

# **LAPORAN TUGAS BESAR**

## **IF2111 Algoritma dan Struktur Data**


### **BNMO**

Dipersiapkan oleh:

Kelompok 2

18221090 Marcheline Fanni Hidayat Putri  
18221124 Jazmy Izzati Alamsyah  
18221126 Karunia Mega Lestari  
18221130 Rayhan Maheswara Pramanda  
18221152 Raditya Azka Prabaswara

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

	<b>Sekolah Teknik Elektro dan Informatika ITB</b>	<b>Nomor Dokumen</b>		<b>Halaman</b>
		<i>IF2111-TB1-2</i>		23
		<i>Revisi</i>	1	2022-10-28



# Daftar Isi

1	Ringkasan	4
2	<b>Penjelasan Tambahan Spesifikasi Tugas</b>	<b>4</b>
2.1	Game Tambahan (bonus)	4
2.2	Paper, Rock, Scissor (bonus)	4
3	<b>Struktur Data (ADT)</b>	<b>5</b>
3.1	ADT ArrayGame	5
3.2	ADT MesinKarakter	6
3.3	ADT MesinKata	6
3.4	ADT QueueGame	7
3.5	ADT QueueDD	8
3.6	ADT Map	9
4	<b>Program Utama</b>	<b>10</b>
5	<b>Algoritma-Algoritma Menarik</b>	<b>11</b>
5.1	Algoritma RNG Versi 1	11
5.2	Algoritma RNG Versi 2	12
5.3	Algoritma IntToWord	12
6	<b>Data Test</b>	<b>12</b>
6.1	Data Test 1	12
6.2	Data Test 2	14
6.3	Data Test 3	15
6.4	Data Test 4	18
7	<b>Test Script</b>	<b>21</b>
8	<b>Pembagian Kerja dalam Kelompok</b>	<b>24</b>
9	<b>Lampiran</b>	<b>25</b>
9.1	Deskripsi Tugas Besar 2	25
9.2	Notulen Rapat	26
9.3	Log Activity Anggota Kelompok	28

# 1 Ringkasan

BNMO adalah sebuah robot video game console milik Indra dan Doni. Di dalamnya terdapat game yang melibatkan RNG (*random number generator*), seperti Diner dash, yaitu sebuah permainan mengantar makanan yang terurut berdasarkan prioritasnya. Namun, pada suatu hari, BNMO mengalami kerusakan. Indra dan Doni telah mencoba memperbaikinya, naas, alih-alih dapat dimainkan kembali, game tersebut justru mendapatkan lebih banyak bug dalam sistemnya. Oleh karena itu, Indra dan Doni meng-hire beberapa programmer handal agar robot tersebut dapat berfungsi sebagaimana mestinya.

Pada dasarnya, diner dash adalah sebuah permainan dengan command-line interface (CLI). Permainan akan dimulai dengan 3 pelanggan. Masing-masing pelanggan hanya dapat memesan satu porsi makanan. Kemudian, akan ditampilkan informasi tentang ID makanan, durasi memasak, harga makanan, serta ketahanan makanan menggunakan RNG. Terdapat dua buah command yang dapat dilakukan pada game, yaitu COOK yang bertujuan untuk memasak makanan dan SERVE yang bertujuan untuk menyajikan makanan kepada pelanggan.

Laporan ini menjelaskan tentang program-program yang telah kami buat. Bagian pertama merupakan ringkasan yang berisi deskripsi umum persoalan dan penjelasan singkat laporan. Bagian kedua berisi tentang penjelasan tambahan spesifikasi tugas besar. Bagian ketiga berisi tentang struktur-struktur data (ADT) yang digunakan oleh kelompok kami. Bagian keempat adalah penjelasan program utama. Dan yang terakhir, ada algoritma-algoritma menarik yang kami gunakan.

Seluruh program dibuat menggunakan bahasa pemrograman C dengan menggunakan ADT yang telah dipelajari pada mata kuliah IF2111 Algoritma dan Struktur Data, antara lain, ADT ArrayGame, ADT MesinKarakter, ADT MesinKata, ADT QueueGame, ADT QueueDD, serta ADT Map.

## 2 Penjelasan Tambahan Spesifikasi Tugas

### 2.1 Game Tambahan (bonus)

Permainan ini tidak menggunakan ADT, namun memanfaatkan RNG untuk membuat angka random sebagai skor akhir. Skor akan dicetak setelah muncul kalimat “GAME OVER!”

### 2.2 Paper, Rock, Scissor (bonus)

Permainan ini menggunakan ADT arrayGame. Permainan akan meminta input dari user berupa paper, rock, atau scissor. Lalu, dengan memanfaatkan RNG akan muncul secara random rock, paper, atau scissor dengan sistem *rock trumps scissor*, *scissor trumps paper*, dan *paper trumps rock*.

### 3 Struktur Data (ADT)

#### 3.1 ADT ArrayGame

ADT arrayGame digunakan untuk menyelesaikan masalah terkait list dan penyimpanan kata pada array, seperti mengecek apakah array kosong, panjang array, apakah suatu elemen adalah member array tersebut atau bukan, insert element, delete element, maupun mencetak array yang kemudian akan diimplementasikan pada fungsi Start, Load, Save, QueueGame, DeleteGame, CreateGame, ListGame, serta ADT MesinKata.

ADT ini diimplementasikan dalam file arrayGame.c, arrayGame.h, dan driver\_arrGame.c.

Sketsa struktur data (dalam C):

```
typedef int IdxType;
typedef Word Game;
typedef struct {
    Game A[100];
    int Neff;
} arrGame;
```

Berikut primitif dalam ADT QueueGame:

1. Konstruktor:  
void MakeArray(arrGame \*arr)
2. Operasi:  
boolean IsEmptyArr(arrGame arr)  
int Length(arrGame arr)  
Game Get(arrGame arr, IdxType i)  
boolean IsMemberArr(arrGame arr, Game el)
3. Penambahan dan pengurangan elemen:  
void InsertAt(arrGame \*arr, Game el, IdxType i)  
void InsertLast(arrGame \*arr, Game el)  
void InsertFirst(arrGame \*arr, Game el)  
void DeleteAt(arrGame \*arr, IdxType i)  
void DeleteLast(arrGame\*arr)  
void DeleteFirst(arrGame \*arr)
4. Penulisan isi queue:  
void PrintArray(arrGame arr)

### 3.2 ADT MesinKarakter

ADT MesinKarakter tidak secara langsung digunakan dalam program, tetapi sebagai dasar bagi ADT MesinKata. MesinKarakter membaca konfigurasi file karakter per karakter sampai bertemu mark yang mana akan berhenti membaca input. ADT ini dideklarasikan di mesinkarakter.h dan diimplementasikan di mesinkarakter.c serta di ADT mesinkata dan fungsi yang lain.

ADT ini diimplementasikan dalam file mesinkarakter.h dan mesinkarakter.c.

Berikut primitif dalam ADT MesinKarakter:

1. Pembacaan karakter:  
void START\_INPUT()  
void ADV\_INPUT()
2. Pembacaan file:  
void START(char\* filename)  
void ADV()
3. Predikat:  
boolean IsEOP()
4. Selektor  
char GetCC()

### 3.3 ADT MesinKata

ADT MesinKata digunakan untuk membaca konfigurasi file serta menerima input command dan mengecek apakah command tersebut sesuai dengan command yang ada. ADT ini diimplementasikan pada command yang memerlukan input kata, hampir semua kecuali command help. Selain itu, RINGGAME, Diner Dash, serta Paper, Rock, Scissor juga menggunakan ADT ini.

ADT ini diimplementasikan dalam file mesinkata.h, mesinkata.c, dan driver\_mesinkata.c  
Sketsa struktur data (dalam C):

```
typedef char string[50];  
typedef struct  
{  
    string TabWord;  
    int Length;  
} Word;
```

Berikut primitif dalam ADT MesinKata:

1. Pembacaan *file*:
 

```
void STARTWORD(char* filename)
void ADVWORD();
void CopyWord();
```
2. Pembacaan kata:
 

```
void IgnoreBlanks()
void IgnoreEnters()
void STARTWORD_INPUT()
void ADVWORD_INPUT()
void CopyWord_Input();
```
3. Pembersihan kata:
 

```
void clearWord()
```
4. Predikat:
 

```
boolean isEndWord()
boolean isKataEqual(Word word, char* string)
boolean isIdentical(Word kata1, Word kata2)
```
5. Operasi tambahan:
 

```
char intToChar(int n)
void CloneWord(Word *word1, Word word2)
void SetWordChar(Word *word, int i, char c)
void Reverse(Word *word)
Word intToWorld(int n)
int wordToInt (Word word)
int wordToIntDD (Word word)
void InsertLastW(Word* word, char* str)
```

### 3.4 ADT QueueGame

ADT queueGame digunakan untuk menyimpan dan memproses elemen pada daftar game atau game yang hendak dimainkan, baik itu berupa enqueue, maupun dequeue. ADT ini diperlukan untuk membuat fungsi QueueGame, DeleteGame, PlayGame, dan SkipGame.

ADT ini diimplementasikan dalam file queueGame.h, queueGame.c, dan driver\_queueGame.c

Sketsa struktur data (dalam C):

```
typedef Word Game;
typedef struct {
    Game A[CAPACITY];
    int idxHead;
    int idxTail;
} queueGame;
```

Berikut primitif dalam ADT QueueGame:

1. Konstruktor:  
void CreateQueue(queueGame \*q)
2. Selektor:  
IDX\_HEAD(q)  
IDX\_TAIL(q)  
HEAD(q)  
TAIL(q)
3. Operasi:  
boolean IsEmptyQ(queueGame q)  
boolean IsFullQ(queueGame q)  
boolean IsMemberQ(queueGame q, Game val)  
int LengthQ(queueGame q)
4. Penambahan dan pengurangan elemen:  
void Enqueue(queueGame \*q, Game val)  
void Dequeue(queueGame \*q, Game \*val)
5. Penghapusan queue:  
void DeleteQueue (queueGame \*q)
6. Penulisan isi queue:  
void PrintQueue(queueGame q)

### 3.5 ADT QueueDD

ADT queueDD pada dasarnya mirip dengan ADT QueueGame, akan tetapi ada perbedaan elemen pada fungsi yang digunakan sehingga tidak dapat mengimplementasikan ADT QueueGame secara langsung, yaitu pada diner dash; dibutuhkan suatu primitif untuk melihat setiap elemen dalam queue.

ADT ini diimplementasikan dalam file queuedd.h, queuedd.c, dan driver\_queuedd.c



Sketsa struktur data (dalam C):

```
typedef struct {
    int key;
    int duration;
    int spoliage;
    int price;
} ElTypedd;
typedef struct {
    ElTypedd buffer[CAPACITY];
    int idxHead;
    int idxTail;
} QueueDD;
```

Berikut primitif dalam ADT QueueDD:

1. Kreator:  
void CreateQueuedd (QueueDD \*q)
2. Prototype:  
boolean isEmptydd (QueueDD q)  
boolean isFullldd (QueueDD q)  
int lengthdd (QueueDD q)
3. Primitif Add/Del:  
void enqueue (QueueDD \*q, int key, int duration, int spoliage, int price)  
void dequeuedd (QueueDD \*q, int \*key, int \*duration, int \*spoliage, int \*price)
4. Menampilkan Queue:  
void displayQueuedd (QueueDD q)
5. Melihat nilai dalam Queue:  
void takeValue (QueueDD q, int key, int \*duration, int \*spoliage, int \*price)

### 3.6 ADT Map

ADT Map digunakan untuk menyelesaikan terkait COOK dan SERVE di program diner dash. Pada gim Diner Dash, makanan yang sedang dimasak dan siap disajikan tidak penting urutannya. Adanya nilai tertentu yang unik memungkinkan penggunaan ADT Map yang memiliki fungsi IsMember.

ADT ini diimplementasikan dalam file map.h, map.c, dan driver\_map.c

Sketsa struktur data (dalam C):

```
typedef int keytype;
typedef int valuetype;
typedef int address;

typedef struct {
    keytype Key;
    valuetype Value;
} infotype;

typedef struct {
    infotype Elements[MaxEl];
    address Count;
} Map;
```

Berikut primitif dalam ADT QueueDD:

1. Konstruktor:  
void CreateEmpty(Map \*M)
2. Predikat:  
boolean IsEmpty(Map M)  
boolean IsFull(Map M)  
boolean IsMember(Map M, keytype k)
3. Operator dasar:  
valuetype Value(Map M, keytype k)  
void Insert(Map \*M, keytype k, valuetype v)  
void Delete(Map \*M, keytype k)

## 4 Program Utama

Pada program utama, program akan mendeklarasi dan membuat *array* dari game serta *queue* dari game yang kosong. Program utama juga memiliki sebuah boolean “play” yang diinisiasi dengan nilai *true*. Ketika dijalankan, program akan menampilkan UI program berupa tulisan BNOMO. Program akan masuk ke dalam sebuah loop selama “play” bernilai *true*. Program akan meminta pengguna untuk memasukkan command dan mesin kata dijalankan untuk menerima input dari pengguna. Program akan menjalankan fungsi atau prosedur sesuai dengan masukan dari pengguna yang dibandingkan dengan command yang tersedia menggunakan ADT mesin kata.

Ketika pengguna memasukkan kata “START”, program akan menjalankan command START yang melakukan *loading default configuration*. Jika pengguna memasukkan kata “LOAD”, program akan meminta masukan lagi dari pengguna lalu menjalankan command LOAD untuk

*loading* file konfigurasi yang namanya sesuai dengan masukan pengguna sebelumnya. Jika pengguna memasukkan kata “SAVE”, program akan meminta masukan nama file dari pengguna lalu menjalankan command SAVE yaitu menyimpan file tersebut.

Jika pengguna memasukkan kata “CREATEGAME”, maka program akan menjalankan command CREATEGAME yang akan menerima input dari pengguna menggunakan mesin kata, kemudian menambahkan nama game tersebut ke dalam daftar (*list*) game. Jika pengguna memasukkan kata “LISTGAME”, program akan menjalankan command LISTGAME yang menampilkan daftar game yang disediakan oleh sistem. Jika pengguna memasukkan kata “DELETGAME”, program akan menjalankan command DELETGAME yang akan menampilkan daftar game yang tersedia, menerima masukan dari pengguna berupa nomor game yang akan dihapus, kemudian menghapus game tersebut. Jika masukan pengguna berupa kata “QUEUEGAME”, maka program akan menjalankan command QUEUEGAME yang akan menampilkan daftar antrian game pengguna serta daftar game yang tersedia, kemudian menerima *input* dari pengguna berupa nomor yang mau ditambahkan ke antrian, lalu memasukkannya ke dalam antrian (*queue*) game. Jika masukan pengguna berupa kata “PLAYGAME”, maka program akan menjalankan command PLAYGAME yang berfungsi untuk memainkan sebuah permainan dengan urutan pertama di antrian game. Jika pengguna memasukkan kata “SKIPGAME”, program akan menerima masukan lagi dengan mesin kata berupa banyaknya permainan yang akan diskip (*n*) kemudian program akan menjalankan command SKIPGAME yang berfungsi untuk melewati permainan sebanyak *n*.

Jika masukan pengguna adalah “HELP”, maka program akan menjalankan command HELP yang berupa bantuan command-command yang tersedia. Jika pengguna memasukkan kata “QUIT”, program akan menjalankan command QUIT yang berfungsi untuk keluar dari program, kemudian nilai dari boolean “play” akan bernilai *false* sehingga program akan keluar dari *while loop*. Apabila masukan tidak sesuai dengan yang telah disebutkan sebelumnya, program akan menjalankan command COMMAND\_LAIN yang meminta pengguna untuk memasukkan command yang lain.

## 5 Algoritma-Algoritma Menarik

### 5.1 Algoritma RNG Versi 1

Algoritma *Random Number Generator* (RNG) versi 1 menarik karena jika fungsinya dipanggil berkali-kali secara berurutan, maka beberapa hasil yang dikeluarkan bernilai sama untuk urutan tertentu. Hal ini dikarenakan fungsinya mengambil *seed* yang bergantung pada time dan clock dan fungsinya mengambil *seed* tiap kali fungsinya dipanggil. Namun, tidak diketahui banyaknya hasil sama tersebut dan terkadang terdapat hasil yang tidak sama secara berurutan.

## 5.2 Algoritma RNG Versi 2

Algoritma *Random Number Generator* (RNG) versi 2 menarik karena dapat digunakan berulang kali dalam waktu yang singkat. Hal tersebut dapat dilakukan karena hanya dilakukan satu kali pengambilan *seed*.

## 5.3 Algoritma IntToWord

Algoritma IntToWord menarik sebab proses konversinya menggunakan modulo dengan division. Namun, hasil dari algoritma tersebut akan menghasilkan urutan angka yang terbalik.

Misalnya 153

$153 \% 10 = 3$  lalu  $153 / 10 = 15$

$15 \% 10 = 5$  lalu  $15 / 10 = 1$

$1 \% 10 = 1$

Jika diinput ke dalam bentuk Word, urutannya akan menjadi 351. Oleh sebab itu, dibuatlah prosedur lain bernama Reverse untuk membalik urutan elemen Word menjadi yang seharusnya.

# 6 Data Test

Program BNMO dapat menjalankan permainan dengan 12 command dan main menu sehingga dapat menjalankan fitur-fiturnya. Pada BNMO, fitur utamanya adalah memainkan game, menambahkan game, menghapus game, serta mengurutkan game yang akan dimainkan. Data testing perlu dilakukan untuk memastikan bahwa main menu serta command-command yang terdapat pada program telah berjalan dengan baik agar fitur dapat berjalan sesuai dengan harapan.

## 6.1 Data Test 1

Pada data test pertama, dilakukan pengecekan untuk fitur penambahan game. Hasil yang diharapkan adalah berhasilnya proses penambahan game yang ditunjukkan dari tersimpannya nama game yang ditambahkan pada default\_config.txt.

- a. Berikut merupakan tampilan default\_config.txt sebelum penambahan game

```
data > default_config.txt
You, last week | 1 author (You)
1 5
2 RNG
3 Diner DASH
4 DINOSAUR IN EARTH
5 RISEWOMAN
6 EIFFEL TOWER
7 .
8
```

- b. Pengecekan dimulai dengan melakukan command START

```

$ src/main.exe
=====SELAMAT DATANG DI=====
/$$$$$$$ /$$ /$$ /$$$$$$ /$$ /$$ /$$$$$$$
| $$__ $$| $$$ |$$ /$__$ $$| $$$ /$$$ /$__$ $$
| $$ \ $$| $$$$| $$| $$ \ $$| $$$$ /$$$$| $$ \ $$
| $$$$$$| $$ $$ $$| $$| $$| $$ $$/$$ $$| $$| $$
| $$__ $$| $$ $$$$| $$| $$| $$ $$$$| $$| $$| $$
| $$ \ $$| $$\ $$$$| $$| $$| $$\ $| $$| $$| $$
| $$$$$$/| $$ \ $$| $$$$$$/| $$ \ /| $$| $$$$$$/
|_____/|_/ \_/ \_/ \_/ \_/ \_/ \_/ \_/ \_/ \_/
=====
MASUKKAN COMMAND: START
Save file berhasil dibaca.

```

- c. Kemudian penambahan game dengan command CREATEGAME lalu memasukkan nama game yang akan ditambahkan

```

MASUKKAN COMMAND: CREATEGAME
Masukkan nama game yang akan ditambahkan: NEWGAME
Game berhasil ditambahkan

```

- d. Setelah game berhasil ditambahkan, command SAVE digunakan untuk menyimpan penambahan game ke dalam file default\_config.txt

```

MASUKKAN COMMAND: SAVE default_config.txt
Save file berhasil disimpan.

```

- e. Command QUIT dilakukan untuk keluar dari program tersebut

```

MASUKKAN COMMAND: QUIT
Anda keluar dari game BNMO.
Bye bye...

```

- f. Berikut hasil dari default\_config.txt setelah penambahan game

```

data > default_config.txt
You, 7 seconds ago | 1 author (You)
1 6
2 RNG
3 Diner DASH
4 DINOSAUR IN EARTH
5 RISEWOMAN
6 EIFFEL TOWER
7 NEWGAME
8 . You, 8 seconds ago • Uncomm

```

## 6.2 Data Test 2

Pada data test kedua, dilakukan pengecekan untuk fitur melihat daftar game yang tersedia, melihat antrian game pengguna, memilih game yang akan dimainkan, serta memainkan game. Hasil yang diharapkan adalah berhasilnya proses yang telah disebutkan sebelumnya yang ditunjukkan dari berhasilnya dimainkan game pada antrian game pengguna.

- a. Menggunakan command LOAD untuk membaca file konfigurasi

```
$ src/main.exe
=====SELAMAT DATANG DI=====
/ $ $ $ $ $ $ / $ $ / $ $ / $ $ $ $ $ / $ $ / $ $ $ $ $
| $ $ _ $ $ | $ $ $ $ | $ $ / $ $ _ $ $ | $ $ $ $ / $ $ $ / $ $ _ $ $
| $ $ \ $ $ | $ $ $ $ | $ $ $ $ \ $ $ | $ $ $ $ / $ $ $ $ | $ $ \ $ $
| $ $ $ $ $ $ | $ $ $ $ $ $ | $ $ | $ $ | $ $ $ $ / $ $ $ $ | $ $ | $ $
| $ $ _ $ $ | $ $ $ $ $ $ | $ $ | $ $ | $ $ $ $ | $ $ | $ $ | $ $
| $ $ \ $ $ | $ $ \ $ $ | $ $ | $ $ | $ $ \ $ | $ $ | $ $ | $ $
| $ $ $ $ $ $ / | $ $ \ $ $ | $ $ $ $ $ $ / | $ $ \ / | $ $ | $ $ $ $ $ /
| _ _ _ _ _ / | _ / _ \ / _ _ _ _ _ / | _ / | _ / _ \ _ _ _ _ _ /
=====
MASUKKAN COMMAND: LOAD config1.txt
Save file berhasil dibaca.
```

- b. Menggunakan command LISTGAME untuk melihat daftar game yang tersedia

```
MASUKKAN COMMAND: LISTGAME
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. PRS
```

- c. Menggunakan command QUEUEGAME untuk melihat daftar antrian game pengguna

```
MASUKKAN COMMAND: QUEUEGAME
Berikut adalah daftar antrian game-mu

Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. PRS
```

- d. Memasukkan game yang ingin ditambahkan ke antrian game pengguna dengan command QUEUEGAME

```
Nomor game yang mau ditambahkan ke antrian: 1
Game berhasil ditambahkan kedalam daftar antrian.
```

- e. Menggunakan command PLAYGAME untuk memainkan game yang ada pada antrian

```
MASUKKAN COMMAND: PLAYGAME
Berikut adalah daftar game yang tersedia
1. RNG

Loading RNG ...
RNG Telah dimulai. Uji keberuntungan Anda dengan menebak X.
5
Lebih Besar
50
Lebih Besar
68
Lebih Kecil
58
Lebih Kecil

X adalah 55.
```

- f. Game pada antrian (RNG) berhasil dimainkan sesuai dengan spesifikasi game  
g. Menggunakan command QUIT untuk keluar dari program setelah selesai memainkan game

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

### 6.3 Data Test 3

Pada data test ketiga, dilakukan pengecekan untuk fitur melihat bantuan, menghapus game, melihat daftar game, kemudian menyimpan perubahan tersebut. Hasil yang diharapkan adalah berhasilnya proses yang telah disebutkan sebelumnya yang ditunjukkan dari berjalannya command-command tersebut serta berhasilnya penghapusan dan penyimpanan yang dapat dilihat dari file konfigurasi setelah proses tersebut dijalankan.

- a. Berikut merupakan tampilan file default\_config.txt sebelum proses dilakukan.

```

data > default_config.txt

1  You, 1 second ago | 1 author (You)
2  6
3  RNG
4  Diner DASH
5  DINOSAUR IN EARTH
6  RISEWOMAN
7  EIFFEL TOWER
8  NEWGAME
9  . You, 1 second ago • Und

```

- b. Menggunakan command START untuk memulai program.

```

$ src/main.exe
=====SELAMAT DATANG DI=====
/##### /## /## /##### /## /## /#####
| ##_ ##| $$$ |## /##_ ##| $$$ /## /##_ ##
| ## \ ##| $$$| ## \ ##| $$$ /#####| ## \ ##
| #####| ## $ $ $| ## |##| ## $ $/## $| ## |##
| ##_ ##| ## $$$$| ## |##| ## $$$| ##| ## |##
| ## \ ##| ## \ $$$| ## |##| ## \ $| ##| ## |##
| #####/| ## \ ##| #####/| ## \/| ##| #####/
|_____/|_/ \_/ \_____/|_/|_/ \_____/
=====
MASUKKAN COMMAND: START
Save file berhasil dibaca.

```

- c. Menggunakan command HELP untuk melihat bantuan. Bantuan berhasil ditampilkan.

```

MASUKKAN COMMAND: a
Command tidak dikenali, silahkan masukkan command yang valid.
Silahkan ketik "HELP" untuk melihat bantuan.
MASUKKAN COMMAND: HELP
=====NINUNINUNINU=====
=====HELP=====
=====NINUNINUNINU=====
1. START - membaca file konfigurasi sistem yang berisi list game yang dapat dimainkan
2. LOAD - membaca save file yang berisi list game yang dapat dimainkan, histori, dan scoreboard game
3. SAVE - menyimpan kondisi state game pemain terkini
4. CREATEGAME - menambahkan game baru pada daftar game
5. LISTGAME - menampilkan daftar game yang ada pada sistem
6. DELETEGAME - menghapus sebuah game dari daftar game
7. QUEUEGAME - mendaftarkan game ke dalam list
8. PLAYGAME - memainkan sebuah permainan
9. SKIPGAME - melewati permainan sebanyak n
10. QUIT - keluar dari game
11. HELP - berisi bantuan atas command-command yang ada
12. COMMAND LAIN - input command tidak valid atau tidak terdapat

```

- d. Menggunakan command DELETEGAME untuk menghapus game. Ditampilkan daftar game yang tersedia.



```
MASUKKAN COMMAND: DELETEDGAME
Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. NEWGAME
```

- e. Dengan command DELETEDGAME, memilih nomor game yang akan dihapus.

```
MASUKKAN COMMAND: DELETEDGAME
Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. NEWGAME

Masukkan nomor game yang akan dihapus: 6
Game berhasil dihapus
```

- f. Menggunakan command LISTGAME untuk melihat daftar game. Dapat dilihat bahwa game berhasil dihapus karena tidak ditampilkan lagi pada daftar game yang tersedia.

```
MASUKKAN COMMAND: LISTGAME
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
```

- g. Menggunakan command SAVE untuk menyimpan perubahan dan command QUIT untuk keluar dari program.

```
MASUKKAN COMMAND: SAVE default_config.txt
Save file berhasil disimpan.
MASUKKAN COMMAND: QUIT
Anda keluar dari game BNMO.
Bye bye...
```

- h. Berikut merupakan tampilan file default\_config.txt setelah dijalankannya proses tersebut. Dapat disimpulkan bahwa proses berhasil dijalankan.

```
data > default_config.txt

      You, 4 seconds ago | 1 author (You)
1      5
2      RNG
3      Diner DASH
4      DINOSAUR IN EARTH
5      RISEWOMAN      You, last week •
6      EIFFEL TOWER
7      .
```

## 6.4 Data Test 4

Pada data test keempat, dilakukan pengecekan untuk fitur menambahkan antrian game, melewati antrian game, memainkan game. Hasil yang diharapkan adalah berhasilnya proses yang telah disebutkan sebelumnya yang ditunjukkan dari daftar antrian yang sesuai penambahan, game berhasil dilewati sesuai masukan pengguna, serta berhasil dimainkannya game.

- a. Menggunakan command LOAD <namafile>

```
$ src/main.exe
=====SELAMAT DATANG DI=====
/##### /## /## /##### /## /## /#####
| ##__ ##| ### | ## /##__ ##| ### /## /##__ ##
| ## \ ##| ####| ##| ## \ ##| #### /####| ## \ ##
| #####| ## $ $| ## | ##| ## $/$$| ## | ##
| ##__ ##| ## ####| ## | ##| ## $$| ##| ## | ##
| ## \ ##| ##\ $$$| ## | ##| ##\ $ | ##| ## | ##
| #####/| ## \ ##| #####/| ## \ | ##| #####/
|_____/|_|/ \_/ \_____/|_|/ \_/ \_____/
=====
MASUKKAN COMMAND: LOAD config2.txt
Save file berhasil dibaca.
```

- b. Menggunakan command QUEUEGAME dan menambahkan game ke dalam antrian sebanyak 4 kali. Dapat dilihat bahwa antrian berhasil ditambahkan

```

MASUKKAN COMMAND: QUEUEGAME
Berikut adalah daftar antrian game-mu

Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. COD

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

Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. COD

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

```

```

MASUKKAN COMMAND: QUEUEGAME
Berikut adalah daftar antrian game-mu
1. RNG
2. Diner DASH

Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. COD

Nomor game yang mau ditambahkan ke antrian: 1
Game berhasil ditambahkan kedalam daftar antrian.
MASUKKAN COMMAND: QUEUEGAME
Berikut adalah daftar antrian game-mu
1. RNG
2. Diner DASH
3. RNG

Berikut adalah daftar game yang tersedia
1. RNG
2. Diner DASH
3. DINOSAUR IN EARTH
4. RISEWOMAN
5. EIFFEL TOWER
6. COD

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

```

- c. Menggunakan command SKIPGAME <n> untuk melewati antrian game sebanyak n

```
MASUKKAN COMMAND: SKIPGAME 3
Berikut adalah daftar Game-mu
1. RNG
2. Diner DASH
3. RNG
4. Diner DASH

Loading Diner DASH ...
Selamat Datang di Diner Dash!

SALDO: 0
Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
M1      | 4              | 2        | 12000
M2      | 4              | 2        | 15000
M3      | 3              | 1        | 22000

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

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

MASUKKAN COMMAND: █
```

- d. Karena antrian terlewati sebanyak 3 kali, maka game yang akan dimainkan adalah game dengan urutan ke-4 dalam antrian, yaitu game Diner Dash

```

MASUKKAN COMMAND: COOK M6

Berhasil memasak M6
=====

SALDO: 21000
Daftar Pesanan
Makanan | Durasi memasak | Ketahanan | Harga
M2      | 5              | 3         | 23000
M3      | 4              | 1         | 21000
M4      | 5              | 1         | 26000
M5      | 4              | 5         | 48000
M6      | 3              | 1         | 45000
M7      | 3              | 3         | 28000
M8      | 5              | 5         | 12000
M9      | 3              | 2         | 11000

Daftar Makanan yang sedang dimasak
Makanan | Sisa durasi memasak
-----
M3      | 1
M5      | 1
M7      | 2
M6      | 3

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

MASUKKAN COMMAND: SKIP

Putaran ini dilewati
Makanan M3 telah selesai dimasak
=====
Jumlah pelanggan yang dilayani : 1
Saldo total : 21000

```

- e. Menggunakan command QUIT untuk keluar dari program setelah game selesai dimainkan

```

MASUKKAN COMMAND: QUIT
Anda keluar dari game BNMO.
Bye bye...

```

## 7 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	START, CREATEG AME, SAVE, QUIT	Memastikan prosedur START bisa memuat isi dari <i>file</i> konfigurasi, prosedur CREATEG AME dapat membuat dan	<ol style="list-style-type: none"> <li>1. Periksa file default_config.txt dan catat isinya</li> <li>2. Jalankan program dengan ./main</li> <li>3. Ketik perintah "START" di terminal</li> <li>4. Ketik perintah "CREATEGAME" di terminal</li> </ol>	Data Test 1	Nama gim yang ditambahkan melalui perintah CREATEG ME yakni NEWGAME terdapat di <i>file</i> default_config.txt	Sesuai ekspektasi

		menambahkan gim ke list, prosedur SAVE dapat menyimpan perubahan yang terjadi pada list ke <i>file</i> default_config.txt	<ol style="list-style-type: none"> <li>Ketik nama gim yang ingin dibuat yakni NEWGAME</li> <li>Ketik perintah "SAVE default_config.txt"</li> <li>Ketik perintah "QUIT"</li> <li>Periksa <i>file</i> default_config.txt, bandingkan hasilnya dengan sebelum tersting, pastikan nama gim baru sudah ada</li> </ol>			
2	LOAD, LISTGAME, QUEUEGAME, PLAYGAME, QUIT	Memastikan program bisa <i>load</i> save file, menampilkan daftar game dari file, menampilkan dan menambahkan gim ke dalam antrian, memainkan gim	<ol style="list-style-type: none"> <li>Jalankan program dengan ./main</li> <li>Ketik perintah "LOAD config1.txt"</li> <li>Ketik perintah "LISTGAME" untuk melihat nama gim yang tersedia</li> <li>Ketik perintah "QUEUEGAME" untuk melihat daftar gim yang terdapat di antrian</li> <li>Ketik nomor gim yang ingin ditambahkan ke antrian berdasarkan urutan dari game yang tersedia yakni 1</li> <li>Ketik perintah "PLAYGAME" untuk memainkan gim pertama di antrian (RNG)</li> <li>Ketik perintah "QUIT"</li> </ol>	Data Test 2	Program memuat daftar gim dari <i>file</i> config1.txt, menuliskan daftar gim, menuliskan antrian gim, menambahkan gim ke antrian yakni gim RNG, dan memainkan gim sesuai urutan antrian (RNG)	Sesuai ekspektasi
3	START, HELP, DELETEGAME,	Memastikan prosedur HELP dapat menampilkan	<ol style="list-style-type: none"> <li>Periksa <i>file</i> default_config.txt dan catat isinya</li> </ol>	Data Test 3	Nama gim yang dihapus (NEWGAME) tidak ada di	Sesuai ekspektasi

	LISTGAME, SAVE, QUIT	n daftar perintah yang ada dan memastikan prosedur DELETEGAME dapat menghapus gim dari list dan hasilnya dapat disimpan oleh prosedur SAVE	<ol style="list-style-type: none"> <li>2. Jalankan program dengan ./main</li> <li>3. Ketik perintah "START" di terminal</li> <li>4. Ketik perintah "HELP" di terminal dan perhatikan hasil yang dituliskan program</li> <li>5. Ketik perintah "DELETEGAME" di terminal</li> <li>6. Ketik nomor gim yang hendak dihapus dari daftar gim yang tersedia yakni 6</li> <li>7. Ketik perintah "LISTGAME" di terminal untuk melihat kondisi terkini dari daftar gim yang tersedia</li> <li>8. Ketik "SAVE default_config.txt" pada terminal</li> <li>9. Ketik "QUIT" pada terminal</li> <li>10. Buka <i>file</i> default_config.txt, bandingkan dengan sebelum <i>testing</i>, pastikan nama gim yang hendak dihapus sudah tidak tercantum lagi (NEWGAME)</li> </ol>		list gim tersedia serta terhapus dari <i>file</i> default_config.txt	
4	LOAD, QUEUEGAME, SKIPGAME, QUIT	Memastikan prosedur LOAD dapat memuat list gim tersedia, prosedur QUEUEGAME dapat menambahkan gim ke antrean,	<ol style="list-style-type: none"> <li>1. Jalankan program dengan ./main</li> <li>2. Ketik perintah "LOAD config.2" di terminal</li> <li>3. Ketik perintah "QUEUEGAME" di terminal</li> <li>4. Ketik nomor gim yang ingin ditambahkan ke antrean</li> </ol>	Data Test 4	Antrean gim bertambah sesuai masukan dan dapat memainkan gim yang seharusnya setelah melakukan perintah SKIPGAME 3	Sesuai ekspektasi

		prosedur SKIPGAME dapat melewati gim sejumlah yang diminta pengguna	5. Ulangi langkah 3 dan 4 sebanyak empat kali dengan nomor gim selang-seling antara 1 atau 2 6. Mainkan gim Diner Dash hingga selesai 7. Ketik perintah "QUIT" pada terminal		yakni Diner Dash	
--	--	---	--	--	------------------	--

## 8 Pembagian Kerja dalam Kelompok

No.	Fitur/ADT	NIM Coder	NIM Tester	Keterangan
1	ADT Mesin Karakter	18221124	18221124	
2	ADT Mesin Kata	18221124, 18221130	18221124	18221130 hanya menambah fungsi wordToInt dan isIdentical
3	ADT Queue Game	18221130	18221124	
4	ADT Array Game	18221124, 18221130	18221124, 18221130	18221130 hanya menambahkan fungsi IsMemberArr
5	ADT Map	18221152	18221152	
6	ADT QueueDD	18221152	18221152	
7	START	18221130	18221124	
8	LOAD	18221124	18221124	
9	SAVE	18221124	18221124	
10	CREATE GAME	18221090	18221124, 18221090	
11	LIST GAME	18221126	18221124	
12	DELETE GAME	18221130	18221124	



13	QUEUE GAME	18221130	18221124,	
14	PLAY GAME	18221090	18221124, 18221090	
15	SKIP GAME	18221090	18221124, 18221090	
16	QUIT	18221126	18221126, 18221124,	18221126 mengetes pada saat program belum disatukan atau saat input masih menggunakan scanf, 18221124 mengetes secara keseluruhan
17	HELP	18221126	18221124	
18	COMMAND LAIN	18221126	18221124	
19	GAME RNG	18221152	18221124, 18221152	
20	GAME DINER DASH	18221152	18221124, 18221152	
21	GAME PRS (BONUS)	18221130	18221130	
22	LAPORAN	18221090, 18221124, 18221126, 18221130, 18221152	-	

## 9 Lampiran

### 9.1 Deskripsi Tugas Besar 2

BNMO (dibaca: Binomo) 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.





## 9.2 Notulen Rapat



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

No. Kelompok/Kelas : 02/K2  
Nama Kelompok : kel 2 alstrukdat letsgoo  
Anggota Kelompok (Nama/NIM) :  
1. Marcheline Fanni Hidayat Putri/18221090  
2. Raditya Azka Prabaswara/18221152  
3. Jazmy Izzati Alamsyah/18221124  
4. Karunia Mega Lestari/18221126  
5. Rayhan Maheswara Pramanda/18221130






Asisten Pembimbing : Afif Fahreza


### Asistensi I

<b>Tanggal : 04/11/2022</b>	<b>Catatan Asistensi:</b>  1. Nanti bakal banyak file.c jadi bikin .shell/makefile 2. Untuk membuat ADT array of strings bagaimana? pakai array char of pointer/matrix 3. Stdlib harusnya boleh; yang perlu diperhatikan adalah tidak boleh menggunakan string.h 4. Di queue, karena gak boleh pake strcpy bisa pake ADT mesin kata (stringtokata) 5. Apabila menggunakan ADT tambahan, dijelaskan di laporan 6. Modifikasi ADT misalnya struct di Queue tidak dianggap ADT baru, ADT custom misalnya ADT Jam 7. Semua yang ada input, pakai ADT mesin kata 8. Beda start sama load -> pakai default config, baru nentuin konfigurasinya pakai file yang mana. make sure di folder data ada default confignya 9. Di asistensi 2 kalau bisa udah ada yang bisa didemoin (sedikit aja juga gapapa)
<b>Tempat : Google Meet</b>	
<b>Kehadiran Anggota Kelompok:</b> No NIM Tanda tangan  1   2   3   4 	

5 	
	<b>Tanda Tangan Asisten:</b>   (1)

### Asistensi II

<b>Tanggal : 11/11/2022</b>	<b>Catatan Asistensi:</b>  1. Kenapa pembacaan file sering gagal ketika run make shell? -> tinggal benerin directorynya 2. selanjutnya asisten memberitahu cara makefile 3. Kenapa kadang kala fungsi rng tidak mengembalikan nilai yang sama berturut-turut meskipun dipanggil dengan selang waktu yang kecil sekali? Jawaban: Karena fungsi rng mengambil seed dari clock, kadang kala clock speed bisa berubah-ubah sehingga menghasilkan nilai yang diluar perkiraan.
<b>Tempat : Labtek V</b>	
<b>Kehadiran Anggota Kelompok:</b> No NIM Tanda tangan  1   2   3   4   5 	

	<b>Tanda Tangan Asisten:</b>  

### 9.3 Log Activity Anggota Kelompok

Waktu	NIM	Keterangan
2021-10-31 11:00	18221090, 18221124 , 18221126, 18221130, 18221152	Rapat perdana penentuan pembagian tugas
2022-10-31 18.00	18221130	Pembuatan repository Git
2022-11-01	18221124	Menambahkan folder ADT, docs, dan src
2022-11-02	18221126	Membuat command QUIT dan COMMANDLAIN
2022-11-03	18221124	Menambahkan ADT mesin karakter dan mesin kata
2022-11-03	18221090, 18221124 , 18221126, 18221130, 18221152	Membuat dan merevisi command masing-masing selagi menunggu ADT
2022-11-04 09.00	18221090, 18221124 , 18221126, 18221130, 18221152	Asistensi pertama
2022-11-04	18221152	Membuat RNG dan test
2022-11-04	18221124	Membuat default_config.txt
2022-11-05	18221124	Membuat ADT word, revisi ADT mesin kata, membuat ADT arrayGame

2022-11-07	18221126	Merevisi command QUIT
2022-11-08	18221124	Merevisi ADT dan membuat command LOAD
2022-11-09	18221090, 18221126	Merancang dan membagi tugas laporan
2022-11-09	18221130	Membuat dan merevisi command DELETEDGAME, START dan QUIT. Merevisi command commandlain, merevisi ADT mesin kata, menambahkan README. Membuat driver array game, membuat dan merevisi ADT queueGame serta command QUEUEGAME.
2022-11-09	18221124	Memperbaiki ADT arrayGame, mesin karakter, mesin kata, dan program utama (main). Membuat command SAVE dan LOAD. Menambahkan spesifikasi tugas besar.
2022-11-10	18221126	Merevisi command QUIT, membuat command LISTGAME
2022-11-10	18221152	Membuat test, merevisi Diner Dash,
2022-11-10	18221090	Membuat dan merevisi command SKIPGAME, PLAYGAME, CREATEGAME, dan game tambahan
2022-11-10	18221130	Merevisi command QUEUEGAME, menambahkan dan merevisi fungsi IsMember pada ADT array dan ADT queue, menambahkan fungsi isIdentical pada ADT mesin kata, menambahkan library. Membuat game bonus PRS ( <i>paper, rock, scissor</i> )
2022-11-10	18221090, 18221124, 18221126, 18221130, 18221152	Memperbaiki dan memperbaiki fungsi-fungsi serta ADT-nya kemudian di-run bersamaan
2022-11-10	18221124	Membuat program utama (main), membuat makefile, <i>testing</i> program utama

2022-11-11 09.00	18221090, 18221124 , 18221126, 18221130, 18221152	Asistensi Kedua
2022-11-11	18221090 , 18221126, 18221130, 18221152 , 18221124	Melanjutkan laporan, merevisi fungsi-fungsi dan memfinalisasi program utama
2022-11-11	18221126, 18221090	Mengerjakan laporan
2022-11-11	18221124, 18221130, 18221152	Membuat driver