

LAPORAN TUGAS BESAR
IF2111/Algoritma dan Struktur Data
Mobitangga




Dipersiapkan oleh:

Kelompok 11:

18220075 - Faizah Kamilah Setiawan
18220035 - I Dewa Gde Widhy Suryana
18220096 - Aufa Fauqi Ardhiqi
18220016 - Salsabila Asyifa Shalehah
18220065 - Hefin Immanuel Ginting

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

	Sekolah Elektro Informatika ITB	Teknik dan	Nomor Dokumen		Halaman
			<i>IF2111-TB-11</i>		<i>16</i>
			<i>Revisi</i>	<i>1</i>	<i>28 November 2021</i>

Daftar Isi

Daftar Isi	2
Ringkasan	3
Penjelasan Tambahan Spesifikasi Tugas	3
Skill Mesin Waktu	3
Skill Baling-Baling Bambu	3
Implementasi 4 Player	4
Command	4
Struktur Data (ADT)	4
ADT Map	4
ADT Mesinkar	4
ADT Mesinkata	5
ADT Player	5
ADT Skill	5
ADT Stack	6
Program Utama	6
Algoritma-Algoritma Menarik	8
Ranking Pemain	8
Print Skill	8
Pembagian Kerja dalam Kelompok	8
Lampiran	9
Deskripsi Tugas Besar	9
Notulen Rapat	10
Log Activity Anggota Kelompok.	10
Form Asistensi	13

1 Ringkasan

Mobitangga merupakan permainan yang mensimulasikan permainan ular tangga. Permainan ini akan dimainkan oleh 2-4 pemain yang dimulai dari petak yang sama (petak 1) dan berlomba-lomba untuk mencapai petak N. Dalam keberjalanan permainan, pemain akan menemukan teleporter yang bisa mencegah mencapai tujuan (bekerja seperti ular di ular tangga) dan atau mendapatkan skill untuk memudahkan mencapai tujuan. Pemenang adalah pemain yang telah mencapai petak N paling awal, selanjutnya posisi pemain saat permainan berakhir akan dapat dilihat.

Berdasarkan deskripsi permainan tersebut, kami membuat program permainan yang terdiri atas 7 komponen, yaitu Main Menu, Peta, Roll, Teleporter, Skill, Buff dan Command.

Laporan yang kami buat akan mencakup deskripsi umum persoalan, penjelasan tambahan spesifikasi, penjelasan struktur data yang digunakan, penjelasan program utama, penjelasan algoritma yang kami anggap menarik, pembagian tugas, notulensi dan lampiran lainnya.

Program dibuat menggunakan bahasa C dengan memanfaatkan ADT yang sudah dipelajari pada mata kuliah IF2111-Algoritma dan Struktur Data. Game ini dimulai dengan menampilkan main menu yang memberi pilihan pada pemain untuk melakukan New Game, Load Game, atau Exit. Pada setiap giliran, pemain dapat melakukan command secara bergiliran. Command yang valid pada game ini adalah *skill*, *map*, *buff*, *inspect*, *roll*, *save*, *endturn* dan *undo*. Permainan akan terus berlangsung bergiliran hingga ada pemain yang telah mencapai petak N. Bonus yang berhasil dibuat oleh kelompok kami adalah skill mesin waktu dan baling baling bambu, juga dukungan untuk 3 sampai 4 pemain.

2 Penjelasan Tambahan Spesifikasi Tugas

2.1 Skill Mesin Waktu

Membuat pemain lain mundur antara 1 dan MaxRoll langkah. Angka akan dirandom ketika mendapatkan skill ini. Skill ini dapat memicu teleporter bagi pemain lain, dan pemain lain dapat memilih untuk menggunakan status **Imunitas Teleport**. Skill ini tidak dapat mendorong pemain ke petak terlarang dan skill ini tetap akan terpakai.

2.2 Skill Baling-Baling Bambu

Membuat pemain lain maju antara 1 dan MaxRoll langkah. Angka akan dirandom ketika mendapatkan skill ini. Skill ini dapat memicu teleporter bagi pemain lain, dan

pemain lain dapat memilih untuk menggunakan status **Imunitas Teleport**. Skill ini tidak dapat mendorong pemain ke petak terlarang dan skill ini tetap akan terpakai.

2.3 Implementasi 4 Player

Supaya permainan Mobitangga ini dapat dimainkan oleh 4 player. Array of players dibuat untuk memudahkan menampung data dari setiap player. Array ini ada di dalam ADT Player. Di dalam ADT ini ada struktur data Players dan Player. Players berisi jumlah player dan array of Player. Struktur data Players ini memungkinkan program ini untuk menampung data per player secara individual di dalam sebuah array.

2.4 Command

Sketsa struktur data dari command terdiri atas prosedur-prosedur yang berisi perintah PrintLokasiPlayer, MAP, InfoPetak, INSPECT, Teleport, ROLL, UseSkill, skill, buff, dan UNDO.

Persoalan yang diselesaikan merupakan command yang akan di-input oleh user selama game berlangsung dan dampak jika command itu dijalankan. Alasan Pemilihannya adalah agar program lebih terstruktur, terpusat, dan jelas. Diimplementasikan sebagai ADT command dengan nama file header “command.h”.

3 Struktur Data (ADT)

3.1 ADT Map

Sketsa struktur data ADT Map : Struktur dari elemen Map info_petak terdiri dari petak (char) dan tp (integer). Sementara struktur Map terdiri atas map[MaxPetak] (info_petak) dan Panjang petak Neff (integer). Prototype pada ADT ini memuat prosedur pembuatan Map, pencetakan Map, serta set teleport.

Persoalan yang diselesaikan oleh ADT Map adalah membentuk Map, menampilkan map yang terdefinisi, dan mengeset lokasi in and out teleport pada petak. Diimplementasikan sebagai ADT Map dengan nama file header “array.h”

3.2 ADT Mesinkar

Sketsa struktur data pada ADT Mesin Karakter, terdapat sebuah variabel extern bertipe char dengan nama CC. Prototype yang ada dalam ADT ini yaitu START() dan ADV(). Persoalan yang diselesaikan adalah ADT Mesinkata. ADT ini dipilih karena

mesinkata membutuhkan mesinkar dalam fungsi-fungsi *prototype*-nya. Diimplementasikan sebagai ADT Mesinkar dengan nama file header “mesin_kar.h”

3.3 ADT Mesinkata

Sketsa struktur data pada Mesinkata adalah, struktur Kata yang terdiri atas TabKata dengan tipe data *array of char* dan Length (*integer*). BLANK yang didefinisikan dalam ADT ini yaitu ‘ ’ (spasi) dan MARK yang didefinisikan sebagai ‘\n’ (*newline*). Prototype yang terdapat pada ADT ini terdiri dari prosedur-prosedur dasar, seperti prosedur untuk mengabaikan *blank*, memulai pembacaan kata, mengakuisisi kata, menyalin kata, menulis kata ke layar, hingga fungsi-fungsi yang spesifik untuk membandingkan kata dengan *char*, mengecek apakah kata merupakan *int*, membaca kata menjadi *map*, dan membaca kata menjadi *char*.

3.4 ADT Player

Sketsa struktur data pada ADT Player adalah, struktur Player yang terdiri atas name dengan tipe data *char*, menyimpan panjang nama & posisi dalam tipe data *int*, ListSkills dengan tipe data List dan struktur Players yang menyimpan jumlah player dalam tipe data *int*. Persoalan yang diselesaikan oleh ADT ini adalah membentuk array players yang kosong, menambahkan player, mengisi data-data seperti nama, posisi, dan list skill ke player, menampilkan nama player, dan menampilkan urutan pemain. ADT ini dipilih karena diperlukan untuk menyimpan segala komponen informasi yang berkaitan dengan pemain tersebut. Diimplementasikan sebagai ADT player dengan nama file header “player.h”

3.5 ADT Skill

Sketsa struktur data list pada ADT skill terdiri atas address First, sementara untuk struktur elemennya (ElmtList) terdiri atas info (dengan tipe data urutan (*integer*)) dan next yang bertipe data address. Prototype dalam ADT ini meliputi pengecekan list kosong, pembuatan list kosong, manajemen memori alokasi dan dealokasi, pencarian elemen dalam list, penambahan elemen sebagai elemen pertama list, penambahan elemen sebagai elemen terakhir list, penghapusan elemen pertama, terakhir, dan posisi tertentu pada list, serta penghapusan skill. Selain itu terdapat proses semua elemen list berupa pencetakan list dan daftar skill ke layer, dan penghitungan jumlah elemen list. Pemrosesan list meliputi penggabungan dua list berbeda, penghasiian nilai random variabel, mencetakan skill, menu skill, dan info.

Persoalan yang diselesaikan adalah penyimpanan daftar skill yang dimiliki masing-masing player. Diimplementasikan sebagai ADT list linier dengan nama file header “listlinier.h”.

3.6 ADT Stack

Struktur data dari ADT stack: infotype dari elemen stack bertipe integer. Struktur stack terdiri atas infoRonde (infotype) dan TOP (addressAtack). Prototype manajemen memori dari ADT ini memuat fungsi dan prosedur untuk menyalin data player ke dalam list baru, menyalin skill, dealokasi skill, serta alokasi dan dealokasi stack. Selanjutnya terdapat prototype representasi logik stack dari ADT ini yang memuat fungsi/prosedur untuk pengecekan stack kosong, membuat stack kosong, menambahkan elemen stack, serta menghapus elemen stack.

Persoalan yang diselesaikan adalah fitur undo. Alasan pemilihan adalah ADT stack dapat melakukan push setiap kali ada perubahan game states dan pop ketika dilakukan Undo. Diimplementasikan sebagai ADT stack dengan nama file header “stacklist.h”.

4 Program Utama

Program utama dengan nama file main.c akan meng-include semua file header dari semua ADT. Program utama pertama akan menampilkan ucapan “WELCOME TO MOBITANGGA!!!” dan menampilkan menu utama yang dapat dipilih user (*new game dan exit*) berdasarkan indeksinya. Setelah itu program melakukan *looping* yang pertama untuk melakukan validasi terhadap input menu yang dipilih user. Pembacaan input dilakukan menggunakan mesin kata. *Looping* ini akan selesai jika user memasukkan input yang valid (bilangan 1 sampai 2). Jika input yang dimasukkan user tidak valid program akan menampilkan pesan “Input tidak valid” dan meminta user memasukkan input kembali.

Jika user memasukkan input 2 (*exit*), program akan menampilkan pesan “TERIMA KASIH MAU MENCoba GAME INI ^.^ ” ke layer. Sementara jika user memasukkan input 1 (*new game*), program akan memulai permainan baru. Kemudian program menanyakan user terlebih dahulu berapa jumlah player yang ingin bermain. Selanjutnya program akan masuk ke *loop* untuk melakukan validasi terhadap input user. Pembacaan input dilakukan dengan menggunakan mesin kata. *Looping* ini akan selesai jika user memasukkan input yang valid (bilangan 1 sampai 4). Jika input tidak valid program akan menampilkan pesan “Minimum Player 2 dan Maximum Player 4” dan meminta input kembali kepada user. Selanjutnya program akan memanggil *procedure*

AddPlayer dari ADT Player yang akan meminta input nama player dari user dan menyimpannya ke dalam array player. Setelah itu program akan melakukan voting apakah player setuju untuk menggunakan command UNDO dalam permainan. Voting dilakukan dengan memanggil fungsi VotingUndo dari ADT Command. Kemudian program meminta input berupa file konfigurasi permainan yang ingin digunakan kepada user. Lalu program utama akan mulai mengimport data konfigurasi permainan dengan membaca file eksternal config dengan memanfaatkan fungsi dan prosedur dari ADT mesin kata. Setelah selesai, program akan memanggil fungsi dari ADT stack untuk membuat sebuah stack kosong dan memulai ronde awal game.

Memasuki bagian utama permainan, dilakukan inisiasi ronde ke 1 dan variabel currentPlayer ke 0. Variabel currentPlayer dipakai untuk mengecek apakah semua player telah mengakhiri giliran sehingga ronde dapat diakhiri dan masuk ke ronde berikutnya. terdapat sebuah *loop* yang akan terus berulang selama permainan belum berakhir (salah satu pemain telah mencapai petak akhir). Kemudian program akan melakukan *looping* yang akan terus berulang selama permainan belum berakhir (salah satu pemain telah mencapai petak akhir). Di awal game dan setiap giliran akan ditampilkan posisi setiap player dengan memanggil prosedur MAP dari ADT Command. Selanjutnya ditampilkan juga giliran player yang bermain. Program kemudian memanggil prosedur RandomSkill dari ADT Skill untuk memberikan salah satu skill kepada player. Setelah itu program meminta input berupa command yang ingin digunakan user. Program pun akan masuk ke sebuah *loop* yang mengatasi input command dari user sebelum fase ROLL dijalankan. Command-command yang dapat dijalankan sebelum ROLL ialah INSPECT, SKILL, MAP, BUFF, dan UNDO. Kemudian ada pula *loop* yang mengatasi input command dari user setelah fase ROLL dijalankan. Command-command yang dapat dijalankan setelah ROLL ialah INSPECT, MAP, BUFF, UNDO, dan ENDTURN. Jika user memasukkan command “INSPECT”, program akan meminta input petak yang ingin di-*inspect* kepada user dan memanggil fungsi INSPECT dari ADT Command. Dengan demikian mekanisme inspect pun dijalankan. Jika user memasukkan command “SKILL”, program akan menampilkan daftar skill yang dimiliki user dan memanggil prosedur skills dari ADT Command. Dengan demikian mekanisme skill pun dijalankan. Jika user memasukkan command “MAP”, program akan menampilkan posisi setiap player dengan memanggil prosedur MAP dari ADT Command. Jika user memasukkan command “BUFF”, program akan memanggil prosedur Buff dari ADT Command dan mekanisme buff pun dijalankan. Jika user memasukkan command “UNDO”, program akan memanggil prosedur UNDO dari ADT command. Fungsi ini akan melakukan pop stack untuk mengembalikan game ke state sebelumnya. Jika user memasukkan command “ROLL”, program akan memanggil prosedur ROLL dari ADT command. dan mekanisme

roll pun dijalankan. Setelah itu dilakukan pengecekan apakah sudah ada player yang mencapai petak akhir. Jika ada player yang telah mencapai petak akhir, EndGame di-assign True dan program keluar dari *loop*. Jika user memasukkan command “ENDTURN”, buff Cermin Pengganda, Senter Pembesar Hoki dan Senter Pengecil Hoki akan di-reset dan giliran player saat itu akan berakhir dan berganti ke giliran player selanjutnya. Jika permainan telah mencapai akhir (telah ada player yang mencapai petak akhir) maka program akan menampilkan ucapan selamat kepada pemenang dan menampilkan ranking dari setiap player. Lalu akan menampilkan kembali main menu.

5 Algoritma-Algoritma Menarik

5.1 Ranking Pemain

Algoritma ini berada di dalam ADT Player. Algoritma ini digunakan sesaat salah satu dari pemain sudah mencapai ujung map dan program menampilkan ranking pemain berdasarkan posisi terakhirnya di map. Algoritma ini menarik karena menggunakan algoritma sorting untuk mengurutkan posisi player dari array player yang telah didefinisikan sebelumnya. Ranking player disusun dari posisi yang paling tinggi ke paling rendah. Selagi sorting posisi, dilakukan juga sorting urutan player yang akan ditampilkan di layar sehingga menghasilkan array yang terurut dari posisi yang terbesar ke yang terkecil sekaligus mengurutkan player.

5.2 Print Skill

Algoritma ini berada di dalam ADT Skill. Algoritma ini digunakan untuk menampilkan skill berdasarkan info yang ada di dalam suatu node list. Algoritma ini menarik karena diperlukan pembagian chance dari tiap skillnya. Pembagian chance dibagi menjadi beberapa range info untuk menentukan skill mana yang berlaku di range tersebut. Ketika program menjalankan fungsi tersebut, program akan mengecek setiap info listnya satu per satu untuk mengetahui jenis skillnya kemudian menampilkannya ke layar.

6 Pembagian Kerja dalam Kelompok

No.	Fitur/ADT	NIM Coder	NIM Tester
1	Konfigurasi	18220096	18220035
3	ADT_Map	18220096	18220035
4	ADT_Mesinkar	18220096	18220035
5	ADT_MesinKata	18220096	18220035
6	ADT_Player	18220075, 18220096, 18220035	18220035
7	ADT_Skill	18220035	18220096

8	Fitur Buff	18220035	18220096
9	ADT_Stack	18220016, 18220096	18220096
10	ART	18220035	18220096
11	Fitur command	18220096, 18220035	18220096
12	Fitur Roll	18220065	18220096

7 Lampiran

7.1 Deskripsi Tugas Besar

Sebuah Institut Teknologi tertentu sedang mengadakan lomba *game dev* dengan tema membuat board game digital terbaik se-kecamatan. Mendengar hal tersebut, Borakemon dan Mobita bersekongkol untuk membuat sebuah board game digital terasyik. Mereka kepikiran untuk menggabungkan game ular tangga dengan modifikasi-modifikasi yang dapat mengganggu lawan sehingga lahirlah ide Mobitangga.

Namun sayangnya, Mobita tidak memiliki kemampuan maupun niat memprogram Mobitangga. Borakemon, kucing robot Mobita, juga belum memiliki kemampuan untuk memprogram karena belum belajar terlalu *deep*. Oleh karena itu, Borakemon menculik sekelompok *programmer* dari dimensi lain agar dapat membantu mereka membuat program Mobitangga agar dapat memenangkan lomba *game dev* itu.

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

Permainan ini merupakan permainan yang berbasis dari permainan ular tangga. Permainan dilakukan oleh 2 **sampai 4 (Bonus)** pemain di suatu papan satu dimensi yang memiliki panjang N. Pada awal permainan, setiap pemain akan mulai dari petak 1 dan berlomba-lomba untuk mencapai petak N.

Saat permainan berlangsung, terdapat teleporter dan skill yang dapat digunakan oleh pemain untuk mencapai tujuan, atau untuk mencegah pemain lain mencapai tujuan. Permainan akan berakhir jika sudah ada satu pemain yang mencapai petak N. Peringkat pemain lain akan dilihat berdasarkan posisi pemain saat pemenang ditemukan.

7.2 Notulen Rapat

Tanggal	Deskripsi
5 November 2021	Membahas setelah Asistensi dan membuat keputusan untuk melakukan percobaan program terlebih dahulu tiap orang
12 November 2021	Pembagian tugas per orang

7.3 Log Activity Anggota Kelompok






Anggota	Tanggal	Deskripsi
18220035	2021-11-05	Initial commit ke Github dan memasukkan ADT yang dibuat dari praktikum
18220096	2021-11-06	Mengerjakan ADT_Array
18220096	2021-11-08	Mengerjakan ADT_Mesinkar dan ADT_Mesinkata
18220035	2021-11-12	Melakukan pembagian tugas
18220035	2021-11-14	Mengerjakan fungsi skill
18220065	2021-11-15	Mengerjakan fitur ROLL
18220075	2021-11-21	Mengerjakan ADT player


18220035	2021-11-22	Merapikan command line dan fix bug pada fungsi roll. Menambahkan fungsi baru untuk memudahkan debugging
18220096	2021-11-22	Menambahkan beberapa di ADT_Player
18220016	2021-11-22	Mengerjakan ADT stack
18220035	2021-11-23	Membuat validasi input untuk fungsi skill, melengkapi komentar, dan merapikan struktur data
18220096	2021-11-23	Mengerjakan beberapa Command seperti MAP, INSPECT
18220035	2021-11-24	Mengerjakan fungsi buff
18220096	2021-11-24	Menambahkan beberapa di ADT_Stack
18220035	2021-11-25	Merapikan print dan mengerjakan bonus skill
18220035	2021-11-26	Mencoba program di linux, membuat fungsi ranking, dan melengkapi komentar
18220065	2021-11-26	Mengerjakan laporan bagian Program Utama
18220075	2021-11-26	Mengerjakan laporan bagian ringkasan, ADT Mesinkar, ADT Mesinkata, ADT Player
18220035	2021-11-27	Menghias main menu dan command
18220096	2021-11-27	Memperbaiki bug Mesinkata

18220016	2021-11-27	Mengerjakan laporan bagian ADT Map, ADT Skill, ADT Stack, ADT Command
18220035	2021-11-28	Merapikan program final dan foldernya. Mengetes program di linux. Merapikan laporan dan mengumpulkan
18220096	2021-11-28	Merapikan main code dan memperbaiki UNDO






7.4 Form Asistensi


Asistensi I

Tanggal : Jumat, 5 November 2021	Catatan Asistensi: <ul style="list-style-type: none"> - spek yang belum jelas bikin asumsi sendiri dulu aja - teknologi gagal mau masukin daftar skill atau ngga
Tempat : Google Meet	
Kehadiran Anggota Kelompok: No NIM Tanda tangan 1 18220075  Faiza 2 18220035  3 18220096  4 18220016  5 18220065 	
	Tanda Tangan Asisten:

	 Jumat, 5 Nov 2021
--	---

Asistensi II

Tanggal : Jumat, 19 November 2021	Catatan Asistensi: ADT / Command >80% <ul style="list-style-type: none">- ADT Map sudah berjalan dengan baik namun belum bisa membaca maxroll dan teleporter. ADT / Command On Progress <ul style="list-style-type: none">- ADT Skill masih butuh perbaikan karena tidak bisa memakai skill di elemen pertama dan menghapus skill di elemen pertama. Sisanya sudah berjalan dengan baik.- ADT Player baru bisa membaca nama player dan menampilkan posisi player. ADT / Command yang Belum Dikerjakan <ul style="list-style-type: none">- ADT Undo belum dikerjakan karena ADT player belum jadi
Tempat : Google Meet	
Kehadiran Anggota Kelompok:	
No	
NIM	
Tanda tangan	
1	
18220075	
	
Faizah	
2	
18220035	
	
3	
18220096	
	
4	
18220016	
	
5	
18220065	
	

	<p>Tanda Tangan Asisten:</p>  <p>Jumat, 19 November 2021</p>