasi ilmiah dan teknis yang melibatkan probabilitas dan statistika (Bezanson *et al.* 2017) antara lain;

1. Simulasi Monte Carlo
Dalam analisis risiko dan perencanaan stokastik, distribusi probabilitas sering digunakan untuk mensimulasikan skenario berdasarkan ketidakpastian input data. *Distributions.jl* memfasilitasi pengambilan sampel acak dari berbagai distribusi untuk menjalankan simulasi Monte Carlo yang efisien.
2. Pemodelan Bayesian
Paket ini sangat berguna dalam pemodelan Bayesian, di mana distribusi prior dan posterior dapat dimodelkan dan diintegrasikan menggunakan distribusi probabilitas yang disediakan oleh *Distributions.jl*. Integrasi dengan *Turing.jl* memungkinkan pelaksanaan inferensi Bayesian yang lebih mendalam (Gelman *et al.* 2013).
3. Finansial dan Ekonomi
Dalam aplikasi finansial, distribusi probabilitas digunakan untuk memodelkan risiko dan volatilitas. *Distributions.jl* menyediakan alat yang diperlukan untuk menghitung distribusi keuntungan, kerugian, dan estimasi risiko menggunakan distribusi probabilitas yang relevan, seperti log-normal atau Pareto.
4. Statistika dan Data Sains
Paket ini digunakan untuk estimasi parameter distribusi dari data yang ada, serta pengujian hipotesis berbasis distribusi, baik untuk populasi diskret maupun kontinu.

Menurut Bezanson *et al.* (2017), *Distributions.jl* memiliki kinerja tinggi berkat JIT compilation, memungkinkan performa mendekati bahasa berkompilasi seperti C. Paket ini juga menawarkan kemudahan penggunaan dengan sintaksis

intuitif memudahkan ilmuwan data dan peneliti dalam menerapkan distribusi probabilitas tanpa harus memahami algoritma komputasi secara mendalam. Keunggulan lainnya adalah integrasi yang kuat, karena *Distributions.jl* dapat digunakan bersama berbagai pustaka dan alat dalam ekosistem Julia, sehingga meningkatkan fleksibilitas dalam eksperimen dan pengembangan aplikasi. Namun, terdapat keterbatasan, seperti cakupan distribusi yang terbatas, dalam hal ini beberapa distribusi khusus tidak tersedia dalam pustaka standar, serta kurva pembelajaran Julia, yang memerlukan waktu adaptasi bagi pengguna baru.

Distributions.jl adalah salah satu paket penting dalam ekosistem Julia untuk pemodelan distribusi probabilitas. Dengan menyediakan berbagai distribusi probabilitas, dari yang sederhana hingga yang kompleks, serta kemampuan sampling yang cepat dan efisien, paket ini sangat cocok untuk aplikasi-aplikasi yang memerlukan analisis probabilitas dan statistik intensif. Dengan kecepatan dan fleksibilitas yang ditawarkan oleh Julia, *Distributions.jl* memberikan solusi efisien bagi peneliti dan praktisi di berbagai bidang yang melibatkan ketidakpastian dan variabilitas data.

2.7 Paket *ConcurrentSim.jl*

ConcurrentSim.jl adalah pustaka Julia yang dirancang untuk memfasilitasi simulasi sistem dinamis yang berjalan secara paralel dan serentak. Pustaka ini sangat berguna dalam memodelkan dan menganalisis sistem yang melibatkan interaksi kompleks antara berbagai komponen yang beroperasi secara asinkron. Artikel ini membahas arsitektur *ConcurrentSim.jl*, fitur utama yang mendukung paralelisme, serta aplikasi dalam berbagai domain, termasuk jaringan komunikasi, sistem manufaktur, dan simulasi lalu lintas.

ConcurrentSim.jl bekerja berdasarkan prinsip pemodelan peristiwa diskret, di mana setiap entitas dalam sistem beroperasi secara independen sesuai dengan peristiwa yang terjadi (Law & Kelton 2014). Simulasi ini mencakup tiga komponen utama: penjadwalan peristiwa, pemrosesan paralel, dan sinkronisasi waktu.

Dalam penjadwalan peristiwa, entitas berinteraksi melalui peristiwa yang dijadwalkan dalam antrean waktu yang dioptimalkan untuk menangani banyak peristiwa secara simultan. Eksekusinya menggunakan *heap-based priority queue* agar peristiwa diproses sesuai urutan waktu. Pemrosesan paralel memanfaatkan kemampuan Julia untuk menjalankan peristiwa dari berbagai entitas secara bersamaan, selama tidak ada ketergantungan waktu, sehingga meningkatkan efisiensi simulasi pada perangkat keras yang mendukung paralelisme. Sinkronisasi waktu diterapkan melalui skema *time-stepping*, di mana setiap entitas memiliki lingkup waktunya sendiri tetapi tetap terhubung dalam sistem waktu global guna menjaga konsistensi simulasi.

Paket *ConcurrentSim.jl* telah digunakan dalam berbagai aplikasi, terutama yang melibatkan simulasi sistem dinamis yang berjalan secara paralel (Law dan Kelton 2014; Railsback dan Grimm 2019). Beberapa aplikasi utama meliputi:

1. Jaringan Komunikasi

Dalam pemodelan jaringan komputer, paket ini dapat digunakan untuk menyimulasikan pengiriman paket data melalui jaringan, termasuk peristiwa seperti penundaan paket, kehilangan paket, dan penyusunan ulang paket. Hal

ini sangat berguna dalam simulasi protokol komunikasi dan analisis performa jaringan.

2. Sistem Manufaktur

Dalam lingkungan manufaktur, mesin dan sumber daya bekerja secara bersamaan, *ConcurrentSim.jl* dapat digunakan untuk mengatur simulasi jalur produksi dan alokasi sumber daya yang optimal.

3. Simulasi Lalu Lintas

Pada sistem transportasi, seperti simulasi lalu lintas kendaraan di kota, paket ini membantu memodelkan interaksi antar kendaraan dan pengaturan sinyal lalu lintas, sehingga memungkinkan analisis terhadap efisiensi lalu lintas dan pengurangan kemacetan.

4. Simulasi Agen Berbasis Proses (*Agent-Based Simulation*)

Sistem yang melibatkan entitas agen yang bekerja secara otonom dapat disimulasikan menggunakan *ConcurrentSim.jl*, termasuk dalam bidang sosial, ekonomi, dan biologi.

Menurut Law dan Kelton (2014) serta JuliaLang Documentation, *ConcurrentSim.jl* memiliki beberapa keunggulan dan keterbatasan dalam pemodelan simulasi berbasis peristiwa diskret. Dari segi keunggulan, paket ini menawarkan kinerja tinggi dengan dukungan paralelisme, memungkinkan eksekusi peristiwa secara simultan, terutama dalam simulasi skala besar. Selain itu, pemodelan *asynchronous* dalam sistem berbasis peristiwa memungkinkan penggunaan sumber daya yang lebih efisien dengan hanya memproses peristiwa yang diperlukan, sehingga mengurangi overhead komputasi dibandingkan simulasi waktu kontinu. Fleksibilitas pemodelan juga menjadi keunggulan utama, karena *ConcurrentSim.jl* mendukung simulasi berbagai jenis sistem, mulai dari diskret hingga kompleks, dengan pendekatan berbasis entitas dan penjadwalan peristiwa yang adaptif.

Namun, paket ini juga memiliki beberapa keterbatasan. Salah satunya adalah kurva pembelajaran yang curam, di mana pengguna baru mungkin menghadapi tantangan dalam memahami konsep seperti penjadwalan peristiwa, paralelisme, dan sinkronisasi waktu. Selain itu, dalam simulasi kecil dengan sedikit entitas atau peristiwa, overhead paralelisme dan sinkronisasi waktu dapat mengurangi efisiensi, membuatnya lebih lambat dibandingkan simulasi sekuensial.

ConcurrentSim.jl adalah alat yang kuat untuk menyimulasikan sistem dinamis yang berjalan secara paralel dan serentak. Dengan dukungan penuh terhadap pemodelan asinkron dan berbasis peristiwa, paket ini memungkinkan pengguna untuk melakukan simulasi skala besar dengan efisien. Berkat dukungan paralelisme dan sinkronisasi waktu, *ConcurrentSim.jl* sangat cocok untuk aplikasi dalam simulasi jaringan, manufaktur, lalu lintas, dan banyak lagi. Meski ada tantangan dalam mempelajari penggunaannya, potensi *ConcurrentSim.jl* untuk meningkatkan kinerja simulasi membuatnya menjadi alat penting dalam dunia komputasi ilmiah dan teknik.

III HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas mengenai penggunaan paket-paket dalam bahasa pemrograman Julia yang akan digunakan untuk menyelesaikan beberapa contoh permasalahan yang akan disajikan. Contoh permasalahan yang akan dibahas meliputi dua topik, yaitu masalah antrean truk-truk di sebuah pertambangan yang mengantre untuk mengangkut material dari alat gali ke mesin penghancur dan masalah antrean pesawat yang akan melakukan inspeksi terhadap mesin pada setiap pesawat di layanan servis. Paket-paket bahasa pemrograman Julia yang akan digunakan antara lain meliputi, *ConcurrentSim.jl*, *Distributions.jl*, *Random.jl*, dan *ResumableFunctions.jl*. Kedua permasalahan ini dipilih untuk memperlihatkan cara sintaks tersebut diterapkan sesuai dengan alur masing-masing. Selain itu, pembahasan dua permasalahan memberikan referensi lebih luas bagi pembaca yang mungkin membutuhkan contoh sintaks yang serupa dengan skenario berbeda.

3.1 Masalah Antrean Truk di Pertambangan

Di sebuah tambang, truk-truk mengangkut batubara dari tiga alat gali (*shovels*) ke satu mesin penghancur (*crusher*). Setiap truk ditugaskan ke alat gali tertentu sehingga truk selalu kembali ke alat gali yang ditugaskan setelah pengosongan muatan di mesin penghancur. Ada dua kapasitas truk yang digunakan, yaitu 20 ton dan 50 ton. Kapasitas truk memengaruhi waktu pemuatan di alat gali, waktu perjalanan ke mesin penghancur, waktu pengosongan di mesin penghancur, dan waktu perjalanan kembali dari mesin penghancur ke alat gali, seperti yang dijelaskan pada Tabel 1 (semua waktu dalam satuan menit).

Tabel 1 Perlakuan dari setiap kapasitas truk

Waktu	Kapasitas Truk	
	20 ton	50 ton
Memuat	Terdistribusi eksponensial dengan rata-rata 5	Terdistribusi eksponensial dengan rata-rata 10
Perjalanan	Konstan 2.5	Konstan 3
Mengosongkan	Terdistribusi eksponensial dengan rata-rata 2	Terdistribusi eksponensial dengan rata-rata 4
Kembali	Konstan 1.5	Konstan 2

Untuk setiap alat gali, ditugaskan dua truk berkapasitas 20 ton dan satu truk berkapasitas 50 ton. Antrean di alat gali semuanya diatur dengan prinsip FIFO (*First In, First Out*), sedangkan antrean di mesin penghancur diatur berdasarkan kapasitas truk dengan urutan menurun, dengan aturan FIFO jika terdapat kesamaan. Dengan asumsi bahwa pada waktu 0, semua truk berada di alat gali masing-masing, dan truk 50 ton baru mulai dimuat, model simulasi dijalankan selama 8 jam untuk mengestimasi rata-rata waktu antrean di setiap alat gali dan mesin penghancur. Selain itu, waktu penggunaan keempat peralatan yang terlibat juga diestimasi. Gunakan *streams* 1 dan 2 untuk waktu pemuatan truk 20 ton dan 50 ton, dan *streams*

3 dan 4 untuk waktu pengosongan truk 20 ton dan 50 ton. Stream digunakan untuk menunjukkan bahwa angka acak yang digunakan dalam simulasi harus dikelompokkan ke dalam kategori yang berbeda, agar setiap proses (pemuatan dan pengosongan truk) memiliki sumber variabilitasnya sendiri. (Law & Kelton 2014)

Sistem yang disimulasikan adalah sistem antrean di tambang dengan dua jenis truk yang berbeda kapasitasnya (20 ton dan 50 ton) yang memuat batubara dari alat gali (*shovel*) ke penghancur (*crusher*). Berikut adalah pendekatan model matematikanya untuk merumuskannya sebagai model simulasi antrean dengan beberapa komponen.

3.1.1 Model Simulasi Antrean Truk di Alat Gali dan Penghancur

1. Parameter Waktu Proses

Parameter waktu dalam simulasi meliputi beberapa tahap, yaitu waktu memuat, waktu perjalanan, waktu mengosongkan muatan, dan waktu kembali ke alat gali.

- Waktu memuat (*loading time (L)*) berdistribusi eksponensial yang dituliskan sebagai:
Truk 20 ton: $L_{20} \sim \text{Exponential}(5)$
Truk 50 ton: $L_{50} \sim \text{Exponential}(10)$
- Waktu perjalanan ke penghancur (*travel time (T)*):
Truk 20 ton: $T_{20} = 2.5$ menit
Truk 50 ton: $T_{50} = 3$ menit
- Waktu mengosongkan muatan di penghancur (*dumping time (U)*) berdistribusi eksponensial yang dituliskan sebagai:
Truk 20 ton: $U_{20} \sim \text{Exponential}(2)$
Truk 50 ton: $U_{50} \sim \text{Exponential}(4)$
- Waktu kembali ke alat gali (*return time (R)*):
Truk 20 ton: $R_{20} = 1.5$ menit
Truk 50 ton: $R_{50} = 2$ menit

dengan :

n_{20} : Banyaknya truk 20 ton

n_{50} : Banyaknya truk 50 ton

S_i : Alat gali ke- i

Q_{S_i} : Panjang antrean di alat gali ke- i

Q_C : Panjang antrean di penghancur C

2. Formulasi Antrean

Antrean dalam sistem ini memiliki beberapa karakteristik utama, yaitu

a. Waktu Layanan (*Service Time*) di alat gali dan penghancur:

Waktu total (W) yang dibutuhkan truk untuk menyelesaikan satu siklus dapat dihitung sebagai berikut:

- Truk 20 ton: $W_{20} = L_{20} + T_{20} + U_{20} + R_{20}$
- Truk 50 ton: $W_{50} = L_{50} + T_{50} + U_{50} + R_{50}$

b. Antrean dan Prioritas:

- Antrean di alat gali diatur berdasarkan FIFO (*First In, First Out*).
- Antrean di penghancur diatur berdasarkan prioritas kapasitas truk (truk 50 ton lebih diprioritaskan). Jika kapasitas truk sama, digunakan aturan FIFO.

c. Tingkat Utilisasi Peralatan (ρ):

Utilisasi alat gali dan penghancur dapat dihitung sebagai:

- Utilisasi alat gali ke- i

$$\rho S_i = \frac{\text{waktu sibuk alat gali } i}{\text{total waktu simulasi}}$$

- Utilisasi mesin penghancur

$$\rho C = \frac{\text{waktu sibuk penghancur}}{\text{total waktu simulasi}}$$

d. Estimasi Rata-rata Waktu Antrean:

- Waktu rata-rata truk menunggu di antrean alat gali:

$$\bar{W}_{Q_{S_i}} = \frac{\text{total waktu tunggu di alat gali } i}{n_{20} + n_{50}}$$

- Waktu rata-rata truk menunggu di antrean penghancur:

$$\bar{W}_{Q_C} = \frac{\text{total waktu tunggu di mesin penghancur}}{n_{20} + n_{50}}$$

3. Tujuan Simulasi

Simulasi dilakukan selama 8 jam untuk menghitung:

- a. rata-rata waktu antrean di setiap alat gali dan di mesin penghancur.
- b. rata-rata penggunaan (utilisasi) dari alat gali dan mesin penghancur.

Model matematika ini selanjutnya dapat diterapkan pada bahasa pemrograman Julia menggunakan paket *ConcurrentSim.jl* untuk menjalankan simulasi dan mendapatkan hasil estimasi yang diinginkan.

3.1.2 Penulisan Sintaks

Pada subbab ini, akan dijelaskan penulisan sintaks yang digunakan dalam membangun model simulasi antrean truk di alat gali dan mesin penghancur.

[24]:

```
using ConcurrentSim
using Distributions
using ResumableFunctions
using Random
```

Gambar 1 Script untuk pemanggilan pustaka

Pada Gambar 1, ditampilkan perintah **using** dalam bahasa pemrograman Julia, yang digunakan untuk memuat pustaka (*package*) dan diperlukan dalam simulasi. Berikut adalah fungsi dari setiap pustaka yang diimpor:

1. `ConcurrentSim` digunakan untuk membangun simulasi berbasis peristiwa diskret.
2. `Distributions` memungkinkan penggunaan berbagai distribusi probabilitas untuk memodelkan variabel acak dalam simulasi.
3. `ResumableFunctions` memfasilitasi pembuatan fungsi yang dapat dihentikan sementara dan dilanjutkan kembali, yang berguna dalam simulasi berbasis peristiwa.
4. `Random` digunakan untuk menghasilkan angka acak yang diperlukan dalam pemodelan stokastik.

Gambar ini menunjukkan langkah awal dalam implementasi simulasi antrian menggunakan Julia. Dengan memuat pustaka-pustaka ini, pengguna dapat mengakses berbagai fungsi yang diperlukan untuk membangun model simulasi secara efisien.

Paket *ResumableFunctions* bisa digunakan untuk mengelola eksekusi asinkron, misalnya, jika sebuah proses harus menunggu hingga suatu kondisi terpenuhi sebelum dilanjutkan. Dalam simulasi kejadian diskret, seperti simulasi antrian truk di tambang, setiap truk dapat diwakili oleh sebuah proses yang berhenti pada titik-titik tertentu selama proses pemuatan, pengangkutan, atau pengosongan. Dengan menggunakan *ResumableFunctions*, proses-proses tersebut dapat ditunda ketika truk menunggu alat gali atau mesin penghancur menjadi tersedia, dan dilanjutkan kembali ketika sumber daya tersebut siap.

[25]:

```
const seed_1=150
const seed_2=200
const seed_3=500
const seed_4=600
```

Gambar 2 Script untuk penetapan nilai *seed*

Pada Gambar 2, misal pada `const seed_1 = 150`. Di sini, `seed_1` adalah nama variabel yang diberi nilai 150. Dengan memberikan nilai 150 (atau ditentukan secara sembarang) ke `seed_1`, untuk menetapkan nilai awal yang akan digunakan oleh generator angka acak. Kata kunci `const` digunakan untuk menunjukkan bahwa nilai variabel tersebut akan tetap konstan.

Dalam konteks *random number generation*, *seed* adalah nilai awal yang digunakan untuk menginisialisasi algoritma *pseudo-random number generator* (PRNG). PRNG menghasilkan urutan angka yang tampak acak, tetapi sebenarnya dihasilkan oleh algoritma deterministik yang bergantung pada nilai awal atau "*seed*". Dengan menggunakan *seed* yang sama, PRNG akan menghasilkan urutan bilangan acak yang sama setiap kali dijalankan. Ini penting dalam banyak eksperimen ilmiah atau simulasi komputer, di mana memungkinkan untuk mengulang eksperimen yang sama dengan hasil yang identik.

[26]:

```
mutable struct ShovelStruct
    shovel::Resource
    busyTime::Float64
    nTruck::Int64
    nLen::Int64
    areaUnderQ::Float64
    delayTime::Float64
    function ShovelStruct(sim::Simulation)
        ss = new()
        ss.nLen=0
        ss.areaUnderQ = 0
        ss.busyTime=0
        ss.nTruck = 0
        ss.delayTime=0
        ss.shovel = Resource(sim,1)
        return ss
    end
end
```

```
mutable struct CrusherStruct
    nTruck::Int64
    busyTime::Float64
    busy::Bool
    crusher::Resource
    nLen::Int64
    delayTime::Float64
    areaUnderQ::Float64
    function CrusherStruct(sim::Simulation)
        cs = new()
        cs.busyTime = 0
        cs.busy=false
        cs.nTruck = 0
        cs.crusher = Resource(sim,1)
        cs.areaUnderQ=0.0
        cs.delayTime=0
        return cs
    end
end
```

Gambar 3 *Script* untuk inialisasi struktur data ShovelStruct dan CrusherStruct dalam Julia

Proses yang diperlihatkan pada Gambar 3 digunakan untuk mencatat statistika dari alat gali (*shovel*) dan penghancur (*crusher*).

```
[27]: rng_loading_20 = MersenneTwister(seed_1)
      rng_loading_50 = MersenneTwister(seed_2)
      rng_dumping_20 = MersenneTwister(seed_3)
      rng_dumping_50 = MersenneTwister(seed_4)
```

Gambar 4 *Script* untuk inialisasi generator angka acak dengan Mersenne Twister

Variabel `rng_loading_20` pada Gambar 4 menggunakan `MersenneTwister(seed_1)` yang menginisialisasi generator acak dengan *seed* tertentu (misalnya `seed_1`) memastikan bahwa bilangan acak yang dihasilkan akan sama setiap kali kode tersebut dijalankan. `MersenneTwister` adalah salah satu algoritma *pseudo-random number generator* (PRNG) yang paling populer dan sering digunakan dalam pemrograman komputer, terutama untuk simulasi dan aplikasi yang membutuhkan bilangan acak. Bahasa pemrograman Julia menggunakan `MersenneTwister` sebagai salah satu algoritma *default* untuk menghasilkan bilangan acak. `MersenneTwister` diimplementasikan sebagai salah satu dari beberapa pilihan *random number generator* dalam Julia.

```
lengthOfSimulation = 8*60

function getLoadingTime_20()
    # exponentially distributed with mean 5
    return 5*randexp(rng_loading_20,Float64)
end

function getLoadingTime_50()
    # exponentially distributed with mean 10
    return 10*randexp(rng_loading_50,Float64)
end

function getDumpingTime_20()
    # exponentially distributed with mean 2
    return 2*randexp(rng_dumping_20,Float64)
end

function getDumpingTime_50()
    # exponentially distributed with mean 4
    return 4*randexp(rng_dumping_50,Float64)
end
```

Gambar 5 *Script* untuk fungsi penentuan waktu *loading* dan *dumping* dengan distribusi eksponensial

Script yang ada pada Gambar 5, digunakan untuk membangkitkan bilangan acak yang menyebar secara eksponensial dengan masing-masing nilai rata-rata yang telah ditentukan.

Makro `@resumable` di `ConcurrentSim.jl` memungkinkan penulisan fungsi proses yang dapat dijeda (*resumable functions*) dan dilanjutkan kembali pada titik tertentu dalam sebuah simulasi. Makro `@resumable` di `ConcurrentSim.jl` digunakan untuk mendefinisikan proses dalam simulasi kejadian diskret. Proses ini bekerja dengan menghasilkan (*yielding*) objek yang disebut *Event*, yang menunjukkan kejadian tertentu dalam simulasi. Fungsi yang didefinisikan dengan `@resumable` dapat ditangguhkan saat menghasilkan suatu kejadian, lalu dilanjutkan saat kejadian tersebut dipicu. Ini membantu menyimulasikan perilaku sistem yang berjalan secara paralel atau terpisah, seperti kendaraan yang menunggu di antrean

atau mesin yang menunggu perbaikan. Hal ini seperti yang dijelaskan pada Gambar 6 sebagai berikut:

```
@resumable function truck_50(sim::Simulation,name,ss::ShovelStruct,cs::CrusherStruct)

    while true
        # truck arrive at shovel

        truckArrive=now(sim)
        ss.nLen += 1
        println("At time $(now(sim)): $name request shovel")
        @yield request(ss.shovel)
        ss.delayTime += now(sim)-truckArrive
        ss.areaUnderQ += (now(sim)-truckArrive)*ss.nLen
        ss.nLen -= 1

        println("At time $(now(sim)): $(name) starts loading")
        loadingtime = getLoadingTime_50()
        shovel_process = @process shovel_proc(sim,ss, loadingtime)
        @yield shovel_process

        @yield release(ss.shovel)
        println("At time $(now(sim)): $(name) stop loading")
        println("At time $(now(sim)): $(name) traveling to crusher")

        @yield timeout(sim, 3)

        println("At time $(now(sim)): $(name) request crusher")
        truckArrive = now(sim)
        cs.nLen += 1

        @yield request(cs.crusher, priority=1)
        println("At time $(now(sim)): $(name) start dumping")

        cs.delayTime += now(sim)-truckArrive
        cs.areaUnderQ += (now(sim)-truckArrive)*ss.nLen
        cs.nLen -= 1

        crushertime = getDumpingTime_50()
        crusher_process = @process crusher_proc(sim,cs,crushertime )
        @yield crusher_process

        @yield release(cs.crusher)
        println("At time $(now(sim)): $(name) finished dumping")
        println("At time $(now(sim)): $(name) traveling to shovel")

        # return to shovel
        @yield timeout(sim, 2)
    end
end
```

Gambar 6 Script untuk simulasi proses operasional truk 50 ton pada sistem *Shovel-Crusher* menggunakan Julia

Dalam Gambar 6, fungsi `truck_50` ini untuk menangani simulasi truk kapasitas 50 ton yang mengangkut material dari tempat penggalian (*shovel*) ke penghancur (*crusher*). Siklus `while true` ini menunjukkan bahwa truk akan terus

melakukan siklus mengambil material dari penggalian ke penghancur secara terus-menerus. Pengelolaan antrean melibatkan beberapa langkah, yaitu: `request(ss.shovel)` untuk meminta *shovel* dan menunggu hingga *shovel* tersedia, serta `request(cs.crusher, priority=1)` untuk meminta akses ke *crusher* dengan prioritas utama. Waktu proses dihitung dengan berbagai mekanisme, seperti `loadingtime` dan `crushertime` untuk membangkitkan waktu yang dibutuhkan untuk memuat dan mengangkut material, serta `timeout` untuk menunggu proses dalam waktu yang telah ditentukan. Misalkan `timeout(sim, 2)` berarti menunggu proses selama 2 menit.

```
@resumable function truck_20(sim::Simulation,name,
                             ss::ShovelStruct,cs::CrusherStruct)

  while true
    truckArrive = now(sim)

    ss.nLen += 1
    println("At time $(now(sim)): $(name) request shovel")
    @yield request(ss.shovel)
    ss.delayTime += now(sim)-truckArrive
    ss.areaUnderQ += (now(sim)-truckArrive)*ss.nLen
    ss.nLen -= 1

    println("At time $(now(sim)): $(name) starts loading")
    shovel_process = @process shovel_proc(sim, ss, getLoadingTime_20())
    @yield shovel_process
    @yield release(ss.shovel)

    println("At time $(now(sim)): $(name) stop loading")
    println("At time $(now(sim)): $(name) traveling to crusher")

    @yield timeout(sim, 2.5)
    println("At time $(now(sim)): $(name) request crusher")

    println("At time $(now(sim)): $(name) request crusher")
    truckArrive = now(sim)
    cs.nLen += 1

    @yield request(cs.crusher,priority=2)
    println("At time $(now(sim)): $(name) start dumping")

    cs.delayTime += now(sim)-truckArrive
    cs.areaUnderQ += (now(sim)-truckArrive)*ss.nLen
    cs.nLen -= 1
    crushertime = getDumpingTime_20()
    crusher_process = @process crusher_proc(sim,cs,crushertime )
    @yield crusher_process

    @yield release(cs.crusher)
    println("At time $(now(sim)): $(name) finished dumping")
    println("At time $(now(sim)): $(name) traveling to shovel")

    # return to shovel
    @yield timeout(sim, 1.5)

  end
end
```

Gambar 7 Script untuk simulasi proses operasional truk 20 ton pada sistem *Shovel-Crusher* menggunakan Julia

Fungsi `truck_20` memiliki struktur yang mirip dengan fungsi `truck_50`, tetapi dengan beberapa perubahan pada waktu yang diambil dan prioritas saat meminta akses ke *crusher*. Prioritas saat meminta akses ke *crusher* untuk truk kapasitas 50 ton lebih diutamakan dibanding untuk truk kapasitas 20 ton.

```
@resumable function crusher_proc(sim::Simulation,cs::CrusherStruct,crushertime)

    cs.busyTime += crushertime
    cs.nTruck += 1
    @yield timeout(sim,crushertime)
end

@resumable function shovel_proc(sim::Simulation,ss::ShovelStruct,shoveltime)
    ss.busyTime += shoveltime
    ss.nTruck += 1
    @yield timeout(sim,shoveltime)
end
```

Gambar 8 Script untuk proses simulasi penggunaan *shovel* dan *crusher* dalam sistem transportasi material

Dalam Gambar 8, fungsi `crusher_proc` dan `shovel_proc` di atas, keduanya didefinisikan sebagai fungsi `@resumable`, yang berarti bahwa mereka dapat dihentikan dan dilanjutkan kembali di titik tertentu (dalam hal ini, saat menunggu). Keduanya mencatat waktu sibuk (`busyTime`) dan jumlah truk yang terlayani (`nTruck`), memberikan informasi penting untuk analisis kinerja sistem.

```
function sim_quarry()
    sim = Simulation()
    ss1 = ShovelStruct(sim)
    ss2 = ShovelStruct(sim)
    ss3 = ShovelStruct(sim)

    cs = CrusherStruct(sim)

    # shovel 1
    @process truck_50(sim,"truck50_1", ss1,cs)
    @process truck_20(sim,"truck20a_1", ss1,cs)
    @process truck_20(sim,"truck20b_1",ss1,cs)

    # shovel 2
    @process truck_50(sim,"truck50_2", ss2,cs)
    @process truck_20(sim,"truck20a_2", ss2,cs)
    @process truck_20(sim,"truck20b_2", ss2,cs)
```

```

# shovel 3
@process truck_50(sim,"truck50_3",ss3,cs)
@process truck_20(sim,"truck20a_3", ss3,cs)
@process truck_20(sim,"truck20b_3", ss3,cs)

msg = run(sim,lengthOfSimulation)

println("Waktu Simulasi: $(lengthOfSimulation) menit")
printQ("Shovel 1", ss1)
printQ("Shovel 2",ss2)
printQ("Shovel 3",ss3)

printQ("Crusher",cs)

stop_time = now(sim)
println("At time $stop_time: $msg")
stop_time
end

function printQTruckAtCrusher(cTitle,s)
println(cTitle)
println("Total truck: $(s.nTruck)")
println("rata-rata waktu delay: $(s.delayTime/s.nTruck)")
println("rata-rata panjang antrian: $(s.areaUnderQ/lengthOfSimulation) trucks")
println("")
end

function printQ(cTitle,s)
println(cTitle)
println("Total truck: $(s.nTruck)")
println("rata-rata waktu delay: $(s.delayTime/s.nTruck)")
println("rata-rata panjang antrian: $(s.areaUnderQ/lengthOfSimulation) trucks")
println("rata-rata utilisasi server : $(s.busyTime/lengthOfSimulation) ")
println("")
end

printQ (generic function with 1 method)

sim_quarry()

```

Gambar 9 Script untuk simulasi proses pengangkutan material di tambang menggunakan Julia

Dalam Gambar 9, fungsi `sim_quarry`, menggunakan `@process` untuk memulai proses pada truk yang berbeda. Setiap *shovel* memiliki beberapa truk yang beroperasi. Misalnya, `@process truck_50(sim,"truck50_1", ss1, cs)` untuk truk kapasitas 50 ton yang menggunakan *shovel* 1 dan *crusher*. Setelah menjalankan simulasi selama waktu tertentu (`lengthOfSimulation`), dilakukan proses mencetak waktu simulasi dan statistik untuk setiap *shovel* dan *crusher* menggunakan fungsi `printQ` dan `printQTruckAtCrusher`.

Fungsi `sim_quarry` dengan baik mengorganisir proses yang berbeda dan mengelola sumber daya dengan menggunakan struktur data yang sesuai. Fungsi-fungsi untuk mencetak laporan memberikan informasi yang jelas dan membantu dalam analisis hasil simulasi. *Output* yang dihasilkan dapat dilihat pada Lampiran 1, sedangkan hasil yang didapatkan ada pada Tabel 2 sebagai berikut:

Tabel 2 Hasil yang didapatkan setelah dilakukan simulasi selama waktu 8 jam

	<i>Shovel 1</i>	<i>Shovel 2</i>	<i>Shovel 3</i>	<i>Crusher</i>
Total truk yang dilayani (unit)	55	56	45	148
Rata-rata waktu <i>delay</i> (menit/truk)	4.72099	3.99654	6.52148	9.70248
Rata-rata panjang antrean (unit)	0.80226	0.65977	1.05066	0.33213
Rata-rata utilisasi server (persen)	69.33%	62.37%	71.62%	89.70%

Tabel 2 menunjukkan bahwa total truk yang diproses selama 8 jam adalah 148 truk. Berarti dari ketiga *shovel* yang tersedia, *crusher* hanya melayani sebanyak 148 truk dengan rata-rata waktu *delay* selama 9.70 menit per truk, antrean sepanjang 0.33 truk dan utilitas server mencapai 89.70%. Berdasarkan hasil ini, diperlukan evaluasi kapasitas *Crusher* untuk mengurangi waktu *delay* yang tinggi. Optimalisasi ini diharapkan dapat meningkatkan efisiensi sistem secara keseluruhan.

3.2 Masalah Antrean Pesawat pada Fasilitas Inspeksi/Perbaikan

Mesin pada pesawat jet harus diperiksa secara berkala dan, jika diperlukan, diperbaiki. Sebuah fasilitas inspeksi/perbaikan di sebuah bandara besar menangani tujuh jenis pesawat jet yang berbeda, seperti yang dijelaskan dalam Tabel 3. Waktu antara kedatangan berturut-turut pesawat jenis ke- i (dengan $i = 1, 2, \dots, 7$) didistribusikan secara eksponensial dengan *mean* $a(i)$, seperti yang diberikan dalam Tabel 3 dan semua waktu dalam satuan hari.

Misalkan ada n stasiun layanan paralel, yang masing-masing secara berurutan menangani inspeksi dan perbaikan semua mesin pada sebuah pesawat, tetapi hanya dapat menangani satu mesin pada satu waktu. Sebagai contoh, pesawat tipe 2 memiliki tiga mesin, jadi ketika masuk ke layanan, setiap mesinnya harus menjalani proses inspeksi dan perbaikan lengkap (seperti dijelaskan di bawah) sebelum mesin berikutnya pada pesawat ini dapat mulai dilayani, dan semua tiga mesin harus diperiksa dan (jika perlu) diperbaiki sebelum pesawat meninggalkan stasiun layanan.

Tabel 3 Data untuk setiap jenis pesawat

Jenis Pesawat (i)	Jumlah Mesin	$a(i)$	$A(i)$	$B(i)$	$p(i)$	$r(i)$	$c(i)$
1	4	8.1	0.7	2.1	0.30	2.1	2.1
2	3	2.9	0.9	1.8	0.26	1.8	1.7
3	2	3.6	0.8	1.6	0.18	1.6	1.0
4*	4	8.4	1.9	2.8	0.12	3.1	3.9
5	4	10.9	0.7	2.2	0.36	2.2	1.4
6	2	6.7	0.9	1.7	0.14	1.7	1.1
7*	3	3.0	1.6	2.0	0.21	2.8	3.7

Setiap stasiun layanan mampu menangani semua jenis pesawat. Seperti biasa, pesawat yang tiba dan menemukan stasiun layanan yang kosong akan langsung

masuk ke layanan, sementara pesawat yang tiba dan menemukan semua stasiun layanan sedang sibuk harus bergabung dalam satu antrian. Dua dari tujuh jenis pesawat diklasifikasikan sebagai pesawat berbadan lebar (ditandai dengan (*) di Tabel 3), sementara lima lainnya diklasifikasikan sebagai pesawat reguler. Dua disiplin untuk antrian yang menarik adalah:

- (i) FIFO sederhana dengan semua jenis pesawat dicampur dalam antrian yang sama.
- (ii) Prioritas tetap diberikan kepada pesawat berbadan lebar, dengan aturan FIFO dalam klasifikasi berbadan lebar dan reguler.

Untuk setiap mesin pada sebuah pesawat (secara independen), proses berikut terjadi (i menunjukkan jenis pesawat):

- Mesin diperiksa terlebih dahulu, yang memakan waktu yang didistribusikan secara uniform antara $A(i)$ dan $B(i)$.
- Keputusan diambil apakah perbaikan diperlukan, probabilitas bahwa perbaikan diperlukan adalah $p(i)$. Jika tidak ada perbaikan yang diperlukan, inspeksi mesin berikutnya dimulai atau jika ini adalah mesin terakhir, pesawat meninggalkan fasilitas.
- Jika perbaikan diperlukan, perbaikan dilakukan, yang memakan waktu yang didistribusikan sebagai variabel acak 2-Erlang dengan rata-rata $r(i)$.
- Setelah perbaikan, inspeksi lain dilakukan, yang memakan waktu yang didistribusikan secara uniform antara $A(i)/2$ dan $B(i)/2$ (yaitu, setengah dari waktu inspeksi awal, karena pembongkaran sudah dilakukan). Probabilitas bahwa mesin memerlukan perbaikan lebih lanjut adalah $p(i)/2$.
- Jika perbaikan awal berhasil, mesin selesai. Jika mesin masih gagal dalam inspeksi, mesin memerlukan perbaikan lebih lanjut, yang memakan waktu yang didistribusikan sebagai 2-Erlang dengan rata-rata $r(i)/2$, setelah itu mesin diperiksa lagi, memakan waktu yang didistribusikan secara *uniform* antara $A(i)/2$ dan $B(i)/2$, mesin gagal dalam inspeksi ini dengan probabilitas $p(i)/2$, dan perlu perbaikan lebih lanjut, yang akan memakan waktu 2-Erlang dengan mean $r(i)/2$. Prosedur ini berlanjut hingga mesin akhirnya lulus inspeksi. Rata-rata waktu perbaikan tetap di $r(i)/2$, probabilitas kegagalan tetap di $p(i)/2$, dan waktu inspeksi tetap antara $A(i)/2$ dan $B(i)/2$.

Biaya sebesar $c(i)$ (diukur dalam puluhan ribu dolar) dikeluarkan untuk setiap hari penuh pesawat tipe i tidak beroperasi, yaitu, berada dalam antrian atau dalam layanan. Biaya *downtime* harian adalah biaya yang dikeluarkan untuk setiap hari penuh ketika pesawat tidak beroperasi karena masih dalam antrian atau dalam layanan perbaikan. Gagasan umumnya adalah untuk mempelajari pengaruh total biaya *downtime* harian rata-rata (yang dijumlahkan untuk semua tipe pesawat) bergantung pada jumlah stasiun layanan, n . Awalnya, sistem dalam keadaan kosong dan tidak aktif, dan simulasi akan berjalan selama 365 hari penuh. Amati rata-rata keterlambatan dalam antrian untuk setiap tipe pesawat dan rata-rata keterlambatan keseluruhan dalam antrian untuk semua tipe pesawat, rata-rata jumlah pesawat dalam antrian, rata-rata jumlah pesawat yang tidak beroperasi untuk setiap tipe pesawat secara terpisah, dan total rata-rata biaya *downtime* harian untuk semua pesawat secara keseluruhan. Cobalah berbagai nilai n untuk memahami perilaku sistem. Rekomendasikan pilihan untuk n , serta disiplin antrian mana (i) atau (ii) di atas yang tampaknya menghasilkan operasi yang paling hemat biaya. Gunakan *streams* 1 hingga 7 untuk waktu antar kedatangan tipe pesawat $i = 1$ hingga $i = 7$,

masing-masing, dan *streams* 8 hingga 14 untuk waktu yang diperlukan untuk perbaikan. Waktu inspeksi masing-masing (pertama atau berikutnya), *streams* 15 hingga 21 untuk menentukan apakah mereka memerlukan perbaikan (tambahan), dan *streams* 22 hingga 28 untuk waktu perbaikan mereka (pertama atau berikutnya).

Sistem yang disimulasikan adalah fasilitas inspeksi dan perbaikan mesin pesawat di sebuah bandara besar yang menangani tujuh jenis pesawat dengan jumlah mesin yang bervariasi.

3.2.1 Model Simulasi Antrean Inspeksi dan Perbaikan Pesawat

1. Parameter Waktu dan Biaya

- Waktu antar kedatangan pesawat tipe i berdistribusi eksponensial yang dapat dituliskan sebagai:

$$IA_i \sim \text{Exponential}(a(i))$$

- Waktu inspeksi awal untuk pesawat tipe i :

$$IT_1(i) \sim \text{Uniform}(A(i), B(i))$$

- Probabilitas perbaikan untuk pesawat tipe i yang diperlukan setelah inspeksi:

$$P(i)$$

- Waktu perbaikan untuk pesawat tipe i berdistribusi Erlang yang dituliskan sebagai:

$$RT_1(i) \sim \text{Erlang}\left(2, \frac{1}{r(i)}\right)$$

- Waktu inspeksi ulang untuk pesawat tipe i setelah perbaikan:

$$IT_2(i) \sim \text{Uniform}\left(\frac{A(i)}{2}, \frac{B(i)}{2}\right)$$

- Probabilitas perbaikan lebih lanjut untuk pesawat tipe i setelah inspeksi ulang:

$$P(i)/2$$

- Waktu perbaikan lebih lanjut untuk pesawat tipe i berdistribusi Erlang yang dituliskan sebagai:

$$RT_2(i) \sim \text{Erlang}\left(2, \frac{1}{r(i)/2}\right)$$

- Biaya *downtime* untuk setiap pesawat tipe i per hari:

$$C(i)$$

2. Model Matematika

a. Fungsi Waktu Layanan (*Service Time*)

Total waktu yang diperlukan pesawat i untuk menyelesaikan seluruh proses inspeksi dan perbaikan (termasuk inspeksi ulang jika diperlukan):

$$W_i = IT_1(i) + [\text{jika diperlukan perbaikan: } RT_1(i) + IT_2(i) + [\text{jika diperlukan perbaikan: } RT_2(i)]]$$

b. Sistem Antrean

Terdapat dua strategi antrean yang dipertimbangkan:

- FIFO (*First In, First Out*): Semua jenis pesawat menggunakan satu antrean tunggal.
- Prioritas Tetap: Pesawat berbadan lebar (tipe 4 dan 7) memiliki prioritas lebih tinggi dibandingkan pesawat reguler.

c. Waktu rata-rata pesawat menunggu di antrean:

$$\bar{W}_{Q_i} = \frac{\text{total waktu tunggu pesawat jenis } i}{\text{jumlah pesawat jenis } i}$$

d. Tingkat Utilisasi Stasiun Layanan

Tingkat utilisasi setiap stasiun

$$\rho = \frac{\text{total waktu sibuk stasiun}}{\text{total waktu simulasi}}$$

e. Biaya Downtime Harian

Total biaya *downtime* harian untuk pesawat jenis *i*:

$$Cost_i = \frac{\text{total waktu pesawat jenis } i \text{ dalam antrian dan layanan} \times C(i)}{365}$$

Total biaya *downtime* harian keseluruhan:

$$Total\ Cost = \sum_{i=1}^7 Cost_i$$

3. Tujuan Simulasi

Simulasi dilakukan selama 365 hari untuk mengestimasi:

- Rata-rata waktu tunggu di antrian untuk setiap jenis pesawat.
- Rata-rata waktu *downtime* dan biaya *downtime* harian.
- Membandingkan efisiensi antara dua strategi antrian:
 - a. FIFO sederhana.
 - b. Prioritas tetap untuk pesawat berbadan lebar.

Model ini dapat diimplementasikan menggunakan bahasa pemrograman Julia dengan paket *ConcurrentSim.jl* atau *SimJulia*, di mana simulasi dapat dilakukan untuk mengevaluasi berbagai nilai *n* (jumlah stasiun layanan) untuk mencari konfigurasi yang paling optimal dari sisi biaya dan efisiensi operasi.

3.2.2 Penulisan Sintaks

3.2.2.1 Kondisi (i) : Strategi antrian dengan FIFO Sederhana

Simulasi ini berfokus pada antrian layanan untuk pesawat berdasarkan tipe mesin, di mana setiap pesawat melalui proses inspeksi dan perbaikan dengan strategi antrian FIFO. Berikut adalah penjelasan mengenai komponen-komponen utama dalam kode:

```
using ConcurrentSim, Random, ResumableFunctions, Distributions
```

```
mutable struct StatStruct
    dur ::Float64
    delay ::Float64
    nPlane ::Int
    tPlane ::Float64
    nPlaneSSQ ::Int
    tDown::Float64
    nDownSSQ::Int
end
```

```

function StatStruct()
    st = new()
    st.dur = 0.0
    st.delay = 0.0
    st.nPlane = 0
    st.tPlane = 0.0
    st.nPlaneSSQ = 0
    st.tDown=0.0
    st.nDownSSQ=0
    return st
end
end

```

Gambar 10 Script struktur data untuk statistik simulasi antrean pesawat dalam Julia

Dalam Gambar 10, mutable struct `StatStruct` mendefinisikan struktur data yang dapat diubah setelah dibuat. `StatStruct` adalah struktur data yang menyimpan statistik dari simulasi, seperti durasi total layanan (`dur`), total penundaan (`delay`), jumlah pesawat (`nPlane`), serta beberapa variabel terkait dengan penghitungan statistik pesawat dalam sistem antrean.

```

aseed = [1234, 34456, 2345, 123, 33, 55, 67,
        6464646, 88888, 109854, 568, 98766324, 6464646, 333324242,
        84746, 787, 55562, 109345, 5456, 635353, 3334,
        8762, 2879, 3908, 24536, 52638, 66673, 8889]

a = [8.1, 2.9, 3.6, 8.4, 10.9, 6.7, 3.0]

A = [0.7, 0.9, 0.8, 1.9, 0.7, 0.9, 1.6]
B = [2.1, 1.8, 1.6, 2.8, 2.2, 1.7, 2.0]
p = [0.3, 0.26, 0.18, 0.12, 0.36, 0.14, 0.21]
r = [2.1, 1.8, 1.6, 3.1, 2.2, 1.7, 2.8]
c = [2.1, 1.7, 1.0, 3.9, 1.4, 1.1, 3.7]

nEngine = [4 3 2 4 4 2 3]

rng = []
for i=1:28
    push!(rng, MersenneTwister(aseed[i]))
end

```

Gambar 11 Script untuk inisialisasi parameter dan generator angka acak dalam simulasi Julia

Sintaks pada Gambar 11 digunakan untuk menetapkan beberapa nilai acak menggunakan *array* `aseed` dan menginisialisasi generator acak `MersenneTwister` untuk setiap nilai.

```

function getNextIA(planeType::Int)
    return a[planeType]*randexp(rng[planeType], Float32)
end

```

```

function getNextIT1(planeType::Int)
    return rand(rng[7+planeType],Uniform(A[planeType],B[planeType]))
end

function getNextIT2(planeType::Int)
    return rand(rng[7+planeType],Uniform(A[planeType]/2,B[planeType]/2))
end

function getNextDec1(planeType)
    return rand(rng[14+planeType],Bernoulli(p[planeType]))
end

function getNextDec2(planeType)
    return rand(rng[14+planeType],Bernoulli(p[planeType]/2))
end

function getNextRep1(planeType)
    return rand(rng[21+planeType],Erlang(2,1/r[planeType]))
end

function getNextRep2(planeType)
    return rand(rng[21+planeType],Erlang(2,1/(r[planeType]/2)))
end

```

Gambar 12 *Script* untuk fungsi pembangkitan variabel acak untuk simulasi pergerakan pesawat

Sintaks pada Gambar 12 digunakan untuk menghasilkan berbagai variabel acak dalam simulasi pergerakan pesawat. Waktu antar kedatangan pesawat dihasilkan menggunakan distribusi eksponensial dengan rata-rata yang ditentukan dalam a , dengan `randexp` dikalikan nilai a untuk mendapatkan waktu yang sesuai. Waktu inspeksi pesawat diperoleh menggunakan distribusi *uniform* antara $A[planeType]$ dan $B[planeType]$, sementara keputusan seperti menentukan apakah perbaikan diperlukan, dihasilkan menggunakan distribusi Bernoulli dengan probabilitas $p[planeType]$. Selain itu, waktu perbaikan dihitung menggunakan distribusi Erlang dengan parameter $r[planeType]$, yang mencerminkan durasi perbaikan yang bervariasi bergantung pada jenis pesawat.

```

@resumable function regularPlane(sim::Simulation,planeType::Int,
    serviceStation::Resource, st::StatStruct)
    start_time = now(sim)
    st.nPlane += 1
    st.nPlaneSSQ += 1
    st.nDownSSQ += 1
    @yield request(serviceStation)
    st.delay += now(sim)-start_time
    st.tPlane += st.nPlaneSSQ * (now(sim)-start_time)
    st.nPlaneSSQ -= 1

```

```

for i=1:nEngine[planeType]
    firstInspect = true
    firstRepair = true
    engineDone = false
    while !engineDone
        # inspecting engine
        if firstInspect
            @yield timeout(sim,getNextIT1(planeType))
            firstInspect = false
        else
            @yield timeout(sim,getNextIT2(planeType))
        end
        if firstRepair
            needRepair = getNextDec1(planeType)
        else
            needRepair = getNextDec2(planeType)
        end

        if needRepair
            # repair
            if firstRepair
                @yield timeout(sim,getNextRep1(planeType))
                firstRepair = false
            else
                @yield timeout(sim,getNextRep2(planeType))
            end
        else
            engineDone = true
        end
    end
end
@yield release(serviceStation)
st.tDown += st.nDownSSQ*(now(sim)-start_time)
st.nDownSSQ -= 1
st.dur += now(sim)-start_time
end

```

Gambar 13 Script untuk implementasi proses inspeksi dan perbaikan pesawat dalam simulasi

Fungsi `regularPlane` pada Gambar 13 memodelkan proses pemeliharaan pesawat dengan menggunakan metode simulasi berbasis peristiwa. Ini adalah fungsi *resumable* yang berarti bisa berhenti sementara untuk menunggu waktu tertentu dalam simulasi. Di sini, pesawat masuk antrean untuk diperiksa, dan jika perbaikan diperlukan, pesawat diperbaiki sebelum dilepaskan dari layanan. Ini mencakup:

- mengatur kedatangan dan layanan pesawat,
- mengelola proses pemeriksaan dan perbaikan mesin,
- memperbarui statistik yang relevan untuk analisis lebih lanjut.

Pendekatan ini memungkinkan untuk melakukan simulasi kompleks yang mencakup berbagai aspek dari operasi pemeliharaan pesawat, termasuk keterlambatan, waktu layanan, dan efisiensi proses.

```
@resumable function planeGenerator(sim::Simulation, planeType::Int,
    serviceStation::Resource, stat::StatStruct)
    while true
        ia = getNextIA(planeType)
        @yield timeout(sim, ia)
        @process regularPlane(sim, planeType, serviceStation, stat)
    end
end
```

Gambar 14 Script untuk fungsi generator pesawat dalam simulasi

Fungsi `planeGenerator` pada Gambar 14 bertanggung jawab untuk menghasilkan pesawat secara berkala sesuai dengan waktu antar kedatangan yang dihasilkan oleh `getNextIA(planeType)`.

```
function sim_3(n)
    sim = Simulation()
    serviceStation = Resource(sim, n)
    aStat = []
    for i=1:7
        stat = StatStruct()
        @process planeGenerator(sim, i, serviceStation, stat)
        push!(aStat, stat)
    end
    run(sim, 365)
```

Gambar 15 Script untuk fungsi simulasi proses perawatan pesawat

Pada Gambar 15, fungsi `sim_3(n)` mensimulasikan proses untuk n stasiun layanan selama 365 hari. Beberapa statistik ditampilkan seperti jumlah pesawat yang perlu diperbaiki, rata-rata penundaan, dan biaya total rata-rata harian untuk perbaikan pesawat.

```
avrDelay = Float64[]
totCost = Float64[]

println("Simulation with number of service station $(n) duration 365 days")

for i=1:7
    println("Plane Type $(i)")
    println("number of planes down $(aStat[i].nPlane) planes")
    println("the time-average number of planes in queue $(aStat[i].tPlane/365) per days")
    println("the time-average number of planes down $(aStat[i].tDown/365) per days")
    println("The average delay $(aStat[i].delay/aStat[i].nPlane) days")
    println("The total average daily downtime cost $((aStat[i].dur*c[i])/365) tens thousand dollars")
    push!(totCost, (aStat[i].dur*c[i])/365)
    push!(avrDelay, aStat[i].delay/aStat[i].nPlane)
    println()
end

println()
println("the overall average delay $(sum(avrDelay)/7)")
println("the overall daily downtime cost $(sum(totCost)) tens thousand dollars")
end
```

Gambar 16 *Script* untuk perhitungan statistik dan biaya *downtime* dalam simulasi perawatan pesawat

Pada Gambar 16, `avrDelay` dan `totCost` membuat daftar untuk menyimpan rata-rata keterlambatan dan biaya total, sedangkan *Loop* dari `i = 1:7`, menghitung dan mencetak statistik untuk setiap jenis pesawat seperti jumlah pesawat yang tidak beroperasi (`nPlane`), rata-rata jumlah pesawat yang tidak beroperasi per hari, rata-rata keterlambatan dalam hari, dan biaya rata-rata *downtime* harian.

3.2.2.2 Kondisi (ii) : Strategi antrean dengan Prioritas Tetap

Kondisi (ii) masih menggunakan sintaks yang sama hanya saja yang membedakan ada tambahan perbedaan prioritas untuk tipe pesawat, yaitu reguler dan berbadan lebar (*widebody*).

```
@resumable function regulerPlane(sim::Simulation,planeType::Int,
    serviceStation::Resource, st::StatStruct)
    start_time = now(sim)
    st.nPlane += 1
    st.nPlaneSSQ += 1
    st.nDownSSQ += 1
    @yield request(serviceStation; priority=2 ) # lower priority
    st.delay += now(sim)-start_time
    st.tPlane += st.nPlaneSSQ * (now(sim)-start_time)
    st.nPlaneSSQ -= 1

    for i=1:nEngine[planeType]
        firstInspect = true
        firstRepair = true
        engineDone = false
        while !engineDone
            # inspecting engine
            if firstInspect
                @yield timeout(sim,getNextIT1(planeType))
                firstInspect = false
            else
                @yield timeout(sim,getNextIT2(planeType))
            end
            if firstRepair
                needRepair = getNextDec1(planeType)
            else
                needRepair = getNextDec2(planeType)
            end

            if needRepair
                # repair
                if firstRepair
                    @yield timeout(sim,getNextRep1(planeType))
                    firstRepair = false
                else
                    @yield timeout(sim,getNextRep2(planeType))
                end
            else
                engineDone = true
            end
        end
    end
end
```

```

        end
    end
    @yield release(serviceStation)
    st.tDown += st.nDownSSQ*(now(sim)-start_time)
    st.nDownSSQ -= 1
    st.dur += now(sim)-start_time
end

```

Gambar 17 Script untuk simulasi pemeliharaan pesawat pada *regular plane* dalam Julia

Pada Gambar 17, fungsi `regularPlane`, pesawat reguler meminta layanan dengan prioritas lebih rendah (`priority=2`).

```

@resumable function widebodyPlane(sim::Simulation, planeType::Int,
    serviceStation::Resource, st::StatStruct)
    start_time = now(sim)
    st.nPlane += 1
    st.nPlaneSSQ += 1
    st.nDownSSQ += 1
    @yield request(serviceStation; priority=1)
    st.delay += now(sim)-start_time
    st.tPlane += st.nPlaneSSQ * (now(sim)-start_time)
    st.nPlaneSSQ -= 1

    for i=1:nEngine[planeType]
        firstInspect = true
        firstRepair = true
        engineDone = false
        while !engineDone
            # inspecting engine
            if firstInspect
                @yield timeout(sim, getNextIT1(planeType))
                firstInspect = false
            else
                @yield timeout(sim, getNextIT2(planeType))
            end
            if firstRepair
                needRepair = getNextDec1(planeType)
            else
                needRepair = getNextDec2(planeType)
            end

            if needRepair
                # repair
                if firstRepair
                    @yield timeout(sim, getNextRep1(planeType))
                    firstRepair = false
                else
                    @yield timeout(sim, getNextRep2(planeType))
                end
            else
                engineDone = true
            end
        end
    end
    @yield release(serviceStation)
    st.tDown += st.nDownSSQ*(now(sim)-start_time)
    st.nDownSSQ -= 1
    st.dur += now(sim)-start_time
end

```

Gambar 18 *Script* untuk simulasi pemeliharaan pesawat pada berbadan lebar (*widebody*) *plane* dalam Julia

Menurut Gambar 18, fungsi *widebodyPlane* hampir identik dalam cara mereka menangani inspeksi dan perbaikan mesin dengan fungsi *regulerPlane*, tetapi ada perbedaan dalam prioritas permintaan layanan di stasiun layanan. Pada fungsi *regulerPlane*, pesawat *reguler* meminta layanan dengan prioritas lebih rendah (*priority=2*). Pada fungsi *widebodyPlane*, pesawat tipe berbadan lebar (*widebody*) (dalam hal ini tipe 4 dan tipe 7) meminta layanan dengan prioritas lebih tinggi (*priority=1*). Ini berarti ketika ada pesawat berbadan lebar (*widebody*) dan pesawat *reguler* yang bersaing untuk sumber daya yang sama, pesawat berbadan lebar (*widebody*) akan dilayani terlebih dahulu, sesuai dengan prioritas yang lebih tinggi.

```
@resumable function planeGenerator(sim::Simulation, planeType::Int,
    serviceStation::Resource, stat::StatStruct)
    while true
        ia = getNextIA(planeType)
        @yield timeout(sim, ia)
        if (planeType==4) || (planeType==7)
            @process widebodyPlane(sim, planeType, serviceStation, stat)
        else
            @process regulerPlane(sim, planeType, serviceStation, stat)
        end
    end
end

function sim_3(n)
    sim = Simulation()
    serviceStation = Resource(sim, n)
    aStat = []
    for i=1:7
        stat = StatStruct()
        @process planeGenerator(sim, i, serviceStation, stat)
        push!(aStat, stat)
    end
    run(sim, 365)
    avrDelay = Float64[]
    totCost = Float64[]

    println("Simulation with number of service station $(n) duration 365 days")

    for i=1:7
        println("Plane Type $(i)")
        println("number of planes down $(aStat[i].nPlane) planes")
        println("the time-average number of planes in queue $(aStat[i].tPlane/365) per days")
        println("the time-average number of planes down $(aStat[i].tDown/365) per days")
        println("The average delay $(aStat[i].delay/aStat[i].nPlane) days")
        println("The total average daily downtime cost $((aStat[i].dur*c[i])/365) tens thousand dollars")
        push!(totCost, (aStat[i].dur*c[i])/365)
        push!(avrDelay, aStat[i].delay/aStat[i].nPlane)
        println()
    end
end
```

```
println()
println("the overall average delay $(sum(avrDelay)/7)")
println("the overall daily downtime cost $(sum(totCost)) tens thousand dollars")
end
```

Gambar 19 Script untuk simulasi pemeliharaan pesawat (model perawatan dan analisis kinerja stasiun servis)

Dalam Gambar 19, fungsi `planeGenerator` memutuskan tipe pesawat mana yang akan digunakan berdasarkan `planeType`:

- Pesawat tipe 4 dan 7 diklasifikasikan sebagai berbadan lebar (*widebody*).
- Pesawat dengan tipe selain 4 dan 7 diklasifikasikan sebagai reguler.

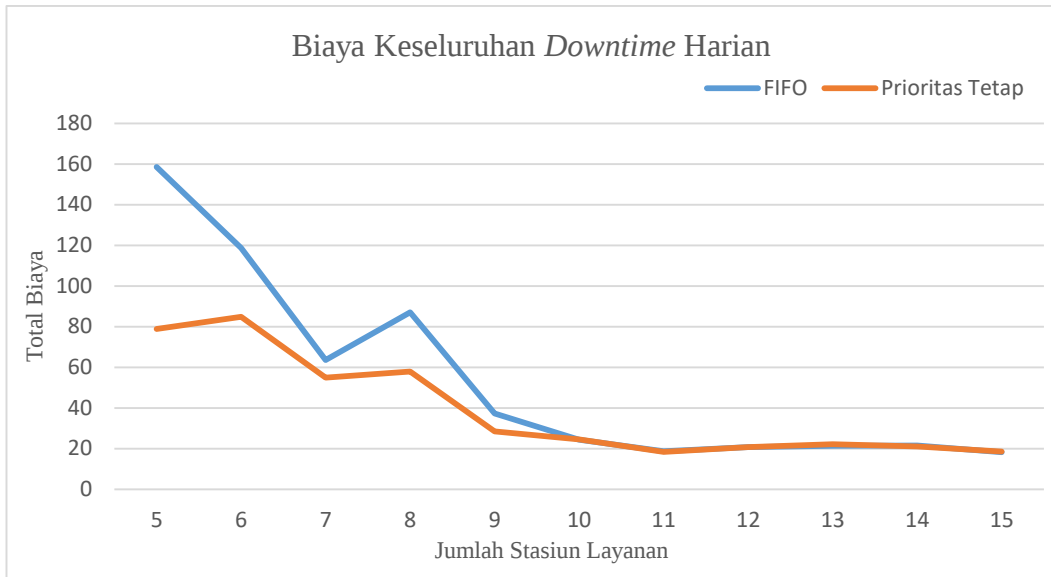
Simulasi berjalan selama 365 hari, pesawat reguler dan berbadan lebar (*widebody*) yang dihasilkan secara berkala oleh `planeGenerator`. Penghitungan statistik seperti banyaknya pesawat yang diinspeksi, durasi waktu inspeksi, waktu rata-rata penundaan, dan total biaya waktu tidak aktif.

3.2.3 Hasil Perbandingan antara Dua Strategi Antrean serta Uji Statistik

Simulasi selama 365 hari dilakukan untuk menganalisis dan membandingkan hasil guna menentukan skenario operasi yang paling efisien dan hemat biaya. Perbandingan total biaya *downtime* harian antara dua strategi antrean dijelaskan pada Tabel 4 dan Gambar 20, sebagai berikut:

Tabel 4 Perbandingan total biaya *downtime* harian

Jumlah Stasiun Layanan (n)	Strategi Antrean	
	FIFO	Prioritas Tetap
5	158.54170160650762	78.88676657489444
6	118.69726714042908	84.80603209771722
7	63.63811615090181	54.96606860972138
8	86.99695441843208	57.85472920561651
9	37.26560144012093	28.41769742132476
10	24.40329907519988	24.55143272557347
11	18.70359117021295	18.44314830639980
12	20.71150263558360	20.77592303950918
13	21.38333551677335	22.11754715329957
14	21.56899466889098	21.03463101346694
15	18.16443437805388	18.59921208328010



Gambar 20 Grafik perbandingan total biaya *downtime* harian

Setelah dilakukan percobaan dengan jumlah stasiun layanan (n) yang berbeda, terlihat pada grafik di Gambar 20 yang menggunakan strategi antrean prioritas tetap, menghasilkan operasi yang paling hemat biaya di dibandingkan dengan menggunakan strategi antrean FIFO. Jika pada $5 < n < 8$, strategi FIFO memiliki biaya yang sangat tinggi dibandingkan dengan Prioritas Tetap, yang lebih unggul karena secara konsisten menghasilkan biaya *downtime* yang lebih rendah. Pada $9 < n < 10$, perbedaan biaya antara kedua strategi mulai mengecil, meskipun FIFO masih lebih tinggi dibandingkan Prioritas Tetap. Selain itu, pada $n = 9$, biaya mulai mencapai titik yang cukup rendah, sehingga penambahan jumlah stasiun lebih lanjut mungkin tidak memberikan penghematan biaya yang signifikan. Sementara itu, jika pada $n > 10$, kedua strategi menunjukkan biaya *downtime* yang hampir sama, dan peningkatan jumlah stasiun di atas $n = 10$ tidak lagi memberikan pengurangan biaya yang signifikan, sehingga menambah n lebih dari 10 mungkin menjadi tidak efisien.

Untuk menguji signifikansi perbedaan biaya operasional antara kedua strategi tersebut, dilakukan uji-T guna menentukan perbedaan yang diamati bersifat signifikan secara statistik atau hanya terjadi secara kebetulan. Uji statistik (uji-T) ini diambil dari hasil simulasi selama 365 hari dengan berbagai nilai n , di mana data yang dianalisis adalah biaya operasional yang dihasilkan oleh strategi antrean prioritas tetap dan strategi antrean FIFO. Secara spesifik, uji-T dilakukan terhadap rata-rata biaya operasional harian dari kedua strategi untuk melihat apakah ada perbedaan yang signifikan secara statistik. Hasil dari uji statistik dijelaskan dalam Tabel 5 sebagai berikut:

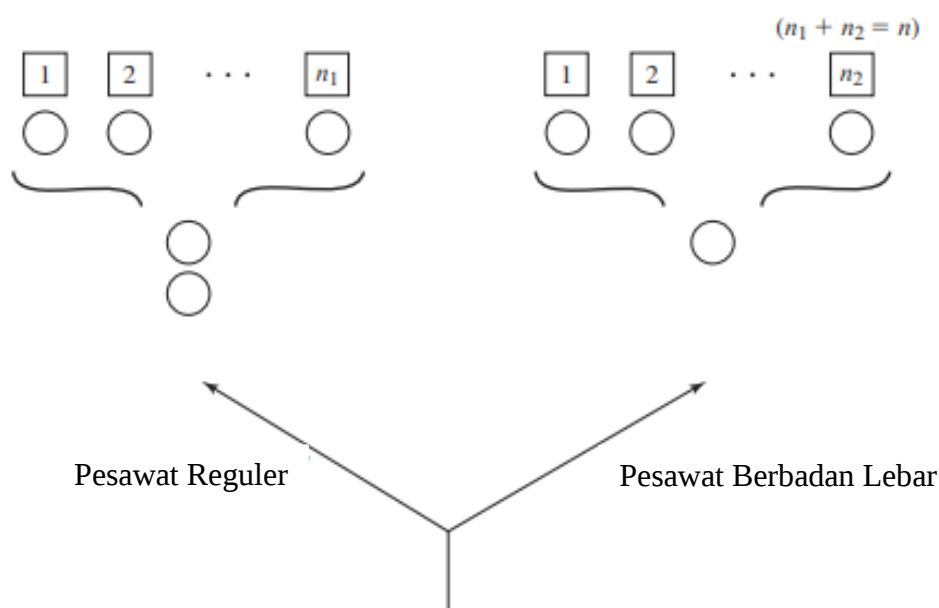
Dari Tabel 5, untuk $n = 5, 6, 7$ memiliki keputusan H_0 ditolak maka ada perbedaan yang signifikan antara strategi FIFO dan Prioritas Tetap yang menunjukkan bahwa strategi FIFO memang lebih mahal secara signifikan, sedangkan untuk $n = 8, \dots, 15$ memiliki keputusan H_0 diterima yang berarti tidak ada perbedaan biaya yang signifikan. Nilai n yang optimal untuk efisiensi biaya tampaknya berada di sekitar 9 atau 10, di mana biaya *downtime* sudah sangat rendah dan tambahan stasiun tidak memberikan keuntungan besar.

Tabel 5 Uji statistika (Uji-T)

Jumlah Stasiun Layanan (n)	s_1^2	s_2^2	$\bar{X}_1$	$\bar{X}_2$	s_{gab}^2	SE	t	df	t_{tabel}	Keputusan
5	161.68	105.45	145.81	74.94	133.57	7.31	9.70	8	2.31	H0 ditolak
6	442.63	74.24	124.30	73.80	258.43	9.28	5.44	10	2.23	H0 ditolak
7	408.20	134.89	105.50	73.17	271.55	8.81	3.67	12	2.18	H0 ditolak
8	503.01	227.01	65.76	50.66	365.01	9.55	1.58	14	2.15	H0 diterima
9	32.29	31.30	31.37	29.81	31.79	2.66	0.59	16	2.12	H0 diterima
10	37.49	44.52	26.63	26.10	41.00	2.86	0.19	18	2.10	H0 diterima
11	1.85	2.29	22.84	22.68	2.07	0.61	0.25	20	2.09	H0 diterima
12	2.21	2.70	21.51	21.47	2.46	0.64	0.06	22	2.07	H0 diterima
13	2.36	2.43	20.63	20.79	2.39	0.61	-0.27	24	2.06	H0 diterima
14	1.63	1.58	20.28	20.24	1.60	0.48	0.07	26	2.06	H0 diterima
15	0.90	0.70	19.72	19.69	0.80	0.33	0.07	28	2.05	H0 diterima

3.2.4 Cara Alternatif

Alternatif ini memisahkan layanan pesawat berbadan lebar dan reguler ke stasiun yang berbeda untuk meningkatkan efisiensi operasional. Dengan pemisahan ini, waktu tunggu dapat berkurang karena antrean menjadi lebih lancar tanpa hambatan dari perbedaan jenis pesawat. Selain itu, efisiensi sumber daya meningkat karena setiap stasiun dapat disesuaikan dengan kebutuhan spesifik pesawat yang dilayaninya. Dampak lainnya adalah pengurangan *downtime* dan biaya operasional, karena layanan yang lebih cepat dapat mengurangi keterlambatan. Dari segi manajemen, pendekatan ini juga memudahkan pemeliharaan dan pengalokasian tenaga kerja yang lebih terfokus. Namun, perlu dilakukan optimasi dalam pembagian jumlah stasiun antara pesawat berbadan lebar dan pesawat reguler agar kapasitas tetap seimbang. Simulasi lebih lanjut diperlukan untuk menilai efektivitas strategi ini dibandingkan dengan metode antrean sebelumnya.



Gambar 21 Tata letak alternatif untuk fasilitas perbaikan pesawat

Sebagai alternatif untuk tata letak di atas, pertimbangkan untuk memisahkan sepenuhnya layanan pesawat jenis “berbadan lebar (*widebody*)” (pesawat berbadan lebar) dan pesawat reguler. Artinya, ambil n_2 dari n stasiun dan kirim semua pesawat berbadan lebar ke sana (dengan satu antrean khusus untuk pesawat jenis “berbadan lebar (*widebody*)” (pesawat berbadan lebar) yang melayani semua n_2 stasiun), dan sisa $n_1 = n - n_2$ stasiun hanya untuk pesawat reguler; lihat Gambar 21. Untuk menentukan tata letak alternatif yang lebih baik dan efisien, gunakan parameter dan penugasan aliran yang sama seperti di atas (sintaks bisa dilihat di Lampiran 2). Hasil yang diperoleh dijelaskan dalam Tabel 6 berikut:

Tabel 6 Data total biaya *downtime* harian

n_1	n_2	Total Biaya <i>Downtime</i> Harian
5	1	82.72351
5	2	99.74875
5	3	67.65933
5	4	51.25576
5	5	33.98835
6	1	74.32311
6	2	61.57760
6	3	52.00206
6	4	31.92133
6	5	24.81399
6	6	21.27570
7	1	73.95875
7	2	95.15161
7	3	48.96818
7	4	27.61192
7	5	24.89490
7	6	24.35201
7	7	24.04171

Berdasarkan Tabel 6, kita dapat menganalisis total biaya *downtime* harian untuk berbagai kombinasi nilai n_1 (stasiun layanan untuk pesawat reguler) dan n_2 (stasiun layanan untuk pesawat berbadan lebar (*widebody*)), untuk menentukan model yang paling efisien.

Pada total biaya *downtime* harian, model yang paling baik adalah ketika $n_1 = 6$ dan $n_2 = 6$, dengan total biaya *downtime* harian terendah sebesar 21.28 seperti yang dijelaskan pada Tabel 6. Ini menunjukkan bahwa model dengan 6 stasiun layanan untuk pesawat reguler dan 6 stasiun layanan untuk pesawat berbadan lebar (*widebody*) menghasilkan biaya *downtime* yang paling efisien. Secara umum, semakin banyak stasiun yang digunakan, semakin rendah biaya *downtime*, yang berarti kapasitas sistem lebih tinggi dan lebih banyak pesawat dapat diproses dalam waktu yang lebih singkat.

Namun, perlu dicatat bahwa penurunan biaya *downtime* tidak selalu sebanding dengan jumlah stasiun yang ditambahkan, karena penambahan stasiun di atas titik tertentu (seperti pada $n_1 = 7$ dan $n_2 = 4$) tidak memberikan penurunan biaya yang signifikan, bahkan bisa meningkatkan biaya seperti yang terlihat pada kombinasi $n_1 = 7$ dan $n_2 = 2$, yang memiliki total biaya *downtime* 95.15. Hal ini mengindikasikan bahwa ada titik optimal di mana penambahan lebih banyak stasiun tidak memberikan banyak manfaat tambahan.

Kesimpulannya, model $n_1 = 6$ dan $n_2 = 6$ adalah yang paling efisien, memberikan biaya *downtime* harian terendah, yang menunjukkan bahwa sistem dengan jumlah stasiun yang seimbang antara layanan dan peralatan lainnya menghasilkan kinerja terbaik.

@Hak cipta milik IPB University

IPB University



IV SIMPULAN DAN SARAN

4.1. Simpulan

Penelitian ini menunjukkan bahwa pemodelan simulasi menggunakan bahasa pemrograman Julia dan pustaka-pustaka pendukungnya, seperti *Distributions.jl* dan *ConcurrentSim.jl*, dapat memberikan solusi efektif untuk memecahkan permasalahan dalam pengelolaan sistem dinamis. Studi kasus menunjukkan bahwa pendekatan simulasi berbasis peristiwa diskret mampu meminimalkan waktu tunggu dan mengoptimalkan pemanfaatan sumber daya. Selain itu, implementasi Julia memungkinkan fleksibilitas dalam analisis skenario dan pengambilan keputusan berdasarkan data simulasi. Dengan demikian, Julia menjadi alternatif yang kompetitif untuk pemodelan simulasi dalam berbagai aplikasi ilmiah dan teknis.

4.2. Saran

Penelitian ini dapat diperluas untuk diterapkan pada sistem lain, seperti logistik dan kesehatan. Dengan menerapkan metode yang serupa, analisis efisiensi dan optimasi sistem antrian di berbagai bidang dapat dilakukan untuk meningkatkan kinerja operasional dan pengelolaan sumber daya.

DAFTAR PUSTAKA

- Banks J, Carson JS, Nelson BL, Nicol DM. 2005. *Discrete-Event System Simulation*. Ed ke-4. Upper Saddle River: Pearson.
- Bezanson J, Karpinski S, Shah VB, Edelman A. 2012. Julia: A Fast Language for Technical Computing. arXiv: 1209.5145v1 [cs.PL].
- Bezanson J, Edelman A, Karpinski S, Shah VB. 2017. Julia: A Fresh Approach to Numerical Computing. *SIAM Review*. 59(1): 65-98. doi: 10.1137/141000671.
- Feldt R. 2017. "BlackBoxOptim.jl". URL: <https://github.com/robertfeldt/BlackBoxOptim.jl>.
- [FRBNY] Federal Reserve Bank of New York. 2019. DSGE.jl: Solve and Estimate Dynamic Stochastic General Equilibrium Models (Including the New York Fed DSGE). URL: <https://github.com/FRBNY-DSGE.jl>.
- Fishman GS. 2001. *Discrete-Event Simulation: Modeling, Programming, and Analysis*. Springer.
- Fowler M. 2010. *Domain-Specific Language*. Boston: Pearson Education.
- Gelman A, Carlin JB, Stern H, Dunson DB, Vehtari A, Robin D B. 2013. *Bayesian Data Analysis*. Ed ke-3. Boca Raton: CRC Press.
- Gordon AD, Henzinger TA, Nori AV, Rajamani SK. 2014. Probability Programming. In *Proceedings of The On Future of Software Engineering*. Pp 167-181. New York (NY): Association for Computing Machinery(ACM).
- Grinstead CM, Snell JL. 1997. *Introduction to Probability*. Providence: American Mathematical Society.
- Harrel C. 2004. *Simulation Using ProModel*. New York (NY): McGraw-Hill.
- JuliaDynamics. [tahun terbit tidak diketahui]. *ConcurrentSim.jl*: A Julia package for concurrent simulations. *GitHub*. <https://github.com/JuliaDynamics/ConcurrentSim.jl>.
- Jaynes ET. 2003. *Probability Theory: The Logic of Science*. Cambridge: Cambridge University Press.
- Law AM, Kelton WD. 2014. *Simulation Modeling and Analysis*. Ed ke-5. New York (NY): McGraw-Hill Education.
- Lyon S, Sargent TJ, Stachurski J. 2017. QuantEcon.jl. URL: <https://github.com/QuantEcon/QuantEcon.jl>.
- Montgomery DC, Runger GC. 2014. *Applied Statistics and Probability for Engineers*. Ed ke-6. Hoboken: Wiley.
- Railsback SF, Grimm V. 2019. *Agent-Based and Individual-Based Modeling: A Practical Introduction*. Princeton: Princeton University Press.
- Ross SM. 2014. *Introduction to Probability Models*. Ed ke-11. Cambridge: Academic Press.
- Smith BJ. 2018. Mamba.jl: Markov Chain Monte Carlo (MCMC) for Bayesian Analysis in Julia (Version 0.12) [Computer software] URL: <https://github.com/brian-j-smith/Mamba.jl>.
- Sterman JD. 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. New York (NY): McGraw-Hill.

LAMPIRAN

Lampiran 1 *Output Software Julia* untuk kasus 1

```

At time 0.0: truck50_1 request shovel
At time 0.0: truck20a_1 request shovel
At time 0.0: truck20b_1 request shovel
At time 0.0: truck50_2 request shovel
At time 0.0: truck20a_2 request shovel
At time 0.0: truck20b_2 request shovel
At time 0.0: truck50_3 request shovel
At time 0.0: truck20a_3 request shovel
At time 0.0: truck20b_3 request shovel
At time 0.0: truck50_1 starts loading
At time 0.0: truck50_2 starts loading
At time 0.0: truck50_3 starts loading
At time 1.0655501924352035: truck50_2 stop loading
At time 1.0655501924352035: truck50_2 traveling to crusher
At time 1.0655501924352035: truck20a_2 starts loading
At time 4.065550192435204: truck50_2 request crusher
At time 4.065550192435204: truck50_2 start dumping
At time 4.35222552584646: truck50_2 finished dumping
At time 4.35222552584646: truck50_2 traveling to shovel
At time 6.025201562903684: truck50_1 stop loading
At time 6.025201562903684: truck50_1 traveling to crusher
At time 6.025201562903684: truck20a_1 starts loading
At time 6.35222552584646: truck50_2 request shovel
At time 7.787443235674708: truck20a_1 stop loading
At time 7.787443235674708: truck20a_1 traveling to crusher
At time 7.787443235674708: truck20b_1 starts loading
At time 9.025201562903684: truck50_1 request crusher
At time 9.025201562903684: truck50_1 start dumping
At time 9.635561915911197: truck20b_1 stop loading
At time 9.635561915911197: truck20b_1 traveling to crusher
At time 10.287443235674708: truck20a_1 request crusher
At time 10.287443235674708: truck20a_1 request crusher
At time 11.57209274397608: truck20a_2 stop loading
At time 11.57209274397608: truck20a_2 traveling to crusher
At time 11.57209274397608: truck20b_2 starts loading
At time 12.135561915911197: truck20b_1 request crusher
At time 12.135561915911197: truck20b_1 request crusher
At time 12.569715808103282: truck50_3 stop loading
At time 12.569715808103282: truck50_3 traveling to crusher
At time 12.569715808103282: truck20a_3 starts loading
At time 12.614423642893525: truck20b_2 stop loading
At time 12.614423642893525: truck20b_2 traveling to crusher
At time 12.614423642893525: truck50_2 starts loading
At time 14.07209274397608: truck20a_2 request crusher
At time 14.07209274397608: truck20a_2 request crusher
At time 14.313394950966533: truck50_2 stop loading
At time 14.313394950966533: truck50_2 traveling to crusher
At time 15.114423642893525: truck20b_2 request crusher
At time 15.114423642893525: truck20b_2 request crusher
At time 15.569715808103282: truck50_3 request crusher
At time 15.754836342667186: truck50_1 finished dumping
At time 15.754836342667186: truck50_1 traveling to shovel
At time 15.754836342667186: truck50_3 start dumping
At time 16.26525409719606: truck50_3 finished dumping
At time 16.26525409719606: truck50_3 traveling to shovel
At time 16.26525409719606: truck20a_1 start dumping
At time 17.313394950966533: truck50_2 request crusher
At time 17.754836342667186: truck50_1 request shovel
At time 17.754836342667186: truck50_1 starts loading
At time 18.26525409719606: truck50_3 request shovel
At time 18.435600200911466: truck20a_1 finished dumping
At time 18.435600200911466: truck20a_1 traveling to shovel
At time 18.435600200911466: truck50_2 start dumping
At time 18.821590444776234: truck20a_3 stop loading

```

At time 18.821590444776234: truck20a_3 traveling to crusher
 At time 18.821590444776234: truck20b_3 starts loading
 At time 19.254190788915: truck50_2 finished dumping
 At time 19.254190788915: truck50_2 traveling to *shovel*
 At time 19.254190788915: truck20b_1 start dumping
 At time 19.935600200911466: truck20a_1 request *shovel*
 At time 20.029350215650073: truck20b_3 stop loading
 At time 20.029350215650073: truck20b_3 traveling to crusher
 At time 20.029350215650073: truck50_3 starts loading
 At time 20.31281799845205: truck20b_1 finished dumping
 At time 20.31281799845205: truck20b_1 traveling to *shovel*
 At time 20.31281799845205: truck20a_2 start dumping
 At time 21.076799826264143: truck20a_2 finished dumping
 At time 21.076799826264143: truck20a_2 traveling to *shovel*
 At time 21.076799826264143: truck20b_2 start dumping
 At time 21.254190788915: truck50_2 request *shovel*
 At time 21.254190788915: truck50_2 starts loading
 At time 21.321590444776234: truck20a_3 request crusher
 At time 21.321590444776234: truck20a_3 request crusher
 At time 21.542039700848978: truck50_2 stop loading
 At time 21.542039700848978: truck50_2 traveling to crusher
 At time 21.81281799845205: truck20b_1 request *shovel*
 At time 22.529350215650073: truck20b_3 request crusher
 At time 22.529350215650073: truck20b_3 request crusher
 At time 22.576799826264143: truck20a_2 request *shovel*
 At time 22.576799826264143: truck20a_2 starts loading
 At time 24.542039700848978: truck50_2 request crusher
 At time 25.793033497735063: truck50_1 stop loading
 At time 25.793033497735063: truck50_1 traveling to crusher
 At time 25.793033497735063: truck20a_1 starts loading
 At time 26.084737531351784: truck20a_2 stop loading
 At time 26.084737531351784: truck20a_2 traveling to crusher
 At time 26.5450979293218: truck20b_2 finished dumping
 At time 26.5450979293218: truck20b_2 traveling to *shovel*
 At time 26.5450979293218: truck50_2 start dumping
 At time 28.0450979293218: truck20b_2 request *shovel*
 At time 28.0450979293218: truck20b_2 starts loading
 At time 28.1221939069367: truck50_2 finished dumping
 At time 28.1221939069367: truck50_2 traveling to *shovel*
 At time 28.1221939069367: truck20a_3 start dumping
 At time 28.257217017577517: truck20a_3 finished dumping
 At time 28.257217017577517: truck20a_3 traveling to *shovel*
 At time 28.257217017577517: truck20b_3 start dumping
 At time 28.584737531351784: truck20a_2 request crusher
 At time 28.584737531351784: truck20a_2 request crusher
 At time 28.793033497735063: truck50_1 request crusher
 At time 29.757217017577517: truck20a_3 request *shovel*
 At time 30.1221939069367: truck50_2 request *shovel*
 At time 30.79063397272143: truck20b_3 finished dumping
 At time 30.79063397272143: truck20b_3 traveling to *shovel*
 At time 30.79063397272143: truck50_1 start dumping
 At time 31.637792057438404: truck20a_1 stop loading
 At time 31.637792057438404: truck20a_1 traveling to crusher
 At time 31.637792057438404: truck20b_1 starts loading
 At time 31.68133983821685: truck20b_2 stop loading
 At time 31.68133983821685: truck20b_2 traveling to crusher
 At time 31.68133983821685: truck50_2 starts loading
 At time 31.891548860903747: truck20b_1 stop loading
 At time 31.891548860903747: truck20b_1 traveling to crusher
 At time 32.29063397272143: truck20b_3 request *shovel*
 At time 33.02482606054429: truck50_1 finished dumping
 At time 33.02482606054429: truck50_1 traveling to *shovel*
 At time 33.02482606054429: truck20a_2 start dumping
 At time 34.1377920574384: truck20a_1 request crusher
 At time 34.1377920574384: truck20a_1 request crusher
 At time 34.18133983821685: truck20b_2 request crusher
 At time 34.18133983821685: truck20b_2 request crusher
 At time 34.21227595368816: truck20a_2 finished dumping
 At time 34.21227595368816: truck20a_2 traveling to *shovel*
 At time 34.21227595368816: truck20a_1 start dumping
 At time 34.36064993091278: truck50_3 stop loading

@Hak cipta milik IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 34.36064993091278: truck50_3 traveling to crusher
 At time 34.36064993091278: truck20a_3 starts loading
 At time 34.39154886090375: truck20b_1 request crusher
 At time 34.39154886090375: truck20b_1 request crusher
 At time 35.02482606054429: truck50_1 request shovel
 At time 35.02482606054429: truck50_1 starts loading
 At time 35.71227595368816: truck20a_2 request shovel
 At time 35.792590642438185: truck50_2 stop loading
 At time 35.792590642438185: truck50_2 traveling to crusher
 At time 35.792590642438185: truck20a_2 starts loading
 At time 35.9667935938874: truck20a_1 finished dumping
 At time 35.9667935938874: truck20a_1 traveling to shovel
 At time 35.9667935938874: truck20b_2 start dumping
 At time 37.36064993091278: truck50_3 request crusher
 At time 37.4667935938874: truck20a_1 request shovel
 At time 37.57977642011483: truck20b_2 finished dumping
 At time 37.57977642011483: truck20b_2 traveling to shovel
 At time 37.57977642011483: truck50_3 start dumping
 At time 38.792590642438185: truck50_2 request crusher
 At time 39.07977642011483: truck20b_2 request shovel
 At time 39.87187210694298: truck20a_2 stop loading
 At time 39.87187210694298: truck20a_2 traveling to crusher
 At time 39.87187210694298: truck20b_2 starts loading
 At time 40.78093381292863: truck20a_3 stop loading
 At time 40.78093381292863: truck20a_3 traveling to crusher
 At time 40.78093381292863: truck20b_3 starts loading
 At time 40.855545679201036: truck20b_2 stop loading
 At time 40.855545679201036: truck20b_2 traveling to crusher
 At time 41.392545825725136: truck50_3 finished dumping
 At time 41.392545825725136: truck50_3 traveling to shovel
 At time 41.392545825725136: truck50_2 start dumping
 At time 41.87145681892961: truck20b_3 stop loading
 At time 41.87145681892961: truck20b_3 traveling to crusher
 At time 42.37187210694298: truck20a_2 request crusher
 At time 42.37187210694298: truck20a_2 request crusher
 At time 43.28093381292863: truck20a_3 request crusher
 At time 43.28093381292863: truck20a_3 request crusher
 At time 43.355545679201036: truck20b_2 request crusher
 At time 43.355545679201036: truck20b_2 request crusher
 At time 43.392545825725136: truck50_3 request shovel
 At time 43.392545825725136: truck50_3 starts loading
 At time 44.37145681892961: truck20b_3 request crusher
 At time 44.37145681892961: truck20b_3 request crusher
 At time 45.035796384649665: truck50_2 finished dumping
 At time 45.035796384649665: truck50_2 traveling to shovel
 At time 45.035796384649665: truck20b_1 start dumping
 At time 45.534130699015904: truck20b_1 finished dumping
 At time 45.534130699015904: truck20b_1 traveling to shovel
 At time 45.534130699015904: truck20a_2 start dumping
 At time 46.25108797177192: truck20a_2 finished dumping
 At time 46.25108797177192: truck20a_2 traveling to shovel
 At time 46.25108797177192: truck20a_3 start dumping
 At time 46.78343713612673: truck20a_3 finished dumping
 At time 46.78343713612673: truck20a_3 traveling to shovel
 At time 46.78343713612673: truck20b_2 start dumping
 At time 47.034130699015904: truck20b_1 request shovel
 At time 47.035796384649665: truck50_2 request shovel
 At time 47.035796384649665: truck50_2 starts loading
 At time 47.75108797177192: truck20a_2 request shovel
 At time 48.28343713612673: truck20a_3 request shovel
 At time 49.54751120427764: truck20b_2 finished dumping
 At time 49.54751120427764: truck20b_2 traveling to shovel
 At time 49.54751120427764: truck20b_3 start dumping
 At time 50.949929753741074: truck20b_3 finished dumping
 At time 50.949929753741074: truck20b_3 traveling to shovel
 At time 51.04751120427764: truck20b_2 request shovel
 At time 52.449929753741074: truck20b_3 request shovel
 At time 56.33517301344436: truck50_2 stop loading
 At time 56.33517301344436: truck50_2 traveling to crusher
 At time 56.33517301344436: truck20a_2 starts loading

@Hak cipta milik IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 56.71056966104811: truck20a_2 stop loading
 At time 56.71056966104811: truck20a_2 traveling to crusher
 At time 56.71056966104811: truck20b_2 starts loading
 At time 59.21056966104811: truck20a_2 request crusher
 At time 59.21056966104811: truck20a_2 request crusher
 At time 59.21056966104811: truck20a_2 start dumping
 At time 59.33517301344436: truck50_2 request crusher
 At time 61.05280343394721: truck20a_2 finished dumping
 At time 61.05280343394721: truck20a_2 traveling to shovel
 At time 61.05280343394721: truck50_2 start dumping
 At time 61.31923122936804: truck20b_2 stop loading
 At time 61.31923122936804: truck20b_2 traveling to crusher
 At time 62.55280343394721: truck20a_2 request shovel
 At time 62.55280343394721: truck20a_2 starts loading
 At time 63.262483799267685: truck50_3 stop loading
 At time 63.262483799267685: truck50_3 traveling to crusher
 At time 63.262483799267685: truck20a_3 starts loading
 At time 63.81923122936804: truck20b_2 request crusher
 At time 63.81923122936804: truck20b_2 request crusher
 At time 66.26248379926768: truck50_3 request crusher
 At time 66.50394431239677: truck50_2 finished dumping
 At time 66.50394431239677: truck50_2 traveling to shovel
 At time 66.50394431239677: truck50_3 start dumping
 At time 67.3728368859688: truck20a_3 stop loading
 At time 67.3728368859688: truck20a_3 traveling to crusher
 At time 67.3728368859688: truck20b_3 starts loading
 At time 68.50394431239677: truck50_2 request shovel
 At time 69.60947811189456: truck20b_3 stop loading
 At time 69.60947811189456: truck20b_3 traveling to crusher
 At time 69.8728368859688: truck20a_3 request crusher
 At time 69.8728368859688: truck20a_3 request crusher
 At time 69.89736959244041: truck50_3 finished dumping
 At time 69.89736959244041: truck50_3 traveling to shovel
 At time 69.89736959244041: truck20b_2 start dumping
 At time 70.2145526288518: truck50_1 stop loading
 At time 70.2145526288518: truck50_1 traveling to crusher
 At time 70.2145526288518: truck20a_1 starts loading
 At time 70.6816063878444: truck20b_2 finished dumping
 At time 70.6816063878444: truck20b_2 traveling to shovel
 At time 70.6816063878444: truck20a_3 start dumping
 At time 71.7705915158534: truck20a_3 finished dumping
 At time 71.7705915158534: truck20a_3 traveling to shovel
 At time 71.89736959244041: truck50_3 request shovel
 At time 71.89736959244041: truck50_3 starts loading
 At time 72.09957491812034: truck20a_2 stop loading
 At time 72.09957491812034: truck20a_2 traveling to crusher
 At time 72.09957491812034: truck50_2 starts loading
 At time 72.10947811189456: truck20b_3 request crusher
 At time 72.10947811189456: truck20b_3 request crusher
 At time 72.10947811189456: truck20b_3 start dumping
 At time 72.1816063878444: truck20b_2 request shovel
 At time 73.2145526288518: truck50_1 request crusher
 At time 73.2705915158534: truck20a_3 request shovel
 At time 73.62026442125708: truck20a_1 stop loading
 At time 73.62026442125708: truck20a_1 traveling to crusher
 At time 73.62026442125708: truck20b_1 starts loading
 At time 74.59957491812034: truck20a_2 request crusher
 At time 74.59957491812034: truck20a_2 request crusher
 At time 74.60148485071524: truck20b_1 stop loading
 At time 74.60148485071524: truck20b_1 traveling to crusher
 At time 74.93108824048582: truck20b_3 finished dumping
 At time 74.93108824048582: truck20b_3 traveling to shovel
 At time 74.93108824048582: truck50_1 start dumping
 At time 76.12026442125708: truck20a_1 request crusher
 At time 76.12026442125708: truck20a_1 request crusher
 At time 76.43108824048582: truck20b_3 request shovel
 At time 76.78586443745853: truck50_1 finished dumping
 At time 76.78586443745853: truck50_1 traveling to shovel
 At time 76.78586443745853: truck20a_2 start dumping
 At time 77.10148485071524: truck20b_1 request crusher

@Hak cipta milik IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 77.10148485071524: truct20b_1 request crusher
 At time 78.17720888890379: truck50_3 stop loading
 At time 78.17720888890379: truck50_3 traveling to crusher
 At time 78.17720888890379: truck20a_3 starts loading
 At time 78.78586443745853: truck50_1 request shovel
 At time 78.78586443745853: truck50_1 starts loading
 At time 79.72714890684912: truck50_2 stop loading
 At time 79.72714890684912: truck50_2 traveling to crusher
 At time 79.72714890684912: truck20b_2 starts loading
 At time 79.93694424859405: truck20a_2 finished dumping
 At time 79.93694424859405: truck20a_2 traveling to shovel
 At time 79.93694424859405: truct20a_1 start dumping
 At time 80.14539231570373: truck20b_2 stop loading
 At time 80.14539231570373: truck20b_2 traveling to crusher
 At time 80.38776769376634: truct20a_1 finished dumping
 At time 80.38776769376634: truct20a_1 traveling to shovel
 At time 80.38776769376634: truck20b_1 start dumping
 At time 81.06953012620616: truck20a_3 stop loading
 At time 81.06953012620616: truck20a_3 traveling to crusher
 At time 81.06953012620616: truck20b_3 starts loading
 At time 81.17720888890379: truck50_3 request crusher
 At time 81.43694424859405: truck20a_2 request shovel
 At time 81.43694424859405: truck20a_2 starts loading
 At time 81.62378279756399: truck20b_1 finished dumping
 At time 81.62378279756399: truck20b_1 traveling to shovel
 At time 81.62378279756399: truck50_3 start dumping
 At time 81.88776769376634: truck20a_1 request shovel
 At time 82.64539231570373: truck20b_2 request crusher
 At time 82.64539231570373: truck20b_2 request crusher
 At time 82.72714890684912: truck50_2 request crusher
 At time 83.12378279756399: truck20b_1 request shovel
 At time 83.36553635932452: truck20b_3 stop loading
 At time 83.36553635932452: truck20b_3 traveling to crusher
 At time 83.56953012620616: truck20a_3 request crusher
 At time 83.56953012620616: truck20a_3 request crusher
 At time 84.05099569200998: truck20a_2 stop loading
 At time 84.05099569200998: truck20a_2 traveling to crusher
 At time 85.86553635932452: truck20b_3 request crusher
 At time 85.86553635932452: truck20b_3 request crusher
 At time 86.21886364877336: truck50_3 finished dumping
 At time 86.21886364877336: truck50_3 traveling to shovel
 At time 86.21886364877336: truck50_2 start dumping
 At time 86.55099569200998: truck20a_2 request crusher
 At time 86.55099569200998: truck20a_2 request crusher
 At time 88.21886364877336: truck50_3 request shovel
 At time 88.21886364877336: truck50_3 starts loading
 At time 90.83740254734946: truck50_2 finished dumping
 At time 90.83740254734946: truck50_2 traveling to shovel
 At time 90.83740254734946: truck20b_2 start dumping
 At time 91.39675294099102: truck20b_2 finished dumping
 At time 91.39675294099102: truck20b_2 traveling to shovel
 At time 91.39675294099102: truck20a_3 start dumping
 At time 92.63671363098268: truck50_3 stop loading
 At time 92.63671363098268: truck50_3 traveling to crusher
 At time 92.83740254734946: truck50_2 request shovel
 At time 92.83740254734946: truck50_2 starts loading
 At time 92.89675294099102: truck20b_2 request shovel
 At time 95.01814647211752: truck20a_3 finished dumping
 At time 95.01814647211752: truck20a_3 traveling to shovel
 At time 95.01814647211752: truck20b_3 start dumping
 At time 95.63671363098268: truck50_3 request crusher
 At time 96.51814647211752: truck20a_3 request shovel
 At time 96.51814647211752: truck20a_3 starts loading
 At time 98.61156542902263: truck20a_3 stop loading
 At time 98.61156542902263: truck20a_3 traveling to crusher
 At time 101.04026394829074: truck20b_3 finished dumping
 At time 101.04026394829074: truck20b_3 traveling to shovel
 At time 101.04026394829074: truck50_3 start dumping
 At time 101.11156542902263: truck20a_3 request crusher
 At time 101.11156542902263: truck20a_3 request crusher
 At time 102.54026394829074: truck20b_3 request shovel

@Hak cipta milik IPB University

IPB University

Hak Cipta Dilindungi Undang-undang
 1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 102.54026394829074: truck20b_3 starts loading
 At time 104.92044376743077: truck20b_3 stop loading
 At time 104.92044376743077: truck20b_3 traveling to crusher
 At time 107.42044376743077: truck20b_3 request crusher
 At time 107.42044376743077: truck20b_3 request crusher
 At time 107.93934578009527: truck50_1 stop loading
 At time 107.93934578009527: truck50_1 traveling to crusher
 At time 107.93934578009527: truck20a_1 starts loading
 At time 109.23478416071778: truck20a_1 stop loading
 At time 109.23478416071778: truck20a_1 traveling to crusher
 At time 109.23478416071778: truck20b_1 starts loading
 At time 109.5788724720318: truck50_3 finished dumping
 At time 109.5788724720318: truck50_3 traveling to shovel
 At time 109.5788724720318: truck20a_2 start dumping
 At time 110.64164457630307: truck50_2 stop loading
 At time 110.64164457630307: truck50_2 traveling to crusher
 At time 110.64164457630307: truck20b_2 starts loading
 At time 110.93934578009527: truck50_1 request crusher
 At time 111.5788724720318: truck50_3 request shovel
 At time 111.5788724720318: truck50_3 starts loading
 At time 111.73478416071778: truck20a_1 request crusher
 At time 111.73478416071778: truck20a_1 request crusher
 At time 112.01576821905839: truck20a_2 finished dumping
 At time 112.01576821905839: truck20a_2 traveling to shovel
 At time 112.01576821905839: truck50_1 start dumping
 At time 113.51576821905839: truck20a_2 request shovel
 At time 113.64164457630307: truck50_2 request crusher
 At time 113.9483840078194: truck20b_1 stop loading
 At time 113.9483840078194: truck20b_1 traveling to crusher
 At time 114.05458859694687: truck50_1 finished dumping
 At time 114.05458859694687: truck50_1 traveling to shovel
 At time 114.05458859694687: truck50_2 start dumping
 At time 114.46733412789185: truck50_2 finished dumping
 At time 114.46733412789185: truck50_2 traveling to shovel
 At time 114.46733412789185: truck20a_3 start dumping
 At time 116.05458859694687: truck50_1 request shovel
 At time 116.05458859694687: truck50_1 starts loading
 At time 116.4483840078194: truck20b_1 request crusher
 At time 116.4483840078194: truck20b_1 request crusher
 At time 116.46733412789185: truck50_2 request shovel
 At time 117.93530418155574: truck20a_3 finished dumping
 At time 117.93530418155574: truck20a_3 traveling to shovel
 At time 117.93530418155574: truck20b_3 start dumping
 At time 118.19865338890986: truck50_1 stop loading
 At time 118.19865338890986: truck50_1 traveling to crusher
 At time 119.43530418155574: truck20a_3 request shovel
 At time 119.85134608235958: truck50_3 stop loading
 At time 119.85134608235958: truck50_3 traveling to crusher
 At time 119.85134608235958: truck20a_3 starts loading
 At time 120.511738246746: truck20b_3 finished dumping
 At time 120.511738246746: truck20b_3 traveling to shovel
 At time 120.511738246746: truck20a_1 start dumping
 At time 120.57108665950403: truck20a_1 finished dumping
 At time 120.57108665950403: truck20a_1 traveling to shovel
 At time 120.57108665950403: truck20b_1 start dumping
 At time 121.18400164346237: truck20b_2 stop loading
 At time 121.18400164346237: truck20b_2 traveling to crusher
 At time 121.18400164346237: truck20a_2 starts loading
 At time 121.19865338890986: truck50_1 request crusher
 At time 121.6875709688753: truck20a_3 stop loading
 At time 121.6875709688753: truck20a_3 traveling to crusher
 At time 121.9121658009647: truck20b_1 finished dumping
 At time 121.9121658009647: truck20b_1 traveling to shovel
 At time 121.9121658009647: truck50_1 start dumping
 At time 122.011738246746: truck20b_3 request shovel
 At time 122.011738246746: truck20b_3 starts loading
 At time 122.07108665950403: truck20a_1 request shovel
 At time 122.07108665950403: truck20a_1 starts loading
 At time 122.46791203432237: truck20a_1 stop loading
 At time 122.46791203432237: truck20a_1 traveling to crusher
 At time 122.60996684470385: truck50_1 finished dumping

@Hak cipta milik IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 122.60996684470385: truck50_1 traveling to *shovel*
 At time 122.85134608235958: truck50_3 request crusher
 At time 122.85134608235958: truck50_3 start dumping
 At time 123.03065746018558: truck20a_2 stop loading
 At time 123.03065746018558: truck20a_2 traveling to crusher
 At time 123.03065746018558: truck50_2 starts loading
 At time 123.4121658009647: truct20b_1 request *shovel*
 At time 123.4121658009647: truct20b_1 starts loading
 At time 123.44023312116455: truck50_3 finished dumping
 At time 123.44023312116455: truck50_3 traveling to *shovel*
 At time 123.68400164346237: truck20b_2 request crusher
 At time 123.68400164346237: truck20b_2 request crusher
 At time 123.68400164346237: truck20b_2 start dumping
 At time 124.14286650449463: truck20b_2 finished dumping
 At time 124.14286650449463: truck20b_2 traveling to *shovel*
 At time 124.1875709688753: truck20a_3 request crusher
 At time 124.1875709688753: truck20a_3 request crusher
 At time 124.1875709688753: truck20a_3 start dumping
 At time 124.60996684470385: truck50_1 request *shovel*
 At time 124.96791203432237: truct20a_1 request crusher
 At time 124.96791203432237: truct20a_1 request crusher
 At time 125.44023312116455: truck50_3 request *shovel*
 At time 125.53065746018558: truck20a_2 request crusher
 At time 125.53065746018558: truck20a_2 request crusher
 At time 125.64286650449463: truck20b_2 request *shovel*
 At time 126.61233738401131: truck20a_3 finished dumping
 At time 126.61233738401131: truck20a_3 traveling to *shovel*
 At time 126.61233738401131: truct20a_1 start dumping
 At time 127.11049327719064: truct20b_1 stop loading
 At time 127.11049327719064: truct20b_1 traveling to crusher
 At time 127.11049327719064: truck50_1 starts loading
 At time 128.1123373840113: truck20a_3 request *shovel*
 At time 128.89604642092206: truck50_1 stop loading
 At time 128.89604642092206: truck50_1 traveling to crusher
 At time 129.61049327719064: truct20b_1 request crusher
 At time 129.61049327719064: truct20b_1 request crusher
 At time 131.89604642092206: truck50_1 request crusher
 At time 134.1820895836834: truct20a_1 finished dumping
 At time 134.1820895836834: truct20a_1 traveling to *shovel*
 At time 134.1820895836834: truck50_1 start dumping
 At time 134.20238521785126: truck50_1 finished dumping
 At time 134.20238521785126: truck50_1 traveling to *shovel*
 At time 134.20238521785126: truck20a_2 start dumping
 At time 134.67392259501747: truck20a_2 finished dumping
 At time 134.67392259501747: truck20a_2 traveling to *shovel*
 At time 134.67392259501747: truct20b_1 start dumping
 At time 134.8178819533597: truct20b_1 finished dumping
 At time 134.8178819533597: truct20b_1 traveling to *shovel*
 At time 135.6820895836834: truct20a_1 request *shovel*
 At time 135.6820895836834: truct20a_1 starts loading
 At time 136.17392259501747: truck20a_2 request *shovel*
 At time 136.20238521785126: truck50_1 request *shovel*
 At time 136.3178819533597: truct20b_1 request *shovel*
 At time 139.71232185959795: truct20a_1 stop loading
 At time 139.71232185959795: truct20a_1 traveling to crusher
 At time 139.71232185959795: truck50_1 starts loading
 At time 139.9376858909952: truck50_1 stop loading
 At time 139.9376858909952: truck50_1 traveling to crusher
 At time 139.9376858909952: truct20b_1 starts loading
 At time 140.657788021658: truck50_2 stop loading
 At time 140.657788021658: truck50_2 traveling to crusher
 At time 140.657788021658: truck20b_2 starts loading
 At time 141.22208524213687: truck20b_2 stop loading
 At time 141.22208524213687: truck20b_2 traveling to crusher
 At time 141.22208524213687: truck20a_2 starts loading
 At time 142.21232185959795: truct20a_1 request crusher
 At time 142.21232185959795: truct20a_1 request crusher
 At time 142.21232185959795: truct20a_1 start dumping
 At time 142.5735650083086: truck20b_3 stop loading
 At time 142.5735650083086: truck20b_3 traveling to crusher
 At time 142.5735650083086: truck50_3 starts loading

At time 142.85501287087152: truct20b_1 stop loading
 At time 142.85501287087152: truct20b_1 traveling to crusher
 At time 142.9376858909952: truck50_1 request crusher
 At time 143.657788021658: truck50_2 request crusher
 At time 143.72208524213687: truck20b_2 request crusher
 At time 143.72208524213687: truck20b_2 request crusher
 At time 144.15503319396666: truct20a_1 finished dumping
 At time 144.15503319396666: truct20a_1 traveling to shovel
 At time 144.15503319396666: truck50_1 start dumping
 At time 144.29290631743524: truck20a_2 stop loading
 At time 144.29290631743524: truck20a_2 traveling to crusher
 At time 145.0735650083086: truck20b_3 request crusher
 At time 145.0735650083086: truck20b_3 request crusher
 At time 145.35501287087152: truct20b_1 request crusher
 At time 145.35501287087152: truct20b_1 request crusher
 At time 145.65503319396666: truct20a_1 request shovel
 At time 145.65503319396666: truct20a_1 starts loading
 At time 146.79290631743524: truck20a_2 request crusher
 At time 146.79290631743524: truck20a_2 request crusher
 At time 149.16981000473916: truct20a_1 stop loading
 At time 149.16981000473916: truct20a_1 traveling to crusher
 At time 149.45848445872335: truck50_1 finished dumping
 At time 149.45848445872335: truck50_1 traveling to shovel
 At time 149.45848445872335: truck50_2 start dumping
 At time 151.45848445872335: truck50_1 request shovel
 At time 151.45848445872335: truck50_1 starts loading
 At time 151.66981000473916: truct20a_1 request crusher
 At time 151.66981000473916: truct20a_1 request crusher
 At time 152.33299016763692: truck50_2 finished dumping
 At time 152.33299016763692: truck50_2 traveling to shovel
 At time 152.33299016763692: truck20b_2 start dumping
 At time 153.75985356692905: truck20b_2 finished dumping
 At time 153.75985356692905: truck20b_2 traveling to shovel
 At time 153.75985356692905: truck20b_3 start dumping
 At time 154.33299016763692: truck50_2 request shovel
 At time 154.33299016763692: truck50_2 starts loading
 At time 154.663393684536: truck20b_3 finished dumping
 At time 154.663393684536: truck20b_3 traveling to shovel
 At time 154.663393684536: truck20b_1 start dumping
 At time 155.25985356692905: truck20b_2 request shovel
 At time 155.91945134098015: truck50_1 stop loading
 At time 155.91945134098015: truck50_1 traveling to crusher
 At time 156.163393684536: truck20b_3 request shovel
 At time 156.2651899179669: truct20b_1 finished dumping
 At time 156.2651899179669: truct20b_1 traveling to shovel
 At time 156.2651899179669: truck20a_2 start dumping
 At time 157.7651899179669: truct20b_1 request shovel
 At time 157.7651899179669: truct20b_1 starts loading
 At time 158.7899444977159: truck20a_2 finished dumping
 At time 158.7899444977159: truck20a_2 traveling to shovel
 At time 158.7899444977159: truct20a_1 start dumping
 At time 158.91945134098015: truck50_1 request crusher
 At time 159.2309135021516: truct20a_1 finished dumping
 At time 159.2309135021516: truct20a_1 traveling to shovel
 At time 159.2309135021516: truck50_1 start dumping
 At time 160.14937821189335: truck50_1 finished dumping
 At time 160.14937821189335: truck50_1 traveling to shovel
 At time 160.2899444977159: truck20a_2 request shovel
 At time 160.7309135021516: truct20a_1 request shovel
 At time 162.14937821189335: truck50_1 request shovel
 At time 165.57369153951527: truck50_3 stop loading
 At time 165.57369153951527: truck50_3 traveling to crusher
 At time 165.57369153951527: truck20a_3 starts loading
 At time 166.99458126652948: truct20b_1 stop loading
 At time 166.99458126652948: truct20b_1 traveling to crusher
 At time 166.99458126652948: truct20a_1 starts loading
 At time 168.57369153951527: truck50_3 request crusher
 At time 168.57369153951527: truck50_3 start dumping
 At time 169.49458126652948: truct20b_1 request crusher
 At time 169.49458126652948: truct20b_1 request crusher
 At time 169.54381646133987: truck50_2 stop loading

@Hak cipta milk IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 169.54381646133987: truck50_2 traveling to crusher
 At time 169.54381646133987: truck20b_2 starts loading
 At time 169.7681119545634: truck50_3 finished dumping
 At time 169.7681119545634: truck50_3 traveling to shovel
 At time 169.7681119545634: truck20b_1 start dumping
 At time 170.75302515340718: truck20b_1 finished dumping
 At time 170.75302515340718: truck20b_1 traveling to shovel
 At time 171.7681119545634: truck50_3 request shovel
 At time 172.25302515340718: truck20b_1 request shovel
 At time 172.54381646133987: truck50_2 request crusher
 At time 172.54381646133987: truck50_2 start dumping
 At time 172.71420311555678: truck20b_2 stop loading
 At time 172.71420311555678: truck20b_2 traveling to crusher
 At time 172.71420311555678: truck20a_2 starts loading
 At time 173.40126367063115: truck20a_3 stop loading
 At time 173.40126367063115: truck20a_3 traveling to crusher
 At time 173.40126367063115: truck20b_3 starts loading
 At time 174.16464360082549: truck20a_2 stop loading
 At time 174.16464360082549: truck20a_2 traveling to crusher
 At time 174.60811947823643: truck50_2 finished dumping
 At time 174.60811947823643: truck50_2 traveling to shovel
 At time 175.21420311555678: truck20b_2 request crusher
 At time 175.21420311555678: truck20b_2 request crusher
 At time 175.21420311555678: truck20b_2 start dumping
 At time 175.33511705482056: truck20a_1 stop loading
 At time 175.33511705482056: truck20a_1 traveling to crusher
 At time 175.33511705482056: truck50_1 starts loading
 At time 175.7607667671425: truck20b_2 finished dumping
 At time 175.7607667671425: truck20b_2 traveling to shovel
 At time 175.90126367063115: truck20a_3 request crusher
 At time 175.90126367063115: truck20a_3 request crusher
 At time 175.90126367063115: truck20a_3 start dumping
 At time 176.40366332816313: truck20b_3 stop loading
 At time 176.40366332816313: truck20b_3 traveling to crusher
 At time 176.40366332816313: truck50_3 starts loading
 At time 176.60811947823643: truck50_2 request shovel
 At time 176.60811947823643: truck50_2 starts loading
 At time 176.66464360082549: truck20a_2 request crusher
 At time 176.66464360082549: truck20a_2 request crusher
 At time 176.73981463117582: truck50_1 stop loading
 At time 176.73981463117582: truck50_1 traveling to crusher
 At time 176.73981463117582: truck20b_1 starts loading
 At time 176.97129244853326: truck50_2 stop loading
 At time 176.97129244853326: truck50_2 traveling to crusher
 At time 177.2607667671425: truck20b_2 request shovel
 At time 177.2607667671425: truck20b_2 starts loading
 At time 177.83511705482056: truck20a_1 request crusher
 At time 177.83511705482056: truck20a_1 request crusher
 At time 178.164094472257: truck20b_2 stop loading
 At time 178.164094472257: truck20b_2 traveling to crusher
 At time 178.90366332816313: truck20b_3 request crusher
 At time 178.90366332816313: truck20b_3 request crusher
 At time 179.73981463117582: truck50_1 request crusher
 At time 179.79244312148177: truck20a_3 finished dumping
 At time 179.79244312148177: truck20a_3 traveling to shovel
 At time 179.79244312148177: truck50_1 start dumping
 At time 179.97129244853326: truck50_2 request crusher
 At time 180.51329133570314: truck20b_1 stop loading
 At time 180.51329133570314: truck20b_1 traveling to crusher
 At time 180.5668721519566: truck50_1 finished dumping
 At time 180.5668721519566: truck50_1 traveling to shovel
 At time 180.5668721519566: truck50_2 start dumping
 At time 180.664094472257: truck20b_2 request crusher
 At time 180.664094472257: truck20b_2 request crusher
 At time 181.12109062620914: truck50_2 finished dumping
 At time 181.12109062620914: truck50_2 traveling to shovel
 At time 181.12109062620914: truck20a_2 start dumping
 At time 181.29244312148177: truck20a_3 request shovel
 At time 182.5668721519566: truck50_1 request shovel
 At time 182.5668721519566: truck50_1 starts loading
 At time 183.00191434290957: truck50_3 stop loading

@Hak cipta milik IPB University

IPB University

At time 183.00191434290957: truck50_3 traveling to crusher
 At time 183.00191434290957: truck20a_3 starts loading
 At time 183.01329133570314: truck20b_1 request crusher
 At time 183.01329133570314: truck20b_1 request crusher
 At time 183.12109062620914: truck50_2 request shovel
 At time 183.12109062620914: truck50_2 starts loading
 At time 183.75195166594702: truck20a_2 finished dumping
 At time 183.75195166594702: truck20a_2 traveling to shovel
 At time 183.75195166594702: truck20a_1 start dumping
 At time 185.25195166594702: truck20a_2 request shovel
 At time 185.610346362291: truck50_1 stop loading
 At time 185.610346362291: truck50_1 traveling to crusher
 At time 185.7639131659145: truck50_2 stop loading
 At time 185.7639131659145: truck50_2 traveling to crusher
 At time 185.7639131659145: truck20a_2 starts loading
 At time 186.00191434290957: truck50_3 request crusher
 At time 186.64876702021132: truck20a_2 stop loading
 At time 186.64876702021132: truck20a_2 traveling to crusher
 At time 188.38746875467504: truck20a_1 finished dumping
 At time 188.38746875467504: truck20a_1 traveling to shovel
 At time 188.38746875467504: truck50_3 start dumping
 At time 188.610346362291: truck50_1 request crusher
 At time 188.7639131659145: truck50_2 request crusher
 At time 189.14876702021132: truck20a_2 request crusher
 At time 189.14876702021132: truck20a_2 request crusher
 At time 189.88746875467504: truck20a_1 request shovel
 At time 189.88746875467504: truck20a_1 starts loading
 At time 192.4528604756736: truck20a_1 stop loading
 At time 192.4528604756736: truck20a_1 traveling to crusher
 At time 192.7304221499031: truck20a_3 stop loading
 At time 192.7304221499031: truck20a_3 traveling to crusher
 At time 194.9528604756736: truck20a_1 request crusher
 At time 194.9528604756736: truck20a_1 request crusher
 At time 195.2304221499031: truck20a_3 request crusher
 At time 195.2304221499031: truck20a_3 request crusher
 At time 206.9019969161908: truck50_3 finished dumping
 At time 206.9019969161908: truck50_3 traveling to shovel
 At time 206.9019969161908: truck50_1 start dumping
 At time 208.9019969161908: truck50_3 request shovel
 At time 208.9019969161908: truck50_3 starts loading
 At time 209.42369830530845: truck50_1 finished dumping
 At time 209.42369830530845: truck50_1 traveling to shovel
 At time 209.42369830530845: truck50_2 start dumping
 At time 211.42369830530845: truck50_1 request shovel
 At time 211.42369830530845: truck50_1 starts loading
 At time 212.68105965215335: truck50_2 finished dumping
 At time 212.68105965215335: truck50_2 traveling to shovel
 At time 212.68105965215335: truck20b_3 start dumping
 At time 213.45058565096238: truck50_1 stop loading
 At time 213.45058565096238: truck50_1 traveling to crusher
 At time 214.68105965215335: truck50_2 request shovel
 At time 214.68105965215335: truck50_2 starts loading
 At time 216.45058565096238: truck50_1 request crusher
 At time 216.64434279420945: truck20b_3 finished dumping
 At time 216.64434279420945: truck20b_3 traveling to shovel
 At time 216.64434279420945: truck50_1 start dumping
 At time 218.14434279420945: truck20b_3 request shovel
 At time 218.80559718649917: truck50_1 finished dumping
 At time 218.80559718649917: truck50_1 traveling to shovel
 At time 218.80559718649917: truck20b_2 start dumping
 At time 218.82061852197535: truck20b_2 finished dumping
 At time 218.82061852197535: truck20b_2 traveling to shovel
 At time 218.82061852197535: truck20b_1 start dumping
 At time 220.16608907341939: truck20b_1 finished dumping
 At time 220.16608907341939: truck20b_1 traveling to shovel
 At time 220.16608907341939: truck20a_2 start dumping
 At time 220.32061852197535: truck20b_2 request shovel
 At time 220.80559718649917: truck50_1 request shovel
 At time 220.80559718649917: truck50_1 starts loading
 At time 221.3228284759001: truck50_2 stop loading
 At time 221.3228284759001: truck50_2 traveling to crusher

@Hak cipta milik IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 221.3228284759001: truck20b_2 starts loading
 At time 221.66608907341939: truck20b_1 request shovel
 At time 223.50332479042785: truck20a_2 finished dumping
 At time 223.50332479042785: truck20a_2 traveling to shovel
 At time 223.50332479042785: truck20a_1 start dumping
 At time 223.88328002120454: truck20b_2 stop loading
 At time 223.88328002120454: truck20b_2 traveling to crusher
 At time 224.2615780400599: truck20a_1 finished dumping
 At time 224.2615780400599: truck20a_1 traveling to shovel
 At time 224.2615780400599: truck20a_3 start dumping
 At time 224.3228284759001: truck50_2 request crusher
 At time 225.00332479042785: truck20a_2 request shovel
 At time 225.00332479042785: truck20a_2 starts loading
 At time 225.7615780400599: truck20a_1 request shovel
 At time 225.86815935847045: truck50_1 stop loading
 At time 225.86815935847045: truck50_1 traveling to crusher
 At time 225.86815935847045: truck20b_1 starts loading
 At time 226.38328002120454: truck20b_2 request crusher
 At time 226.38328002120454: truck20b_2 request crusher
 At time 226.90944742046875: truck20a_2 stop loading
 At time 226.90944742046875: truck20a_2 traveling to crusher
 At time 228.86815935847045: truck50_1 request crusher
 At time 229.3999092455379: truck20b_1 stop loading
 At time 229.3999092455379: truck20b_1 traveling to crusher
 At time 229.3999092455379: truck20a_1 starts loading
 At time 229.40944742046875: truck20a_2 request crusher
 At time 229.40944742046875: truck20a_2 request crusher
 At time 231.8999092455379: truck20b_1 request crusher
 At time 231.8999092455379: truck20b_1 request crusher
 At time 232.05197087296273: truck20a_1 stop loading
 At time 232.05197087296273: truck20a_1 traveling to crusher
 At time 233.40807529880888: truck20a_3 finished dumping
 At time 233.40807529880888: truck20a_3 traveling to shovel
 At time 233.40807529880888: truck50_2 start dumping
 At time 234.55197087296273: truck20a_1 request crusher
 At time 234.55197087296273: truck20a_1 request crusher
 At time 234.65792762594182: truck50_2 finished dumping
 At time 234.65792762594182: truck50_2 traveling to shovel
 At time 234.65792762594182: truck50_1 start dumping
 At time 234.90807529880888: truck20a_3 request shovel
 At time 236.65792762594182: truck50_2 request shovel
 At time 236.65792762594182: truck50_2 starts loading
 At time 237.9729717993018: truck50_1 finished dumping
 At time 237.9729717993018: truck50_1 traveling to shovel
 At time 237.9729717993018: truck20b_2 start dumping
 At time 239.64520488051267: truck20b_2 finished dumping
 At time 239.64520488051267: truck20b_2 traveling to shovel
 At time 239.64520488051267: truck20a_2 start dumping
 At time 239.9729717993018: truck50_1 request shovel
 At time 239.9729717993018: truck50_1 starts loading
 At time 240.4159669888176: truck20a_2 finished dumping
 At time 240.4159669888176: truck20a_2 traveling to shovel
 At time 240.4159669888176: truck20b_1 start dumping
 At time 241.05103068605447: truck50_2 stop loading
 At time 241.05103068605447: truck50_2 traveling to crusher
 At time 241.14520488051267: truck20b_2 request shovel
 At time 241.14520488051267: truck20b_2 starts loading
 At time 241.9159669888176: truck20a_2 request shovel
 At time 243.54702895117634: truck50_1 stop loading
 At time 243.54702895117634: truck50_1 traveling to crusher
 At time 244.05103068605447: truck50_2 request crusher
 At time 244.4518908375516: truck20b_1 finished dumping
 At time 244.4518908375516: truck20b_1 traveling to shovel
 At time 244.4518908375516: truck50_2 start dumping
 At time 245.9518908375516: truck20b_1 request shovel
 At time 245.9518908375516: truck20b_1 starts loading
 At time 246.09050095308072: truck20b_2 stop loading
 At time 246.09050095308072: truck20b_2 traveling to crusher
 At time 246.09050095308072: truck20a_2 starts loading
 At time 246.54702895117634: truck50_1 request crusher
 At time 246.73190713813324: truck20a_2 stop loading

@Hak cipta milk IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 246.73190713813324: truck20a_2 traveling to crusher
 At time 246.98123837008816: truck50_2 finished dumping
 At time 246.98123837008816: truck50_2 traveling to shovel
 At time 246.98123837008816: truck50_1 start dumping
 At time 247.1443018824285: truck50_1 finished dumping
 At time 247.1443018824285: truck50_1 traveling to shovel
 At time 247.1443018824285: truck20a_1 start dumping
 At time 247.43726437992132: truck20a_1 finished dumping
 At time 247.43726437992132: truck20a_1 traveling to shovel
 At time 248.59050095308072: truck20b_2 request crusher
 At time 248.59050095308072: truck20b_2 request crusher
 At time 248.59050095308072: truck20b_2 start dumping
 At time 248.93726437992132: truck20a_1 request shovel
 At time 248.98123837008816: truck50_2 request shovel
 At time 248.98123837008816: truck50_2 starts loading
 At time 249.1443018824285: truck50_1 request shovel
 At time 249.2110582903564: truck50_3 stop loading
 At time 249.2110582903564: truck50_3 traveling to crusher
 At time 249.2110582903564: truck20b_3 starts loading
 At time 249.23190713813324: truck20a_2 request crusher
 At time 249.23190713813324: truck20a_2 request crusher
 At time 249.48125886469836: truck20b_2 finished dumping
 At time 249.48125886469836: truck20b_2 traveling to shovel
 At time 249.48125886469836: truck20a_2 start dumping
 At time 249.7554250717336: truck20a_2 finished dumping
 At time 249.7554250717336: truck20a_2 traveling to shovel
 At time 250.98125886469836: truck20b_2 request shovel
 At time 251.2554250717336: truck20a_2 request shovel
 At time 252.2110582903564: truck50_3 request crusher
 At time 252.2110582903564: truck50_3 start dumping
 At time 253.97894191229287: truck20b_3 stop loading
 At time 253.97894191229287: truck20b_3 traveling to crusher
 At time 253.97894191229287: truck20a_3 starts loading
 At time 254.3119619839493: truck50_3 finished dumping
 At time 254.3119619839493: truck50_3 traveling to shovel
 At time 255.3652913204638: truck20b_1 stop loading
 At time 255.3652913204638: truck20b_1 traveling to crusher
 At time 255.3652913204638: truck20a_1 starts loading
 At time 256.3119619839493: truck50_3 request shovel
 At time 256.47894191229284: truck20b_3 request crusher
 At time 256.47894191229284: truck20b_3 request crusher
 At time 256.47894191229284: truck20b_3 start dumping
 At time 256.692113606732: truck20b_3 finished dumping
 At time 256.692113606732: truck20b_3 traveling to shovel
 At time 257.8652913204638: truck20b_1 request crusher
 At time 257.8652913204638: truck20b_1 request crusher
 At time 257.8652913204638: truck20b_1 start dumping
 At time 258.192113606732: truck20b_3 request shovel
 At time 258.38683910742964: truck20a_1 stop loading
 At time 258.38683910742964: truck20a_1 traveling to crusher
 At time 258.38683910742964: truck50_1 starts loading
 At time 260.4215617807219: truck20b_1 finished dumping
 At time 260.4215617807219: truck20b_1 traveling to shovel
 At time 260.88683910742964: truck20a_1 request crusher
 At time 260.88683910742964: truck20a_1 request crusher
 At time 260.88683910742964: truck20a_1 start dumping
 At time 261.1883573466092: truck20a_1 finished dumping
 At time 261.1883573466092: truck20a_1 traveling to shovel
 At time 261.9215617807219: truck20b_1 request shovel
 At time 262.6883573466092: truck20a_1 request shovel
 At time 262.9736102914716: truck20a_3 stop loading
 At time 262.9736102914716: truck20a_3 traveling to crusher
 At time 262.9736102914716: truck50_3 starts loading
 At time 265.0571354643896: truck50_2 stop loading
 At time 265.0571354643896: truck50_2 traveling to crusher
 At time 265.0571354643896: truck20b_2 starts loading
 At time 265.4736102914716: truck20a_3 request crusher
 At time 265.4736102914716: truck20a_3 request crusher
 At time 265.4736102914716: truck20a_3 start dumping
 At time 266.0168478327776: truck50_3 stop loading
 At time 266.0168478327776: truck50_3 traveling to crusher

@Hak cipta milik IPB University

IPB University

Hak Cipta Dilindungi Undang-undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 266.0168478327776: truck20b_3 starts loading
 At time 266.6301696849826: truck20a_3 finished dumping
 At time 266.6301696849826: truck20a_3 traveling to *shovel*
 At time 268.0571354643896: truck50_2 request crusher
 At time 268.0571354643896: truck50_2 start dumping
 At time 268.1301696849826: truck20a_3 request *shovel*
 At time 268.17642822810984: truck20b_3 stop loading
 At time 268.17642822810984: truck20b_3 traveling to crusher
 At time 268.17642822810984: truck20a_3 starts loading
 At time 268.8482621365276: truck20b_2 stop loading
 At time 268.8482621365276: truck20b_2 traveling to crusher
 At time 268.8482621365276: truck20a_2 starts loading
 At time 269.0168478327776: truck50_3 request crusher
 At time 269.7491305175091: truck50_1 stop loading
 At time 269.7491305175091: truck50_1 traveling to crusher
 At time 269.7491305175091: truck20b_1 starts loading
 At time 270.67642822810984: truck20b_3 request crusher
 At time 270.67642822810984: truck20b_3 request crusher
 At time 270.98534209923423: truck20a_3 stop loading
 At time 270.98534209923423: truck20a_3 traveling to crusher
 At time 271.3482621365276: truck20b_2 request crusher
 At time 271.3482621365276: truck20b_2 request crusher
 At time 272.7491305175091: truck50_1 request crusher
 At time 273.48464422864964: truck20b_1 stop loading
 At time 273.48464422864964: truck20b_1 traveling to crusher
 At time 273.48464422864964: truck20a_1 starts loading
 At time 273.48534209923423: truck20a_3 request crusher
 At time 273.48534209923423: truck20a_3 request crusher
 At time 274.9338943732584: truck20a_1 stop loading
 At time 274.9338943732584: truck20a_1 traveling to crusher
 At time 275.98464422864964: truck20b_1 request crusher
 At time 275.98464422864964: truck20b_1 request crusher
 At time 276.1051554341216: truck20a_2 stop loading
 At time 276.1051554341216: truck20a_2 traveling to crusher
 At time 277.4338943732584: truck20a_1 request crusher
 At time 277.4338943732584: truck20a_1 request crusher
 At time 278.6051554341216: truck20a_2 request crusher
 At time 278.6051554341216: truck20a_2 request crusher
 At time 280.5852151920581: truck50_2 finished dumping
 At time 280.5852151920581: truck50_2 traveling to *shovel*
 At time 280.5852151920581: truck50_3 start dumping
 At time 282.5852151920581: truck50_2 request *shovel*
 At time 282.5852151920581: truck50_2 starts loading
 At time 290.7488465273512: truck50_2 stop loading
 At time 290.7488465273512: truck50_2 traveling to crusher
 At time 291.7681036565523: truck50_3 finished dumping
 At time 291.7681036565523: truck50_3 traveling to *shovel*
 At time 291.7681036565523: truck50_1 start dumping
 At time 292.00208224962194: truck50_1 finished dumping
 At time 292.00208224962194: truck50_1 traveling to *shovel*
 At time 292.00208224962194: truck20b_3 start dumping
 At time 293.3967910595665: truck20b_3 finished dumping
 At time 293.3967910595665: truck20b_3 traveling to *shovel*
 At time 293.3967910595665: truck20b_2 start dumping
 At time 293.7488465273512: truck50_2 request crusher
 At time 293.7681036565523: truck50_3 request *shovel*
 At time 293.7681036565523: truck50_3 starts loading
 At time 294.00208224962194: truck50_1 request *shovel*
 At time 294.00208224962194: truck50_1 starts loading
 At time 294.8967910595665: truck20b_3 request *shovel*
 At time 297.6139203035932: truck50_3 stop loading
 At time 297.6139203035932: truck50_3 traveling to crusher
 At time 297.6139203035932: truck20b_3 starts loading
 At time 297.6509225007704: truck20b_2 finished dumping
 At time 297.6509225007704: truck20b_2 traveling to *shovel*
 At time 297.6509225007704: truck50_2 start dumping
 At time 299.1509225007704: truck20b_2 request *shovel*
 At time 299.1509225007704: truck20b_2 starts loading
 At time 300.2843507451012: truck20b_2 stop loading
 At time 300.2843507451012: truck20b_2 traveling to crusher
 At time 300.366060825147: truck20b_3 stop loading

@Hak cipta milik IPB University

IPB University

- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 300.3660660825147: truck20b_3 traveling to crusher
 At time 300.6139203035932: truck50_3 request crusher
 At time 302.7843507451012: truck20b_2 request crusher
 At time 302.7843507451012: truck20b_2 request crusher
 At time 302.8660660825147: truck20b_3 request crusher
 At time 302.8660660825147: truck20b_3 request crusher
 At time 304.49522874004646: truck50_1 stop loading
 At time 304.49522874004646: truck50_1 traveling to crusher
 At time 307.49522874004646: truck50_1 request crusher
 At time 309.30815903127456: truck50_2 finished dumping
 At time 309.30815903127456: truck50_2 traveling to shovel
 At time 309.30815903127456: truck50_3 start dumping
 At time 311.30815903127456: truck50_2 request shovel
 At time 311.30815903127456: truck50_2 starts loading
 At time 311.8595808030233: truck50_3 finished dumping
 At time 311.8595808030233: truck50_3 traveling to shovel
 At time 311.8595808030233: truck50_1 start dumping
 At time 312.703204146754: truck50_2 stop loading
 At time 312.703204146754: truck50_2 traveling to crusher
 At time 313.8595808030233: truck50_3 request shovel
 At time 313.8595808030233: truck50_3 starts loading
 At time 314.380913288315: truck50_1 finished dumping
 At time 314.380913288315: truck50_1 traveling to shovel
 At time 314.380913288315: truck20a_3 start dumping
 At time 315.703204146754: truck50_2 request crusher
 At time 316.380913288315: truck50_1 request shovel
 At time 316.380913288315: truck50_1 starts loading
 At time 316.6317581642844: truck20a_3 finished dumping
 At time 316.6317581642844: truck20a_3 traveling to shovel
 At time 316.6317581642844: truck50_2 start dumping
 At time 317.3293426159153: truck50_3 stop loading
 At time 317.3293426159153: truck50_3 traveling to crusher
 At time 318.1317581642844: truck20a_3 request shovel
 At time 318.1317581642844: truck20a_3 starts loading
 At time 318.3689010458438: truck50_2 finished dumping
 At time 318.3689010458438: truck50_2 traveling to shovel
 At time 318.3689010458438: truck20b_1 start dumping
 At time 320.31643403535793: truck20a_3 stop loading
 At time 320.31643403535793: truck20a_3 traveling to crusher
 At time 320.3293426159153: truck50_3 request crusher
 At time 320.3689010458438: truck50_2 request shovel
 At time 320.3689010458438: truck50_2 starts loading
 At time 321.4461592218987: truck50_1 stop loading
 At time 321.4461592218987: truck50_1 traveling to crusher
 At time 321.6984136662123: truck20b_1 finished dumping
 At time 321.6984136662123: truck20b_1 traveling to shovel
 At time 321.6984136662123: truck50_3 start dumping
 At time 322.74493510482046: truck50_2 stop loading
 At time 322.74493510482046: truck50_2 traveling to crusher
 At time 322.81643403535793: truck20a_3 request crusher
 At time 322.81643403535793: truck20a_3 request crusher
 At time 323.1984136662123: truck20b_1 request shovel
 At time 323.1984136662123: truck20b_1 starts loading
 At time 323.88878989038636: truck50_3 finished dumping
 At time 323.88878989038636: truck50_3 traveling to shovel
 At time 323.88878989038636: truck20a_1 start dumping
 At time 324.4461592218987: truck50_1 request crusher
 At time 325.74493510482046: truck50_2 request crusher
 At time 325.88878989038636: truck50_3 request shovel
 At time 325.88878989038636: truck50_3 starts loading
 At time 326.4069674863255: truck20a_1 finished dumping
 At time 326.4069674863255: truck20a_1 traveling to shovel
 At time 326.4069674863255: truck50_1 start dumping
 At time 327.9069674863255: truck20a_1 request shovel
 At time 330.83313885082646: truck50_3 stop loading
 At time 330.83313885082646: truck50_3 traveling to crusher
 At time 333.83313885082646: truck50_3 request crusher
 At time 335.6909387441323: truck20b_1 stop loading
 At time 335.6909387441323: truck20b_1 traveling to crusher
 At time 335.6909387441323: truck20a_1 starts loading
 At time 337.33109561432224: truck50_1 finished dumping

@Hak cipta milik IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 337.33109561432224: truck50_1 traveling to *shovel*
 At time 337.33109561432224: truck50_2 start dumping
 At time 338.1909387441323: truct20b_1 request crusher
 At time 338.1909387441323: truct20b_1 request crusher
 At time 339.33109561432224: truck50_1 request *shovel*
 At time 340.44935705640614: truct20a_1 stop loading
 At time 340.44935705640614: truct20a_1 traveling to crusher
 At time 340.44935705640614: truck50_1 starts loading
 At time 342.5904545038684: truck50_2 finished dumping
 At time 342.5904545038684: truck50_2 traveling to *shovel*
 At time 342.5904545038684: truck50_3 start dumping
 At time 342.94935705640614: truct20a_1 request crusher
 At time 342.94935705640614: truct20a_1 request crusher
 At time 344.42442158090205: truck50_1 stop loading
 At time 344.42442158090205: truck50_1 traveling to crusher
 At time 344.5904545038684: truck50_2 request *shovel*
 At time 344.5904545038684: truck50_2 starts loading
 At time 345.97231976467987: truck50_3 finished dumping
 At time 345.97231976467987: truck50_3 traveling to *shovel*
 At time 345.97231976467987: truck20a_2 start dumping
 At time 346.4407720698822: truck20a_2 finished dumping
 At time 346.4407720698822: truck20a_2 traveling to *shovel*
 At time 346.4407720698822: truck20b_2 start dumping
 At time 346.47075413961113: truck20b_2 finished dumping
 At time 346.47075413961113: truck20b_2 traveling to *shovel*
 At time 346.47075413961113: truck20b_3 start dumping
 At time 347.42442158090205: truck50_1 request crusher
 At time 347.9407720698822: truck20a_2 request *shovel*
 At time 347.97075413961113: truck20b_2 request *shovel*
 At time 347.97231976467987: truck50_3 request *shovel*
 At time 347.97231976467987: truck50_3 starts loading
 At time 351.7509907553673: truck20b_3 finished dumping
 At time 351.7509907553673: truck20b_3 traveling to *shovel*
 At time 351.7509907553673: truck50_1 start dumping
 At time 353.02580813838966: truck50_3 stop loading
 At time 353.02580813838966: truck50_3 traveling to crusher
 At time 353.2509907553673: truck20b_3 request *shovel*
 At time 353.2509907553673: truck20b_3 starts loading
 At time 354.2501151498755: truck50_1 finished dumping
 At time 354.2501151498755: truck50_1 traveling to *shovel*
 At time 354.2501151498755: truck20a_3 start dumping
 At time 354.33533501549704: truck20a_3 finished dumping
 At time 354.33533501549704: truck20a_3 traveling to *shovel*
 At time 354.33533501549704: truct20b_1 start dumping
 At time 355.69327628213136: truct20b_1 finished dumping
 At time 355.69327628213136: truct20b_1 traveling to *shovel*
 At time 355.69327628213136: truct20a_1 start dumping
 At time 355.83533501549704: truck20a_3 request *shovel*
 At time 355.8992766545913: truct20a_1 finished dumping
 At time 355.8992766545913: truct20a_1 traveling to *shovel*
 At time 356.02580813838966: truck50_3 request crusher
 At time 356.02580813838966: truck50_3 start dumping
 At time 356.2326940820991: truck20b_3 stop loading
 At time 356.2326940820991: truck20b_3 traveling to crusher
 At time 356.2326940820991: truck20a_3 starts loading
 At time 356.2501151498755: truck50_1 request *shovel*
 At time 356.2501151498755: truck50_1 starts loading
 At time 357.123283997251: truck50_3 finished dumping
 At time 357.123283997251: truck50_3 traveling to *shovel*
 At time 357.19327628213136: truct20b_1 request *shovel*
 At time 357.3992766545913: truct20a_1 request *shovel*
 At time 358.7326940820991: truck20b_3 request crusher
 At time 358.7326940820991: truck20b_3 request crusher
 At time 358.7326940820991: truck20b_3 start dumping
 At time 359.123283997251: truck50_3 request *shovel*
 At time 359.9525602408145: truck20b_3 finished dumping
 At time 359.9525602408145: truck20b_3 traveling to *shovel*
 At time 360.26906573358275: truck50_1 stop loading
 At time 360.26906573358275: truck50_1 traveling to crusher
 At time 360.26906573358275: truct20b_1 starts loading
 At time 360.5270447847431: truck50_2 stop loading

At time 360.5270447847431: truck50_2 traveling to crusher
 At time 360.5270447847431: truck20a_2 starts loading
 At time 360.8127131039637: truck20a_2 stop loading
 At time 360.8127131039637: truck20a_2 traveling to crusher
 At time 360.8127131039637: truck20b_2 starts loading
 At time 361.4525602408145: truck20b_3 request *shovel*
 At time 363.26906573358275: truck50_1 request crusher
 At time 363.26906573358275: truck50_1 start dumping
 At time 363.3127131039637: truck20a_2 request crusher
 At time 363.3127131039637: truck20a_2 request crusher
 At time 363.37731526759353: truck20b_2 stop loading
 At time 363.37731526759353: truck20b_2 traveling to crusher
 At time 363.5270447847431: truck50_2 request crusher
 At time 364.80799930946154: truck50_1 finished dumping
 At time 364.80799930946154: truck50_1 traveling to *shovel*
 At time 364.80799930946154: truck50_2 start dumping
 At time 365.403211680521: truck20b_1 stop loading
 At time 365.403211680521: truck20b_1 traveling to crusher
 At time 365.403211680521: truck20a_1 starts loading
 At time 365.54218244095495: truck20a_1 stop loading
 At time 365.54218244095495: truck20a_1 traveling to crusher
 At time 365.87731526759353: truck20b_2 request crusher
 At time 365.87731526759353: truck20b_2 request crusher
 At time 365.9048746058315: truck20a_3 stop loading
 At time 365.9048746058315: truck20a_3 traveling to crusher
 At time 365.9048746058315: truck50_3 starts loading
 At time 366.80799930946154: truck50_1 request *shovel*
 At time 366.80799930946154: truck50_1 starts loading
 At time 367.903211680521: truck20b_1 request crusher
 At time 367.903211680521: truck20b_1 request crusher
 At time 368.04218244095495: truck20a_1 request crusher
 At time 368.04218244095495: truck20a_1 request crusher
 At time 368.4048746058315: truck20a_3 request crusher
 At time 368.4048746058315: truck20a_3 request crusher
 At time 375.6330594075503: truck50_2 finished dumping
 At time 375.6330594075503: truck50_2 traveling to *shovel*
 At time 375.6330594075503: truck20a_2 start dumping
 At time 375.7640716779856: truck20a_2 finished dumping
 At time 375.7640716779856: truck20a_2 traveling to *shovel*
 At time 375.7640716779856: truck20b_2 start dumping
 At time 377.2640716779856: truck20a_2 request *shovel*
 At time 377.2640716779856: truck20a_2 starts loading
 At time 377.6330594075503: truck50_2 request *shovel*
 At time 378.2580659793041: truck20b_2 finished dumping
 At time 378.2580659793041: truck20b_2 traveling to *shovel*
 At time 378.2580659793041: truck20b_1 start dumping
 At time 379.7580659793041: truck20b_2 request *shovel*
 At time 379.8899290136891: truck20b_1 finished dumping
 At time 379.8899290136891: truck20b_1 traveling to *shovel*
 At time 379.8899290136891: truck20a_1 start dumping
 At time 381.3899290136891: truck20b_1 request *shovel*
 At time 385.66614731451597: truck20a_2 stop loading
 At time 385.66614731451597: truck20a_2 traveling to crusher
 At time 385.66614731451597: truck50_2 starts loading
 At time 387.99396536521675: truck20a_1 finished dumping
 At time 387.99396536521675: truck20a_1 traveling to *shovel*
 At time 387.99396536521675: truck20a_3 start dumping
 At time 388.16614731451597: truck20a_2 request crusher
 At time 388.16614731451597: truck20a_2 request crusher
 At time 389.49396536521675: truck20a_1 request *shovel*
 At time 390.91552920813587: truck50_2 stop loading
 At time 390.91552920813587: truck50_2 traveling to crusher
 At time 390.91552920813587: truck20b_2 starts loading
 At time 391.45349572875796: truck20a_3 finished dumping
 At time 391.45349572875796: truck20a_3 traveling to *shovel*
 At time 391.45349572875796: truck20a_2 start dumping
 At time 391.645315891964: truck50_1 stop loading
 At time 391.645315891964: truck50_1 traveling to crusher
 At time 391.645315891964: truck20b_1 starts loading
 At time 392.73528413072864: truck20b_1 stop loading
 At time 392.73528413072864: truck20b_1 traveling to crusher

@Hak cipta milik IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 392.73528413072864: truck20a_1 starts loading
 At time 392.95349572875796: truck20a_3 request shovel
 At time 393.46831833951506: truck20a_2 finished dumping
 At time 393.46831833951506: truck20a_2 traveling to shovel
 At time 393.91552920813587: truck50_2 request crusher
 At time 393.91552920813587: truck50_2 start dumping
 At time 394.645315891964: truck50_1 request crusher
 At time 394.72001502787174: truck20b_2 stop loading
 At time 394.72001502787174: truck20b_2 traveling to crusher
 At time 394.96831833951506: truck20a_2 request shovel
 At time 394.96831833951506: truck20a_2 starts loading
 At time 395.23528413072864: truck20b_1 request crusher
 At time 395.23528413072864: truck20b_1 request crusher
 At time 397.22001502787174: truck20b_2 request crusher
 At time 397.22001502787174: truck20b_2 request crusher
 At time 397.3801091024485: truck50_2 finished dumping
 At time 397.3801091024485: truck50_2 traveling to shovel
 At time 397.3801091024485: truck50_1 start dumping
 At time 399.3801091024485: truck50_2 request shovel
 At time 400.0806895050963: truck50_3 stop loading
 At time 400.0806895050963: truck50_3 traveling to crusher
 At time 400.0806895050963: truck20b_3 starts loading
 At time 402.84649913798626: truck20a_2 stop loading
 At time 402.84649913798626: truck20a_2 traveling to crusher
 At time 402.84649913798626: truck50_2 starts loading
 At time 403.0806895050963: truck50_3 request crusher
 At time 403.2928949072291: truck20b_3 stop loading
 At time 403.2928949072291: truck20b_3 traveling to crusher
 At time 403.2928949072291: truck20a_3 starts loading
 At time 404.779315766101: truck20a_1 stop loading
 At time 404.779315766101: truck20a_1 traveling to crusher
 At time 405.34649913798626: truck20a_2 request crusher
 At time 405.34649913798626: truck20a_2 request crusher
 At time 405.7928949072291: truck20b_3 request crusher
 At time 405.7928949072291: truck20b_3 request crusher
 At time 407.279315766101: truck20a_1 request crusher
 At time 407.279315766101: truck20a_1 request crusher
 At time 412.55357885142985: truck20a_3 stop loading
 At time 412.55357885142985: truck20a_3 traveling to crusher
 At time 414.0092880834503: truck50_2 stop loading
 At time 414.0092880834503: truck50_2 traveling to crusher
 At time 415.05357885142985: truck20a_3 request crusher
 At time 415.05357885142985: truck20a_3 request crusher
 At time 417.0092880834503: truck50_2 request crusher
 At time 417.29754629154434: truck50_1 finished dumping
 At time 417.29754629154434: truck50_1 traveling to shovel
 At time 417.29754629154434: truck50_3 start dumping
 At time 419.29754629154434: truck50_1 request shovel
 At time 419.29754629154434: truck50_1 starts loading
 At time 419.3883411883791: truck50_3 finished dumping
 At time 419.3883411883791: truck50_3 traveling to shovel
 At time 419.3883411883791: truck50_2 start dumping
 At time 421.3883411883791: truck50_3 request shovel
 At time 421.3883411883791: truck50_3 starts loading
 At time 429.6879886859161: truck50_2 finished dumping
 At time 429.6879886859161: truck50_2 traveling to shovel
 At time 429.6879886859161: truck20b_1 start dumping
 At time 429.8675774979: truck50_1 stop loading
 At time 429.8675774979: truck50_1 traveling to crusher
 At time 431.6879886859161: truck50_2 request shovel
 At time 431.6879886859161: truck50_2 starts loading
 At time 432.7007052703993: truck20b_1 finished dumping
 At time 432.7007052703993: truck20b_1 traveling to shovel
 At time 432.7007052703993: truck20b_2 start dumping
 At time 432.8675774979: truck50_1 request crusher
 At time 433.20932187022197: truck20b_2 finished dumping
 At time 433.20932187022197: truck20b_2 traveling to shovel
 At time 433.20932187022197: truck50_1 start dumping
 At time 434.2007052703993: truck20b_1 request shovel
 At time 434.2007052703993: truck20b_1 starts loading
 At time 434.70932187022197: truck20b_2 request shovel

At time 434.87019479403773: truct20b_1 stop loading
 At time 434.87019479403773: truct20b_1 traveling to crusher
 At time 437.32973337156847: truck50_3 stop loading
 At time 437.32973337156847: truck50_3 traveling to crusher
 At time 437.37019479403773: truct20b_1 request crusher
 At time 437.37019479403773: truct20b_1 request crusher
 At time 440.32973337156847: truck50_3 request crusher
 At time 442.30110479787834: truck50_1 finished dumping
 At time 442.30110479787834: truck50_1 traveling to shovel
 At time 442.30110479787834: truck50_3 start dumping
 At time 444.30110479787834: truck50_1 request shovel
 At time 444.30110479787834: truck50_1 starts loading
 At time 446.0046514021132: truck50_2 stop loading
 At time 446.0046514021132: truck50_2 traveling to crusher
 At time 446.0046514021132: truck20b_2 starts loading
 At time 446.34596761401775: truck50_3 finished dumping
 At time 446.34596761401775: truck50_3 traveling to shovel
 At time 446.34596761401775: truck20a_2 start dumping
 At time 447.1175658578616: truck50_1 stop loading
 At time 447.1175658578616: truck50_1 traveling to crusher
 At time 447.25259903210326: truck20a_2 finished dumping
 At time 447.25259903210326: truck20a_2 traveling to shovel
 At time 447.25259903210326: truck20b_3 start dumping
 At time 448.34596761401775: truck50_3 request shovel
 At time 448.34596761401775: truck50_3 starts loading
 At time 448.75259903210326: truck20a_2 request shovel
 At time 449.0046514021132: truck50_2 request crusher
 At time 449.4669755626033: truck20b_3 finished dumping
 At time 449.4669755626033: truck20b_3 traveling to shovel
 At time 449.4669755626033: truck50_2 start dumping
 At time 449.7288773601377: truck20b_2 stop loading
 At time 449.7288773601377: truck20b_2 traveling to crusher
 At time 449.7288773601377: truck20a_2 starts loading
 At time 450.1175658578616: truck50_1 request crusher
 At time 450.9669755626033: truck20b_3 request shovel
 At time 451.6200796492843: truck20a_2 stop loading
 At time 451.6200796492843: truck20a_2 traveling to crusher
 At time 451.8787809837141: truck50_2 finished dumping
 At time 451.8787809837141: truck50_2 traveling to shovel
 At time 451.8787809837141: truck50_1 start dumping
 At time 452.2288773601377: truck20b_2 request crusher
 At time 452.2288773601377: truck20b_2 request crusher
 At time 453.8787809837141: truck50_2 request shovel
 At time 453.8787809837141: truck50_2 starts loading
 At time 454.1200796492843: truck20a_2 request crusher
 At time 454.1200796492843: truck20a_2 request crusher
 At time 456.6102415622555: truck50_3 stop loading
 At time 456.6102415622555: truck50_3 traveling to crusher
 At time 456.6102415622555: truck20b_3 starts loading
 At time 457.5138974056779: truck50_1 finished dumping
 At time 457.5138974056779: truck50_1 traveling to shovel
 At time 457.5138974056779: truck20a_1 start dumping
 At time 459.5138974056779: truck50_1 request shovel
 At time 459.5138974056779: truck50_1 starts loading
 At time 459.6102415622555: truck50_3 request crusher
 At time 460.0716304348185: truck20b_3 stop loading
 At time 460.0716304348185: truck20b_3 traveling to crusher
 At time 460.436877166015: truck20a_1 finished dumping
 At time 460.436877166015: truck20a_1 traveling to shovel
 At time 460.436877166015: truck50_3 start dumping
 At time 461.936877166015: truck20a_1 request shovel
 At time 462.5716304348185: truck20b_3 request crusher
 At time 462.5716304348185: truck20b_3 request crusher
 At time 466.69086315056074: truck50_3 finished dumping
 At time 466.69086315056074: truck50_3 traveling to shovel
 At time 466.69086315056074: truck20a_3 start dumping
 At time 466.7779502223933: truck50_1 stop loading
 At time 466.7779502223933: truck50_1 traveling to crusher
 At time 466.7779502223933: truck20a_1 starts loading
 At time 468.69086315056074: truck50_3 request shovel
 At time 468.69086315056074: truck50_3 starts loading

@Hak cipta milik IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

At time 469.0120867960769: truck20a_3 finished dumping
 At time 469.0120867960769: truck20a_3 traveling to *shovel*
 At time 469.0120867960769: truck20b_1 start dumping
 At time 469.24619545252176: truck50_3 stop loading
 At time 469.24619545252176: truck50_3 traveling to crusher
 At time 469.6742390943587: truck20b_1 finished dumping
 At time 469.6742390943587: truck20b_1 traveling to *shovel*
 At time 469.6742390943587: truck20b_2 start dumping
 At time 469.7779502223933: truck50_1 request crusher
 At time 470.5120867960769: truck20a_3 request *shovel*
 At time 470.5120867960769: truck20a_3 starts loading
 At time 471.1742390943587: truck20b_1 request *shovel*
 At time 472.24619545252176: truck50_3 request crusher
 At time 472.2500193071898: truck20b_2 finished dumping
 At time 472.2500193071898: truck20b_2 traveling to *shovel*
 At time 472.2500193071898: truck50_1 start dumping
 At time 473.1539398363433: truck20a_3 stop loading
 At time 473.1539398363433: truck20a_3 traveling to crusher
 At time 473.7500193071898: truck20b_2 request *shovel*
 At time 474.04445455197896: truck20a_1 stop loading
 At time 474.04445455197896: truck20a_1 traveling to crusher
 At time 474.04445455197896: truck20b_1 starts loading
 At time 474.3540084640895: truck50_1 finished dumping
 At time 474.3540084640895: truck50_1 traveling to *shovel*
 At time 474.3540084640895: truck50_3 start dumping
 At time 474.3748551742679: truck50_2 stop loading
 At time 474.3748551742679: truck50_2 traveling to crusher
 At time 474.3748551742679: truck20b_2 starts loading
 At time 475.2530059841809: truck20b_2 stop loading
 At time 475.2530059841809: truck20b_2 traveling to crusher
 At time 475.6539398363433: truck20a_3 request crusher
 At time 475.6539398363433: truck20a_3 request crusher
 At time 476.3540084640895: truck50_1 request *shovel*
 At time 476.54445455197896: truck20a_1 request crusher
 At time 476.54445455197896: truck20a_1 request crusher
 At time 477.3748551742679: truck50_2 request crusher
 At time 477.7530059841809: truck20b_2 request crusher
 At time 477.7530059841809: truck20b_2 request crusher
 At time 478.1258993213056: truck20b_1 stop loading
 At time 478.1258993213056: truck20b_1 traveling to crusher
 At time 478.1258993213056: truck50_1 starts loading
 Waktu Simulasi: 480 menit
Shovel 1
 Total truck: 55
 rata-rata waktu delay: 4.720993653185327
 rata-rata panjang antrean: 0.8022600175986926 trucks
 rata-rata utilisasi server : 0.693280866436338

Shovel 2
 Total truck: 56
 rata-rata waktu delay: 3.996544709146877
 rata-rata panjang antrean: 0.6597720557245933 trucks
 rata-rata utilisasi server : 0.6236694431599302

Shovel 3
 Total truck: 45
 rata-rata waktu delay: 6.521485894478452
 rata-rata panjang antrean: 1.0506623086650233 trucks
 rata-rata utilisasi server : 0.7161995793742485

Crusher
 Total truck: 148
 rata-rata waktu delay: 9.702485659461836
 rata-rata panjang antrean: 0.3321314144159838 trucks
 rata-rata utilisasi server : 0.897003784913983

At time 480.0: nothing

480.0

Lampiran 2 Sintaks untuk Kasus 2 menggunakan cara alternatif

```
using ConcurrentSim, Random, ResumableFunctions, Distributions
```

```
mutable struct StatStruct
    dur ::Float64
    delay ::Float64
    nPlane ::Int
    tPlane ::Float64
    nPlaneSSQ ::Int
    nDownSSQ::Int
    tDown::Float64

    function StatStruct()
        st = new()
        st.dur = 0.0
        st.delay = 0.0
        st.nPlane = 0
        st.tPlane = 0.0
        st.nPlaneSSQ =0
        st.nDownSSQ=0
        st.tDown=0.0
        return st
    end
end
```

```
aseed = [1234,34456,2345,123,33,55,67,
        6464646,88888,109854,568,98766324,6464646,333324242,
        84746,787,55562,109345,5456,635353,3334,
        8762,2879,3908,24536,52638,66673,8889]
```

```
a = [8.1,2.9,3.6,8.4,10.9,6.7,3.0]
```

```
A = [0.7,0.9,0.8,1.9,0.7,0.9,1.6]
```

```
B = [2.1,1.8,1.6,2.8,2.2,1.7,2.0]
```

```
p = [0.3,0.26,0.18,0.12, 0.36,0.14,0.21]
```

```
r = [2.1,1.8,1.6,3.1,2.2,1.7,2.8]
```

```
c = [2.1,1.7,1.0,3.9, 1.4, 1.1,3.7]
```

```
nEngine = [4 3 2 4 4 2 3 ]
```

```
rng = []
for i=1:28
    push!(rng,MersenneTwister(aseed[i]))
end
```

```
function getNextIA(planeType::Int)
    return a[planeType]*randexp(rng[planeType],Float32)
end
```

```
function getNextIT1(planeType::Int)
    return rand(rng[7+planeType],Uniform(A[planeType],B[planeType]))
end
```

```
function getNextIT2(planeType::Int)
    return rand(rng[7+planeType],Uniform(A[planeType]/2,B[planeType]/2))
end
```

```

function getNextDec1(planeType)
    return rand(rng[14+planeType],Bernoulli(p[planeType]))
end

function getNextDec2(planeType)
    return rand(rng[14+planeType],Bernoulli(p[planeType]/2))
end

function getNextRep1(planeType)
    return rand(rng[21+planeType],Erlang(2,1/r[planeType]))
end

function getNextRep2(planeType)
    return rand(rng[21+planeType],Erlang(2,1/(r[planeType]/2)))
end

@resumable function regularPlane(sim::Simulation,planeType::Int,
    serviceStation::Resource, st::StatStruct)
    start_time = now(sim)
    st.nPlane += 1
    st.nPlaneSSQ += 1
    st.nDownSSQ += 1
    @yield request(serviceStation; priority=2 ) # lower priority
    st.delay += now(sim)-start_time
    st.tPlane += st.nPlaneSSQ * (now(sim)-start_time)
    st.nPlaneSSQ -= 1

    for i=1:nEngine[planeType]
        firstInspect = true
        firstRepair = true
        engineDone = false
        while !engineDone
            # inspecting engine
            if firstInspect
                @yield timeout(sim,getNextIT1(planeType))
                firstInspect = false
            else
                @yield timeout(sim,getNextIT2(planeType))
            end
            if firstRepair
                needRepair = getNextDec1(planeType)
            else
                needRepair = getNextDec2(planeType)
            end

            if needRepair
                # repair
                if firstRepair
                    @yield timeout(sim,getNextRep1(planeType))
                    firstRepair = false
                else
                    @yield timeout(sim,getNextRep2(planeType))
                end
            else
                engineDone = true
            end
        end
    end
end

```

```

@yield release(serviceStation)
st.tDown += st.nDownSSQ*(now(sim)-start_time)
st.nDownSSQ -= 1
st.dur += now(sim)-start_time
end

@resumable function widebodyPlane(sim::Simulation,planeType::Int,
    serviceStation::Resource, st::StatStruct)
    start_time = now(sim)
    st.nPlane += 1
    st.nPlaneSSQ += 1
    st.nDownSSQ += 1
    @yield request(serviceStation; priority=1)
    st.delay += now(sim)-start_time
    st.tPlane += st.nPlaneSSQ * (now(sim)-start_time)
    st.nPlaneSSQ -= 1

    for i=1:nEngine[planeType]
        firstInspect = true
        firstRepair = true
        engineDone = false
        while !engineDone
            # inspecting engine
            if firstInspect
                @yield timeout(sim,getNextIT1(planeType))
                firstInspect = false
            else
                @yield timeout(sim,getNextIT2(planeType))
            end
            if firstRepair
                needRepair = getNextDec1(planeType)
            else
                needRepair = getNextDec2(planeType)
            end

            if needRepair
                # repair
                if firstRepair
                    @yield timeout(sim,getNextRep1(planeType))
                    firstRepair = false
                else
                    @yield timeout(sim,getNextRep2(planeType))
                end
            else
                engineDone = true
            end
        end
    end
    @yield release(serviceStation)
    st.tDown += st.nDownSSQ*(now(sim)-start_time)
    st.nDownSSQ -= 1
    st.dur += now(sim)-start_time
end

```

@Hak cipta milik IPB University

- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

```

@resumable function planeGenerator(sim::Simulation, planeType::Int,
                                   serviceStation1::Resource,
                                   serviceStation2::Resource, stat::StatStruct)

    while true
        ia = getNextIA(planeType)
        @yield timeout(sim,ia)
        if (planeType==4) || (planeType==7)
            @process widebodyPlane(sim,planeType,serviceStation1,stat)
        else
            @process regulerPlane(sim,planeType,serviceStation2,stat)
        end
    end
end

function sim_3(n1,n2)
    sim = Simulation()
    serviceStation1 = Resource(sim,n1)
    serviceStation2 = Resource(sim,n2)
    aStat = []
    for i=1:7
        stat = StatStruct()
        @process planeGenerator(sim,i,serviceStation1,serviceStation2,stat)
        push!(aStat,stat)
    end
end

```

@Hak cipta milik IPB University

IPB University



- Hak Cipta Dilindungi Undang-undang
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah
 - b. Pengutipan tidak merugikan kepentingan yang wajar IPB University.
 2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin IPB University.

RIWAYAT HIDUP

Penulis dilahirkan di Kabupaten Kulonprogo pada tanggal 1 Januari 2000 sebagai anak pertama dari pasangan bapak Supriyono dan ibu Sumijah. Pendidikan sekolah menengah atas (SMA) ditempuh di sekolah SMAN 10 Bogor dan lulus pada tahun 2018. Pada tahun 2018, penulis diterima sebagai mahasiswa program sarjana (S-1) di Program Studi Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam di IPB.

Selama mengikuti program S-1, penulis aktif mengikuti kegiatan organisasi seperti Serambi Ruhiyah Mahasiswa FMIPA (Serum-G) pada periode 1441 H sebagai Staf Divisi Kemuslimahan, dan periode 1442 H sebagai Bendahara Divisi Kemuslimahan. Adapun kegiatan kepanitiaan, beberapa di antaranya adalah Staf Divisi Desain, Dekorasi dan Dokumentasi (DDD) (2019) dan Staf Divisi Koordinator (2020) pada Pesta Sains Nasional (PSN).