

LAPORAN TUGAS BESAR

IF2111 Algoritma dan Struktur Data


BNMO

Dipersiapkan oleh:

Kelompok 04

Muhammad Rafi Haidar / 18221134
Priscilla Auleader Napitupulu / 18221098
Laurensia Audrey Pramashinta / 1822114
Darren / 18221050
Razan Aditya Putra / 18221132

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung
Jl. Ganesha 10, Bandung 40132

	Sekolah Teknik Elektro dan Informatika ITB	Nomor Dokumen		Halaman
		<i>IF2111-TB1-04</i>		39
		<i>Revisi</i>	0	11 November 2022

Daftar Isi

1 Ringkasan	4
2 Penjelasan Tambahan Spesifikasi Tugas	4
2.1 START	4
2.2 LOAD	4
2.3 CREATEGAME	4
2.4 LISTGAME	5
2.5 DELETGAME	5
2.6 QUEUEGAME	5
2.7 PLAYGAME	5
2.8 SKIPGAME	5
2.9 QUIT	5
2.10 TIC TAC TOE	5
3 Struktur Data	6
3.1 Array	7
3.2 Queue	7
3.3 Mesin Karakter	8
3.4 Mesin Kata	8
3.5 Mesin Load	8
4 Program Utama	9
5 Algoritma-Algoritma Menarik	10
5.1 Random Number Generator	10
6 Data Test	10
6.1 START	10
6.2 LOAD	11
6.3 SAVE	11
6.4 CREATEGAME	12
6.5 LISTGAME	12
6.6 DELETGAME	13
6.7 QUEUEGAME	14
6.8 PLAYGAME	15
6.9 SKIPGAME	16
6.10 QUIT	16
6.11 HELP	16
6.12 COMMAND LAIN	17
6.13 RNG	17
6.14 DINER DASH	18
6.15 TIC TAC TOE	24
6.16 GAME TAMBAHAN	25
7 Test Script	26
8 Pembagian Kerja dalam Kelompok	30
9 Lampiran	31
9.1 Deskripsi Tugas Besar 2	31
9.2 Notulen Rapat	32

1 Ringkasan

Program BNMO merupakan program permainan berbasis *command-line interface* yang dibuat menggunakan bahasa C. BNMO mempunyai beberapa fitur utama, yaitu memainkan game, menambahkan game, menghapus game, dan mengurutkan game yang akan dimainkan. Awalnya pengguna akan masuk ke dalam main menu dan dari situ dapat memasukkan command dengan cara mengetiknya untuk menggunakan fitur-fitur BNMO.

Laporan ini berisikan penjelasan mengenai spesifikasi-spesifikasi tambahan yang ada di program, penjelasan mengenai ADT dan fungsinya di program ini, penjelasan mengenai program utama, penjelasan mengenai algoritma-algoritma menarik yang kami temukan saat mengerjakan tugas ini, data test beserta test script yang digunakan, pembagian kerja kelompok, dan lampiran yang diminta.

Tugas besar pertama mata kuliah IF2111 Algoritma dan Struktur Data meminta mahasiswa untuk mengimplementasikan hal-hal yang telah dipelajari di kelas dan praktikum mengenai kemampuan untuk membuat program kompleks di bahasa C secara modular dan pengetahuan seputar beberapa ADT yang telah diajarkan, seperti ADT Array, ADT List, ADT Queue, ADT Mesin Karakter, dan ADT Mesin Kata, serta implementasinya di bahasa C.

2 Penjelasan Tambahan Spesifikasi Tugas

2.1 START

Start memanggil fungsi load dengan file konfigurasi “config.txt”. Start hanya dapat dipanggil satu kali di awal program.

2.2 LOAD

LOAD merupakan salah satu command yang dimasukkan pertama kali oleh pemain ke BNMO. Memiliki dua argumen yaitu filename yang merepresentasikan suatu save file yang ingin dibuka , dan sebuah Array game untuk menyimpan daftar game yang telah di load. Load hanya dapat dipanggil satu kali di awal program.

2.3 CREATEGAME

Command CREATEGAME menambahkan game baru pada daftar game. Sistem akan meminta input nama game, apabila game yang ingin ditambahkan sudah ada di dalam daftar game, maka sistem akan menampilkan pesan error. Apabila belum ada, sistem akan menambahkan game tersebut ke dalam daftar game pada urutan paling terakhir.

2.4 LISTGAME

Command LISTGAME menampilkan daftar game yang disediakan oleh sistem maupun game yang ditambahkan oleh pengguna.

2.5 DELETEGAME

Command DELETEGAME menghapus game berdasarkan masukan pengguna. Namun terdapat beberapa syarat untuk melakukan penghapusan pada game yaitu apabila game yang ingin dihapus merupakan game yang disediakan oleh sistem atau game masih berada dalam command QUEUEGAME ataupun termasuk pada lima game pertama yang terdapat pada file konfigurasi tidak dapat dihapus.

2.6 QUEUEGAME

Command QUEUEGAME menambahkan game ke dalam antrian. Sistem akan terlebih dahulu menampilkan isi dari daftar antrian game. Apabila antrian kosong, maka akan menampilkan pesan error. Pada command QUEUEGAME akan dipanggil command LISTGAME yang berfungsi menampilkan daftar game yang tersedia. Akan diminta input berupa nomor game dari daftar yang ingin ditambahkan ke antrian, apabila nomor tidak valid maka akan menampilkan pesan error.

2.7 PLAYGAME

Command PLAYGAME akan menjalankan permainan. Apabila antrian game kosong, maka akan menampilkan pesan error. Sistem akan menambahkan list game pada antrian secara berurutan. Setelah itu, akan diminta input game apa yang ingin dimainkan. Apabila tidak valid, akan ditampilkan pesan error.

2.8 SKIPGAME

SKIPGAME adalah command yang digunakan untuk melewati permainan sebanyak n. Apabila queue game antrian pribadi pemain kosong maka program akan menampilkan pesan "Tidak ada permainan di dalam daftar game-mu". Apabila jumlah n melebihi jumlah permainan yang ada di daftar antrian permainan, maka program akan menampilkan pesan "Tidak ada lagi permainan di daftar game-mu". Apabila jumlah n kurang dari jumlah permainan yang ada daftar antrian permainan, akan dimainkan game dengan queue teratas saat ini setelah n game dilewatkan.

2.9 QUIT

QUIT adalah command yang digunakan untuk keluar dari permainan. Apabila menjalankan command quit maka program akan menampilkan pesan "Anda keluar dari game BNO" "Bye bye.."

2.10 TIC TAC TOE

Tic Tac Toe adalah program yang dibuat sebagai pemenuhan spesifikasi bonus merupakan sebuah game yang memungkinkan pengguna untuk melawan komputer (non-player) di dalam permainan tic tac toe. Pengguna dan komputer diminta untuk membuat sebuah garis yang berisikan 3 O / 3 X bisa secara horizontal, vertikal, dan diagonal. Pemenang dari permainan ini adalah siapa yang berhasil melakukan instruksi tersebut sebelum lawannya.

Komputer menggunakan elemen random number generator dalam menentukan gerakan selanjutnya. Pengguna akan diminta input berupa baris dan kolom oleh program pada setiap gilirannya dan program akan menampilkan grafis yang merupakan representasi dari state permainan saat itu.

3 Struktur Data (ADT)

3.1 Array

ADT Array merupakan kumpulan elemen yang tersusun secara kontigu di memori mesin. ADT ini terkandung di dalam file array.h dengan implementasinya di file array.c. ADT Array terdiri dari array statis TI yang memiliki kapasitas sebesar $\text{IdxMax} - \text{IdxMin} + 1$ untuk ElType dan nilai efektifnya, yaitu Neff.

```
#define IdxMax 100
#define IdxMin 1
#define IdxUndef -999 /* indeks tak terdefinisi*/

/* Definisi elemen dan koleksi objek */
typedef int IdxType;
typedef char* ElType;

typedef struct
{
    ElType TI [IdxMax-IdxMin+1]; /* memori tempat penyimpan elemen (container) */
    int Neff; /* banyaknya elemen efektif */
} Array;
```

Primitif-primitif yang ada di ADT ini adalah MakeEmpty untuk membuat array kosong, NbElmt yang mengembalikan jumlah elemen array, MaxNbEl yang mengembalikan jumlah elemen maksimal yang bisa dikandung array, GetFirstIdx yang mengembalikan indeks pertama array, GetLastIdx yang mengembalikan indeks terakhir array, GetElmt yang mengembalikan elemen indeks ke-i array, SetTab untuk menyalin array satu ke array lainnya, SetEl untuk mengubah nilai elemen indeks ke-i array, SetNeff untuk mengubah nilai efektif array, IsIdxValid untuk mengetahui apakah indeks ke-i valid, IsIdxEff untuk mengetahui apakah indeks ke-i di dalam range nilai efektif array, IsEmpty untuk mengetahui apakah array kosong, IsFull untuk mengetahui apakah array penuh, TulisIsi untuk mencetak array ke layar, ValMax beserta IdxMaxTab untuk mengetahui nilai maksimum elemen array dan indeksinya, dan ValMin beserta IdxMinTab untuk mengetahui nilai minimum elemen array dan indeksinya.

ADT Array digunakan di program ini untuk menyelesaikan permasalahan yang membutuhkan penyimpanan elemen-elemen secara kontigu seperti array secara umumnya. ElType untuk ADT Array di program ini adalah pointer to char karena ADT ini digunakan untuk menyimpan elemen berbentuk string seperti halnya di Array game di program utama yang menyimpan daftar game dari file konfigurasi di beserta informasi mengenai jumlah game yang dikandung di Neff-nya. Bentuk ADT Array yang dimodifikasi juga dapat ditemukan di ddash.h dimana ElType berupa integer dan array statis TI yang memiliki kapasitas 100 elemen.

3.2 Queue

ADT Queue merupakan kumpulan elemen yang diatur penyimpanan dan pengambilannya dengan aturan First In First Out (FIFO). ADT berada di dalam queue.h dengan implementasinya di queue.c dimana struktur Queue terdiri dari array statik bernama buffer yang memiliki kapasitas sebesar CAPACITY untuk Eltype dan direpresentasikan eksplisit dengan cara menyimpan nilai idxTail beserta idxHead.

```
#define IDX_UNDEF -1
#define CAPACITY 100

/* Definisi elemen dan address */
#define ElType char *
typedef struct {
    ElType buffer[CAPACITY];
    int idxHead;
    int idxTail;
} Queue;
```

Primitif-primitif yang ada di ADT ini adalah CreateQueue untuk membuat queue kosong, isEmpty untuk mengetahui apakah queue kosong, isFull untuk mengetahui apakah queue penuh, length untuk mengetahui panjang queue, enqueue untuk memasukkan elemen ke akhir queue, dequeue untuk mengeluarkan elemen terdepan queue, dan displayQueue untuk mencetak queue ke layar.

ADT ini diperlukan untuk membantu persoalan-persoalan yang membutuhkan struktur data yang dapat merepresentasikan antrian elemen dengan prinsip FIFO seperti yang ditemukan di fitur QUEUEGAME, PLAYGAME dan SKIPGAME. Di ketiga fitur tersebut, game disusun layaknya antrian dan pengguna hanya dapat memainkan game yang berada di antrian paling depan sebelum pindah ke game setelahnya.

Bentuk ADT Queue yang dimodifikasi juga dapat ditemukan di ddash.h dimana ElType berupa struktur buatan order dan beberapa primitif diubah sehingga memudahkan proses enqueue di game tersebut. terkait pembuatan sebuah q kosong serta melakukan alokasi, menentukan apakah sebuah tabel penampung elemen q penuh, mengetahui banyaknya elemen q, menambahkan sebuah val pada q, menghapus val pada q, serta menuliskan isi dari queue.

3.3 Mesin Karakter

ADT Mesin Karakter merupakan ADT yang digunakan untuk menerima dan membaca filestream yang nantinya akan digunakan di ADT mesin kata. Filestream dapat berupa file ataupun standard input berupa ketikan pengguna di keyboardnya. Mesin Karakter pertama akan diinisiasi dengan prosedur START() yang akan membaca filestream. Kemudian pengguna dapat menjalankan

prosedur ADV() yang akan membaca karakter dan memasukkannya ke currentCharacter. ADV() dapat terus dijalankan sampai program menemukan EOP dengan fungsi isEOP() di filestream.

```
char currentChar;
boolean EOP;

static FILE *pita;
static int retval;
```

Primitif tambahan yang ditambahkan adalah fungsi READINPUT yang membaca input pengguna dan mengembalikannya sebagai pointer to char.

Mesin Karakter digunakan di program ini untuk menerima input pengguna berupa command beserta argumennya seperti di fungsi utama main() dan prosedur game RNG, Diner Dash, dan Tic Tac Toe. Mesin Karakter juga digunakan untuk membaca isi file konfigurasi dan savefile di prosedur START() dan LOAD() menggunakan ADT Mesin Load.

3.4 Mesin Kata

ADT Mesin Kata merupakan ADT yang digunakan untuk mendapatkan sebuah kata yang akan disimpan di struktur Word dari filestream dengan memanfaatkan ADT Mesin Karakter. Mesin Kata pertama akan diinisiasi dengan STARTWORD() yang akan menjalankan prosedur START() Mesin Karakter. Setelah itu, Mesin Kata akan menyalin kata yang didapat ke dalam currentWord menggunakan CopyWord(). Pengguna dapat menjalankan prosedur ADVWORD() untuk maju ke kata selanjutnya yang dipisahkan oleh blank atau spasi yang ditangani oleh IgnoreBlanks(). ADVWORD() dapat terus dijalankan sampai program menemukan EOP di filestream yang akan mengubah boolean EndWord menjadi true.

```
#define NMax 50
#define BLANK ' '

typedef struct
{
    char TabWord[NMax]; /* container penyimpan kata, indeks yang dipakai [0..NMax-1] */
    int Length;
} Word;

/* State Mesin Kata */
extern boolean EndWord;
extern Word currentWord;
```

Beberapa primitif tambahan yang dibuat adalah SalinWord yang menyalin word satu ke word lainnya, blank_count yang menghitung jumlah blank, StrToInt dan StrToInt_input yang mengubah string menjadi integer, str_len yang menghitung panjang suatu string, KataPertama dan KataKedua yang mendapatkan kata pertama dan kedua dari suatu string, str_comp yang

mengecek apakah dua string identik, dan filetodir yang digunakan untuk mengubah string menjadi letak suatu direktori.

Mesin Kata digunakan di program ini di fitur LOAD dengan menggunakan bentuk yang telah dimodifikasi, yaitu Mesin Load.

3.5 Mesin Load

ADT Mesin Load merupakan modifikasi dari ADT Mesin Kata yang menggunakan fopen() untuk membuka file dan menjadikannya sebagai filestream.

```
static FILE *pita;  
static int retval;
```

Primitif-primitif ADT ini adalah StartLoad(), IgnoreNewLine(), ADVWORDLOAD(), CopyWordLOAD(), dan ADVLOAD() yang merupakan modifikasi dari primitif Mesin Kata untuk menangani file .txt.

4 Program Utama

Program utama berupa fungsi main yang merupakan driver dari console.h dan implementasinya di console.c yang berisikan prosedur-prosedur yang akan dieksekusi sesuai dengan input pengguna. Fungsi main mempunyai variabel global berupa Array game yang berisikan daftar game dari file konfigurasi yang di-load pengguna, Queue antrian yang berisikan antrian game yang akan dimainkan oleh pengguna, string input yang menampung input pengguna yang nantinya akan dipecah menjadi string command dan arg yang berisikan perintah dan argumen dari input pengguna, dan boolean load dan over yang masing-masing menandakan apakah pengguna telah me-load file konfigurasi dan apakah pengguna telah keluar dari loop utama.

Program akan menjalankan loop pertama yang akan meminta input pemain berupa perintah untuk me-load file konfigurasi atau file savefile menggunakan START atau LOAD. Pengguna tidak akan masuk ke loop utama sampai me-load salah satu dari file tersebut. Perintah lain yang dapat dijalankan pengguna saat ini hanyalah HELP.

Di loop utama, pengguna dapat menggunakan fitur utama dengan memasukan perintah yang akan menjalankan prosedur-prosedur yang tersedia, yaitu CREATE GAME, LIST GAME, DELETE GAME, QUEUE GAME, PLAY GAME, dan HELP. Pemain akan keluar dari loop utama dan dari program secara keseluruhan apabila memasukkan perintah QUIT yang akan mengubah nilai over menjadi true dan menghentikan loop.

5 Algoritma-Algoritma Menarik

5.1 Random Number Generator

Algoritma random number generator yang digunakan di program ini merupakan fungsi rand() dari library stdlib.h. Namun, angka yang diberikan bersifat pseudo random sehingga perlu menggunakan srand() untuk memberikan seed kepada fungsi rand(). Agar seed benar-benar random, kami menggunakan fungsi time() dari library time.h sebagai seed di srand() yang menghitung waktu yang terlewat dari epoch. Dengan ini, angka yang diberikan fungsi rand() akan bersifat random setiap kali program dieksekusi.

Algoritma ini digunakan di setiap game yang dapat dimainkan oleh pengguna. Di game RNG, algoritma ini digunakan untuk membangkitkan angka random yang akan ditebak oleh pemain. Di game Diner Dash, algoritma ini digunakan untuk membangkitkan waktu memasak, ketahanan, dan harga makanan setiap kali program melakukan enqueue pesanan pengguna ke antrian pesanan. Di game Tic Tac Toe, algoritma ini digunakan untuk menentukan kolom dan baris mana yang akan dipilih oleh komputer sebagai gerakan selanjutnya sebagai lawan dari pemain manusia. Di saat pengguna memainkan game buatannya, algoritma ini akan membangkitkan skor akhir pengguna.

Algoritma ini menarik karena kami belajar bahwa mesin hanya dapat membangkitkan nilai yang bersifat pseudo random dan memerlukan input seed dari pengguna ataupun algoritma lainnya agar menjadi random, atau setidaknya mendekati random.

6 Data Test

6.1 START

Fitur yang akan di test ialah start. Ketika diterima input yang valid, maka program akan mulai dijalankan. Apabila input tidak valid, maka akan ditampilkan pesan “File konfigurasi sistem berhasil dibaca. BNMO berhasil dijalankan.”

Berikut daftar test dari fitur start :

1. Apabila pengguna memasukkan input valid, program akan menjalankan fitur START, LOAD, atau HELP

```
Selamat datang di BNMO ^w^  
ENTER COMMAND: START  
File konfigurasi sistem berhasil dibaca. BNMO berhasil dijalankan.
```

2. Apabila pengguna memasukkan input invalid, program akan meminta pengguna untuk memasukkan input valid

```
Selamat datang di BNMO ^w^  
ENTER COMMAND: STARTT  
Perintah yang dapat digunakan hanya START, LOAD, atau HELP
```

6.2 LOAD

Berikut daftar test dari fitur load :

1. Apabila pengguna memasukkan input filename yang valid, program akan menjalankan LOAD dan save file berhasil dibaca.

```
Selamat datang di BNMO ^w^  
ENTER COMMAND: LOAD savefile.txt  
Save file berhasil dibaca. BNMO berhasil dijalankan.
```

2. Apabila pengguna memasukan input filename yang tidak valid, program akan gagal menjalankan LOAD dan akan meminta input COMMAND lagi.

```
Selamat datang di BNMO ^w^  
ENTER COMMAND: LOAD savefile.txtttt  
Save file gagal dibaca. Coba lagi
```

6.3 SAVE

Berikut daftar test dari fitur load :

Apabila pengguna memasukkan input filename yang valid(yang ada di folder data), program akan menjalankan SAVE dan file .txt yang diinput akan di-update.

Catatan tambahan: apabila pengguna memasukkan input filename yang tidak ada di folder data, maka program akan secara otomatis membuat file baru sesuai filename yang diinput.

Tampilan di program, TEST SAVE merupakan game yang dibuat pengguna menggunakan command CREATEGAME.

```
ENTER COMMAND: LIST GAME  
1. RNG  
2. Diner DASH  
3. TIC TAC TOE  
4. DINOSAUR IN EARTH  
5. RISEWOMAN  
6. EIFFEL TOWER  
7. TEST SAVE  
ENTER COMMAND: SAVE savefile.txt  
Save file berhasil disimpan.
```

Tampilan sesudah pengguna menjalankan command SAVE di file savefile.txt.

```
gas-Besar-1 > data > savefile.txt
1 7
2 RNG
3 Diner DASH
4 TIC TAC TOE
5 DINOSAUR IN EARTH
6 RISEWOMAN
7 EIFFEL TOWER
8 TEST SAVE
```

6.4 CREATEGAME

Fitur yang akan dites adalah game create game. Berikut merupakan daftar tes dengan hasil yang seharusnya dikeluarkan oleh program dan hasil yang diberikan program:

Program akan meminta input pengguna berupa nama game yang akan ditambahkan dan menambahkan game tersebut.

```
ENTER COMMAND: CREATE GAME
Masukkan nama game yang akan ditambahkan: TEST CREATE
ENTER COMMAND: LIST GAME
1. RNG
2. Diner DASH
3. TIC TAC TOE
4. DINOSAUR IN EARTH
5. RISEWOMAN
6. EIFFEL TOWER
7. TEST SAVE
8. TEST CREATE
```

Hasil tes create game jika berhasil.

```
ENTER COMMAND: CREATE GAME
Masukkan nama game yang akan ditambahkan: RNG
Nama game sudah ada!
```

Hasil tes create game jika nama game sudah ada.

```
ENTER COMMAND: CREATE GAME 1
Command tidak dikenali, silahkan masukkan command yang valid.
```

6.5 LISTGAME

Fitur yang akan dites adalah game list game. Berikut merupakan daftar tes dengan hasil yang seharusnya dikeluarkan oleh program dan hasil yang diberikan program:

Program akan menampilkan list game yang dimiliki oleh pengguna.

```
ENTER COMMAND: LIST GAME
1. RNG
2. Diner DASH
3. TIC TAC TOE
4. DINOSAUR IN EARTH
5. RISEWOMAN
6. EIFFEL TOWER
7. TEST SAVE
8. TEST CREATE
```

Hasil tes command list game.

```
ENTER COMMAND: LIST GAME 1
Command tidak dikenali, silahkan masukkan command yang valid.
```

6.6 **DELETEGAME**

Fitur yang akan dites adalah game create game. Berikut merupakan daftar tes dengan hasil yang seharusnya dikeluarkan oleh program dan hasil yang diberikan program:

Program akan menghapus game yang dimiliki oleh pengguna, jika game berhasil dihapus akan mengeluarkan pesan “Game berhasil dihapus”, sedangkan jika tidak sesuai dengan ketentuan yang terdapat pada program maka game akan gagal dihapus dan mengeluarkan pesan “Game gagal dihapus!”

```
ENTER COMMAND: DELETE GAME
1. RNG
2. Diner DASH
3. TIC TAC TOE
4. DINOSAUR IN EARTH
5. RISEWOMAN
6. EIFFEL TOWER
7. TEST SAVE
8. TEST CREATE
Masukkan nomor game yang akan dihapus: 8
Game berhasil dihapus
ENTER COMMAND: LIST GAME
1. RNG
2. Diner DASH
3. TIC TAC TOE
4. DINOSAUR IN EARTH
5. RISEWOMAN
6. EIFFEL TOWER
7. TEST SAVE
```

Hasil tes adalah game berhasil dihapus oleh pengguna apabila game ada di daftar game.

```

ENTER COMMAND: DELETE GAME
1. RNG
2. Diner DASH
3. TIC TAC TOE
4. DINOSAUR IN EARTH
5. RISEWOMAN
6. EIFFEL TOWER
7. LALA
Masukkan nomor game yang akan dihapus: 6
Game gagal dihapus!

```

Hasil tes adalah game gagal dihapus oleh pengguna apabila game tidak ada di daftar game.

6.7 QUEUEGAME

Fitur yang akan di test adalah queue game. Dalam fitur ini, akan diminta input berupa nomor game yang ingin ditambahkan ke dalam antrian. Apabila input valid, maka game akan dimasukkan ke dalam antrian sesuai urutan. Apabila nomor yang dimasukkan tidak valid, maka akan menampilkan pesan “Nomor permainan tidak valid, silahkan masukkan nomor game pada list.”

Berikut daftar test dari fitur queue game:

```

ENTER COMMAND: QUEUE GAME
Berikut adalah daftar antrian game-mu
Tidak ada game dalam daftar antrian

1. RNG
2. Diner DASH
3. TIC TAC TOE
4. DINOSAUR IN EARTH
5. RISEWOMAN
6. EIFFEL TOWER
7. LALA

Nomor Game yang mau ditambahkan ke antrian:1
Game berhasil ditambahkan kedalam daftar antrian.
ENTER COMMAND: QUEUE GAME
Berikut adalah daftar antrian game-mu
1. RNG

1. RNG
2. Diner DASH
3. TIC TAC TOE
4. DINOSAUR IN EARTH
5. RISEWOMAN
6. EIFFEL TOWER
7. LALA

Nomor Game yang mau ditambahkan ke antrian:2
Game berhasil ditambahkan kedalam daftar antrian.

```

Apabila pengguna mencoba untuk menambahkan nomor game yang ada di antrian, program akan menambahkan game tersebut ke dalam antrian..

```

ENTER COMMAND: QUEUE GAME
Berikut adalah daftar antrian game-mu
1. RNG
2. Diner DASH

1. RNG
2. Diner DASH
3. TIC TAC TOE
4. DINOSAUR IN EARTH
5. RISEWOMAN
6. EIFFEL TOWER
7. LALA

Nomor Game yang mau ditambahkan ke antrian:8
Nomor permainan tidak valid, silahkan masukkan nomor game pada list.

```

Apabila pengguna mencoba untuk menambahkan nomor game yang tidak ada di daftar game ke dalam antrian, program akan mengeluarkan pesan bahwa nomor game tersebut tidak valid.

6.8 PLAYGAME

Fitur yang akan di test ialah play game. Fitur ini akan memainkan game yang ada di dalam daftar. Ketika input nama game yang diterima ialah “RNG”, “Diner DASH”, maupun “TIC TAC TOE”, maka program akan memainkan game tersebut. Apabila input nama game tidak valid, maka akan ditampilkan pesan “Game masih dalam maintenance, belum dapat dimainkan. Silahkan pilih game lain.

Berikut daftar test dari fitur play game :

Program akan menjalankan game pertama yang ada di antrian buatan command QUEUEGAME.

```

ENTER COMMAND: PLAY GAME
Berikut adalah daftar Game-mu
1. RNG
2. Diner DASH

Loading RNG . . .
RNG Telah dimulai. Uji keberuntungan Anda dengan menebak X
Tebakan: 

```

Menjalankan game yang ada di file config.txt selain RNG, Diner Dash, dan Tic Tac Toe akan membuat program mengeluarkan pesan bahwa game tersebut masih dalam maintenance.

```

ENTER COMMAND: PLAY GAME
Berikut adalah daftar Game-mu
1. EIFFEL TOWER
2. RISEWOMAN
3. TEST QUEUE

Game EIFFEL TOWER masih dalam maintenance, belum dapat dimainkan.
Silahkan pilih game lain.

```

6.9 SKIPGAME

Apabila pengguna memasukkan command SKIPGAME, program akan melewati game di antrian sesuai dengan input pengguna dan langsung memainkan game setelahnya.

```
ENTER COMMAND: SKIPGAME 2
Berikut adalah daftar Game-mu:
1. RNG
2. Diner DASH
3. TIC TAC TOE

Berikut adalah daftar Game-mu
1. TIC TAC TOE

Loading TIC TAC TOE . . .
TICTACTOE
| | |
---|---|---
| | |
---|---|---
| | |
Masukan row(1-3):
```

Apabila pengguna mencoba untuk melewati game yang tidak ada di antrian game dengan command SKIPGAME, program akan mengeluarkan pesan error.

```
ENTER COMMAND: SKIPGAME 6
Berikut adalah daftar Game-mu:
1. RNG
2. Diner DASH
3. TIC TAC TOE
4. RISEWOMAN

Tidak ada permainan di dalam daftar Game-mu.
```

6.10 QUIT

Apabila pengguna memasukkan command QUIT, program akan mengeluarkan pesan bahwa pengguna telah keluar dari program dan program akan berakhir.

```
ENTER COMMAND: QUIT
Anda keluar dari game BNMO.
Bye bye ...
```

6.11 HELP

Command HELP akan mencetak daftar command dan kegunaannya untuk menolong pengguna.


```

ENTER COMMAND: HELP
COMMANDS :
START      Membaca file konfigurasi default yang berisi list game yang dapat dimainkan
LOAD       Membaca save file yang berisi list game yang dapat dimainkan, histori dan scoreboard game
SAVE       Merepresentasikan nama file yang akan disimpan pada disk
CREATE GAME Menambahkan game baru pada daftar game
LIST GAME  Menampilkan daftar game yang disediakan oleh sistem
DELETE GAME Menghapus sebuah game dari daftar game
QUEUE GAME Mendaftarkan permainan kedalam list
PLAY GAME  Memainkan sebuah permainan
SKIPGAME <n> Melewatkan permainan sebanyak n
QUIT       Keluar dari Program
HELP       Menampilkan daftar perintah

```

6.12 COMMAND LAIN

Apabila pengguna memasukkan command yang tidak dikenal, program akan mengeluarkan pesan error dan meminta pengguna untuk memasukkan command yang valid.

```

ENTER COMMAND: COMMAND_ANEH
Command tidak dikenali, silahkan masukkan command yang valid.
ENTER COMMAND: ALALA
Command tidak dikenali, silahkan masukkan command yang valid.

```

6.13 RNG

```

Loading RNG . . .
RNG Telah dimulai. Uji keberuntungan Anda dengan menebak X
Tebakan: 20000
Lebih besar
Tebakan: 25000
Lebih besar
Tebakan: 27500
Lebih besar
Tebakan: 30000
Lebih kecil
Tebakan: 28500
Lebih kecil
Tebakan: 27600
Lebih besar
Tebakan: 27700
Lebih besar
Tebakan: 27900
Lebih kecil
Tebakan: 27800
Lebih kecil
Tebakan: 27750
Lebih besar
Tebakan: 2775
Lebih besar
Tebakan: 27775
Lebih kecil
Tebakan: 27760
Lebih besar
Tebakan: 27766
Lebih besar
Tebakan: 27770
Lebih besar
Tebakan: 27771
Lebih besar
Tebakan: 27773
Lebih besar
Tebakan: 27774
Ya, X adalah 27774
Score anda adalah = 83

```

Number X di generate secara random, setelah itu user menebak X , user diberi kesempatan 100 kali dengan score terpotong 1 setiap kali tebakan salah. Tidak ada batas untuk angka random yang di generate. Data test di atas adalah dimana user berhasil menebak setelah 27 langkah, tidak ada data test salah 100 kali karena terlalu panjang untuk di Screenshot.

6.14 DINER DASH

Fitur yang akan dites adalah game diner dash. Berikut merupakan daftar tes dengan hasil yang seharusnya dikeluarkan oleh program dan hasil yang diberikan program:

1. Memasak pesanan yang ada di daftar pesanan

Program akan memasak pesanan yang tertulis di daftar pesanan.

```
Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
-----
M0      | 3              | 4          | 12000
M1      | 5              | 2          | 23000
M2      | 2              | 5          | 34000

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----
|

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----
|

MASUKKAN COMMAND: COOK M0

Berhasil memasak M0
=====

SALDO: 0

Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
-----
M1      | 5              | 2          | 23000
M2      | 2              | 5          | 34000
M3      | 3              | 2          | 13000

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----
M0      | 3
```

2. Memasak pesanan yang belum ada di daftar pesanan

Program tidak akan memasak pesanan yang belum ada di daftar pesanan.

```

Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
-----
M1      | 5      | 2      | 23000
M2      | 2      | 5      | 34000
M3      | 3      | 2      | 13000

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----
M0      | 3

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----

MASUKKAN COMMAND: COOK M5

Tidak ada pesanan M5

MASUKKAN COMMAND: 

```

3. Memasak pesanan yang sudah disajikan
Program tidak akan memasak pesanan yang sudah disajikan.

```

Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
-----
M4      | 1      | 1      | 36000
M5      | 1      | 1      | 43000
M6      | 3      | 5      | 45000
M7      | 1      | 1      | 37000

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----
M1      | 2
M3      | 2

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----
M2      | 5

MASUKKAN COMMAND: COOK M0

M0 sudah dimasak

MASUKKAN COMMAND: 

```

4. Menyajikan pesanan yang sudah selesai
Program akan menyajikan pesanan apabila pesanan telah selesai memasak dan ketahanannya > 0 .

```

SALDO: 0

Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
-----
M4      | 1      | 1      | 36000
M5      | 1      | 1      | 43000
M6      | 3      | 5      | 45000

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----
M1      | 3
M2      | 1
M3      | 3

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----
M0      | 4

MASUKKAN COMMAND: SERVE M0

Berhasil mengantarkan M0
Makanan M2 telah selesai dimasak
=====

SALDO: 12000

```

5. Menyajikan pesanan yang belum selesai
Program tidak akan menyajikan pesanan yang belum dimasak atau belum selesai dimasak.

```

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----
M1      | 1
M3      | 1
M4      | 1

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----
M2      | 4

MASUKKAN COMMAND: SERVE M1

M1 belum selesai dimasak

MASUKKAN COMMAND: 

```

6. Menyajikan pesanan yang tidak berurutan
Mesin hanya akan menyajikan pesanan apabila pesanan sebelumnya telah disajikan sesuai dengan prinsip FIFO.

```

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----
M1      | 1
M3      | 1
M4      | 1

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----
M2      | 4

MASUKKAN COMMAND: SERVE M1

M1 belum selesai dimasak

MASUKKAN COMMAND: SERVE M2

M2 belum dapat disajikan karena M1 belum selesai

MASUKKAN COMMAND: 

```

7. Menyajikan pesanan yang sudah disajikan
Mesin tidak akan menyajikan pesanan yang telah disajikan.

```

SALDO: 12000

Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
-----
M6      | 3              | 5         | 45000
M7      | 1              | 1         | 37000
M8      | 3              | 1         | 19000
M9      | 1              | 1         | 11000

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----
M5      | 1

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----
M1      | 2
M2      | 3
M3      | 2
M4      | 1

MASUKKAN COMMAND: SERVE M0

M0 sudah disajikan

MASUKKAN COMMAND: 

```

8. Melakukan command SKIP
Mesin akan lanjut ke putaran selanjutnya tanpa melakukan aksi apapun dari pengguna.

```

Makanan | Sisa ketahanan makanan
-----
M1      | 2
M2      | 3
M3      | 2
M4      | 1

MASUKKAN COMMAND: SERVE M0

M0 sudah disajikan

MASUKKAN COMMAND: SKIP

Makanan M5 telah selesai dimasak
Makanan M4 telah membusuk
=====

SALDO: 12000

Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
-----
M4      | 1              | 1         | 36000
M6      | 3              | 5         | 45000
M7      | 1              | 1         | 37000
M8      | 3              | 1         | 19000
M9      | 1              | 1         | 11000
M10     | 3              | 1         | 24000

```

9. Ketahanan makanan bernilai 0

Pesanan yang siap disajikan akan kembali ke daftar pesanan yang perlu dimasak apabila ketahanannya sama dengan nol.

```

Makanan | Sisa ketahanan makanan
-----
M1      | 2
M2      | 3
M3      | 2
M4      | 1

MASUKKAN COMMAND: SERVE M0

M0 sudah disajikan

MASUKKAN COMMAND: SKIP

Makanan M5 telah selesai dimasak
Makanan M4 telah membusuk
=====

SALDO: 12000

Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
-----
M4      | 1              | 1         | 36000
M6      | 3              | 5         | 45000
M7      | 1              | 1         | 37000
M8      | 3              | 1         | 19000
M9      | 1              | 1         | 11000
M10     | 3              | 1         | 24000

```

10. Pengguna memasukkan command invalid

Program akan meminta pengguna untuk memasukkan ulang command sampai masukan valid.

```
SALDO: 0

Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
-----
M0      | 3      | 3      | 23000
M1      | 5      | 5      | 45000
M2      | 2      | 2      | 17000

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----
|

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----
|

MASUKKAN COMMAND: INVALID

Command invalid, mohon masukkan command COOK M., SERVE M., atau SKIP
MASUKKAN COMMAND: 
```

11. Jumlah pesanan lebih dari tujuh

Program akan mengeluarkan pesan bahwa pemain kalah dan program akan keluar

```
SALDO: 0

Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
-----
M0      | 4      | 5      | 47000
M1      | 2      | 4      | 43000
M2      | 5      | 3      | 39000
M3      | 3      | 5      | 9000
M4      | 1      | 1      | 36000
M5      | 5      | 1      | 29000
M6      | 2      | 2      | 40000

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----

MASUKKAN COMMAND: SKIP

Kamu kalah :<
```

6.15 TIC TAC TOE

```
Loading TIC TAC TOE . . .
TICTACTOE
| | |
---|---|---
| | |
---|---|---
| | |
Masukan row(1-3): 2
Masukan column(1-3): 2
| | |
---|---|---
0 | X | 
---|---|---
| | |
Masukan row(1-3): 3
Masukan column(1-3): 2
| | |
---|---|---
0 | X | 
---|---|---
0 | X | 
Masukan row(1-3): 1
Masukan column(1-3): 2
| X | 
---|---|---
0 | X | 
---|---|---
0 | X | 
Anda Menang!
Score : 100
Terima Kasih telah bermain TICTACTOE!
```

Tic tac toe merupakan sebuah game berlawanan dengan computer(non-player) untuk membuat sebuah garis yang berisikan 3 O / 3 X bisa secara horizontal, vertikal, dan diagonal. User hanya dapat membuat 1 O / 1 X untuk setiap giliran. Screenshot di atas merupakan contoh di saat user yang menang.


```

Loading TIC TAC TOE . . .
TICTACTOE
| | |
---|---|---
| | |
---|---|---
| | |

Masukan row(1-3): 1
Masukan column(1-3): 1
X | | 
---|---|---
| | 0
---|---|---
| | |

Masukan row(1-3): 3
Masukan column(1-3): 2
X | | 
---|---|---
| | 0
---|---|---
| X | 0

Masukan row(1-3): 3
Masukan column(1-3): 1
X | | 
---|---|---
| 0 | 0
---|---|---
X | X | 0

Masukan row(1-3): 1
Masukan column(1-3): 2
X | X | 
---|---|---
0 | 0 | 0
---|---|---
X | X | 0

Anda Kalah! Computer Menang!
Score : 0
Terima Kasih telah bermain TICTACTOE!

```

Data test dimana computer menang

6.16 GAME BUATAN

Game yang ditambahkan oleh pengguna melalui command CREATEGAME akan langsung mengeluarkan pesan game over beserta skor akhir yang merupakan integer random.

```

ENTER COMMAND: PLAY GAME
Berikut adalah daftar Game-mu
1. LALA

Loading LALA . . .
GAME OVER
SKOR AKHIR: 7268

```

7 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	START	Mengetahui apakah fitur START, dapat berjalan dengan baik dan pengguna dapat masuk ke program utama dengan lancar.	Memberikann command START	Data test dapat dilihat di 6.1	Pengguna dapat masuk ke program utama.	Pengguna berhasil masuk ke program utama.
2	LOAD	Menjalankan tes untuk mengetahui apakah fitur load dapat berjalan dengan baik	Memberikan command LOAD diikuti dengan nama file yang valid	Data test dapat dilihat di 6.2	Memuat program dan savefile berhasil dibaca	Berhasil memuat program dan savefile berhasil dibaca
3	SAVE	Menjalankan tes untuk mengetahui apakah fitur save dapat berjalan dengan baik	Mengetik SAVE <filename> pada command setelah START dan LOAD	Data test dapat dilihat di 6.3	Menyimpan state game pengguna saat ini ke dalam file.	Berhasil menyimpan state game pengguna saat ini ke dalam file.
4	CREATEGAME	Melakukan tes untuk mengetahui apakah fitur create game berjalan dengan baik	Mengetik CREATE GAME pada command setelah START atau LOAD lalu input nama game yang ingin ditambahkan ke list	Data test dapat dilihat di 6.4	Mendapat input pengguna dan menambahkan game ke koleksi game pengguna	Berhasil mendapat input pengguna dan menambahkan game ke koleksi game pengguna
5	LISTGAME	Melakukan tes untuk mengetahui apakah fitur list game berjalan dengan baik	Mengetik LIST GAME pada command setelah START atau LOAD	Data test dapat dilihat di 6.5	Menampilkan game yang dimiliki pengguna ke layar	Berhasil menampilkan game yang dimiliki pengguna ke layar

6	DELETEGAME	Melakukan tes untuk mengetahui apakah fitur delete game berjalan dengan baik	Mengetik DELETE GAME pada command setelah START atau LOAD lalu input nomor game sesuai list game untuk game yang ingin dihapus	Data test dapat dilihat di 6.6	Menghapus game yang dimiliki sesuai input pengguna dengan ketentuan pada spesifikasi	Berhasil menghapus game yang dimiliki sesuai input pengguna dengan ketentuan pada spesifikasi
7	QUEUEGAME	Menjalankan test untuk mengetahui apakah fitur queue game dapat berjalan dengan baik	Mengetik QUEUE GAME pada command setelah START atau LOAD lalu input angka game yang ingin dimasukkan ke antrian	Data test dapat dilihat di 6.7	Menerima input pengguna dan menambahkan game tersebut ke dalam antrian.	Berhasil menerima input pengguna dan menambahkan game tersebut ke dalam antrian.
8	PLAYGAME	Menjalankan test untuk mengetahui apakah fitur play game dapat berjalan dengan baik	Mengetik PLAY GAME pada command setelah menambahkan game dalam QUEUE GAME	Data test dapat dilihat di 6.8	Menerima input nama game dari pengguna dan memainkan game tersebut.	Berhasil menerima input nama game dari pengguna dan memainkan game tersebut.
9	SKIPGAME	Menjalankan test untuk mengetahui apakah fitur skipgame dapat berjalan dengan baik	Mengetik SKIP GAME <n> pada command dengan n sebagai jumlah game yang ingin dilewati, setelah menambahkan game dalam QUEUE GAME	Data test dapat dilihat di 6.9	Menerima inputan n berupa banyak game yang ingin di skip kemudian masuk ke game selanjutnya	Berhasil menerima inputan n berupa banyak game yang ingin di skip kemudian masuk ke game selanjutnya
10	QUIT	Menjalankan test untuk mengetahui apakah fitur quitgame	Mengetik QUIT pada command setelah	Data test dapat dilihat di 6.10	Menampilkan pesan keluar dari permainan	Berhasil menampilkan pesan keluar dari permainan

		dapat berjalan dengan baik	START atau LOAD			
11	HELP	Menjalankan test untuk mengetahui apakah fitur help dapat berjalan dengan baik	Mengetik HELP pada command setelah START atau LOAD	Data test dapat dilihat di 6.11	Menampilkan bantuan kegunaan command yang tersedia	Berhasil menampilkan bantuan kegunaan command yang tersedia
12	COMMAND LAIN	Menjalankan test untuk mengetahui apakah program dapat mengetahui input yang dimasukkan salah	Memasukkan command yang berada di luar menu help	Data test dapat dilihat 6.12	Menampilkan pesan jika terdapat command yang berada di luar list yang sudah ditentukan	Berhasil menampilkan pesan jika terdapat command yang berada di luar list yang sudah ditentukan
13	RNG	Menjalankan tes untuk mengetahui apakah fitur game RNG dapat berjalan dengan baik	Mengetik PLAY GAME setelah memasukkan game RNG je dalam QUEUE GAME dan game RNG berada di urutan pertama QUEUE GAME	Data test dapat dilihat 6.13	Menerima input sampai 100 kali untuk menebak angka X, ketika X sudah ditebak maka akan ditampilkan skor pengguna.	Berhasil menerima input sampai 100 kali untuk menebak angka X, ketika X sudah ditebak, skor pengguna ditampilkan
14	DINER DASH	Menjalankan tes untuk mengetahui apakah fitur game Diner Dash dapat berjalan dengan baik	Mengetik PLAY GAME setelah memasukkan game DINNER DASH je dalam QUEUE GAME dan game DINNER DASH berada di urutan pertama QUEUE GAME	Data test dapat dilihat di 6.14		
15	TIC TAC TOE	Menjalankan tes untuk mengetahui	Mengetik PLAY GAME setelah	Data test dapat dilihat 6.15	Menerima input baris dan kolom	Berhasil menerima input baris

		apakah fitur game TIC TAC TOE dapat berjalan dengan baik	memasukkan game TIC TAC TOE ke dalam QUEUE GAME dan game TIC TAC TOE berada di urutan pertama QUEUE GAME		dari pengguna dan menampilkan pesan apakah pengguna menang atau kalah di akhir permainan.	dan kolom dari pengguna dan menampilkan pesan apakah pengguna menang atau kalah di akhir permainan.
16	GAME BUATAN	Menjalankan tes untuk mengetahui apakah program dapat menjalankan game yang dibuat pengguna dapat berjalan dengan baik	Mengetik PLAY GAME setelah memasukkan game yang dibuat di fitur CREATE GAME ke dalam QUEUE GAME dan game TIC TAC TOE berada di urutan pertama QUEUE GAME	Data test dapat dilihat 6.16	Menerima input baris dan kolom dari pengguna dan menampilkan pesan game over dan skor akhir yang merupakan integer random	Berhasil menerima input baris dan kolom dari pengguna dan menampilkan pesan game over dan skor akhir yang merupakan integer random

8 Pembagian Kerja dalam Kelompok

No	Nama - NIM	Tugas
1	Darren - 18221050	<ul style="list-style-type: none"> • Membuat main.c • Membuat command START • Membuat command LOAD • Membuat command SAVE • Membuat command CREATE GAME • Membuat command LIST GAME • Membuat command DELETE GAME • Membuat command QUEUE GAME • Membuat command PLAY GAME • Membuat command SKIP GAME • Membuat game RNG • Membuat game TIC TAC TOE • Membuat ADT Queue • Membuat ADT Mesin Load • Membuat ADT Array • Membuat ADT Mesin Karakter • Membuat ADT Mesin Kata • Membuat MAKEFILE • Melakukan pengetesan keseluruhan program dan data test
2	Priscilla - 18221098	<ul style="list-style-type: none"> • Membuat command SKIPGAME • Membuat command HELP • Membuat command QUIT • Membuat fitur yang menangani command lain • Formatting Laporan
3	Razan - 18221132	<ul style="list-style-type: none"> • Membuat command CREATE GAME • Membuat command LIST GAME • Membuat command DELETE GAME • Membuat fungsi compare
4	L.Audrey - 18221114	<ul style="list-style-type: none"> • Membuat command START • Membuat command QUEUE GAME • Membuat command PLAY GAME • Menambahkan fungsi str_comp
5.	M. Rafi Haidar - 18221134	<ul style="list-style-type: none"> • Membuat game Diner Dash • Membuat driver ADT Array • Membuat driver ADT Queue • Membuat driver ADT Mesin Karakter

		<ul style="list-style-type: none"> • Membuat bagian PLAY GAME yang menangani game buatan • Membuat repository di Github • Menjadi narahubung antara kelompok dan asisten • Membuat laporan bagian 1, 3, 4, dan 5 • Membantu membuat main.c
--	--	---

9 Lampiran

9.1 Deskripsi Tugas Besar 1

BNMO adalah sebuah robot video game console yang dimiliki oleh Indra dan Doni. Dua bulan yang lalu, ia mengalami kerusakan dan telah berhasil diperbaiki. Sayangnya, setelah diperbaiki ia justru mendapatkan lebih banyak bug dalam sistemnya. Oleh karena itu, Indra dan Doni mencari programmer lain yang lebih handal untuk ulang memprogram robot video game console kesayangannya.

BNMO merupakan suatu robot game console yang dapat menjalankan permainan. BNMO memiliki beberapa fitur utama, yaitu:

- 1) Memainkan game
- 2) Menambahkan game
- 3) Menghapus game
- 4) Mengurutkan game yang akan dimainkan

Ketika program pertama kali dijalankan, BNMO akan memperlihatkan main menu yang berisi welcome page dan beberapa menu pilihan yaitu START dan LOAD. Setelah itu, main menu akan menerima input commands yang akan dijelaskan pada bagian berikutnya.

a. START

START merupakan salah satu command yang dimasukkan pertama kali oleh pemain ke BNMO. Setelah menekan Enter, dibaca file konfigurasi default yang berisi list game yang dapat dimainkan.

b. LOAD <filename>

LOAD merupakan salah satu command yang dimasukkan pertama kali oleh pemain ke BNMO. Memiliki satu argumen yaitu filename yang merepresentasikan suatu *save file* yang ingin dibuka. Setelah menekan Enter, akan dibaca save file <filename> yang berisi list game yang dapat dimainkan, histori dan scoreboard game.

c. SAVE <filename>

SAVE merupakan command yang digunakan untuk menyimpan state game pemain saat ini ke dalam suatu file. Command SAVE memiliki satu argumen yang merepresentasikan nama file yang akan disimpan pada disk.

d. CREATEGAME

CREATEGAME merupakan command yang digunakan untuk menambahkan game baru pada daftar game.

e. LISTGAME

LISTGAME merupakan command yang digunakan untuk menampilkan daftar game yang disediakan oleh sistem.

f. DELETGAME

DELETGAME merupakan command yang digunakan untuk menghapus sebuah game dari daftar game. Adapun aturan penghapusan game adalah:

- Game yang dapat dihapus hanya game yang dibuat secara custom oleh pengguna.
- 5 game pertama pada file konfigurasi tidak dapat dihapus.
- Game yang saat itu terdapat di dalam queue game tidak dapat dihapus.

g. QUEUEGAME

QUEUEGAME merupakan command yang digunakan untuk mendaftarkan permainan kedalam list. List dalam queue akan hilang ketika pemain menjalankan command QUIT.

h. PLAYGAME

PLAY GAME merupakan command yang digunakan untuk memainkan sebuah permainan. Game yang dimainkan adalah game dengan urutan pertama di antrian game.

i. SKIPGAME <n>

SKIPGAME merupakan command yang digunakan untuk melewati permainan sebanyak n.

j. QUIT

Keluar dari program.

k. HELP

Bantuan command-command yang disebutkan di atas. Tampilan dan kata-kata dibebaskan.

l. COMMAND LAIN

Command-command lain selain yang disebutkan diatas tidak valid. Keluar dari program.

9.2 Notulen Rapat

Form Asistensi Tugas Besar







STEI- ITB	IF2111-TB1-04	Halaman 32 dari 39 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.		

IF2110/Algoritma dan Struktur Data
Sem. 1 2022/2023






No. Kelompok/Kelas : 4/ K02
Nama Kelompok :
Anggota Kelompok (Nama/NIM) :
1. Priscilla Auleader Napitupulu - 18221098
2. Muhammad Rafi Haidar - 18221134
3. Darren - 18221050
4. Razan Aditya Putra - 18221132
5. Laurensia Audrey Pramashinta - 18221114


Asisten Pembimbing : Cynthia Rusadi

Asistensi I

Tanggal : 03 November 2022	Catatan Asistensi: Makefile fungsinya apa? makefile berfungsi untuk mengcompile banyak file .c secara langsung. Sehingga tidak usah manual gcc mesinkarakter.c Kalau untuk fscanf apakah boleh digunakan di main.c? fscanf hanya dapat digunakan dalam ADT , karena membaca command dalam main menggunakan mesinkata. Diperbolehkan menggunakan fgets.
Tempat : Google meet	
Kehadiran Anggota Kelompok: No NIM Tanda tangan 1 18221098  2 18221134  3 18221050  4 18221132  5 18221114 	
	Tanda Tangan Asisten: 

Asistensi II

Tanggal : 10 November 2022	Catatan Asistensi: Untuk game bonus apa perlu dimasukkan ke config.txt? Ya masukkan saja Untuk data test di laporan masukkan apa aja? Sesuai yang diminta aja , screenshot dll Untuk Penjelasan ADT dimasukkan apa? Penjelasan dan screenshot type struct nya
Tempat : Google Meet	
Kehadiran Anggota Kelompok:	
No	
NIM	
Tanda tangan	
1	
18221098	
	
2	
18221134	
	
3	
18221050	
	
4	
18221132	
	
5	
18221114	
	
	Tanda Tangan Asisten:

	
--	---

9.3 Log Activity Anggota Kelompok

Commits activity anggota kelompok dapat dilihat di link di bawah ini:

<https://github.com/haidar03/Tugas-Besar-Algoritma-dan-Struktur-Data/commits/main>

No	Tanggal	Nama	NIM	Aktivitas
1	28 October 2022	Muhammad Rafi Haidar	18221134	Membuat repository github
2	30 October 2022	Muhammad Rafi Haidar	18221134	Push update diner dash
3	31 October 2022	Muhammad Rafi Haidar	18221134	Memperbaiki program diner dash
4	01 November 2022	Darren	18221050	Push rng yang belum finished
5	01 November 2022	Priscilla Auleader Napitupulu	18221098	Penambahan npc
6	01 November 2022	Priscilla Auleader Napitupulu	18221098	Delete map.c
7	01 November 2022	Darren	18221050	Penambahan ADT Arraydin
8	02 November 2022	Priscilla Auleader Napitupulu	18221098	Push validator (command lain)
9	02 November 2022	Laurensia Audrey Pramashinta	1822114	Push command queue game
10	02 November 2022	Darren	18221050	Penambahan score pada rng.c

11	02 November 2022	Muhammad Rafi Haidar	18221134	Revisi Diner Dash
12	03 November 2022	Darren	18221050	Create main untuk menggabungkan program
13	03 November 2022	Darren	18221050	Penambahan fungsi ReadInput & StrToInt
14	03 November 2022	Muhammad Rafi Haidar	18221134	Push driver ddash
15	03 November 2022	Razan Aditya Putra	18221132	Update create game
16	03 November 2022	Darren	18221050	Update rng.c
17	05 November 2022	Laurensia Audrey Pramashinta	1822114	Update stringcomp.c
18	05 November 2022	Laurensia Audrey Pramashinta	1822114	Update playgame.h
19	05 November 2022	Darren	18221050	Update load.c
20	05 November 2022	Darren	18221050	Penambahan command startgame
21	05 November 2022	Darren	18221050	Pemindahan load.c ke console.c
22	05 November 2022	Muhammad Rafi Haidar	18221134	Menggabungkan di main
23	05 November 2022	Razan Aditya Putra	18221132	Update list game
24	05 November 2022	Razan Aditya Putra	18221132	Update delete game
25	06 November 2022	Darren	18221050	Penambahan save

26	06 November 2022	Darren	18221050	Update fix DeleteGame
27	06 November 2022	Darren	18221050	Update fix CreateGame
28	06 November 2022	Darren	18221050	Update console.c
29	06 November 2022	Darren	18221050	Update mesinkarakter.c
30	06 November 2022	Darren	18221050	Mengganti input rng
31	06 November 2022	Darren	18221050	Delete queuegame.c
32	06 November 2022	Laurensia Audrey Pramashinta	1822114	Delete queuegame.h
33	06 November 2022	Darren	18221050	Fix queuegame dan playgame
34	06 November 2022	Darren	18221050	Update console.c
35	08 November 2022	Darren	18221050	Fix start game
36	08 November 2022	Muhammad Rafi Haidar	18221134	Update diner dash
37	08 November 2022	Muhammad Rafi Haidar	18221134	Menggabungkan di main
38	08 November 2022	Darren	18221050	Update fix delete game
39	08 November 2022	Darren	18221050	Update console.h
40	08 November 2022	Darren	18221050	Update revisi dinner dash
41	08 November 2022	Darren	18221050	Update main.c
42	08 November 2022	Darren	18221050	Update console.c
43	09 November 2022	Darren	18221050	Update

				console.h
44	09 November 2022	Darren	18221050	Penambahan game tic tac toe
45	09 November 2022	Darren	18221050	Update main.c
46	09 November 2022	Darren	18221050	Update console.c
47	09 November 2022	Priscilla Auleader Napitupulu	18221098	Menambahkan skip game fix
48	09 November 2022	Priscilla Auleader Napitupulu	18221098	Menambahkan quit game fix
49	09 November 2022	Muhammad Rafi Haidar	18221134	Menggabungkan merge di main
50	09 November 2022	Muhammad Rafi Haidar	18221134	Update ddash.h
51	10 November 2022	Darren	18221050	Fix load dan playgame
52	10 November 2022	Darren	18221050	Push driver ADT
53	10 November 2022	Darren	18221050	Push make file
54	11 November 2022	Darren	18221050	Update driver_mesinkat a.c
55	11 November 2022	Darren	18221050	Update driver_mesinload.c
56	11 November 2022	Priscilla Auleader Napitupulu	18221098	Update README.txt
57	11 November 2022	Muhammad Rafi Haidar	18221134	Revisi terakhir program