

LAPORAN TUGAS BESAR

IF2110/Algoritma dan Struktur Data

MOBILITA KURIR


Dipersiapkan oleh:

Kelompok 9 K03

1. 13520122 – Alifia Rahmah
2. 13520133 – Jevant Jedidia Augustine
3. 13520140 – Febryola Kurnia Putri
4. 13520153 – Vito Ghifary
5. 13520160 – Willy Wilsen

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

	Sekolah Teknik Elektro dan Informatika ITB	Nomor Dokumen		Halaman
		<i>IF2110-TB-09-03</i>		<i>18</i>
		<i>Revisi</i>	<i>0</i>	<i>18 November 2021</i>

Daftar Isi

1 Ringkasan.....	3
2 Penjelasan Tambahan Spesifikasi Tugas	3
2.1 VIP	3
2.2 Return to Sender	3
2.3 Return.....	3
2.4 Exit.....	4
3 Struktur Data.....	4
3.1 Point	4
3.2 List Statis	4
3.3 List Dinamis.....	4
3.4 Matriks	4
3.5 Mesin Karakter & Mesin Kata.....	5
3.6 Queue	5
3.7 Stack.....	5
3.8 Linked List.....	5
3.9 Item	5
3.10 Pesanan	6
3.11 Gadget.....	6
4 Program Utama	6
5 Data Test	7
5.1 File Configuration.....	7
5.2 End of Game	7
6 Test Script.....	7
7 Pembagian Kerja dalam Kelompok	13
8 Lampiran.....	14
8.1 Deskripsi Tugas Besar 2	14
8.2 Notulen Rapat	14
8.3 Log Activity Anggota Kelompok	15
8.4 Form Asistensi Tubes	15

1 Ringkasan

Akibat adanya pandemi COVID-19, penghasilan orang tua Mobita terus merosot. Mobita memikirkan cara untuk membantu orang tuanya di situasi seperti ini. Dengan adanya keringanan PPKM, akhirnya Mobita berakhir menjadi seorang kurir untuk membantu kedua orang tuanya. Kamu diminta untuk membantu Mobita membuat aplikasi yang akan sangat membantunya dalam menjadi kurir.

Dalam aplikasi ini, mula-mula Mobita sudah mempersiapkan terlebih dahulu hal-hal yang dibutuhkan seperti ukuran peta, koordinat headquarters, jumlah lokasi, karakter lokasi dan koordinat lokasi, matriks hubungan lokasi (*adjacency matrix*), jumlah pesanan, dan jenis pesanan yang akan masuk sebelum waktu dimulai.

Setelah waktu dimulai, program dapat mensimulasikan kegiatan-kegiatan layaknya seorang kurir seperti berpindah, melacak pesanan, navigasi peta, mengambil dan menurunkan barang.

Dalam tugas besar ini, kami membuat sebuah permainan berbasis *command-line interface* (CLI) berupa simulasi pengantaran barang menggunakan bahasa C. Dalam permainan ini, kami membuat mekanisme untuk melacak pesanan, melakukan navigasi dengan peta, mengambil, dan menurunkan barang. Permainan ini dibuat dengan memanfaatkan struktur data ADT Point, ADT List Statis, ADT List Dinamis, ADT Matriks, ADT Mesin Kata, ADT Stack, ADT Queue, dan ADT Linked List.

Program aplikasi yang telah dibuat dapat melaksanakan sebagian besar mekanisme yang diminta dengan baik.

2 Penjelasan Tambahan Spesifikasi Tugas

2.1 VIP

VIP merupakan salah satu dari jenis Pesanan. Pada saat Pesanan tipe VIP masuk ke To Do List, Pesanan tipe VIP harus segera dilayani, sehingga Mobita hanya dapat melakukan pick up pada Pesanan bertipe VIP. Apabila berhasil diantarkan, Pesanan bertipe VIP akan memberikan reward Return to Sender dan uang sebesar 600 Yen.

2.2 Return to Sender

Return to Sender merupakan reward yang didapat setelah Mobita berhasil mengantarkan Pesanan bertipe VIP. Setelah Return to Sender didapatkan, command RETURN dapat diakses pada pemilihan command.

2.3 Return

Command RETURN akan mengembalikan Pesanan teratas pada tas Mobita kepada lokasi pick up awalnya dan khusus untuk Pesanan bertipe perishable item, mengembalikan waktu expirinya menjadi semula. Command RETURN akan gagal bila Pesanan yang berada di atas tas Mobita bertipe VIP.

2.4 Exit

Command EXIT dapat dilakukan di dalam permainan. Bila dieksekusi, akan mengakhiri bermain Mobita.

3 Struktur Data (ADT)

3.1 Point

Struktur data Point berisikan tipe POINT yang terdiri atas X (absis) bertipe integer, Y (ordinat) bertipe integer, dan name bertipe char. ADT Point digunakan untuk merepresentasikan koordinat bangunan pada peta. ADT Point dipilih karena dapat merepresentasikan bangunan dan koordinatnya dengan baik pada peta yang merupakan data tipe Matrix. Struktur data POINT diimplementasikan sebagai ADT Point dengan nama file point.c dan point.h.

3.2 List Statis

Struktur data List Statis berisikan tipe ListGadget yang terdiri dari array of Gadget. Tipe dari Gadget itu sendiri merupakan char. ADT List Statis dengan representasi secara implisit tidak rata kiri (menggunakan GADGET_UNDEF sebagai mark) digunakan untuk merepresentasikan inventory gadget yang digunakan untuk menampung gadget yang telah dibeli. Gadget yang berada di inventory kemudian dapat di pilih untuk diaktivasi. ADT List Statis dipilih untuk merepresentasikan inventory gadget karena jumlah kapasitas dari inventory gadget selalu tetap (berjumlah 5) dan penempatan element Gadget pada List Statis menyerupai penempatan gadget yang sudah dibeli pada inventory. Struktur data List Statis diimplementasikan sebagai ADT ListGadget dengan nama file listgadget.h dan listgadget.c, sedangkan tipe Gadget diimplementasikan dengan nama file gadget.h dan gadget.c.

3.3 List Dinamis

Struktur data ADT list dinamis digunakan untuk menyimpan daftar bangunan yang tersedia pada peta. ADT list dinamis dipilih untuk bangunan karena daftar dan letak bangunan pada peta akan berbeda dan dapat berubah sesuai dengan kebutuhan. Oleh karena itu, ADT list dinamis lah yang paling sesuai untuk merepresentasikan masalah penyimpanan daftar bangunan. Elemen pada list dinamis ini berupa ListDinEltype yang bertipe data POINT. Struktur data list dinamis ini diimplementasikan pada ADT listdin dengan header listdin.h dan file c dalam bentuk listdin.c.

3.4 Matriks

Struktur data Matriks berisikan tipe MatElType yaitu char, tipe Matrix yang terdiri atas rowEff bertipe integer, colEff bertipe integer, dan array of array of MatElType. ADT Matriks digunakan untuk merepresentasikan peta dan *adjacency matrix*. ADT Matriks dipilih karena cocok untuk merepresentasikan peta dimana peta berisikan bangunan-bangunan yang direpresentasikan dengan alfabet, sedangkan untuk *adjacency matrix*, matriks hanya berisikan '0' dan '1'. Struktur data Matriks diimplementasikan sebagai ADT Matrix dengan nama file matrix.c dan matrix.h.

3.5 Mesin Karakter dan Mesin Kata

Struktur data Mesin Karakter dan Mesin Kata berisikan tipe Word yang terdiri dari length bertipe integer dan array of char contents. Struktur data ini digunakan untuk membaca input file dan mengisi peta, adjMat, dan pesanan sesuai dengan input file dan membaca command yang dimasukan oleh user. Struktur data ini dipilih karena dapat membaca input file dan memindahkan data dari file ke tipe bentukannya masing-masing. Struktur data ini juga dipilih karena dapat membaca input dari user sehingga input dapat diproses oleh program. Struktur data Mesin Karakter dan Mesin Kata diimplementasikan sebagai ADT Word Machine dan ADT Char Machine dengan nama file wordmachine.h, wordmachine.c, charmachine.h, dan charmachine.c.

3.6 Queue

Struktur data Queue berisikan tipe Pesanan yang terdiri dari time bertipe integer, pickup bertipe POINT, dropoff bertipe POINT, item bertipe Item, dan Ptime bertipe integer; dan tipe Queue yang terdiri dari idxHead bertipe integer, idxTail bertipe integer, dan buffer yang merupakan array of Pesanan. ADT Queue digunakan untuk merepresentasikan pesanan yang akan masuk secara berurutan berdasarkan komponen time pada Pesanan. ADT Queue dipilih karena mekanisme dari masuknya Pesanan mirip dengan antrian dimana bagian HEAD dari Queue akan di dequeue apabila waktu di dalam game nilainya sama dengan komponen time pada Pesanan. Struktur data Queue diimplementasikan sebagai ADT Queue dengan nama file pesanan.c dan pesanan.h.

3.7 Stack

Struktur data Stack berisikan tipe Stack yang terdiri atas idxTop bertipe integer, tasCap bertipe integer, dan array of Pesanan buffer. ADT Stack digunakan untuk merepresentasikan tas dari Mobita yang berisikan Pesanan yang sudah di pick up. ADT Stack dipilih karena konsep stack menyerupai konsep tas Mobita dimana benda yang pertama masuk akan menjadi benda yang pertama keluar (LIFO), serta benda yang dapat diproses hanyalah benda yang berada di tumpukan paling atas. Struktur data Stack diimplementasikan sebagai ADT Stack dengan nama file tas.c dan tas.h.

3.8 Linked List

Struktur data Linked List berisikan LList yang bertipe Address. Address itu sendiri bertipe pointer to node dan node merupakan tipe bentukan yang terdiri dari info bertipe Pesanan dan next bertipe Address. ADT Linked List digunakan untuk merepresentasikan todo, sebuah array yang memberi tahu pesanan yang dapat di pick up dan harus diantarkan oleh Mobita, dan inprogress, sebuah array yang memberi tahu pesanan yang sedang dikerjakan Mobita (isi dari tas Mobita). ADT Linked List dipilih karena kemudahannya untuk menghapus dan menambah elemen pertama pada list. Struktur data Linked List diimplementasikan sebagai ADT To Do dengan nama file pesanan_todo.c, pesanan_todo.h, node.c, dan node.h.

3.9 Item

Struktur data Item berisikan tipe Item yang terdiri dari type bertipe char dan expired bertipe integer. ADT Item digunakan untuk merepresentasikan item yang di akan di pick up dan

diantarkan oleh Mobita. ADT Item dipilih karena dapat merepresentasikan tipe dari item dan waktu expirenya (terdefinisi untuk perishable item dan tidak terdefinisi untuk type selain perishable item) dengan baik. Struktur data Item diimplementasikan sebagai ADT Item dengan nama file `item.c` dan `item.h`.

3.10 Pesanan

Struktur data Pesanan berikan waktu pesanan masuk bertipe integer, lokasi pick-up bertipe Point, lokasi drop-off bertipe Point, tipe Item yang dibawa, perish time bertipe integer, dan default perish time bertipe integer. Perish time digunakan sebagai countdown perishable item dari tiap pesanan, yang akan berkurang setiap berjalannya waktu. Jika countdown perish time sudah mencapai 0, akan dilakukan penghapusan pesanan dari in-progress list dan pengeluaran barang dari tas.

3.11 Gadget

Struktur data Gadget diimplementasikan sebagai ADT Gadget dengan nama file `gadget.c` dan `gadget.h`. ADT gadget direpresentasikan sebagai struktur data yang sederhana karena hanya menampilkan gadget sebagai karakter, karena dalam spesifikasi gadget langsung diinisialisasi dengan kode gadgetnya, tidak seperti ADT item yang dikelompokkan berdasarkan tipenya dan memiliki expired time.

4 Program Utama

Program utama yang telah kami buat berada di dalam satu direktori bernama *src*. Direktori *src* berisi `read_file` untuk membaca file konfigurasi, *main*, *main_menu*, dan sebuah *makefile* untuk melakukan *compile*. Direktori *src* mempunyai tiga direktori lain di dalamnya.

Direktori yang pertama yaitu *commands*. Direktori ini terdiri atas file-file `c` yang berisi perintah untuk keberjalanan permainan. Direktori ini berisi 10 command, di antaranya adalah *buy*, *pick up*, dan *drop-off*. Berikutnya adalah *config*. Fungsi direktori ini adalah untuk melakukan *load* file konfigurasi contohnya file `config.txt`. Direktori yang ketiga adalah *modules*, yaitu direktori berisi file-file ADT dan file lain yang dimanfaatkan oleh *commands*.

Pada saat dimulainya permainan, program akan masuk ke main menu, yaitu menu untuk memilih antara melakukan New Game atau Exit. Jika memilih New Game, pengguna akan diarahkan untuk memasukkan nama file konfigurasi yang terdapat pada direktori *config*. Jika file tidak ditemukan, pengguna harus mengulang masukan hingga nama file yang benar dimasukkan. Setelah itu, pengguna diarahkan ke permainan utama. Pada permainan utama, pengguna dapat memilih command yang tersedia. Pengguna memainkan game ini sebagai Mobita yang harus mengantarkan semua pesanan. Kondisi awal game ini adalah Mobita tidak mempunyai uang dan harus mengantarkan pesanan yang datang pada waktu tertentu. Target Mobita di sini adalah menyelesaikan semua pesanan dalam waktu yang singkat. Pesanan yang dapat diantarkan Mobita ada beberapa jenis, yaitu normal item, heavy item, perishable item, dan VIP item. Pengguna dapat memasukkan input seperti *BUY*, *MOVE*, *MAP*, *DROP_OFF*, *PICK_UP*, dan lain sebagainya untuk menunjang sistem pengantaran pesanan. Pada sistem *pick up*, Mobita hanya

dapat mengambil tiga barang sekaligus yang muat dalam tasnya, kecuali dengan gadget yang berfungsi seperti power-up. Contoh gadget lainnya adalah kain pembungkus waktu yang akan me-reset timer barang perishable. Jika pesanan telah habis, permainan berakhir dan akan ditampilkan jumlah pesanan yang berhasil dan waktu yang dilampaui.

5 Data Test

5.1 File Configuration (Read File)

Test ini menguji fungsi read_file, yang membaca data dari pembacaan file konfigurasi. Keluaran yang ingin ditampilkan pada test ini adalah peta dan daftar pesanan yang masuk. Input yang diberikan yaitu nama file konfigurasi yang berada di src/config, serta melakukan perintah MAP dan MOVE untuk menampilkan pesanan sesuai waktu.

5.2 End of Game

Test ini menguji akhir dari permainan yang ditandai dengan input command EXIT, atau ketika pesanan, To Do List, In Progress, dan tas kosong, serta Mobita berada di HQ. Input yang diberikan yaitu nama file konfigurasi yang berada di src/config, lalu melakukan perintah EXIT. Test kemudian dilakukan lagi dengan memberikan input nama file konfigurasi, melakukan perintah MOVE ke lokasi PICK_UP dan mengantarkan semua pesanan ke lokasi DROP_OFF. Setelah semua pesanan berhasil diantarkan, akan dilakukan perintah MOVE ke HQ.

6 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	Read File	Memeriksa konfigurasi pada awal permainan	Membuka file konfigurasi (dalam .txt), melihat Map, dan melakukan Move beberapa kali untuk melihat pesanan masuk	1 config.txt MAP MOVE 1 TO_DO	Mobita berada di titik 8(1,1) Waktu: 0 Uang yang dimiliki: 0 Yen Banyak pesanan VIP: 0 ***** *8 C D B* * E * *F G H* * I * * J * * K * * L * * M N O* * P * *A Q *	Mobita berada di titik 8(1,1) Waktu: 0 Uang yang dimiliki: 0 Yen Banyak pesanan VIP: 0 ***** *8 C D B* * E * *F G H* * I * * J * * K * * L * * M N O* * P * *A Q *

					<p>*****</p> <p>Posisi yang dapat dicapai:</p> <ol style="list-style-type: none"> 1. C (1,9) 2. E (2,3) 3. F (3,1) <p>Posisi yang dapat dipilih? (Ketik 0 jika ingin kembali)</p> <p>ENTER</p> <p>COMMAND: 1</p> <p>2 pesanan baru masuk!</p> <p>Pesanan pada To Do List:</p> <ol style="list-style-type: none"> 1. G -> N (Normal Item) 2. M -> B (Heavy Item) 	<p>*****</p> <p>Posisi yang dapat dicapai:</p> <ol style="list-style-type: none"> 1. C (1,9) 2. E (2,3) 3. F (3,1) <p>Posisi yang dapat dipilih? (Ketik 0 jika ingin kembali)</p> <p>ENTER</p> <p>COMMAND: 1</p> <p>2 pesanan baru masuk!</p> <p>Pesanan pada To Do List:</p> <ol style="list-style-type: none"> 1. G -> N (Normal Item) 2. M -> B (Heavy Item)
2	End of game	Memeriksa akhir dari game lewat command EXIT	Membuka file konfigurasi (dalam .txt), melakukan EXIT	1 test.txt EXIT	<p>Mobita berada di titik 8(1,1)</p> <p>Waktu: 0</p> <p>Uang yang dimiliki: 0 Yen</p> <p>Banyak pesanan VIP: 0</p> <p>TERIMA KASIH SUDAH BERMAIN</p> <p>-----Statistik Akhir-----</p> <p>Jumlah pesanan yang berhasil diantar: 0</p>	<p>Mobita berada di titik 8(1,1)</p> <p>Waktu: 0</p> <p>Uang yang dimiliki: 0 Yen</p> <p>Banyak pesanan VIP: 0</p> <p>TERIMA KASIH SUDAH BERMAIN</p> <p>-----Statistik Akhir-----</p> <p>Jumlah pesanan yang berhasil diantar: 0</p>

					Waktu yang terlampaui: 0	Waktu yang terlampaui: 0
3	End of Game (goal akhir tercapai)	Memeriksa akhir game dengan menyelesaikan semua pesanan	Membuka file konfigurasi (dalam .txt), Melakukan MOVE ke lokasi pick up pesanan dan mengantar kannya, setelah pesanan habis, kembali ke HQ	1 test.txt MOVE 1 PICK _UP MOVE 2 DROP_ OFF PICK _UP MOVE 1 DROP _OFF MOVE 1	Mobita berada di titik 8(1,1) Waktu: 0 Uang yang dimiliki: 0 Yen Banyak pesanan VIP: 0 Posisi yang dapat dicapai: 1. A (5,5) Posisi yang dapat dipilih? (Ketik 0 jika ingin kembali) 1 pesanan baru masuk! Mobita berada di titik A(5,5) Waktu: 1 Uang yang dimiliki: 0 Yen Banyak pesanan VIP: 0 Pesanan berupa Heavy Item berhasil diambil! Tujuan Pesanan: B Mobita berada di titik A(5,5)	Mobita berada di titik 8(1,1) Waktu: 0 Uang yang dimiliki: 0 Yen Banyak pesanan VIP: 0 Posisi yang dapat dicapai: 1. A (5,5) Posisi yang dapat dipilih? (Ketik 0 jika ingin kembali) 1 pesanan baru masuk! Mobita berada di titik A(5,5) Waktu: 1 Uang yang dimiliki: 0 Yen Banyak pesanan VIP: 0 Pesanan berupa Heavy Item berhasil diambil! Tujuan Pesanan: B Mobita berada di titik A(5,5)

					<p>Waktu: 1</p> <p>Uang yang dimiliki: 0 Yen</p> <p>Banyak pesanan VIP: 0</p> <p>Posisi yang dapat dicapai:</p> <p>1. HQ (1,1)</p> <p>2. B (10,10)</p> <p>Posisi yang dapat dipilih? (Ketik 0 jika ingin kembali)</p> <p>1 pesanan baru masuk!</p> <p>Mobita berada di titik B(10,10)</p> <p>Waktu: 3</p> <p>Uang yang dimiliki: 0 Yen</p> <p>Banyak pesanan VIP: 1</p> <p>Pesanan Heavy Item berhasil diantarkan</p> <p>Uang yang didapatkan: 400 Yen</p> <p>Ability yang didapatkan: Speed Boost</p> <p>Karena berhasil</p>	<p>Waktu: 1</p> <p>Uang yang dimiliki: 0 Yen</p> <p>Banyak pesanan VIP: 0</p> <p>Posisi yang dapat dicapai:</p> <p>1. HQ (1,1)</p> <p>2. B (10,10)</p> <p>Posisi yang dapat dipilih? (Ketik 0 jika ingin kembali)</p> <p>1 pesanan baru masuk!</p> <p>Mobita berada di titik B(10,10)</p> <p>Waktu: 3</p> <p>Uang yang dimiliki: 0 Yen</p> <p>Banyak pesanan VIP: 1</p> <p>Pesanan Heavy Item berhasil diantarkan</p> <p>Uang yang didapatkan: 400 Yen</p> <p>Ability yang didapatkan: Speed Boost</p> <p>Karena berhasil</p>
--	--	--	--	--	---	---

					<p>mengantarkan Heavy Item, anda bergerak lebih cepat!</p> <p>Mobita berada di titik B(10,10)</p> <p>Waktu: 3</p> <p>Uang yang dimiliki: 400 Yen</p> <p>Banyak pesanan VIP: 1</p> <p>Pesanan berupa VIP Item berhasil diambil!</p> <p>Tujuan Pesanan: A</p> <p>Mobita berada di titik B(10,10)</p> <p>Waktu: 3</p> <p>Uang yang dimiliki: 400 Yen</p> <p>Banyak pesanan VIP: 1</p> <p>Posisi yang dapat dicapai:</p> <p>1. A (5,5)</p> <p>Posisi yang dapat dipilih? (Ketik 0 jika ingin kembali)</p> <p>Mobita berada di titik A(5,5)</p> <p>Waktu: 3</p>	<p>mengantarkan Heavy Item, anda bergerak lebih cepat!</p> <p>Mobita berada di titik B(10,10)</p> <p>Waktu: 3</p> <p>Uang yang dimiliki: 400 Yen</p> <p>Banyak pesanan VIP: 1</p> <p>Pesanan berupa VIP Item berhasil diambil!</p> <p>Tujuan Pesanan: A</p> <p>Mobita berada di titik B(10,10)</p> <p>Waktu: 3</p> <p>Uang yang dimiliki: 400 Yen</p> <p>Banyak pesanan VIP: 1</p> <p>Posisi yang dapat dicapai:</p> <p>1. A (5,5)</p> <p>Posisi yang dapat dipilih? (Ketik 0 jika ingin kembali)</p> <p>Mobita berada di titik A(5,5)</p> <p>Waktu: 3</p>
--	--	--	--	--	--	--

					<p>Uang yang dimiliki: 400 Yen</p> <p>Banyak pesanan VIP: 1</p> <p>Pesanan VIP Item berhasil diantarkan</p> <p>Uang yang didapatkan: 600 Yen</p> <p>Ability yang didapatkan: Return To Sender</p> <p>Anda dapat menggunakan perintah RETURN!</p> <p>Pesanan sudah habis. Silahkan kembali ke Headquarter.</p> <p>Mobita berada di titik A(5,5)</p> <p>Waktu: 3</p> <p>Uang yang dimiliki: 1000 Yen</p> <p>Banyak pesanan VIP: 0</p> <p>Posisi yang dapat dicapai:</p> <p>1. HQ (1,1)</p> <p>2. B (10,10)</p> <p>Posisi yang dapat dipilih? (Ketik 0 jika</p>	<p>Uang yang dimiliki: 400 Yen</p> <p>Banyak pesanan VIP: 1</p> <p>Pesanan VIP Item berhasil diantarkan</p> <p>Uang yang didapatkan: 600 Yen</p> <p>Ability yang didapatkan: Return To Sender</p> <p>Anda dapat menggunakan perintah RETURN!</p> <p>Pesanan sudah habis. Silahkan kembali ke Headquarter.</p> <p>Mobita berada di titik A(5,5)</p> <p>Waktu: 3</p> <p>Uang yang dimiliki: 1000 Yen</p> <p>Banyak pesanan VIP: 0</p> <p>Posisi yang dapat dicapai:</p> <p>1. HQ (1,1)</p> <p>2. B (10,10)</p> <p>Posisi yang dapat dipilih? (Ketik 0 jika</p>
--	--	--	--	--	--	--

					ingin kembali)	ingin kembali)
					Game selesai!	Game selesai!
					-----Statistik Akhir-----	-----Statistik Akhir-----
					Jumlah pesanan yang berhasil diantar: 2	Jumlah pesanan yang berhasil diantar: 2
					Waktu yang terlampai: 4	Waktu yang terlampai: 4

7 Pembagian Kerja dalam Kelompok

No.	NIM – Nama	Tugas
1	13520122 - Alifia Rahmah	Merencanakan dan mengimplementasikan command BUY
		Merencanakan dan mengimplementasikan command PICK_UP
		Merencanakan dan mengimplementasikan command RETURN
2	13520133 - Jevant Jedidia Augustine	Membuat Main Menu (New Game & Exit)
		Modifikasi mesin kata dan membuat prosedur untuk membaca file
		Merencanakan dan mengimplementasikan command MAP
3	13520140 - Febryola Kurnia Putri	Membentuk ADT baru berupa tipe pesanan (berisi item, lokasi pick-up, drop-off, dan jenis item)
		Membentuk ADT baru tipe gadget (berisi validasi gadget, informasi gadget, dan harga gadget)
		Merencanakan dan mengimplementasikan command TO_DO
4	13520153 - Vito Ghifari	Merencanakan dan mengimplementasikan command IN_PROGRESS
		Merencanakan dan mengimplementasikan command MOVE
		Membuat menu HELP
5	13520160 - Willy Wilsen	Merencanakan dan mengimplementasikan command DROP_OFF
		Merencanakan dan mengimplementasikan command INVENTORY
		Mengimplementasikan efek dari pesanan VIP

8 Lampiran

8.1 Deskripsi Tugas Besar 2

Dengan adanya pandemi COVID-19, usaha orang tua Mobita mengalami penurunan pendapatan. Setelah diringangkannya PPKM, Mobita ingin membantu orang tuanya mendapat penghasilan menjadi seorang kurir. Tugas kalian adalah membantu Mobita membuat aplikasi yang akan membantunya melacak pesanan, navigasi peta, mengambil dan menurunkan barang.

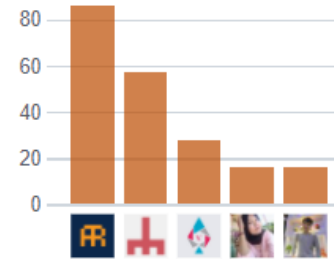
Buatlah sebuah permainan berbasis CLI (command-line interface) tentang pengantaran barang. Permainan ini dibuat dalam bahasa C dengan menggunakan struktur data yang sudah kalian pelajari di mata kuliah ini. Kalian boleh menggunakan struktur data yang sudah kalian buat untuk praktikum pada tugas besar ini.

8.2 Notulen Rapat

1. Tanggal 23 Oktober 2021
 - a. Mendiskusikan jadwal asistensi yang pertama
 - b. Melakukan pembagian kerja antar anggota
 - c. Mendiskusikan gambaran game secara umum
 - d. Membuat gambaran kasar source code di github
2. Tanggal 27 Oktober 2021
 - a. Melakukan asistensi 1 bersama
 - b. Mendiskusikan kelanjutan implementasi tugas besar
 - c. Membahas fungsi-fungsi lain yang diperlukan
 - d. Mendiskusikan jadwal asistensi kedua
3. Tanggal 10 November 2021
 - a. Melakukan asistensi 2 bersama
 - b. Menyatukan seluruh file dengan membuat adt commands.h
 - c. Melakukan fix dan debug program
4. Tanggal 17-18 November 2021
 - a. Menyelesaikan source code program
 - b. Menyelesaikan laporan
 - c. Melakukan finishing terhadap program secara keseluruhan

8.3 Log Activity Anggota Kelompok

Excluding merges, 6 authors have pushed 204 commits to master and 204 commits to all branches. On master, 0 files have changed and there have been 0 additions and 0 deletions.



Keterangan dari kiri ke kanan:

1. alifiarahmah : Alifia Rahmah
2. JevantJedidia : Jevant Jedidia Augustine
3. VanillaMacchiato : Vito Ghifary
4. febryola : Febryola Kurnia Putri
5. TubesForLyfe : Willy Wilsen

8.4 Form Asistensi Tugas Besar

Form Asistensi Tugas Besar
IF2110/Algoritma dan Struktur Data
Sem. 1 2021/2022






No. Kelompok/Kelas : 09/K03
 Nama Kelompok : bismillahKelarr :V
 Anggota Kelompok (Nama/NIM) :
 1. Alifia Rahmah/13520122
 2. Jevant Jedidia Augustine/13520133
 3. Febryola Kurnia Putri/13520140
 4. Vito Ghifari/13520153
 5. Willy Wilsen/13520160


Asisten Pembimbing : Muhammad Kamal Shafi/13518113

Asistensi 1



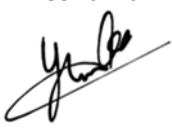

Tanggal : 27 Oktober 2021	Catatan Asistensi:
Tempat : Google Meet	
Kehadiran Anggota Kelompok:	
No	
NIM	
Tanda tangan	
1	
13520122	1. Kalau ada ADT nama yang tidak sesuai dengan namanya (misalkan Queue/Stack) tapi tetap mengimplementasikannya, apakah diperbolehkan? Sebenarnya tidak ada masalah buat itu jadi boleh.
	2. Sebaiknya di-run di linux, berarti nanti hasilnya main dan



STEI- ITB	IF2110-TB-09-03	Halaman 15 dari 18 halaman
Template dokumen ini dan informasi yang dimilikinya adalah milik Sekolah Teknik Elektro dan Informatika ITB dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Sekolah Teknik Elektro dan Informatika ITB.		

 2 13520133  3 13520140  4 13520153  5 13520160 	<p>bukan main.exe? Iya biar standar aja. Harusnya dari compiler nya nggak masalah misalkan versinya sama. Tapi mungkin masalah di formatting linux dan windows. Karena di linux ada \r\n. Biar lebih jelas aja.</p> <p>3. Kalo misal kita ada adt point gitu dan kita mau nambah-nambahin, itu kita perlu buat adt baru atau modif aja ya kak? Itu boleh dua-duanya sih, modifikasi bisa kalo ga mau modifikasi bisa buat adt baru</p> <p>4. Kalo yang implementasi adjacency matrix kan ada dari A ke B, A ke C, nah itu udah pasti simetris nggak matriksnya? Jadi kalo ada A ke B pasti ada B ke A, dst. Jawaban: Anggap aja nggak pasti dua arah. Nanti ditanyain. Dianggap sudah simetri.</p> <p>5. Misalnya beli gadget pembesar di tengah jalan, ke headquarter masih bisa dibeli kan kak? Bisa</p> <p>6. Kalo misal 1 inventory isinya gadget yang sama semua bisa kan kak? Bisa</p> <p>7. Kan dibilang di spesifikasi load game dia mengakses file konfigurasi dan file eksternal. Jadi apa masukin file nya ada 2? Iya ada 2.</p> <p>8. Untuk masukan file konfigurasi apakah formatnya sama seperti dicontoh? Iya, samakan dengan contoh.</p> <p>9. Misalnya perpindahan waktu kan hitungnya 1 waktu itu dari A ke B 1? Nggak ada hubungannya sama jarak, jadi dilihat dari matriksnya</p> <p>10. Dalam ADT mesin kata itu ada tipe data word, isinya ada panjang kata dan kata itu sendiri. Boleh nggak, hanya menggunakan tipe data tetapi prosedur seperti start word tapi copy word nya nggak dipake? Nggak ada larangan, seharusnya boleh.</p> <p>11. Gadget nya fix ada 4/5 aja kan? Iya harusnya 5.</p> <p>12. Beda To do sama in progress tuh gimana ya kak? Jadi, pesanan yang datang di waktu tertentu itu bakal masuk ke to do list, lalu kalo diambil jadi in progress</p>
---	---

	Tanda Tangan Asisten:  Digitally signed by Muhammad Kamal Shafi Date: 2021.11.12 18:25:48 +07'00'
--	--

Asistensi 2

Tanggal : 10 November 2021 Tempat : Google Meet Kehadiran Anggota Kelompok: No NIM Tanda tangan 1 13520122  2 13520133  3 13520140  4 13520153  5 13520160	Catatan Asistensi: <ol style="list-style-type: none"> Progress kalian sejauh ini gimana? Secara umum, command masing-masing udah tinggal bikin flow game sama gabungin semua file, sama fungsi buat baca file belum diimplementasikan. Saran: Kalo bisa dari sekarang coba-coba untuk buat baca filenya soalnya kalo baca dari terminal itu kan beda dengan baca dari file, harapannya bisa eksplor dari sekarang Command exit itu langsung di main menu apa gimana kak? Sesuai spek, langsung di main menu nya. Kalau mau nambahin di gamenya juga bisa. Buat gabungin semua file, apa harus di gcc nya atau .h? Gaada ketentuan, tapi dengan .h bisa jadi double include gitu. Kalo pake .h kalo sulit compilenya bisa bikin scriptnya. Waktu di-compile pake gcc, bisa semua file nya (pake *.c) biar langsung.
--	---

	
	<p>Tanda Tangan Asisten:</p> <div data-bbox="834 537 1179 730">  <p>Digitally signed by Muhammad Kamal Shafi Date: <u>2021.11.12</u> 18:25:48 +07'00'</p> </div>