

**LAPORAN TUGAS BESAR**  
**IF2111 Algoritma dan Struktur Data**

**BNMO**

Dipersiapkan oleh:

**Kelompok 6**

Agnes Tamara Putri (18220010)


Salman Ma'arif Achsien (18221102)

Reswara Trista Aulia C (18221122)

Erensi Ratu Chelsia (18221166)

Rania Sasi Kirana (18221168)

Jl. Ganesha 10, Bandung 40132

	<b>Sekolah Teknik Elektro dan Informatika ITB</b>	<b>Nomor Dokumen</b>		<b>Halaman</b>
		<i>IF2111-TB1-K2-06</i>		39
		<i>Revisi</i>	0	11/11/2022

# Daftar Isi

1	Ringkasan	4
2	<b>Penjelasan Tambahan Spesifikasi Tugas</b>	<b>6</b>
2.1	Battleship	6
3	<b>Struktur Data (ADT)</b>	<b>6</b>
3.1	ADT Point	6
3.2	ADT Battleship	7
3.3	ADT Queue1	8
3.4	ADT Queue2	9
3.5	ADT List/Array	9
3.6	ADT Mesin Karakter	10
3.7	ADT Mesin Kata	11
4	<b>Program Utama</b>	<b>11</b>
5	<b>Algoritma-Algoritma Menarik</b>	<b>12</b>
5.1	Random	12
6	<b>Data Test</b>	<b>13</b>
6.1	Data Test START	13
6.2	Data Test LOAD	13
6.3	Data Test CREATE GAME	14
6.4	Data Test LIST GAME	16
6.5	Data Test DELETE GAME	16
6.6	Data Test QUEUE GAME	17
6.7	Data Test SAVE	18
6.8	Data Test PLAY GAME	19
6.9	Data Test SKIP GAME	20
6.10	Data Test QUIT	21
6.11	Data Test HELP	22
6.12	Data Test Command Lain	23
6.13	Data Test Diner DASH	23
6.14	Data Test RNG	26
6.15	Data Test Game Bonus (Battleship)	28

<b>7 Test Script</b>	<b>31</b>
<b>8 Pembagian Kerja dalam Kelompok</b>	<b>33</b>
9 Lampiran	<b>34</b>
9.1 Deskripsi Tugas Besar 1	34
9.2 Notulen Rapat	34
9.3 Log Activity Anggota Kelompok	39

# 1 Ringkasan

Lima orang mahasiswa Sistem dan Teknologi Informasi diminta untuk membantu Indra dan Doni dalam memperbaiki game console kesayangan mereka, yaitu BNMO. Setelah diperbaiki akibat rusak dua bulan lalu, ternyata terdapat lebih banyak bug di dalam BNMO. Maka dari itu, Indra dan Doni mencari programmer untuk memrogram ulang BNMO. Setelah menemukan lima orang mahasiswa Sistem dan Teknologi Informasi yang handal dalam pemrograman, mereka segera memperbaiki program BNMO.

BNMO merupakan suatu robot game console yang diprogram menggunakan bahasa C, dengan game-game didalamnya merupakan permainan berbasis CLI (Command Line Interface). Program ini memanfaatkan berbagai struktur data, diantaranya adalah list/array, mesin karakter, mesin kata, queue, point, dan juga grid. Program pada BNMO dapat digunakan untuk memainkan game, menambahkan game, menghapus game, dan juga mengurutkan game yang akan dimainkan.

Ketika program BNMO dijalankan, program akan menampilkan welcome page dan juga menu yang bisa dipilih oleh pengguna berupa START dan LOAD. Setelah itu, main menu akan menerima input commands berupa CREATEGAME, LISTGAME, DELETGAME, QUEUEGAME, PLAYGAME, SKIPGAME, QUIT, HELP, atau COMMAND LAIN.

Program ini berisi dua game yang dapat dimainkan, diantaranya adalah Diner Dash dan Random Number Generator. Diner Dash merupakan game mengantar makanan secara terurut berdasarkan prioritasnya. dalam game ini, pengguna diminta untuk memberikan perintah berupa cook, serve, dan skip untuk menyelesaikan permainan. Sedangkan, RNG (Random Number Generator) merupakan permainan menebak angka yang dilakukan dengan memberikan input angka untuk dicek lebih besar atau lebih

kecilnya angka tersebut dibanding *generated number* hingga pengguna dapat menebak *generated number* dari program.

Pengerjaan tugas besar ini dapat membantu mahasiswa untuk memahami dan mempelajari dasar-dasar program bahasa C. Mahasiswa juga dapat mengetahui serta memahami bagaimana membuat program yang cukup kompleks dibantu dengan materi dasar yang sudah dipelajari di kelas dan praktikum. Tugas besar ini juga membantu mahasiswa dalam menemukan ide dan alur dari program tersebut untuk menyelesaikan program. Oleh karena itu, adanya tugas besar ini sangat berguna dan bermanfaat bagi mahasiswa.

## 2 Penjelasan Tambahan Spesifikasi Tugas

### 2.1 Battleship

Battleship merupakan game yang dimainkan oleh dua pemain dengan masing-masing pemain menempatkan terlebih dahulu ship nya. Lalu, dari setiap pemain akan memilih grid dimana akan ditaruh point. Untuk pemain yang berhasil menempatkan point di letak ship yang ditentukan lawan, maka pemain tersebut dapat menempatkan point kembali. Namun, jika pemain tidak menempatkan point di letak ship yang ditentukan lawan, maka pemain lawan yang akan bermain. Untuk pemain yang dapat menyelesaikan permainan dengan menempatkan point di semua ship lawan, maka dialah pemenangnya.

## 3 Struktur Data (ADT)

### 3.1 ADT Point

ADT Point ini ada pada file header point.h

#### 3.1.1 Sketsa struktur data

Pada ADT Point, menyimpan struktur data Point. Pada struktur data Point menyimpan elemen x dan y yang bertipe integer. ADT Point ini berisi selektor untuk akses elemennya dan juga primitif-primitif seperti createPoint, isOrigin, isFirstQuadrant, translate, copyPoint, movePoint.

### 3.1.2 Persoalan yang diselesaikan

ADT Point ini dipakai untuk menyelesaikan program battleship. Pada program battleship dibutuhkan ADT Point contohnya pada battleship perlu dilakukannya input point dari pemain.

### 3.1.3 Alasan pemilihan

ADT ini dipilih karena di program battleship dibutuhkan ADT Point untuk mempermudah input pemain yang bertipe point.

## 3.2 ADT Battleship

ADT Battleship ini ada pada file header battleship.h

### 3.1.1 Sketsa struktur data

Pada ADT Battleship, menyimpan struktur data Tile, Ship, dan Grid. Pada struktur data Grid menyimpan elemen enemy yang bertipe boolean, elemen tiles dengan indeks TILECOUNT yang bertipe Tile, dan elemen ships dengan indeks SHIPCOUNT yang bertipe Ship. Pada struktur data Ship menyimpan elemen name yang bertipe pointer to char, elemen position yang bertipe Point, elemen vertical yang bertipe boolean, elemen sunk yang bertipe boolean, dan elemen length yang bertipe integer. Pada struktur data Tile menyimpan elemen ship yang bertipe boolean dan elemen shot yang bertipe boolean. ADT Battleship ini juga berisi selektor untuk akses elemennya.

### 3.1.2 Persoalan yang diselesaikan

ADT Battleship ini dipakai untuk menyelesaikan program battleship. Pada program battleship dibutuhkan ADT Battleship contohnya pada battleship perlu dilakukannya inisialisasi ukuran grid, membuat ship, menempatkan ship.

### 3.1.3 Alasan pemilihan

ADT ini dipilih karena di program battleship digunakan struktur data Tile, Ship, dan Grid.

## 3.3 ADT Queue1

ADT Queue1 ini ada pada file header queue1.h

### 3.3.1 Sketsa struktur data

Pada ADT Queue1, menyimpan struktur data Queue1 dan QEltype. Pada struktur data Queue1 menyimpan elemen buffer dengan indeks QCAPACITY yang bertipe QEltype, idxHead yang bertipe integer, dan idxTail yang bertipe integer. Pada struktur data QEltype menyimpan elemen foodId, cookTime, stayTime, dan price yang semuanya bertipe integer. ADT Queue1 ini berisi selektor untuk akses elemennya dan juga primitif-primitif seperti createQueue1, isEmptyQ, isFullQ, lengthQ, enqueueQ, dequeueQ, printOrders, printCooking, printServing.

### 3.3.2 Persoalan yang diselesaikan

ADT Queue1 ini dipakai untuk menyelesaikan program dinerdash. Pada program dinerdash dibutuhkan ADT Queue1 ini karena contohnya pada saat pemanggilan command cook dan serve, maka perlu melakukan enqueue ke dalam antrian cook dan serve nya.

### 3.3.3 Alasan pemilihan

ADT Queue1 ini dipilih karena di program dinerdash dibutuhkan ADT Queue alternatif 1 untuk melakukan geser elemen jika ada elemen yang didequeue.



### 3.4 ADT Queue2

ADT Queue2 terdapat pada file header queue2.h

#### 3.4.1 Sketsa struktur data

Pada ADT Queue2, disimpan struktur data Queue, QEltype, dan IdxType. IdxType sendiri bertipe integer dan QElType pada Queue2 bertipe Word yang diambil dari ADT mesinkata. Pada struktur data Queue2, akan disimpan elemen Tab dengan tipe QEltype, HEAD yang bertipe integer, TAIL yang bertipe IdxType, dan MaxEl yang bertipe integer.

ADT Queue2 berisi berbagai primitif, diantaranya adalah primitif kreator berupa createQueue, primitif prototype seperti IsEmpty, IsFull, dan Length, primitif destruktur seperti DeleteQueue dan UpdateCapacity, serta primitif-primitif lain seperti enqueue, dequeue, Front, isSameWord, isMember, dan displayQueue.

#### 3.4.2 Persoalan yang diselesaikan

ADT Queue2 ini digunakan pada program untuk menjalankan command DELETEGAME, PLAYGAME, QUEUEGAME, dan SKIPGAME.

#### 3.4.3 Alasan pemilihan

ADT Queue2 dibuat dengan konsep queue alternatif 2 agar tidak perlu dilakukan pergeseran setiap dilakukan dequeue. pergeseran hanya perlu dilakukan jika queue penuh semu dengan  $IDXHEAD \neq 0$  dan  $IDXTAIL = CAPACITY - 1$ .

### 3.5 ADT List/Array

ADT List/Array terdapat pada file array.h

#### 3.5.1 Sketsa struktur data

Pada ADT List/Array, disimpan struktur data List dan Word dengan Word bertipe ElType. Pada struktur data List, disimpan elemen A dengan tipe ElType, serta nEff dan capacity dengan tipe integer. ADT List/Array berisi berbagai primitif, yaitu primitif createList, deallocateList, isElTypeEqual, isEmpty, isFull, length, getElmt, updateCapacity, insertFirst, insertAt, insertLast, deleteFirst, deleteAt, deleteLast, dan displayList.

### 3.5.2 Persoalan yang diselesaikan

ADT List/Array ini dipakai untuk menjalankan command listGame yang merepresentasikan daftar game yang terdapat pada sistem, serta command saveToFile.

### 3.5.3 Alasan pemilihan

ADT List/Array dipilih karena dengan menggunakan ADT List/Array pengaksesan game dari daftar game akan menjadi lebih mudah karena setiap elemen dapat langsung diakses melalui indeksnya.

## 3.6 ADT Mesin Karakter

ADT Mesin Karakter terdapat pada file mesinkarakter.h

### 3.5.1 Sketsa struktur data

Pada ADT Mesin Karakter memiliki struktur berupa cc (current character) bertipe char, serta isFile dan eoi (end of input) bertipe boolean. Pada ADT Mesin Karakter juga disertakan sebuah INPUTMARK dalam bentuk ‘\n’ sebagai penanda akhir proses pembacaan. ADT Mesin Karakter berisi dua primitif, yaitu start dan adv.

### 3.5.2 Persoalan yang diselesaikan

ADT Mesin Karakter ini digunakan untuk melakukan pembacaan karakter pada suatu file.

### 3.5.3 Alasan pemilihan

ADT Mesin Karakter dipilih untuk dapat melakukan pembacaan file konfigurasi. Setelah itu, akan digunakan lebih lanjut pada ADT Mesin Kata.

## 3.7 ADT Mesin Kata

ADT Mesin Kata terdapat pada file mesinkata.h

### 3.5.1 Sketsa struktur data

ADT Mesin Kata memiliki struktur berupa `endKata` yang bertipe boolean dan `currentKata` bertipe `Word`. Pada ADT Mesin Kata ini juga terdapat sebuah `FILEBLANK` dalam bentuk `'\n'` dan `INPUTBLANK` dalam bentuk `' '`. ADT ini memiliki beberapa prosedur, yaitu `ignoreBlank` untuk mengabaikan `INPUTBLANK`, `startKata` untuk memulai prosedur pembacaan input user atau file, `advKata` untuk mengakuisisi kata selanjutnya, `salinKata` untuk mengakuisisi kata yang dibaca.

### 3.5.2 Persoalan yang diselesaikan

ADT Mesin Kata ini digunakan untuk melakukan pembacaan file konfigurasi yang berisi jumlah permainan yang dimiliki oleh sistem beserta dengan namanya.

### 3.5.3 Alasan pemilihan

ADT Mesin Kata dipilih karena ADT Mesin Kata akan membuat pembacaan file konfigurasi menjadi lebih teratur.

## 4 Program Utama

Program utama yang terdapat pada file `"main.c"` akan *include* semua *file header* dari ADT yang telah dibuat sebelumnya. Saat mulai dijalankan, program utama akan menampilkan *interface* welcome page, list command yang bisa dipakai dalam program, dan pilihan menu, yaitu `START` dan `LOAD`. Setelah itu, program utama akan

meminta input berupa START atau LOAD. Jika user menginput START, program akan membaca file konfigurasi yang berisi jumlah permainan yang dimiliki oleh sistem beserta dengan namanya, sedangkan jika user menginput LOAD, maka program akan membaca save file yang berisi list permainan yang dapat dimainkan, tetapi jika file save file tidak ditemukan, maka program akan meminta user untuk memasukkan input kembali.

Setelah user memasukkan input START dan file konfigurasi telah berhasil dibaca, program akan meminta input berupa command-command yang terdapat pada program,. Jika user memasukkan input CREATE GAME, maka program akan memasukkan nama game yang akan ditambahkan di akhir daftar game. Jika user memasukkan input LIST GAME, maka program akan menampilkan daftar game yang tersedia pada sistem. Jika user memasukkan input DELETE GAME, maka program akan menghapus sebuah game dari daftar game. Jika user memasukkan input QUEUE GAME, maka program akan mendaftarkan pemain ke dalam list game. Jika user memasukkan input PLAY GAME, maka program akan memainkan game yang terdapat pada spesifikasi yang berada pada urutan pertama di antrian game. Jika user memasukkan input SKIP GAME, maka program akan melewati permainan. Jika user memasukkan input QUIT, maka program akan menjalankan prosedur quit untuk keluar dari game. Jika user memasukkan input command yang tidak tersedia pada program, maka program akan menampilkan kalimat “Command tidak dikenali, silahkan masukkan command yang valid”.

## 5 Algoritma-Algoritma Menarik

### 5.1 Random

#### 5.1.1 Algoritma yang dimaksud

Algoritma ini merupakan algoritma untuk menghasilkan angka random yang digunakan pada beberapa fitur program. Algoritma ini digunakan untuk menentukan angka random yang harus ditebak oleh pengguna pada RNG dan untuk menentukan angka random waktu memasak, waktu ketahanan, dan harga

makanan pada Diner Dash. Selain itu, algoritma ini digunakan untuk menentukan gerak lawan pada Battleship.

#### 5.1.2 Letak Algoritma

Algoritma random digunakan pada game RNG dan Diner Dash. Algoritma ini diterapkan pada file `rng.c`, `dinerdash.c`, dan `battleship.c`

#### 5.1.3 Alasan algoritma menarik

Algoritma ini menarik karena pada fungsi `rand` yang merupakan bawaan C untuk random number generator, akan menghasilkan angka random yang berpola. Hal tersebut terjadi karena fungsi `rand` memiliki *seed default* atau posisi awal bawaan dalam sebuah pengacakan. Maka dari itu, dapat dilakukan dengan pembuatan fungsi dengan meminta input nilai minimal dan maksimal untuk *seed default* pengacakan nomor.

## 6 Data Test

Program terdiri dari beberapa command yang dapat diberikan oleh pengguna sehingga dilakukan data testing pada setiap command yang mungkin dimasukkan oleh pengguna untuk mendapatkan keluaran yang diharapkan. Berikut adalah testing yang kami lakukan pada setiap fitur atau command yang ada disertai dengan penjelasannya.

### 6.1 Data Test START

Command “START” adalah command pertama yang dijalankan oleh pengguna. Fitur ini akan membaca file konfigurasi default yang berisi daftar game yang dapat dimainkan. Setelah program membaca file tersebut, program akan menambahkan semua game yang berada di file tersebut ke dalam array of word berisi game di dalam program.



```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
*#%#(
/((*****/#%&@@%#(((#
*@(/((((.....(,
*/(((.....@.....(,
/(&#%#(.....@#.....(
/(&#%#(.....(
/((@&#(((@%#((((&#(
/((%&/((((.....(
%((%&/((&***((%,(((
&(&/*&/((((.....(&#%&(&
(((**#(((.....#(((&(( **@
&&*#(((.....#&((#&(&(.&
*% /#(((
18220010 - Agnes Tamara Putri
18221102 - Salman Ma'arif Achsien
18221122 - Reswara Trista Aulia C
18221166 - Erensi Ratu Chelsia
18221168 - Rania Sasi Kirana

- Tugas Besar 1 IF2111 Algoritma dan Struktur Data STI - Kelompok 06 Kelas 02 -

Selamat Datang di BNMO! Jalankan dalam Linux; jangan ubah ukuran terminal.
Silahkan input [START], [LOAD <FILENAME>], atau [HELP]...
root@BNMO:~$ LOAD fauna.txt
Save file berhasil dibaca. BNMO berhasil dijalankan.
Tekan [ENTER] untuk melanjutkan...

```

Gambar 6.2.1 Tampilan command LOAD jika berhasil

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
*#%#(
/((*****/#%&@@%#(((#
*@(/((((.....(,
*/(((.....@.....(,
/(&#%#(.....@#.....(
/(&#%#(.....(
/((@&#(((@%#((((&#(
/((%&/((((.....(
%((%&/((&***((%,(((
&(&/*&/((((.....(&#%&(&
(((**#(((.....#(((&(( **@
&&*#(((.....#&((#&(&(.&
*% /#(((
18220010 - Agnes Tamara Putri
18221102 - Salman Ma'arif Achsien
18221122 - Reswara Trista Aulia C
18221166 - Erensi Ratu Chelsia
18221168 - Rania Sasi Kirana

- Tugas Besar 1 IF2111 Algoritma dan Struktur Data STI - Kelompok 06 Kelas 02 -

Selamat Datang di BNMO! Jalankan dalam Linux; jangan ubah ukuran terminal.
Silahkan input [START], [LOAD <FILENAME>], atau [HELP]...
root@BNMO:~$ LOAD a
Save file tidak ditemukan! Silahkan ulangi input.
Tekan [ENTER] untuk melanjutkan...

```

Gambar 6.2.2 Tampilan command LOAD jika gagal

## 6.3 Data Test CREATE GAME

Command “CREATE GAME” merupakan fitur untuk menambah game baru pada daftar game sesuai input pengguna. Nama game yang valid akan ditambahkan ke dalam list game yang dapat dimainkan. Nama game yang sudah ada dalam daftar list game bukan nama game yang valid dan jika ditambahkan maka akan keluar pesan kesalahan bahwa game sudah ada dalam daftar game. Jika inputan kosong, maka akan mengeluarkan pesan kesalahan karena nama game tidak boleh kosong. Jika input dari pengguna melebihi batas karakter maksimal nama game maka juga akan mengeluarkan pesan kesalahan. Selain dari

itu, program akan menambahkan input nama game dari pengguna ke daftar game yang dapat dimainkan.

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
Masukkan nama game yang akan ditambahkan: CERES FAUNA <3 DUKE
Game berhasil ditambahkan.
Apakah Anda ingin kembali melakukan input? [Y/N]:

```

Gambar 6.3.1 Tampilan command CREATEGAME jika berhasil

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
Masukkan nama game yang akan ditambahkan:
Nama game tidak boleh kosong!
Apakah Anda ingin kembali melakukan input? [Y/N]:

```

Gambar 6.3.2 Tampilan command CREATEGAME jika gagal

## 6.4 Data Test LIST GAME

Command “LIST GAME” adalah fitur untuk menampilkan daftar game yang dapat dimainkan. Ketika dijalankan, jika daftar game kosong maka akan mengeluarkan pesan kesalahan bahwa tidak ada game yang tersedia. Jika tidak kosong, maka akan mengeluarkan list daftar game yang dapat dimainkan.

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. Battleship
7. CERES FAUNA <3 DUKE
8. Uvuvwevwevwe Onyetenwenvwe Ugwemubwenm OSSAS
Tekan [ENTER] untuk kembali ke menu utama...

```

6.4 Tampilan command LISTGAME jika berhasil

## 6.5 Data Test DELETE GAME

Command “DELETE GAME” menerima input nomor game yang ingin dihapus oleh pengguna dari daftar list game. Pertama akan dilakukan validasi terhadap input dari pengguna. Jika input antara 1 sampai 6, maka akan mengirimkan pesan kesalahan karena



isi list game awal dari file konfigurasi default tidak bisa dihapus. Jika input lebih dari 6 dan kurang dari atau sama dengan jumlah game dalam list game, maka akan dicek apakah game yang berada di nomor input dari pengguna berada di daftar antrian game. Game yang berada di queue game tidak dapat dihapus. Jika tidak ada di queue game, maka akan dihapus game sesuai nomor urutan yang diinput oleh pengguna. Jika input nomor lebih dari jumlah game yang ada di list game, maka akan dikirim pesan kesalahan.

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. Battleship
7. CERES FAUNA <3 DUKE
8. Uvuvewvewvwe Onyetemwemvwe Ugwenubwenm OSSAS
Masukkan nomor game yang akan dihapus: 8
Game berhasil dihapus
Apakah Anda ingin kembali melakukan input? [Y/N]: s

```

Gambar 6.5.1 Tampilan command DELETEDGAME jika berhasil

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. Battleship
7. CERES FAUNA <3 DUKE
Masukkan nomor game yang akan dihapus: 8
Nomor game tidak valid; game gagal dihapus.
Apakah Anda ingin kembali melakukan input? [Y/N]: 

```

Gambar 6.5.2 Tampilan command DELETEDGAME jika gagal

## 6.6 Data Test QUEUE GAME

Command “QUEUE GAME” adalah fitur untuk menambahkan game dari list game ke daftar antrian game. Pertama akan ditampilkan daftar antrian game. Jika daftar antrian game kosong maka akan menampilkan pesan antrian game kosong. Kemudian, ditampilkan daftar game dari list game yang dapat dimainkan. Setelah itu, program akan

meminta input dari pengguna berupa nomor urutan game dari list game yang ingin ditambahkan ke daftar antrian game. Program akan melakukan validasi terhadap inputan pengguna. Jika input nomor lebih dari 0 dan kurang dari atau sama dengan jumlah game dalam daftar list game, maka program akan menambahkan game pada nomor urutan input di list game ke daftar antrian game. Selain itu, program akan memberikan pesan kesalahan karena input yang tidak valid.

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
Antrian anda kosong
Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. Battleship
7. CERES FAUNA <3 DUKE

Nomor Game yang mau ditambahkan ke antrian: 1
Game berhasil ditambahkan ke dalam daftar antrian
Apakah Anda ingin kembali melakukan input? [Y/N]: 

```

Gambar 6.6.1 Tampilan command QUEUE GAME jika berhasil

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
Antrian anda kosong
Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. Battleship
7. CERES FAUNA <3 DUKE

Nomor Game yang mau ditambahkan ke antrian: 8
Nomor permainan tidak valid, silahkan masukkan nomor game pada list
Apakah Anda ingin kembali melakukan input? [Y/N]: 

```

Gambar 6.6.2 Tampilan command QUEUE GAME jika gagal

## 6.7 Data Test SAVE

Command “SAVE “ merupakan fitur untuk menyimpan state game pemain ke dalam suatu save file sesuai input pengguna. Jika dijalankan, maka program akan menyimpan daftar list game pemain saat itu dan juga jumlah game dalam list game pemain ke dalam save file input pengguna.

```
duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
*#%#(
/((*****/#%&@%#(((#
*@(/(.....(,
*/((.....@.....(,
/(&#%#(.....@#.....(
/(&#&/(.....(
/(&#&/(.....(
/((%&/(.....(
%((%&/(.....(
&(&/#/.....(
(((**#(.....(
&#&#(.....(
*#&/#(.....(
18220010 - Agnes Tamara Putri
18221102 - Salman Ma'arif Achsten
18221122 - Reswara Trista Aulia C
18221166 - Erensi Ratu Chelsia
18221168 - Rania Sasi Kirana

- Tugas Besar 1 IF2111 Algoritma dan Struktur Data STI - Kelompok 06 Kelas 02 -

Selamat Datang di BNMO! Jalankan dalam Linux; jangan ubah ukuran terminal.
Silahkan input perintah atau lihat daftarnya menggunakan [HELP]...
root@BNMO:~$ SAVE ceresfaunasayangku.txt
Save file berhasil disimpan
Tekan [ENTER] untuk melanjutkan...
```

Gambar 6.7.1 Tampilan command SAVE jika berhasil

```
duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
*#%#(
/((*****/#%&@%#(((#
*@(/(.....(,
*/((.....@.....(,
/(&#%#(.....@#.....(
/(&#&/(.....(
/(&#&/(.....(
/((%&/(.....(
%((%&/(.....(
&(&/#/.....(
(((**#(.....(
&#&#(.....(
*#&/#(.....(
18220010 - Agnes Tamara Putri
18221102 - Salman Ma'arif Achsten
18221122 - Reswara Trista Aulia C
18221166 - Erensi Ratu Chelsia
18221168 - Rania Sasi Kirana

- Tugas Besar 1 IF2111 Algoritma dan Struktur Data STI - Kelompok 06 Kelas 02 -

Selamat Datang di BNMO! Jalankan dalam Linux; jangan ubah ukuran terminal.
Silahkan input perintah atau lihat daftarnya menggunakan [HELP]...
root@BNMO:~$ SAVE fileaneh
Nama file tidak valid!
Tekan [ENTER] untuk melanjutkan...
```

Gambar 6.7.2 Tampilan command SAVE jika gagal

## 6.8 Data Test PLAY GAME

Command “PLAY GAME” merupakan fitur untuk memainkan permainan yang berada pada urutan pertama di antrian game. Pertama, program akan melakukan pengecekan terhadap antrian game. Jika antrian game kosong, maka program akan menampilkan pesan “Antrian Anda kosong.” karena tidak ada game yang dapat dimainkan. Namun, jika antrian tidak kosong, maka program akan menampilkan daftar game yang dapat dimainkan oleh user dan secara otomatis akan memainkan game yang berada pada urutan pertama dan mengeluarkannya dari daftar game. Jika game pada urutan pertama adalah RNG, Diner DASH, atau Battleship, maka program akan menjalankan prosedur rng, dinerDash, atau battleship. Namun, jika game pada urutan pertama adalah EIFFEL TOWER,

RISEWOMAN, atau DINOSAUR IN EARTH, maka program akan menampilkan pesan bahwa game belum dapat dimainkan.

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
Berikut adalah daftar Game di dalam antrian Anda.
1. RNG
2. DINOSAUR IN EARTH
3. CERES FAUNA <3 DUKE
4. Battleship
Loading RNG ...
Loading selesai! Tekan [ENTER] untuk memasuki permainan...

```

Gambar 6.8.1 Tampilan command PLAYGAME jika berhasil

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
Antrian Anda kosong.
Tekan [ENTER] untuk kembali ke menu utama...

```

Gambar 6.8.2 Tampilan command PLAYGAME jika gagal

```

duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...
=====BNMO=====
Berikut adalah daftar Game di dalam antrian Anda.
1. CERES FAUNA <3 DUKE
2. Battleship
Loading CERES FAUNA <3 DUKE ...
GAME OVER
Score: 80
Tekan [ENTER] untuk kembali ke menu utama...

```

Gambar 6.8.3 Tampilan command PLAYGAME dengan input user

### 6.9 Data Test SKIP GAME

Command “SKIP GAME” merupakan fitur untuk melewati game sebanyak n input pengguna di dalam daftar antrian game dan memainkan game tersebut. Pertama, program akan menampilkan daftar antrian game. Kemudian program akan meminta input dari pengguna berupa integer yang artinya banyak game yang ingin dilewati pada daftar antrian game. Jika input melebihi jumlah game pada daftar antrian game, maka antrian game akan

kosong dan dikeluarkan pesan kesalahan. Jika tidak, program akan mengeluarkan game sebanyak input pengguna dari antrian game kemudian memainkan game selanjutnya pada daftar game tersebut. Jika game yang akan dimainkan adalah RNG, Diner DASH, atau battleship (game bonus) maka program akan memainkan game tersebut. Jika game adalah EIFFEL TOWER, RISEWOMAN, atau DINOSAUR IN EARTH maka akan mengeluarkan pesan game masih dalam maintenance dan belum bisa dimainkan. Jika game merupakan game buatan pengguna dari create game, maka akan mengeluarkan pesan game over dan skor akhir integer random.

```

=====BNMO=====
          *#%(
/((***** /#%&@@%#(((#
*(@/(((.....(,
*/(((%...@.....@..(,
/(&#%#((%.....@#@.....(
/(&#%#/(((&.....(
/(@&#&(((@&%((((((&#&((
/(((%&/((((((((((((((((
%((%&#&/(((&***(((%,(((
&(& /*&/(((((((((((((((&#&@(&
(((**#(((((((((((#((&(( **@
&#&*((((((((((((&((#&(&(.&
          *& /#(((

18220010 - Agnes Tamara Putri
18221102 - Salman Ma'arif Achsien
18221122 - Reswara Trista Aulia C
18221166 - Erensi Ratu Chelsia
18221168 - Rania Sasi Kirana

- Tugas Besar 1 IF2111 Algoritma dan Struktur Data STI - Kelompok 06 Kelas 02 -

Selamat Datang di BNMO! Jalankan dalam Linux; jangan ubah ukuran terminal.
Silahkan input perintah atau lihat daftarnya menggunakan [HELP]...
root@BNMO:~$ SKIP GAME 2

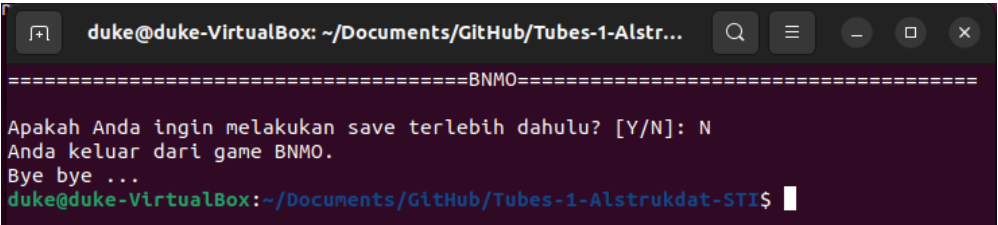
```

Gambar 6.9.1 Tampilan command SKIPGAME

## 6.10 Data Test QUIT

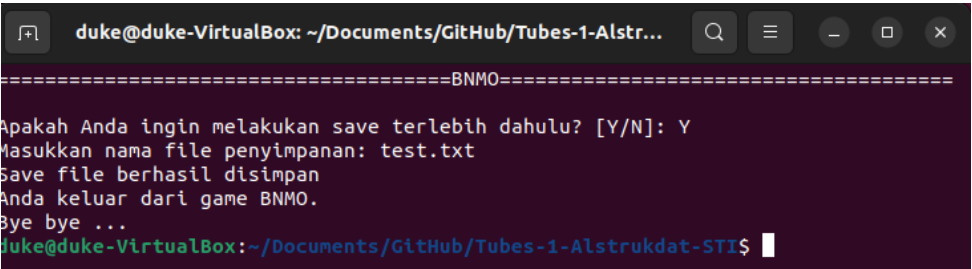
Command “QUIT” merupakan command untuk keluar dari program. Jika melakukan quit, program akan menanyakan apakah pengguna sudah melakukan save. Jika sudah maka program akan menjalankan command quit dan BNMO akan berhenti. Jika belum maka program akan menanyakan apakah pengguna ingin melakukan save terlebih dahulu. Jika iya, maka program akan menjalankan perintah save dan meminta input nama save file dari pengguna kemudian menjalankan perintah quit dan BNMO akan berhenti. Jika tidak,

program akan menjalankan perintah quit tanpa melakukan penyimpanan state game pemain saat itu.



```
=====BNMO=====
Apakah Anda ingin melakukan save terlebih dahulu? [Y/N]: N
Anda keluar dari game BNMO.
Bye bye ...
duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstrukdat-STI$
```

Gambar 6.10.1 Tampilan command QUIT tanpa SAVE

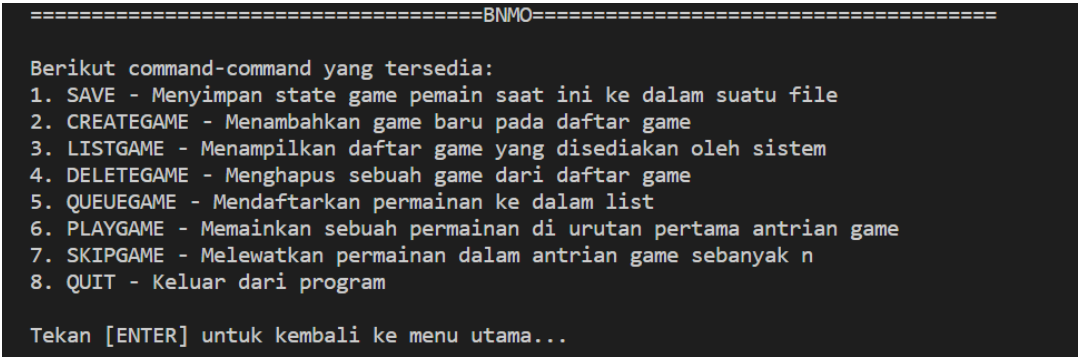


```
=====BNMO=====
Apakah Anda ingin melakukan save terlebih dahulu? [Y/N]: Y
Masukkan nama file penyimpanan: test.txt
Save file berhasil disimpan
Anda keluar dari game BNMO.
Bye bye ...
duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstrukdat-STI$
```

Gambar 6.10.2 Tampilan command QUIT dengan SAVE

6.11 Data Test HELP

Command “HELP” merupakan command untuk menunjukkan daftar command yang dapat digunakan pada BNMO. Command HELP akan dipanggil pada main program. Saat pertama kali program dijalankan, akan dikeluarkan petunjuk command yang dapat digunakan, yaitu start dan load. Jika start atau load sudah dijalankan, maka program akan mengeluarkan petunjuk command yang dapat dijalankan kecuali start dan load.



```
=====BNMO=====

Berikut command-command yang tersedia:
1. SAVE - Menyimpan state game pemain saat ini ke dalam suatu file
2. CREATEGAME - Menambahkan game baru pada daftar game
3. LISTGAME - Menampilkan daftar game yang disediakan oleh sistem
4. DELETEGAME - Menghapus sebuah game dari daftar game
5. QUEUEGAME - Mendaftarkan permainan ke dalam list
6. PLAYGAME - Memainkan sebuah permainan di urutan pertama antrian game
7. SKIPGAME - Melewatkan permainan dalam antrian game sebanyak n
8. QUIT - Keluar dari program

Tekan [ENTER] untuk kembali ke menu utama...
```

Gambar 6.11.1 Tampilan command HELP ketika START atau LOAD sudah dijalankan

```
=====BNMO=====
Berikut command-command yang tersedia:
1. START - Membaca file konfigurasi default yang berisi list game yang dapat dimainkan
2. LOAD - Membaca save file yang ingin dibuka berisi list game yang dapat dimainkan

Tekan [ENTER] untuk kembali ke menu utama...
```

Gambar 6.11.2 Tampilan command HELP ketika START dan LOAD belum dijalankan

## 6.12 Data Test Command Lain

Command “COMMAND LAIN” merupakan command yang akan ditampilkan jika terdapat command aneh atau command yang tidak ada pada program.

```
Command tidak dikenali, silahkan masukkan command yang valid.
```

Gambar 6.12 Tampilan command COMMAND LAIN jika diinputkan command yang tidak tersedia.

## 6.13 Data Test Diner DASH

Game “DINER DASH” merupakan salah satu daftar game yang merupakan permainan mengantar makanan namun terurut berdasarkan prioritasnya. Ada 3 command yang dapat dilakukan pada game diner dash, yaitu COOK, SERVE, dan SKIP. Pada diner dash akan ditampilkan Daftar Pesanan, Daftar Makanan yang sedang dimasak, dan Daftar Makanan yang dapat disajikan. Pada Daftar Pesanan akan ditampilkan ID Makanan, Durasi memasak, Ketahanan, dan Harga. Pada Daftar Makanan yang sedang dimasak akan ditampilkan ID Makanan dan Sisa durasi memasak. Pada Daftar Makanan yang dapat disajikan akan ditampilkan ID Makanan dan Sisa ketahanan makanan.

```

C:\Users\acer\Documents\tubes alstrukdat>.\try
Welcome to diner dash!

SALDO: 0

Daftar Pesanan:
Makanan | Durasi memasak | Ketahanan | Harga
-----
M0      | 5                | 2         | 20000
M1      | 3                | 5         | 25000
M2      | 1                | 4         | 15000

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

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

```

Gambar 6.13.1 Tampilan game diner dash ketika dimulai

```

Masukkan command: COOK M2
Command: COOK
orderId: M2
id: 2
Makanan M2 telah dimasukkan ke dalam antrian

SALDO: 0

Daftar Pesanan:
Makanan | Durasi memasak | Ketahanan | Harga
-----
M0      | 5                | 2         | 20000
M1      | 3                | 5         | 25000
M2      | 1                | 4         | 15000
M3      | 2                | 1         | 30000

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

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

=====

```

Gambar 6.13.2 Tampilan game diner dash setelah command cook berhasil



```

Masukkan command:COOK M1
Command: COOK
orderId: M1
id: 1
Makanan M1 telah dimasukkan ke dalam antrian

SALDO: 0

Daftar Pesanan:
Makanan | Durasi memasak | Ketahanan | Harga
-----
M0      | 5               | 2         | 20000
M1      | 3               | 5         | 25000
M2      | 1               | 4         | 15000
M3      | 2               | 1         | 30000
M4      | 4               | 5         | 30000

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

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----
M2      | 3

=====

```

Gambar 6.13.2 Tampilan game diner dash setelah berhasil cook, masuk ke tabel serve

```

Masukkan command:SERVE M2
Command: SERVE
orderId: M2
id: 2
Makanan M2 tidak dapat disajikan karena M0 belum selesai

SALDO: 0

Daftar Pesanan:
Makanan | Durasi memasak | Ketahanan | Harga
-----
M0      | 5               | 2         | 20000
M1      | 3               | 5         | 25000
M2      | 1               | 4         | 15000
M3      | 2               | 1         | 30000
M4      | 4               | 5         | 30000

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

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
-----
M2      | 2

=====

```

Gambar 6.13.3 Tampilan game diner dash jika gagal serve karena masih ada antrian sebelumnya

```

Masukkan command:SKIP M0
Command: SKIP
orderId: M0
id: 0

SALDO: 0

Daftar Pesanan:
Makanan | Durasi memasak | Ketahanan | Harga
ΓÇö-----
M0      | 5                | 2          | 20000
M1      | 3                | 5          | 25000
M2      | 1                | 4          | 15000
M3      | 2                | 1          | 30000
M4      | 4                | 5          | 30000
M5      | 5                | 3          | 15000

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

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
ΓÇö-----
M2      | 1

=====

```

Gambar 6.13.4 Tampilan game diner dash setelah command skip

```

Masukkan command:COOK M3
Command: COOK
orderId: M3
id: 3
Makanan M3 telah dimasukkan ke dalam antrian

SALDO: 0

Daftar Pesanan:
Makanan | Durasi memasak | Ketahanan | Harga
ΓÇö-----
M0      | 5                | 2          | 20000
M1      | 3                | 5          | 25000
M2      | 1                | 4          | 15000
M3      | 2                | 1          | 30000
M4      | 4                | 5          | 30000
M5      | 5                | 3          | 15000
M6      | 5                | 2          | 15000

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

Daftar Makanan yang dapat disajikan
Makanan | Sisa ketahanan makanan
ΓÇö-----
M2      | 0
M1      | 4

=====
===== GAME OVER =====
SKOR AKHIR : 0

```

Gambar 6.13.5 Tampilan game diner dash jika sudah ada 7 antrian

## 6.14 Data Test RNG

RNG merupakan game tebak-tebakan dimana pengguna menebak angka yang dihasilkan program secara random. Jika input pengguna lebih besar dari angka dari program, maka akan mengeluarkan pesan kesalahan. Begitu juga jika input lebih kecil. Jika input sudah sesuai dengan angka yang dihasilkan program, maka program akan memberikan pesan bahwa input sudah sesuai dan program akan berhenti. Program juga akan mengeluarkan skor akhir yaitu 100 dikurangi jumlah tebakkan yang dilakukan pengguna hingga berhasil. Jika skor akhir kurang dari 0 maka skor akhirnya menjadi 0.

```
duke@duke-VirtualBox: ~/Documents/GitHub/Tubes-1-Alstr...  
=====BNMO=====
```

```
Selamat datang di  
:::~::~ ~:: ~:: ~::~:  
:+:   :+: +++: +: :+:    :+  
+++   +-+ +++++ +-+ +-+  
++++!++#: ++# ++# +#:#  
##+ ##+ ##+ #####+##+ ###+  
##+ ##+ ##+ #####+##+ ###+  
### ## # ## ##+## ##+ ##+  
### ### ### ##### #####  
  
Are you lucky..?
```

```
=====  
=====RNG telah dimulai. Uji keberuntungan Anda dengan menebak X.=====
```

```
Tebakan: 50  
Lebih Kecil  
Tebakan:
```

Gambar 6.14.1 Tampilan RNG dengan tebakan valid





```

=====BNMO=====

PAPAN ANDA SAAT INI
A B C D E F G H I J
0) # # # # . . . . .
1) . . . . .
2) . . . . .
3) . . . . .
4) . . . . .
5) . . . . .
6) . . . . .
7) . . . . .
8) . . . . .
9) . . . . .

Silahkan masukkan posisi dan orientasi kapal Battleship.
Silahkan masukkan koordinat dengan format [A-J][0-9]: A0
Apakah kapal berorientasi vertikal? [Y/N]: N
Posisi kapal tidak valid! Ulangi masukan.
Silahkan masukkan koordinat dengan format [A-J][0-9]: 

```

Gambar 6.15.3 Tampilan battleship jika masukan tidak valid

```

=====BNMO=====

PAPAN ANDA                                PAPAN Yor
A B C D E F G H I J | A B C D E F G H I J
0) # # # # . . . . . | . . . . . (0
1) # # # # . . . . . | . . . . . (1
2) # # # # . . . . . | . . . . . (2
3) # # # # . . . . . | . . . . . (3
4) # # # # . . . . . | . . . . . (4
5) . . . . . | . . . . . (5
6) . . . . . | . . . . . (6
7) . . . . . | . . . . . (7
8) . . . . . | . . . . . (8
9) . . . . . | . . . . . (9
A B C D E F G H I J | A B C D E F G H I J
5/5                                KAPAL TERSISA                                5/5

=====

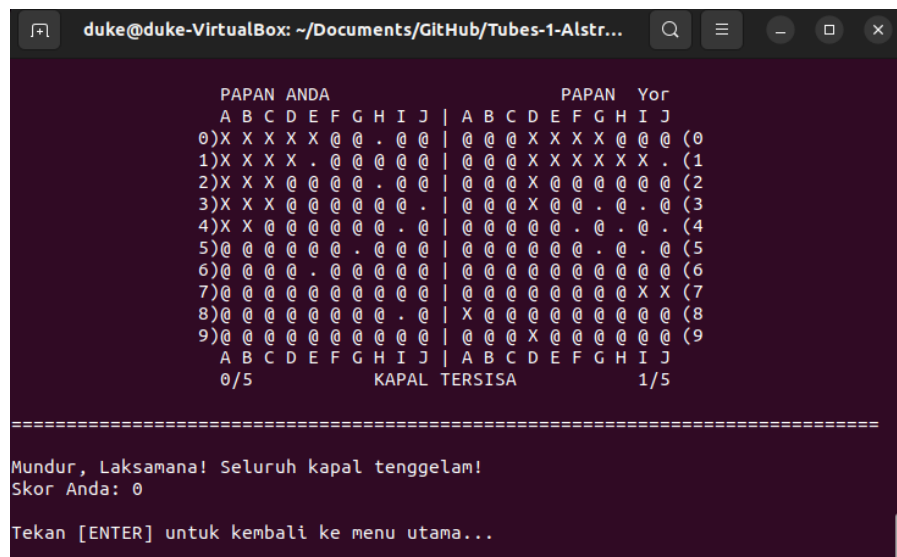
Silahkan tembak salah satu koordinat lawan.
Silahkan masukkan koordinat dengan format [A-J][0-9]: 

```

Gambar 6.15.4 Tampilan battleship untuk menembak



Gambar 6.15.5 Tampilan battleship untuk memasukkan koordinat serangan



Gambar 6.15.6 Tampilan battleship jika sudah selesai

## 7 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	START	Untuk menguji apakah fungsi start dapat mengakses <i>file</i>	Mengetikkan START pada input command	START	Gambar 6.1	Sesuai yang diharapkan

		konfigurasi awal sehingga <i>file</i> dapat dibuka.				
2	LOAD	Untuk menguji apakah program dapat membaca save file yang sudah ada	Mengetikkan LOAD pada input command	LOAD	Gambar 6.2.1 Gambar 6.2.2	Sesuai yang diharapkan
3	CREATE GAME	Untuk menguji apakah program dapat membuat game baru sesuai input pengguna dan menambahkannya pada daftar game sesuai ketentuan	Mengetikkan CREATE GAME pada input command kemudian menginput nama game yang ingin ditambah ke daftar game	CREATE GAME	Gambar 6.3.1 Gambar 6.3.2	Sesuai yang diharapkan
4	LIST GAME	Menampilkan daftar list game yang dapat dimainkan	Mengetikkan LIST GAME pada input command	LIST GAME	Gambar 6.4	Sesuai yang diharapkan
5	DELETE GAME	Untuk menguji apakah command ini dapat digunakan untuk menghapus game pada list game sesuai ketentuan	Mengetikkan DELETE GAME pada input command kemudian input nomor game pada list yang ingin dihapus	DELETE GAME	Gambar 6.5.1 Gambar 6.5.2	Sesuai yang diharapkan
6	QUEUE GAME	Untuk menguji apakah command ini dapat digunakan untuk menambahkan game pada list game ke daftar antrian game sesuai ketentuan	Mengetikkan QUEUE GAME pada input command kemudian input nomor game pada list game yang ingin ditambah ke queue game	QUEUE GAME	Gambar 6.6.1 Gambar 6.6.2	Sesuai yang diharapkan
7	SAVE	Untuk menguji apakah program bisa melakukan penyimpanan pada file txt	Mengetikkan SAVE <filename> pada input command	SAVE <filename>	Gambar 6.7.1 Gambar 6.7.2	Sesuai yang diharapkan
8	PLAY GAME	Untuk menguji apakah program bisa memainkan game pertama	Mengetikkan PLAY GAME pada input command	PLAY GAME	Gambar 6.8.1 Gambar 6.8.2 Gambar 6.8.3	Sesuai yang diharapkan



		pada daftar antrian game				
9	SKIP GAME	Untuk menguji apakah program bisa melewati game sebanyak n pada daftar antrian game kemudian memainkan game tersebut	Mengetikkan SKIP GAME <n> pada input command dengan n adalah banyaknya game yang mau dilewati pada daftar antrian game	SKIP GAME <n>	Gambar 6.9.1	Sesuai yang diharapkan
10	QUIT	Untuk menguji apakah command QUIT bisa dijalankan atau tidak.	Mengetikkan QUIT pada input command.	QUIT	Gambar 6.10.1 Gambar 6.10.2	Sesuai yang diharapkan
11	HELP	Untuk menguji apakah command QUIT bisa dijalankan atau tidak.	Mengetikkan HELP pada input command	HELP	Gambar 6.11.1 Gambar 6.11.2	Sesuai yang diharapkan
12	COMMAND LAIN	Untuk menguji apakah command COMMAND LAIN bisa dijalankan atau tidak.	Mengetikkan command yang tidak valid.	<command_a neh>	Gambar 6.12	Sesuai yang diharapkan
13	DINER DASH	Untuk menjalankan diner dash dengan menginputkan command-command nya yaitu COOK, SERVE, ataupun SKIP.	Mengetikkan DINER DASH dan melakukan input COOK, SERVE, ataupun SKIP.	DINER DASH	Gambar 6.13.1 Gambar 6.13.2 Gambar 6.13.3 Gambar 6.13.4 Gambar 6.13.5	Sesuai yang diharapkan
14	RNG	Untuk menguji apakah game RNG dapat berjalan	Memasukkan input integer	QUEUE GAME 1 PLAY GAME	Gambar 6.14.1 Gambar 6.14.2 Gambar 6.14.3	Sesuai yang diharapkan
15	BATTLESHIP	Untuk menguji apakah game BATTLESHIP dapat berjalan	Mengetik lawan yang diinginkan kemudian menginput koordinat	QUEUE GAME 6 PLAY GAME	Gambar 6.15.1 Gambar 6.15.2 Gambar 6.15.3 Gambar 6.15.4 Gambar 6.15.5 Gambar 6.15.6	Sesuai yang diharapkan

## 8 Pembagian Kerja dalam Kelompok

Nama Anggota	Pembagian Tugas
18220010 - Agnes Tamara Putri	Membuat program queuegame, playgame, skipgame, mengisi laporan bagian ADT, program utama, dan data test
18221102 - Salman Ma'arif Achsien	Membuat program ADT point, ADT Word, game battleship, game rng, splash, fungsi-fungsi tambahan serta menyatukan program di main; dokumen bagian data test.
18221122 - Reswara Trista Aulia C.	Membuat program start, load, dan game diner dash, ADT prioqueue, membuat Makefile, mengisi laporan bagian data test, ringkasan, ADT queue2, algoritma menarik, test script.
18221166 - Erensi Ratu Chelsia	Membuat program help, commandlain, quit, game diner dash, dan ADT prioqueue, serta mengisi laporan bagian battleship, ADT Point, ADT Grid, dan Data Test.
18221168 - Rania Sasi Kirana	Membuat program creatgame, deletgame, listgame, save, driver ADT, ADT queue2, ADT array, menyatukan program di main, mengisi laporan bagian data test dan test script

## 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.

Program ini merupakan permainan berbasis CLI (command-line interface). Sistem ini dibuat dalam bahasa C. BNMO memiliki beberapa fitur utama yaitu memainkan game, menambahkan game, menghapus game, dan mengurutkan game yang dimainkan. Pada program ini ada main menu dan juga command. Main menu ini berisi welcome page dan beberapa menu pilihan start dan load. Main menu ini akan menerima input commands.


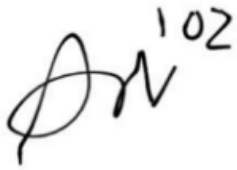
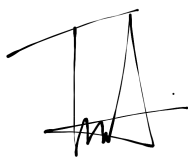

## 9.2 Notulen Rapat



**Form Asistensi Tugas Besar  
IF2110/Algoritma dan Struktur Data  
Sem. 1 2022/2023**

No. Kelompok/Kelas : 06/K02  
Nama Kelompok :  
Anggota Kelompok (Nama/NIM) :  
1. Agnes Tamara Putri (18220010)  
2. Salman Ma' Arif Achsien (18221102)  
3. Reswara Trista Aulia C (18221122)  
4. Erensi Ratu Chelsia (18221166)  
5. Rania Sasi Kirana (18221168)  
  
Asisten Pembimbing : Aditya Bimawan (13519064)


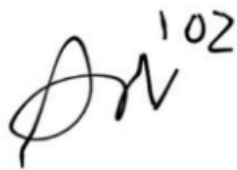


---



# Asistensi I

<b>Tanggal : 03 November 2022</b>	<b>Catatan Asistensi:</b> <ul style="list-style-type: none"> <li>- coba pake wsl untuk ngerunnya, gak perlu linux juga bisa.</li> <li>- inputnya harus di mesin kata.</li> <li>- tidak boleh langsung dari mesin karakter, harus ke mesin kata dulu</li> <li>- quit: gak perlu didequeue untuk semua list antrian yang ada di queue karena kalo programnya udah keluar udah gak ada lagi juga untuk list antriannya.</li> </ul>
<b>Tempat : Zoom Meeting</b>	
<b>Kehadiran Anggota Kelompok:</b> <ol style="list-style-type: none"> <li>1. NIM 18220010 Tanda tangan </li> <li>2. NIM 18221102 Tanda tangan </li> <li>3. NIM 18221122 Tanda tangan </li> <li>4. NIM 18221166 Tanda tangan </li> <li>5. NIM 18221168 Tanda tangan</li> </ol>	

	
Aditya Bimawan (13519064)	<b>Tanda Tangan Asisten:</b> 

## Asistensi II

Tanggal : 10 November 2022	<b>Catatan Asistensi:</b> - boleh include sys gak sih kak? jawaban: kalo dari aku mending gak usah sih  - makefile itu harus dipake gak kak? jawaban: itu cuman untuk bantu compile sih. Jadi, kalo ada waktu untuk bikin makefile, bikin aja untuk mempermudah compilenya.  - nama file dari fungsi load selalu valid gak kak? jawaban: harus ada handle nama file yang salahnya.  Jangan lupa dilengkapin programnya dan bikin driver untuk ADT nya.
Tempat : Zoom Meeting	
Kehadiran Anggota Kelompok:	
1. NIM 18220010 Tanda tangan 	
2. NIM 18221102 Tanda tangan 	
3. NIM 18221122 Tanda tangan 	
4. NIM 18221166 Tanda tangan 	
5. NIM 18221168 Tanda tangan	

	
Aditya Bimawan (13519064)	<b>Tanda Tangan Asisten:</b> 

### 9.3 Log Activity Anggota Kelompok

No	Waktu	Keterangan
1.	30 Oktober 2022	- Meet perdana dan pembagian tugas
2.	3 November 2022	- Asistensi 1
3.	10 November 2022	<ul style="list-style-type: none"> <li>- Asistensi 2</li> <li>- Penyusunan laporan</li> <li>- Menggabungkan seluruh program</li> <li>- Melakukan debugging</li> </ul>