

LAPORAN TUGAS BESAR

IF2111 Algoritma dan Struktur Data

BNMO 2

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

	Sekolah Teknik Elektro dan Informatika ITB	Nomor Dokumen	Halaman
		<i>IF2111-TB2-2</i>	35
		Revisi	1

Daftar Isi

1	Ringkasan	2
2	Penjelasan Tambahan Spesifikasi Tugas	3
2.1	Penambahan fitur pada game Hangman	3
2.2	Penambahan fitur pada game Tower of Hanoi	3
2.3	Penambahan fitur pada game Snake on Meteor	3
3	Struktur Data (ADT)	4
3.1	ADT Map	4
3.2	ADT arrayMap	5
3.3	ADT matrixchar	6
3.4	ADT Point	7
3.5	ADT Stack	7
3.6	ADT StackGame	8
3.7	ADT ListDP	10
4	Program Utama	11
5	Algoritma-Algoritma Menarik	12
5.1	Algoritma Snake on Meteor	12
6	Data Test	13
6.1	Data Test 1	13
6.2	Data Test 2	15
6.3	Data Test 3	17
6.4	Data Test 4	19
6.5	Data Test 5	24
7	Test Script	27
8	Pembagian Kerja dalam Kelompok	29
9	Lampiran	30
9.1	Deskripsi Tugas Besar 2	30
9.2	Notulen Rapat	30
9.3	Log Activity Anggota Kelompok	33

1 Ringkasan

BNMO 2 merupakan lanjutan dari tugas besar BNMO dengan penambahan spek dan perubahan pada ADT serta implementasi fungsi yang dipakai pada BNMO. Pada BNMO 2 terdapat penambahan commands, yaitu scoreboard dan history dari game yang dimainkan beserta perintah untuk mereset commands tersebut. Selain itu, ada beberapa game yang ditambahkan sehingga menambah variasi dari game yang telah ada, seperti Hangman, Tower of Hanoi, dan Snake on Meteor. Tentunya dengan penambahan commands dan games ini memengaruhi hampir seluruh program BNMO yang telah dibuat sebelumnya.

Secara umum, laporan ini menjelaskan lebih detail mengenai spek terbaru dari BNMO serta ADT tambahan yang akan dibahas lebih lanjut pada bagian ketiga, yaitu Struktur Data (ADT). Ada pun, kami juga menjelaskan bagaimana program utama bekerja dan mengintegrasikan seluruh fungsi dan prosedur yang telah kami buat pada bagian keempat laporan ini. Tidak lupa kami cantumkan juga data test, test script pada bagian kelima dan keenam, serta catatan aktivitas kami ketika mengerjakan tugas besar BNMO 2 dalam bentuk lampiran.

Program BNMO 2 dibuat menggunakan bahasa C dengan memanfaatkan ADT yang telah dipelajari di kelas IF2111 Algoritma dan Struktur Data yang kemudian kami modifikasi dan disesuaikan sesuai kebutuhan, antara lain ADT arrayGame, ADT arrayMap, ADT map, ADT matrixchar, ADT mesinkarakter, ADT mesinkata, ADT queueGame, ADT set, ADT stack, ADT stackGame, serta ADT word.

2 Penjelasan Tambahan Spesifikasi Tugas

2.1 Penambahan fitur pada game Hangman

Permainan ini menambahkan beberapa fitur bonus, seperti menambahkan kata-kata ke dalam kamus atau list kata yang berada pada suatu file terpisah bernama kamus.txt. Pada awal permainan, program membaca list kata dari suatu file. Apabila permainan telah berakhir maka list kata disimpan kembali ke file dengan kemungkinan list kata akan bertambah.

2.2 Penambahan fitur pada game Tower of Hanoi

Pada game Tower of Hanoi, pemain dapat memilih berapa jumlah piringan yang akan dimainkan sebelum memainkan game. Score akhir akan ditentukan berdasarkan jumlah piringan tersebut. Semakin banyak jumlah piringan maka akan semakin tinggi juga skor yang didapatkan pemain.

2.3 Penambahan fitur pada game Snake on Meteor

Untuk permainan Snake on Meteor, kelompok kami menambahkan obstacle (rintangan). Di mana obstacle tersebut akan muncul pada awal permainan sehingga tidak dapat dilewati oleh snake. Makanan juga tidak dapat muncul pada titik yang memiliki obstacle. Apabila kepala snack mengenai obstacle, permainan akan berakhir. Selain itu, kami juga menambahkan fitur di mana kepala snake seakan-akan menembus apabila melewati batas peta dan akan muncul di sisi peta yang berseberangan.

3 Struktur Data (ADT)

3.1 ADT Map

ADT Map digunakan untuk menampilkan sebuah array dengan nilai yang berpasangan. Dalam hal ini adalah Nama dan Score, di mana nama merupakan keytype berupa Word dan Score merupakan value berupa int. Nama dan score didapatkan setiap kali permainan berakhir, kemudian akan di-insert ke dalam sebuah Map untuk kemudian diurutkan berdasarkan value dari terbesar ke terkecil. ADT Map diimplementasikan untuk fungsi scoreboard dan resetscoreboard, juga di-include pada header arrayMap.h

Modularitas ADT Map:



Sketsa struktur data (dalam bahasa C):

```
typedef struct {
    key Name;
    value Score;
} info;

typedef struct {
    info Elements[MaxEl];
    Saddress Count;
} scoreMap;
```

Berikut primitif dalam ADT Map:

1. Konstruktor:
void CreateEmptyMap (scoreMap *M);
2. Predikat:
boolean IsEmptyMap (scoreMap M);
boolean IsFullMap (scoreMap M);

3. Operator Dasar:

```

value ValueMap (scoreMap M, key k);
void sortMap (scoreMap *M);
void InsertMap (scoreMap *M, key k, value v);
void DeleteMap (scoreMap *M, key k);
boolean isMemberMap (scoreMap M, key k);
void PrintMap (scoreMap map);
int countMap (scoreMap M);

```

3.2 ADT arrayMap

ADT arrayMap digunakan untuk mengubah Map ke dalam bentuk array atau list. Kami memerlukan ADT ini untuk mengubah pasangan Nama-Score yang berupa Map ke dalam bentuk list atau array untuk ditampilkan dalam program scoreboard. Oleh karena itu, ADT ini juga dapat disebut sebagai ADT List of Map. ADT ini diimplementasikan dalam file arrayMap.h dan arrayMap.c

Modularitas ADT arrayMap:



Sketsa struktur data (dalam C):

```

typedef int Idx;
typedef struct {
    scoreMap A[100];
    int Neff;
} arrScore;

```

Berikut primitif dalam ADT arrayMap:

1. Konstruktor:
`void MakeArrayM (arrScore *arr);`
2. Predikat:
`boolean IsEmptyArrM (arrScore arr);`

STEI- ITB	IF2111_TB2_02_02	Halaman 5 dari 35 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.		

- ```

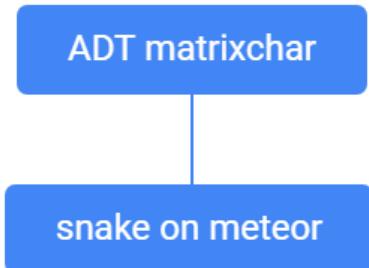
boolean IsMemberArrM (arrScore arr, scoreMap el);
3. Operasi:
int LengthArrM (arrScore arr);
int LengthArrM (arrScore arr);
scoreMap GetArrM (arrScore arr, Idx i);
void setNeffM (arrScore *arr, int n);
4. Penambahan dan Pengurangan:
void InsertAtM (arrScore *arr, scoreMap el, Idx i);
void InsertLastM (arrScore *arr, scoreMap el);
void InsertFirstM (arrScore *arr, scoreMap el);
void DeleteAtM (arrScore *arr, Idx i);
void DeleteLastM (arrScore *arr);

```

### **3.3 ADT matrixchar**

ADT matrixchar digunakan untuk menampilkan peta permainan snake yang berupa garis-garis matriks. ADT ini diimplementasikan pada program Snake on Meteor. ADT ini memanfaatkan ADT lain, yaitu ADT Point dan ADT Word. ADT ini diimplementasikan dalam file matrixchar.h dan matrixchar.c

Modularitas ADT matrixchar:



Sketsa struktur data (dalam C):

```

typedef struct
{
 Word info[20][20];
 int h;
 int w;
} matrixchar;

```

Berikut primitif dalam ADT matrixchar:

1. Operasi:

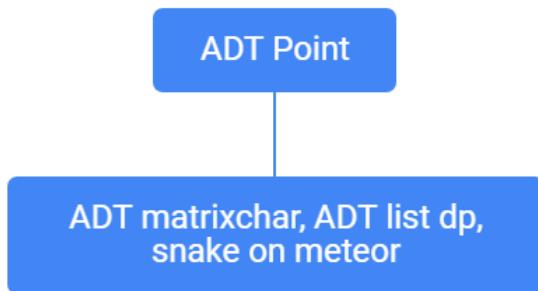
| STEI- ITB                                                                                                                                                                                                                             | IF2111_TB2_02_02 | Halaman 6 dari 35 halaman |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|---------------------------|
| 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. |                  |                           |

```
CreateEmptyMatrixChar (matrixchar *m, int h, int w);
InsertMatrixChar (matrixchar *m, Word c, point p);
void PrintMatrixChar (matrixchar m);
```

### 3.4 ADT Point

ADT Point digunakan untuk menyimpan koordinat di gim Snake on Meteor. Tipe data ini merupakan tipe data yang membentuk isi dari ListDP. ADT Point diimplementasikan juga pada ADT lain, yaitu ADT matrixchar. ADT ini diimplementasikan dalam file point.h

Modularitas ADT Point:



Sketsa struktur data (dalam C):

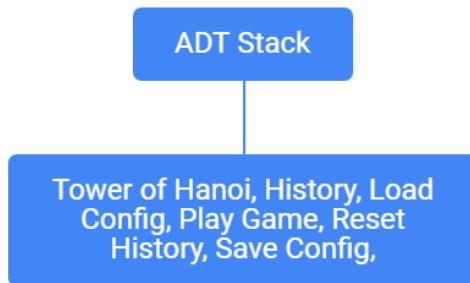
```
#define Xp(P) (P).x
#define Yp(P) (P).y

typedef struct
{
 int x;
 int y;
} point;
```

### 3.5 ADT Stack

ADT Stack digunakan untuk merepresentasikan perilaku piringan di gim Tower of Hanoi. ADT ini menghandle permasalahan bagaimana satu piring dapat berpindah ke tiang lain sesuai dengan urutannya (dari paling atas) dengan primitif utamanya, yaitu Push dan Pop. ADT ini diimplementasikan dalam file stack.h, stack.c

Modularitas ADT Stack:



Sketsa struktur data (dalam C):

```
typedef int stacktype;
typedef int address; /* indeks tabel */

/* Contoh deklarasi variabel bertipe stack dengan ciri TOP : */
/* Versi I : dengan menyimpan tabel dan alamat top secara eksplisit*/
typedef struct {
 stacktype T[SMaxEl]; /* tabel penyimpan elemen */
 address TOP; /* alamat TOP: elemen puncak */
} Stack;
```

Berikut primitif dalam ADT Stack:

1. Konstruktor  
void CreateEmptyS (Stack \*S);
2. Predikat  
boolean IsEmptyS (Stack S);  
boolean IsFullS (Stack S);
3. Penambahan dan Penghapusan  
void Push (Stack \*S, stacktype X);  
void Pop (Stack \*S, stacktype \*X);
4. Operator Tambahan  
void CopyStack (Stack Sin, Stack\* Sout);  
int LengthS (Stack S);  
void printStack (Stack S);

### 3.6 ADT StackGame

ADT StackGame digunakan untuk menjalankan HISTORY dan RESETHISTORY. ADT ini juga digunakan pada program utama untuk membuat *stack* berisi *games* lalu ditambahkan elemennya setiap PLAY GAME dijalankan. ADT ini digunakan pada HISTORY untuk menampilkan game-game yang telah dimainkan dengan urutan teratas merupakan permainan terakhir yang dimainkan. ADT ini juga digunakan pada RESET HISTORY untuk menghapus semua history

permainan yang dimainkan dengan primitif PopSG hingga kosong. ADT ini diimplementasikan dalam file stackGame.h, stackGame.c, dan driver\_stackGame.c

Modularitas ADT StackGame:



Sketsa struktur data (dalam C):

```
typedef Word Game;
typedef int address; /* indeks tabel */

/* Contoh deklarasi variabel bertipe stack dengan ciri TOP : */
/* Versi I : dengan menyimpan tabel dan alamat top secara eksplisit*/
typedef struct {
 Game T[MaxSG]; /* tabel penyimpan elemen */
 address TOP; /* alamat TOP: elemen puncak */
} stackGame;
```

Berikut primitif dalam ADT StackGame:

1. Konstruktor  
void CreateEmptySG (stackGame \*S);
2. Predikat  
boolean IsEmptySG (stackGame S);  
boolean IsFullSG (stackGame S);
3. Penambahan dan Penghapusan  
void PushSG (stackGame \*S, Game game);  
void PopSG (stackGame \*S, Game \*game);
4. Operator Tambahan  
void CopyStackSG (stackGame Sin, stackGame \*Sout);  
int LengthSG (stackGame S);  
void PrintAllSG (stackGame S);  
void PrintSomeSG (stackGame S, int n);  
void reverseSG (stackGame \*S);

### 3.7 ADT ListDP

ADT ListDP digunakan untuk menyimpan koordinat-koordinat kumpulan objek pada gim *Snake on Meteor*. ADT ini diimplementasikan pada program *Snake on Meteor*. ADT ini memanfaatkan ADT lain, yaitu ADT Point. ADT ini diimplementasikan dalam file listdp.h dan listdp.c

Modularitas ListDP:



Sketsa struktur data (dalam C):

```
typedef struct tElmtListdp *addressldp;
typedef struct tElmtListdp {
 point info;
 addressldp next;
 addressldp prev;
} ElmtListdp;
typedef struct {
 addressldp First;
 addressldp Last;
} Listdp;
```

Berikut primitif dalam ADT Map:

1. Makro:

```
#define X(P) (P)->info.x
#define Y(P) (P)->info.y
#define Next(P) (P)->next
#define Prev(P) (P)->prev
#define First(L) ((L).First)
#define Last(L) ((L).Last)
```

2. Konstruktor:

```
void CreateEmptyLDP (Listdp *L);
```

3. Predikat:

|                                                                                                                                                                                                                                       |                  |                            |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|----------------------------|
| STEI- ITB                                                                                                                                                                                                                             | IF2111_TB2_02_02 | Halaman 10 dari 35 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. |                  |                            |

- ```

boolean IsEmptyLDP (Listdp L);
4. Manajemen Memori:
addressldp AlokasiLDP (point N);
void DealokasiLDP (addressldp P);
5. Penambahan dan penghapusan elemen:
void InsVFirstLDP (Listdp *L, point N);
void InsVLastLDP (Listdp *L, point N);
void InsertFirstLDP (Listdp *L, addressldp P);
void InsertLastLDP (Listdp *L, addressldp P);
void InsertAfterLDP (Listdp *L, addressldp P, addressldp Prec);
void InsertBeforeLDP (Listdp *L, addressldp P, addressldp Succ);
void DelVFirstLDP (Listdp *L, point *N);

void DelVLastLDP (Listdp *L, point *N);
void DelFirstLDP (Listdp *L, addressldp *P);
void DelLastLDP (Listdp *L, addressldp *P);
void DelPLDP (Listdp *L, point X);
void DelAfterLDP (Listdp *L, addressldp *Pdel, addressldp Prec);
void DelBeforeLDP (Listdp *L, addressldp *Pdel, addressldp Succ);
6. Pencarian:
addressldp SearchLDP (Listdp L, point N);
7. Pencetak:
void PrintForwardLDP (Listdp L);
void PrintBackwardLDP (Listdp L);

```

4 Program Utama

Pada program utama, program akan memasukkan ADT dan *command* yang terdapat pada file lainnya. Program akan mendeklarasi dan membuat *array* dari *game*, *queue* dari *game*, *array* dari *score*, serta *stack* 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 “CREATE GAME”, 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 dan menambahkan pada *array score*. Jika pengguna memasukkan kata “LIST GAME”, program akan menjalankan command LISTGAME yang menampilkan daftar game yang disediakan oleh sistem. Jika pengguna memasukkan kata “DELETE GAME”, program akan menjalankan command DELETEGAME yang akan menampilkan daftar game yang tersedia, menerima masukan dari pengguna berupa nomor game yang akan dihapus, kemudian menghapus game tersebut serta meng-update *array score*-nya. Jika masukan pengguna berupa kata “QUEUE GAME”, 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 “PLAY GAME”, maka program akan menjalankan command PLAYGAME yang berfungsi untuk memainkan sebuah permainan dengan urutan pertama di antrian game. *Stack game* dan *array score* akan ter-update ketika *game* selesai dimainkan. Jika pengguna memasukkan kata “SKIP GAME”, program akan menerima masukan lagi dengan mesin kata berupa banyaknya permainan yang akan diskip (n) kemudian program akan menjalankan command SKIPGAME yang berfungsi untuk melewaskan permainan sebanyak n.

Jika masukan pengguna adalah “SCOREBOARD”, maka program akan menjalankan command SCOREBOARD yang menampilkan nama serta skor yang diperoleh dari masing-masing game. Apabila pengguna memasukkan “RESET SCOREBOARD”, maka program akan menjalankan command RESET SCOREBOARD dan program akan melakukan *reset* terhadap scoreboard permainan. Reset dapat dilakukan untuk menghapus salah satu atau semua informasi pada permainan. Jika masukan pengguna adalah “HISTORY <n>”, maka program akan menampilkan sejumlah n *game* yang telah dimainkan dengan *game* teratas merupakan *game* yang terakhir dimainkan. Apabila pengguna memasukkan “RESET HISTORY”, maka program akan menghapus semua history permainan yang dimainkan.

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 Snake on Meteor

Pada *game snake on meteor*, ketika dilakukan pengecekan apakah suatu titik terisi atau tidak. Algoritma tersebut dilakukan dengan membuat suatu titik semu kemudian dibandingkan dengan

STEI- ITB	IF2111_TB2_02_02	Halaman 12 dari 35 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.		

semua objek yang ada, apakah ada yang memiliki koordinat yang sama. Pergerakan *snake* dilakukan dengan cara menemukan suatu titik semu yang valid kemudian dilakukan InsertFirst ke *snake* kemudian DeleteLast pada *snake*.

6 Data Test

Program BNMO 2 dapat menjalankan permainan dengan 16 command dan main menu sehingga dapat menjalankan fitur-fiturnya. Pada BNMO 2, fitur utamanya adalah memainkan game, menambahkan game, menghapus game, mengurutkan game yang akan dimainkan, menampilkan game yang telah dimainkan, serta menampilkan scoreboard game. 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 game Hangman. Hasil yang diharapkan adalah berhasilnya proses START program dan berhasil dimainkannya PLAY HANGMAN beserta QUEUE HANGMAN.

Langkah-langkah:

- START program, QUEUE HANGMAN, dan PLAY HANGMAN

```
$ ./src/main.exe
=====
SELAMAT DATANG DI=====
/$$$$$$$/ $$ /$$ /$$$$$$$/ $$ /$$ /$$$$$$$
| $$ _ $$| $$ | $$ /$$ _ $$| $$ /$$ /$$ _ $$
| $$ \ $$| $$| $$ \ $$| $$ \ $$ /$$$$| $$ \ $$
| $$$$/ $$| $$| $$| $$| $$| $$| $$| $$| $$| $$| $$
| $$ _ $$| $$| $$| $$| $$| $$| $$| $$| $$| $$| $$
| $$ \ $$| $$| $$| $$| $$| $$| $$| $$| $$| $$| $$
| $$ \ $$| $$| $$| $$| $$| $$| $$| $$| $$| $$| $$
| $$$$/| $$| $$| $$| $$| $$| $$| $$| $$| $$| $$
| ____/_/ \_/_/ \_/_/ | ____/_/ \_/_/ | ____/_/ \_/_/
=====
MASUKKAN COMMAND: START
Save file berhasil dibaca.
MASUKKAN COMMAND: QUEUE GAME
Berikut adalah daftar antrian game-mu

Berikut adalah daftar game yang tersedia
1. RNG
2. DINER DASH
3. HANGMAN
4. TOWER OF HANOI
5. SNAKE ON METEOR
6. RISEWOMAN
7. PRS

Nomor game yang mau ditambahkan ke antrian: 3
Game berhasil ditambahkan kedalam daftar antrian.
MASUKKAN COMMAND: PLAY GAME
Berikut adalah daftar Game-mu
1. HANGMAN

Loading HANGMAN ...
```

- Pilih MAIN untuk bermain

WELCOME TO HANGMAN

Aturan:

1. Anda punya 10 kesempatan
2. Jawaban dalam huruf kapital
3. Jika ingin berhenti bermain, ketik EXIT

Menu:

1. MAIN
2. TAMBAHKATA
3. EXIT

Pilihan Anda: MAIN

c. Memainkan game Hangman

Tebakan sebelumnya: A B W U

Kata: _____

Kesempatan: 6

Masukkan tebakan: I

Tebakan sebelumnya: A B W U I

Kata: _____

Kesempatan: 5

Masukkan tebakan: P

Tebakan sebelumnya: A B W U I P

Kata: P_____

Kesempatan: 5

Masukkan tebakan: Y

Tebakan sebelumnya: A B W U I P Y

Kata: PY_____

Kesempatan: 5

Masukkan tebakan: T

Tebakan sebelumnya: A B W U I P Y T

Kata: PYT____

Kesempatan: 5

Masukkan tebakan: H

Tebakan sebelumnya: A B W U I P Y T H

Kata: PYTH____

Kesempatan: 5

Masukkan tebakan: O

Tebakan sebelumnya: A B W U I P Y T H O

Kata: PYTHO_

Kesempatan: 5

Masukkan tebakan: N

Kata: PYTHON

Berhasil menebak kata PYTHON! Kamu mendapatkan 6 poin!

Tebakan sebelumnya: -

Kata: _____

Kesempatan: 5

Masukkan tebakan: A

Tebakan sebelumnya: A

Kata: _____

Kesempatan: 4

Masukkan tebakan: J

Tebakan sebelumnya: A J

Kata: _____

Kesempatan: 3

Masukkan tebakan: K

Tebakan sebelumnya: A J K

Kata: _____

Kesempatan: 2

Masukkan tebakan: L

Tebakan sebelumnya: A J K L

Kata: ___L

Kesempatan: 2

Masukkan tebakan: T

Tebakan sebelumnya: A J K L T

Kata: _T_L

Kesempatan: 2

Masukkan tebakan: O

Tebakan sebelumnya: A J K L T O

Kata: _T_L

Kesempatan: 1

Masukkan tebakan: Q

d. Jika kesempatan habis, game over

Game over!
SCORE: 6
Masukkan Nama:Jazmy
Score berhasil ditambahkan!

- e. Masukkan nama (nama harus unik) dan program akan memasukkan score ke Scoreboard
- f. Jika ingin menambahkan Kata pada game, pilih menu TAMBAHKATA

```
WELCOME TO HANGMAN

Aturan:
1. Anda punya 10 kesempatan
2. Jawaban dalam huruf kapital
3. Jika ingin berhenti bermain, ketik EXIT

Menu:
1. MAIN
2. TAMBAHKATA
3. EXIT

Pilihan Anda: TAMBAHKATA

Jika ingin berhenti menambahkan kata, ketik 'STOP'
Masukkan kata yang ingin ditambahkan: KATABARU
Kata berhasil ditambahkan!
Masukkan kata yang ingin ditambahkan: STOP
```

6.2 Data Test 2

Pada data test kedua, dilakukan pengecekan untuk fitur game Tower of Hanoi. Hasil yang diharapkan adalah berhasilnya proses START program dan berhasil dimainkannya PLAY Tower of Hanoi.

Langkah-langkah:

- a. Mulai program
- b. Ketik START
- c. Ketik QUEUE GAME dan masukan angka 4

- d. Masukan jumlah piringan yang diinginkan
 - e. Mainkan gim Tower of Hanoi

- f. Setelah selesai, masukkan nama yang unik, kemudian program akan memasukkan *score* ke *scoreboard*

```

=====
|   *   |
|*****| ***
|       |
=====
A   B   C
=====
TIANG ASAL : A
TIANG TUJUAN: C
=====
|   *   |
|***| *****
|       |
=====
A   B   C
=====
TIANG ASAL : B
TIANG TUJUAN: A
=====
|   *   |
|***| *****
|       |
=====
A   B   C
=====
TIANG ASAL : B
TIANG TUJUAN: C
=====
|   |   |
|   |   ***|
|   |   *****
|       |
=====
A   B   C
=====
TIANG ASAL : A
TIANG TUJUAN: C
=====
|   |   *
|   |   ***
|   |   *****
|       |
=====
A   B   C
=====
SCORE : 10
Masukkan Nama:Jazmy
Score berhasil ditambahkan!
=====
MASUKKAN COMMAND: 

```

6.3 Data Test 3

Pada *data test* ketiga, dilakukan pengecekan untuk fitur *game* Snake on Meteor. Hasil yang diharapkan adalah berhasilnya dimainkan game tersebut dan berjalannya setiap *command* untuk memainkan *game* tersebut.

- START program, QUEUE SNAKE ON METEOR, dan PLAY SNAKE ON METEOR

b. Mainkan game

```

Berhasil bergerak!
Anda Terkena Meteor

+---+---+---+ X |
|   |   |   | |
|   |   |   |
| o | 1 | H |
|   |   |   |
| m | 2 |   |
|   |   |   |
|   |   |   |
|   |   | X | X |
+---+---+---+---+
Silaikan masukkan command anda: s

Berhasil bergerak!

+---+---+---+ X |
|   |   |   | |
|   |   |   |
| o | 2 | 1 |
|   |   |   |
| m |   |   |
|   |   |   |
|   |   | X | X |
+---+---+---+---+
GAME OVER
SCORE : 4
Masukkan Nama:Jazmy
Score berhasil ditambahkan!

```

- c. Masukkan nama dan program akan menambahkan score ke Scoreboard

6.4 Data Test 4

Pada *data test* keempat, dilakukan pengecekan untuk fitur SCOREBOARD dan HISTORY. Hasil yang diharapkan adalah berhasil ditampilkannya SCOREBOARD serta HISTORY sesuai dengan spesifikasi yang diharapkan.

- a. START program, mainkan 1 game dan masukkan nama

```
$ ./src/main.exe
=====SELAMAT DATANG DI=====
/$$$$$$ $ /$$ /$$ /$$$$$$$ /$$ /$$ /$$$$$$
| $$__ $$| $$$ | $$ /$$__ $$| $$$ /$$ /$$__ $$
| $$ \ $$| $$$| $$ \ $$| $$$| $$$ /$$$$| $$ \ $$
| $$$$/ $$| $$ $$ $$| $$| $$| $$ $$/$$ $$| $$| $$| $$
| $$__ $$| $$ $$$$/| $$| $$| $$ $$$| $$| $$| $$| $$
| $$ \ $$| $$\ $$| $$| $$| $$\ $| $$| $$| $$| $$| $$
| $$$$/ $$| $$ \ $$| $$$$/| $$ \ $$| $$| $$$$/| $$| $$$/
|_____| /| /| \_|| /| /| /| \_|| /| /| \_|| /| /| \_||

=====
MASUKKAN COMMAND: START
Save file berhasil dibaca.
MASUKKAN COMMAND: QUEUE GAME
Berikut adalah daftar antrian game-mu

Berikut adalah daftar game yang tersedia
1. RNG
2. DINER DASH
3. HANGMAN
4. TOWER OF HANOI
5. SNAKE ON METEOR
6. RISEWOMAN
7. PRS

Nomor game yang mau ditambahkan ke antrian: 4
Game berhasil ditambahkan kedalam daftar antrian.
MASUKKAN COMMAND: PLAY GAME
Berikut adalah daftar Game-mu
1. TOWER OF HANOI
```

b. Mainkan game

- c. Masukkan nama dan program akan menambahkan score ke scoreboard

```

MASUKKAN COMMAND: SCOREBOARD
***SCOREBOARD RNG***
| NAMA | SKOR |
-----
| SCORE KOSONG |
-----

***SCOREBOARD DINER DASH***
| NAMA | SKOR |
-----
| SCORE KOSONG |
-----

***SCOREBOARD HANGMAN***
| NAMA | SKOR |
-----
| SCORE KOSONG |
-----

***SCOREBOARD TOWER OF HANOI***
| NAMA | SKOR |
-----
| jazmy | 3 |
-----


***SCOREBOARD SNAKE ON METEOR***
| NAMA | SKOR |
-----
| SCORE KOSONG |
-----


***SCOREBOARD RISEWOMAN***
| NAMA | SKOR |
-----
| SCORE KOSONG |
-----


***SCOREBOARD PRS***
| NAMA | SKOR |
-----
| SCORE KOSONG |
-----


MASUKKAN COMMAND: █

```

- d. Jalankan perintah SCOREBOARD untuk melihat scoreboard

```
MASUKKAN COMMAND: HISTORY
```

```
Berikut adalah daftar Game yang telah dimainkan
```

```
1. TOWER OF HANOI
```

```
MASUKKAN COMMAND: [ ]
```

- e. Jalankan perintah HISTORY untuk melihat riwayat game yang dimainkan

```
MASUKKAN COMMAND: HISTORY
```

```
Berikut adalah daftar Game yang telah dimainkan
```

```
1. TOWER OF HANOI
```

```
MASUKKAN COMMAND: [ ]
```

- f. Jika ingin mereset scoreboard, jalankan RESET SCOREBOARD

```
MASUKKAN COMMAND: RESET SCOREBOARD
```

```
DAFTAR SCOREBOARD:
```

```
0. ALL
```

```
1. RNG
```

```
2. DINER DASH
```

```
3. HANGMAN
```

```
4. TOWER OF HANOI
```

```
5. SNAKE ON METEOR
```

```
6. RISEWOMAN
```

```
7. PRS
```

```
SCOREBOARD YANG INGIN DIHAPUS:
```

```
4
```

```
APAKAH ANDA YAKIN INGIN MENGHAPUS SCOREBOARD TOWER OF HANOI? (YA/TIDAK)
```

```
YA
```

```
Scoreboard berhasil di-reset
```

```
MASUKKAN COMMAND: SCOREBOARD
```

```
***SCOREBOARD RNG***
```

NAMA	SKOR
SCORE KOSONG	

```
***SCOREBOARD DINER DASH***
```

NAMA	SKOR
SCORE KOSONG	

```
***SCOREBOARD HANGMAN***
```

NAMA	SKOR
SCORE KOSONG	

```
***SCOREBOARD TOWER OF HANOI***
```

NAMA	SKOR
SCORE KOSONG	

```
***SCOREBOARD SNAKE ON METEOR***
```

NAMA	SKOR
SCORE KOSONG	

```
***SCOREBOARD RISEWOMAN***
```

NAMA	SKOR
SCORE KOSONG	

```
***SCOREBOARD PRS***
```

NAMA	SKOR
SCORE KOSONG	

- g. Jika ingin mereset riwayat, jalankan RESET HISTORY

```

[RECORDS]
MASUKKAN COMMAND: RESET HISTORY
APAKAH KAMU YAKIN INGIN MELAKUKAN RESET HISTORY? (YA/TIDAK) YA
History Game Anda Kosong!
MASUKKAN COMMAND: HISTORY
Berikut adalah daftar Game yang telah dimainkan
History Game Anda KOSONG!

```

6.5 Data Test 5

Pada *data test* kelima, dilakukan pengecekan untuk fitur CREATE GAME dan DELETE GAME. Hasil yang diharapkan adalah berhasilnya proses pembuatan game dan penghapusan game sesuai dengan spek beserta dengan scoreboardnya.

- a. START program, CREATE game baru dan mainkan game tersebut

```

$ ./src/main.exe
=====SELAMAT DATANG DI=====
/$$$$$$/ $$ /$$ /$$$$$/ $$ /$$ /$$$$$/ 
| $$ _ $$| $$ | $$ /$$ _ $$| $$| $$$ /$$ _ $$ 
| $$ \ $$| $$| $$ \ $$| $$| $$$ /$$$$| $$ \ $$ 
|$$$$$$| $$| $$| $$| $$| $$| $$| $$| $$| $$| $$ 
| $$ _ $$| $$| $$| $$| $$| $$| $$| $$| $$| $$| $$ 
| $$ \ $$| $$| $$| $$| $$| $$| $$| $$| $$| $$| $$ 
| $$| $$\| $$| $$| $$| $$| $$\| $| $$| $$| $$| $$ 
|$$$$$$| /| $$| $$\| $$| $$$| /| $$| /| $$| $$$| /| 
|_____| /| /| \_\| /| /| /| /| /| /| \_\| /| /| \_\| /| 
=====

MASUKKAN COMMAND: START
Save file berhasil dibaca.
MASUKKAN COMMAND: CREATE GAME
Masukkan nama game yang akan ditambahkan: COD
Game berhasil ditambahkan
MASUKKAN COMMAND: QUEUE GAME
Berikut adalah daftar antrian game-mu

Berikut adalah daftar game yang tersedia
1. RNG
2. DINER DASH
3. HANGMAN
4. TOWER OF HANOI
5. SNAKE ON METEOR
6. RISEWOMAN
7. PRS
8. COD

Nomor game yang mau ditambahkan ke antrian: 8
Game berhasil ditambahkan kedalam daftar antrian.
MASUKKAN COMMAND: PLAY GAME
Berikut adalah daftar Game-mu
1. COD

Loading COD ...
GAME OVER!
Skor Akhir: 798
Masukkan Nama: Jazmy
Score berhasil ditambahkan!

```

- b. Jalankan perintah SCOREBOARD dan HISTORY

STEI- ITB	IF2111_TB2_02_02	Halaman 24 dari 35 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.		

```

MASUKKAN COMMAND: SCOREBOARD
***SCOREBOARD RNG***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD DINER DASH***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD HANGMAN***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD TOWER OF HANOI***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD SNAKE ON METEOR***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD RISEWOMAN***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD PRS***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD COD***
| NAMA | SKOR |
-----
| jazmy | 798 |

MASUKKAN COMMAND: HISTORY
Berikut adalah daftar Game yang telah dimainkan
1. COD

```

- c. Jalankan perintah SAVE untuk menyimpan keadaan program di sebuah file

```
MASUKKAN COMMAND: SAVE create.txt
Save file berhasil disimpan.

data / create.txt
1 8
2 RNG
3 DINER DASH
4 HANGMAN
5 TOWER OF HANOI
6 SNAKE ON METEOR
7 RISEWOMAN
8 PRS
9 COD
10 1
11 COD
12 0
13 0
14 0
15 0
16 0
17 0
18 0
19 1
20 jazmy 798
21 .
```

- d. Jalankan DELETE GAME untuk menghapus game, lalu cek scoreboardnya

```
MASUKKAN COMMAND: DELETE GAME
Berikut adalah daftar game yang tersedia
1. RNG
2. DINER DASH
3. HANGMAN
4. TOWER OF HANOI
5. SNAKE ON METEOR
6. RISEWOMAN
7. PRS
8. COD

Masukkan nomor game yang akan dihapus: 8
Game berhasil dihapus
```

```

MASUKKAN COMMAND: SCOREBOARD
***SCOREBOARD RNG***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD DINER DASH***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD HANGMAN***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD TOWER OF HANOI***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD SNAKE ON METEOR***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD RISEWOMAN***
| NAMA | SKOR |
-----
| SCORE KOSONG |

***SCOREBOARD PRS***
| NAMA | SKOR |
-----
| SCORE KOSONG |

```

- e. Game COD berhasil dihapus dari program

7 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	HANGMAN	Menguji gim	Mulai program Ketik START	Data Test 1	Gim hangman berjalan sesuai	Sesuai ekspektasi

		hangman dan fitur SCOREBOARD	Ketik QUEUE GAME dan masukkan angka 3 Masukkan menu yang ingin dipilih dan mainkan gim Hangman Setelah selesai, masukkan nama yang unik, kemudian program akan memasukkan score ke <i>scoreboard</i>		mekanisme yang dijelaskan di spesifikasi dan fungsi <i>scoreboard</i> dapat menyimpan skor ke <i>scoreboard</i>	
2	TOWER OF HANOI	Menguji gim Tower of Hanoi dan fungsi SCOREBOARD	Mulai program Ketik START Ketik QUEUE GAME dan masukan angka 4 Masukan jumlah piringan yang diinginkan Mainkan gim Tower of Hanoi Setelah selesai, masukkan nama yang unik, kemudian program akan memasukkan score ke <i>scoreboard</i>	Data Test 2	Gim Tower of Hanoi berjalan sesuai mekanisme yang dijelaskan di spesifikasi dan fungsi <i>scoreboard</i> dapat menyimpan skor ke <i>scoreboard</i>	Sesuai ekspektasi
3	SNAKE ON METEOR	Menguji gim Snake on Meteor dan fungsi SCOREBOARD	Mulai program Ketik START Ketik QUEUE GAME dan masukan angka 5 Mainkan gim Snake on Meteor Setelah selesai, masukkan nama yang unik, kemudian program akan memasukkan score ke <i>scoreboard</i>	Data Test 3	Gim Snake on Meteor berjalan sesuai mekanisme yang dijelaskan di spesifikasi dan fungsi <i>scoreboard</i> dapat menyimpan skor ke <i>scoreboard</i>	Sesuai ekspektasi
4	SCOREBOARD dan HISTORY	Menguji fungsi SCOREBOARD dan fungsi HISTORY	START program Ketik SCOREBOARD Ketik HISTORY maka akan tampil scoreboard kosong apabila belum ada permainan yang dimainkan serta tidak ada permainan yang ditampilkan pada history dan menampilkan score serta history permainan jika ada permainan yang dimainkan	Data Test 4		Sesuai ekspektasi
5	CREATE GAME, DELETE GAME,	Menguji apakah skor gim yang ditambah	START program CREATE game baru bernama COD	Data Test 5	Muncul skor untuk gim yang baru ditambahkan	Sesuai ekspektasi

STEI- ITB	IF2111_TB2_02_02	Halaman 28 dari 35 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.		

	dan SCOREBOARD	dan/atau dihapus terbaharui di fungsi SCOREBOARD	Queue COD dengan QUEUE GAME Mainkan COD dengan mengetik PLAY GAME Setelah <i>gameover</i> , ketik SCOREBOARD dan lihat skor COD Lakukan SAVE Hapus gim COD dengan DELETE GAME Cek <i>scoreboard</i> dengan fungsi SCOREBOARD, pastikan skor gim COD terhapus		di <i>scoreboard</i> dan ketika gim dihapus maka skornya di <i>scoreboard</i> akan terhapus pula	
--	----------------	--	---	--	--	--

8 Pembagian Kerja dalam Kelompok

No.	Fitur/ADT	NIM Coder	NIM Tester	Keterangan
1	Main Menu	18221126, 18221090, 18221130	18221126, 18221090, 18221130	
2	Command Dasar	18221090, 18221124, 18221126, 18221130, 18221152	18221090, 18221124, 18221126, 18221130, 18221152	seperti pada Tugas Besar 1
3	SCOREBOARD	18221126, 18221124	18221126, 18221124	
4	RESET SCOREBOARD	18221126	18221126, 18221124	
5	HISTORY	18221090	18221090, 18221124	
6	RESET HISTORY	18221090	18221090, 18221124	
7	RNG	18221152	18221152	
8	Diner Dash	18221152	18221152	

9	Hangman	18221130	18221130	
10	Tower of Hanoi	18221124	18221124	
11	<i>Snake on Meteor</i>	18221152	18221152	
12	<i>PRS (Paper, Rock, Scissor)</i>	18221130	18221130	
13	Game tambahan	18221090	18221090	
14	ADT Stack	18221124, 18221090	18221090, 18221124	
16	ADT Map	18221090, 18221124, 18221126	18221126, 18221124	
17	ADT Linked List	18221152	18221130	
18	ADT Matrix	18221152, 18221124	18221124	
19	Bonus Fitur Hangman	18221130	18221130	
20	Bonus Fitur Tower of Hanoi	18221124	18221124	
21	Bonus Fitur Snake on Meteor	18221152	18221152	

9 Lampiran

9.1 Deskripsi Tugas Besar 2

BNMO merupakan suatu robot game console yang dapat menjalankan permainan. BNMO memiliki beberapa fitur utama, yaitu memainkan game, menambahkan game, menghapus game, mengurutkan game yang akan dimainkan, menampilkan game yang telah dimainkan, dan menampilkan scoreboard game. 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.

9.2 Notulen Rapat

Form Asistensi Tugas Besar

STEI- ITB	IF2111_TB2_02_02	Halaman 30 dari 35 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 : 02/K2
Nama Kelompok : Nameless
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 III

Tanggal : 23 Nov 2022	Catatan Asistensi: 1. Map tidak wajib di-hash, kalaupun di-hash gak usah ribet-ribet 2. Command harus sesuai dengan spek 3. User tidak boleh menebak huruf yang sudah ditebak. Kalau misalkan dia nebak huruf yang ditebak gimana? Jawaban: Tidak dihitung salah tapi tebakannya invalid 4. Kalau ada yang kena meteor bakal kosong terus sampai snakenya lewat
Tempat : Google Meet	
Kehadiran Anggota Kelompok:	
No	
NIM	
Tanda tangan	
1	
2	
3	
4	
5	

	Tanda Tangan Asisten:
	 (3)

STEI- ITB	IF2111_TB2_02_02	Halaman 32 dari 35 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.		

Asistensi IV

Tanggal : 30 Nov 2022	Catatan Asistensi:
Tempat : Google Meet	
Kehadiran Anggota Kelompok:	
No	1. kalau kepalanya (snake) gerak ke bagian ekor boleh atau tidak? >> kak acil: karena dihitung masih bagian tubuh jadi minta input ulang
NIM	2. jazmy: kalau bikin game baru, scoreboard game nambah lagi? >> kak acil: iya dan kalau delete game ilang
Tanda tangan	3. kan bonusnya hangman bisa nambahin kata, kalau user masukinnya harus kapital, diperiksanya single word kan ya? >> kak acil: semua katanya single word, tidak perlu spasi
1 	4. untuk snake, makanan diletakkan duluan jadi kalo ada makanan, meteornya ga bisa di situ
2 	5. sebenarnya untuk ini (prioritas obstacle) dibebasin, tapi menurut kakak asisten kalo obstacle diletakkan duluan bakal lebih gampang
3 	6. hangman kadang suka ngebug kaya pas awal main salah satu hurufnya udah muncul >> kak acil: kalau array statis coba diisi pakai karakter tertentu aja di awal.
4 	7. >> kayaknya ini magicnya ---missed--- kalau mau array statisnya diisi titik jadi gak mungkin isMember
5 	8. kalau bisa di laporan ada yang menggambarkan hubungan antar modul misal arrayMap sama command scoreboard dibikin diagram
6 	9.
	Tanda Tangan Asisten: 

9.3 Log Activity Anggota Kelompok

Waktu	NIM	Keterangan
2021-11-18 15:00	18221090, 18221124,	Rapat perdana penentuan pembagian tugas

STEI- ITB	IF2111_TB2_02_02	Halaman 33 dari 35 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.		

	18221126, 18221130, 18221152	dan mulai mengerjakan tugas bagian masing-masing.
2022-11-18	18221126	Membuat ADT Set
2022-11-19	18221124	Merevisi ADT Set
2022-11-20	18221130	Mulai membuat Hangman, Update README, Update MesinKata
2022-11-20	18221124	Update Hangman, Playgame, ADT Stack, Membuat Tower of Hanoi
2022-11-23	18221090	Membuat ADT MapHash
2022-11-23	18221124	Merevisi Tower of Hanoi
2022-11-23	18221130	Merevisi Hangman
2022-11-23 19:00	18221090, 18221124, 18221126, 18221130, 18221152	Asistensi pertama
2022-11-24	18221130	Merevisi Hangman
2022-11-25	18221124	Memperbaiki ADT MesinKata dan CreateWord
2022-11-25	18221126	Merevisi ADT Set
2022-11-28	18221126	Merevisi Scoreboard, Membuat ADT Map Score
2022-11-29	18221152	Membuat ADT Matrix Char, Point
2022-11-29	18221090	Membuat ADT Stack Game, Command History, Command Reset History
2022-11-29	18221124	Merevisi ADT Matrix Char, ADT Map Score,
2022-11-29	18221126	Merevisi Reset Scoreboard
2022-11-30	18221090	Merevisi program utama (Main) dan Play Game untuk History dan Reset History

2022-11-30	18221126	Merevisi compile.h dan main.c, ADT yang ada (map, word, arrayMap, arrayGame), Commands (SkipGame, Scoreboard, PlayGame), finalisasi Tower of Hanoi
2022-11-30 19:00	18221090, 18221124, 18221126, 18221130, 18221152	Asistensi kedua
2022-12-01	18221130	Memperbaiki bug pada Hangman
2022-12-01	18221090	Merevisi History dan ResetHistory
2022-12-01	18221152	Membuat dan merevisi Snake on Meteor, merevisi RNG, Diner Dash
2022-12-01	18221124	Memperbaiki ADT, Command, main, dan membuat command baru, snake on meteor
2022-12-01	18221126	Merevisi scoreboard dan ADT Map Score, menambahkan primitif ADT Map Score
2022-12-02	18221124	Merevisi ADT Map Score
2022-12-02	18221126	Membuat Driver Map Score, menambahkan primitif pada ADT Word, merevisi ADT, Snake on Meteor,
2022-12-02	18221152	Merevisi Snake on Meteor, RNG Game, Diner Dash,
2022-12-02	18221090	Membuat Driver Stack Game, merevisi History, menambahkan primitif pada ADT Stack Game
2022-12-02	18221130	Membuat Driver ADT ListDP dan Point
2022-12-02	18221090, 18221124, 18221126, 18221130, 18221152	Merevisi dan memfinalisasi program dan laporan