

LAPORAN TUGAS BESAR

IF2111 Algoritma dan Struktur Data STI

WayangWave

Dipersiapkan oleh:


Kelompok 02

K-03

Adinda Khairunnisa I	18222104
M. Daffa Kusuma A	18222108
Samuel Franciscus T. H.	18222131
M. Faishal Firdaus	18222136
Khayla Belva A	18222138

Sekolah Teknik Elektro dan Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

	Sekolah Teknik Elektro dan Informatika ITB	Nomor Dokumen		Halaman
		<i>IF2111-TB-03-02</i>		28
		<i>Revisi</i>	0	25/10/2023

Daftar Isi

1 Ringkasan	3
2 Penjelasan Tambahan Spesifikasi Tugas	4
2.1 Spesifikasi Fitur Tambahan Fungsi Start	4
2.2 Spesifikasi Fitur Tambahan Fungsi Load	4
2.3 Spesifikasi Fitur Tambahan Fungsi Play	4
3 Struktur Data (ADT)	5
3.1 ADT List	5
3.2 ADT Mesin Karakter	5
3.3 ADT Mesin Kata	5
3.4 ADT Queue	6
3.5 ADT Stack	6
3.6 ADT Set	6
3.7 ADT Map	6
3.8 ADT Linked List	7
4 Program Utama	7
5 Algoritma-Algoritma Menarik	7
5.1 ADT Maps	7
5.2 Algoritma IsCommandEqual	8
6 Data Test	9
6.1 Welcome Page dan Start	9
6.2 Load	10
6.3 Help	10
6.4 List	10
6.5 Play	11
6.6 Queue	11
6.7 Song	11
6.8 Playlist	11
6.9 Status	11
6.10 Save	11
6.11 <Invalid Command>	11
6.12 Quit	11
7 Test Script	12
8 Pembagian Kerja dalam Kelompok	15
9 Lampiran	16

9.1 Deskripsi Tugas Besar	16
Latar Belakang	16
Spesifikasi Umum	16
System Mechanic	17
9.2 Notulen Rapat	20
9.3 Log Activity Anggota Kelompok	24

1 Ringkasan

WayangWave, aplikasi pemutar musik inovatif yang lahir dari keinginan Bondowoso, remaja yang berusaha memenangkan hati Roro, si cantik yang hanya menyukai lagu hip-hop dari *walkman*-nya. Dengan bantuan teman-temannya—makhluk tak kasat mata—yang memiliki keahlian AI, Bondowoso menciptakan WayangWave untuk menggantikan perangkat lunak di *walkman* Roro.

Fitur-fitur utama WayangWave meliputi kemampuan untuk memutar lagu dari daftar penyanyi dan album, membuat serta menghapus playlist, mengatur urutan lagu, dan menampilkan status pemutaran. Program ini dimulai dengan main menu yang menawarkan opsi untuk memulai, memuat, atau meminta bantuan. Dari sini, pengguna dapat memasukkan berbagai perintah untuk menjalankan aplikasi, seperti memutar lagu dari penyanyi atau album tertentu, menampilkan daftar lagu, mengelola queue lagu, berinteraksi dengan playlist, melihat status pemutaran, menyimpan konfigurasi, dan keluar dari aplikasi.

Oleh karenanya, dapat disimpulkan bahwa persoalan yang dialami Bandung Bondowoso dalam upayanya menaklukkan Roro dapat diselesaikan menggunakan ilmu-ilmu yang dipelajari pada mata kuliah IF2111 Algoritma dan Struktur Data,

2 Penjelasan Tambahan Spesifikasi Tugas

2.1 Spesifikasi Fitur Tambahan Fungsi Start

Fungsi Start membaca file konfigurasi yang berisi data penyanyi, album, dan nama lagu kemudian memasukkannya ke dalam struktur ADT. Nama penyanyi dimasukkan ke dalam ADT List Statis. Nama penyanyi dan album yang dibuat oleh penyanyi tersebut disimpan dalam ADT Map. Lalu nama Penyanyi, album yang dibuat oleh penyanyi tersebut dan lagu yang berada di masing-masing album disimpan dalam Struktur ADT Maps.

2.2 Spesifikasi Fitur Tambahan Fungsi Load

Fungsi load memiliki kesamaan pada fungsi start karena membaca file konfigurasi. Namun, fungsi load juga membaca riwayat lagu, antrian, dan playlist yang berada di folder save.

2.3 Spesifikasi Fitur Tambahan Fungsi Play

Fungsi Play memiliki kesamaan pada fungsi list dalam output, namun dalam membentuk *CurrentSong*, algoritma yang kami gunakan memiliki approach yang sedikit berbeda. Kami meng *assign* indeks dari tiap komponen memori, yaitu indeks penyanyi, indeks album, dan indeks lagu. Indeks ini akan kami gunakan untuk mengakses adt dari tiap komponen memori lalu menyimpannya dalam variable word. Setelah itu, kami membuat *CurrentSong* dengan bentuk word dengan titik koma (;) sebagai pembatasnya

(contoh: penyanyi; album; lagu) dengan cara menggabungkan satu persatu variable word komponen dengan fungsi *ConcatWord*.

3 Struktur Data (ADT)

3.1 ADT List

- Sketsa struktur data: Terdapat list statis dan list dinamis yang terdiri dari variabel bertipe integer dan Word. Pada ADT ini didefinisikan panjang maksimum elemen list yaitu *MaxEl* sebanyak 100. Prototype yang digunakan pada command yaitu *InsertLastDynamic* dan *setItem* yang berguna untuk menambahkan data bertipe Word di list.
- Persoalan yang diselesaikan: Menyimpan daftar penyanyi dan playlist.
- Alasan pemilihan: Informasi nama penyanyi dapat disimpan pada array *namaPenyanyi* dan setiap ada pengurangan/penambahan dapat terselesaikan.
- Diimplementasikan sebagai ADT list statis dan list dinamis dengan nama file header “list_dinamis.h” dan “listStatik.h”

3.2 ADT Mesin Karakter

- Sketsa struktur data: Terdapat variable extern yang memiliki tipe char yaitu *currentChar* dan extern bertipe boolean yaitu EOP. Beberapa prototype yang ada di dalam ADT ini adalah *START()* dan *ADV()*.
- Persoalan yang diselesaikan: Melakukan parsing command program dan pembacaan file konfigurasi ke dalam aplikasi.
- Alasan pemilihan: Mesin kata membutuhkan mesin karakter dalam mengimplementasikan fungsi dan prosedur prototype nya.
- Diimplementasikan sebagai ADT mesin karakter dengan nama file header “mesinkarakter.h”

3.3 ADT Mesin Kata

- Sketsa struktur data: Terdapat struct Word yang terdiri dari Length bertipe integer dan *TabWord[NMax]* yang bertipe char. Adapun define spasi (‘ ’) dengan BLANK, ‘\n’ dengan NEWLINE, dan value NMax bertipe integer yaitu 50.
- Persoalan yang diselesaikan: Melakukan parsing command program dan pembacaan file konfigurasi ke dalam aplikasi.
- Alasan pemilihan: Membaca informasi file external memerlukan akuisisi kata demi kata. Pembacaan input command dari user juga diterima dalam bentuk Word sehingga perlu digunakan mesin kata
- Diimplementasikan sebagai ADT mesin kata dengan nama file header “mesinkata.h”

3.4 ADT Queue

- Sketsa struktur data: Terdapat struct Queue yang terdiri dari *idxTail* dan *idxHead* yang bertipe integer serta *buffer[CAPACITY]* yang bertipe Word.
- Persoalan yang diselesaikan: Menentukan urutan lagu yang ingin diputar.
- Alasan pemilihan: Lagu yang akan diputar oleh user direpresentasikan sebagai queue atau diputar secara berurutan (*First In First Out/FIFO*).
- Diimplementasikan sebagai ADT queue dengan nama file header “queue.h”.

3.5 ADT Stack

- Sketsa struktur data: Terdapat struct Stack yang terdiri dari address yang bertipe integer dan *INFOTYPE* bertipe Word. Pada ADT ini didefinisikan panjang maksimum elemen Stack yaitu *MaxElStack* dengan value 100 dan *Undef -1*.
- Persoalan yang diselesaikan: Menentukan tumpukan riwayat atau history daftar lagu
- Alasan pemilihan: Urutan lagu yang telah dimainkan dalam aplikasi akan diletakkan pada elemen TOP yang berlaku LIFO (*Last In Last Out*).
- Diimplementasikan sebagai ADT stack dengan nama file header “stack.h”

3.6 ADT Set

- Sketsa struktur data: Terdapat struct Set yang terdiri dari Count bertipe integer dan *Elements[MaxElSet]* bertipe Word. Pada ADT ini didefinisikan panjang maksimum elemen Set yaitu *MaxElSet* dengan value 100 dan *Undef -1*.
- Persoalan yang diselesaikan: Menyimpan lagu dari suatu album
- Alasan pemilihan: Dengan menggunakan set, lagu yang ada dalam suatu album dipastikan tidak ada yang duplikat.
- Diimplementasikan sebagai ADT set dengan nama file header “set.h”

3.7 ADT Map

- Sketsa struktur data: Terdapat struct Map dan maps. Map terdiri dari Count bertipe integer dan *Elements[MaxElMap]* yang mempunyai *keytype* bertipe Word dan *valuetype* bertipe Set. Sementara struct maps terdiri dari Count bertipe integer dan *Elements[MaxElMap]* bertipe Map.
- Persoalan yang diselesaikan: Menyimpan album untuk tiap penyanyi dan menyimpan lagu untuk tiap album.
- Alasan pemilihan: Dengan menggunakan map, bisa mengakses album apa saja yang dimiliki oleh seorang penyanyi dan lagu apa saja yang dimiliki tiap album.
- Diimplementasikan sebagai ADT map dengan nama file header “map.h”

3.8 ADT Linked List

- Sketsa struktur data: Terdapat struct *LinierList* dan *ElmtList* yang terdiri dari info yang bertipe *Word* dan *First*, *next*, *addressLinier* dengan pointer. Memiliki prototype *InsVLast* yang berguna untuk memasukkan tipe data *Word* ke dalam List Linier di urutan paling akhir.
- Persoalan yang diselesaikan: Menyimpan isi lagu dari suatu daftar playlist
- Alasan pemilihan: Representasi lagu dari sebuah playlist merupakan kumpulan list yang kontigu, sehingga dalam implementasi nya membutuhkan list linier
- Diimplementasikan sebagai ADT list linier dengan nama file header “list_linier.h”

4 Program Utama

Pengaplikasian program utama WayangWave terdapat dalam main.c yang berisikan file header dari ADT, Spesifikasi Program, Driver, beserta file konfigurasinya. Ketika *user* memanggil program, akan ditampilkan *Welcoming Page* untuk menyapa *user*, kemudian, program akan menampilkan input command pertama, yang terbagi menjadi 3 command yaitu, START, LOAD <filename>, dan HELP. Jika *user* memasukkan command yang tidak sesuai dengan ketentuan, maka *user* akan mendapatkan *pop-up notification* bahwa command invalid.

Apabila *user* telah berhasil masuk ke dalam main menu, maka program akan meminta input command lain yang sesuai. Untuk mengakses data command yang tersedia dalam program, *user* dapat memasukkan command HELP, perlu diperhatikan bahwa program tidak akan menerima command dengan penulisan yang tidak sesuai.

5 Algoritma-Algoritma Menarik

5.1 ADT Maps

Algoritma ini merupakan adaptasi dari ADT array dengan modifikasi dibagian *ElType* menjadi ADT Map yang berisi key dan value. Hal ini dapat mempermudah dalam penyimpanan data album dan lagu dari artis yang berbeda.

```

// typedef int bool;
typedef Word keytype;
typedef Set valuetype;
typedef int address;

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

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

typedef struct
{
    Map Elements[MaxElMap];
    address count;
} maps;

```

Gambar 5.1. Algoritma IsCommandEqual

5.2 Algoritma IsCommandEqual

Algoritma IsCommandEqual didefinisikan pada ADT mesinkata.c. Algoritma ini digunakan untuk mengecek apakah command yang diinput oleh user sama dengan character yang ingin kita tentukan. Fungsi inilah yang menjalankan command START, LOAD, QUIT, dan lain-lain sesuai dengan inputan user.


```

boolean IsCommandEqual(Word Input, char * kata)
{
    boolean equal = true;
    if (Input.Length == stringLen(kata))
    {
        int i = 0;
        while (i < Input.Length && equal)
        {
            if (Input.TabWord[i] != kata[i])
            {
                equal = false;
            } else {
                i++;
            }
        }
        return equal;
    }
    else
    {
        return false;
    }
}

```

Gambar 5.2. Algoritma IsCommandEqual

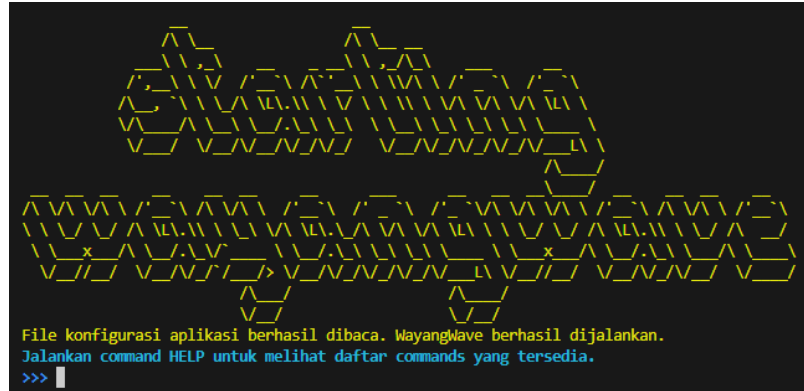
6 Data Test

6.1 Welcome Page dan Start

Test ini dilakukan untuk memastikan program sudah berjalan dengan benar yaitu berhasil membaca file config.txt dan Map yang berisikan penyanyiAlbum sudah tidak kosong.



Gambar 6.1.1. Tampilan main page saat belum dimulai



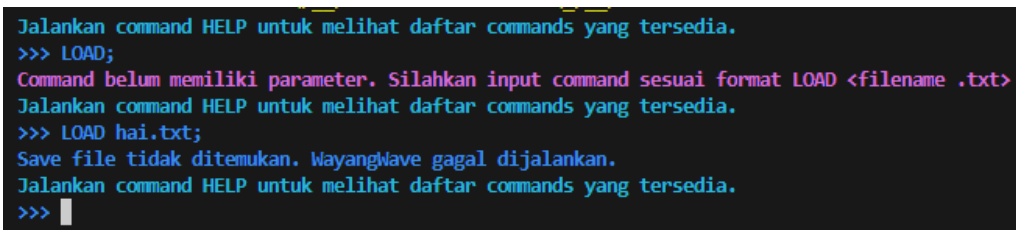
Gambar 6.1.2. Tampilan setelah menerima command “START”

6.2 Load

Tes ini dilakukan untuk menguji command “LOAD” yang sudah tersimpan sebelumnya di folder save.



Gambar 6.2.1. Tampilan saat berhasil membaca LOAD file.txt



Gambar 6.2.2. Tampilan saat LOAD dijalankan tanpa memasukkan filename.txt dan file tidak ada di folder save

6.3 Help

Tes ini dilakukan untuk menguji command help sebelum dan sesudah pengguna memasuki program.

```
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> HELP;
===== [ Menu Help WayangWave ] =====
1. START -> Untuk masuk sesi baru
2. LOAD <filename.txt> -> Untuk memulai sesi berdasarkan file konfigurasi
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> |
```

Gambar 6.3.1 Help sebelum masuk ke dalam sesi

```
File konfigurasi aplikasi berhasil dibaca. WayangWave berhasil dijalankan.
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> HELP;
1. LIST -> Untuk menampilkan daftar lagu, penyanyi, album, dan playlist
2. PLAY -> Untuk memulai suatu lagu
3. QUEUE -> Untuk memanipulasi queue (urutan) lagu
4. SONG -> Menavigasi lagu yang ada pada queue lagu saat ini
5. PLAYLIST -> Melakukan basic command untuk playlist yaitu CREATE, ADD, SWAP, REMOVE dan DELETE.
   5.1 CREATE -> Membuat playlist baru dan ditambahkan pada daftar playlist pengguna.
   5.2 ADD -> Menambahkan lagu pada suatu playlist yang telah ada sebelumnya pada daftar playlist pengguna.
   5.3 SWAP -> Menukar lagu pada urutan ke x dan juga urutan ke y di playlist dengan urutan ke id.
   5.4 REMOVE -> Menghapus lagu dengan urutan n pada playlist dengan index id.
   5.5 DELETE -> Melakukan penghapusan suatu existing playlist dalam daftar playlist pengguna.
6. STATUS -> menampilkan lagu yang sedang dimainkan beserta Queue song yang ada dan dari playlist mana lagu itu diputar.
7. SAVE -> menyimpan state aplikasi terbaru ke dalam suatu file.
8. QUIT -> Untuk keluar dari sesi
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> |
```

Gambar 6.3.2 Help setelah masuk ke dalam sesi

6.4 List

Tes ini dilakukan untuk menguji command list. Command ini terdiri dari “LIST DEFAULT” dan “LIST PLAYLIST”, dimana LIST DEFAULT menampilkan daftar penyanyi, lalu menampilkan album, serta lagu yang dimasukkan oleh pengguna, dan LIST PLAYLIST menampilkan daftar playlist yang dimiliki oleh program.

```
>>> Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> LIST DEFAULT;
Daftar Penyanyi :
1. BLACKPINK
2. Arctic Monkeys

Ingin melihat album yang ada?(Y/N): Y;

Pilih penyanyi untuk melihat album mereka: BLACKPINK;

Daftar Album oleh BLACKPINK :
1. BORN PINK
2. THE ALBUM

Ingin melihat lagu yang ada?(Y/N): Y;

Pilih album untuk melihat lagu yang ada di album : BORN PINK;

Daftar Lagu di BORN PINK :
1. Pink Venom
2. Shut Down
3. Typa Girl
4. Ready For Love
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> |
```

Gambar 6.4.1 List Default

```
>>> LIST PLAYLIST;
Daftar Playlist yang kamu miliki:
1. BLACKPINK My Top Three
2. Arctic Monkeys My Top 5
3. Mixed Playlist
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> █
```

Gambar 6.4.2 List Playlist

6.5 Play

Tes ini dilakukan untuk menguji command play. Command ini terdiri dari “PLAY SONG” dan “PLAY PLAYLIST”, dimana PLAY SONG pertama akan menampilkan daftar penyanyi, lalu menampilkan daftar album dan lagu yang diinginkan pengguna. Ketika pengguna sudah memilih lagu, program akan memutar lagu tersebut dengan mengosongkan antrian (queue) dan riwayat (stack) lagu, serta mengubah state dari *struct* CurrentSong.

```
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> PLAY SONG;
Daftar Penyanyi :
1. BLACKPINK
2. Arctic Monkeys

Masukkan Nama Penyanyi yang dipilih : BLACKPINK;

Daftar Album oleh BLACKPINK :
1. BORN PINK
2. THE ALBUM

Masukkan Nama Album yang dipilih :
BORN PINK;

Daftar Lagu di album BORN PINK :
1. Pink Venom
2. Shut Down
3. Typa Girl
4. Ready For Love

Masukkan ID Lagu yang dipilih :
1;

Memutar lagu Pink Venom oleh BLACKPINK.
```

Gambar 6.5.1 Play Song

```

Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> PLAY PLAYLIST;

Masukkan ID Playlist: 2;

Memutar playlist Arctic Monkeys My Top 5.
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>>

```

Gambar 6.5.2 Play Playlist

6.6 Queue

Test ini menguji fitur QUEUE yang akan menambahkan lagu, album, maupun playlist ke dalam antrian sesuai dengan input pengguna. Setelah memasukkan perintah ini, lagu, album, maupun playlist yang ditambahkan akan tersimpan pada antrian yang sudah ada. Ada dua kemungkinan yang dapat terjadi pada fitur ini, yaitu lagu, album maupun playlist yang dipilih pengguna tidak valid atau tidak ada; dan lagu, album atau playlist berhasil ditambahkan ke dalam queue.

```

Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> QUEUE SONG;
Daftar Penyanyi :
  1. BLACKPINK
  2. Arctic Monkeys
Masukkan Nama Penyanyi : BLACKPINK;
Daftar Album oleh BLACKPINK :
  1. BORN PINK
  2. THE ALBUM
Masukkan Nama Album : THE ALBUM;
Daftar Lagu Album THE ALBUM oleh BLACKPINK :
  1. How You Like That
  2. Ice Cream (with Selena Gomez)
  3. Bet You Wanna (Feat. Cardi B)
Masukkan ID lagu yang dipilih : 1;
Berhasil menambahkan lagu "How You Like That" oleh "BLACKPINK" ke queue.
Berhasil menambahkan lagu ke queueJalankan command HELP untuk melihat daftar commands yang tersedia.
>>> STATUS;
>> STATUS;
Now Playing :
Arctic Monkeys-AM-R U Mine?
Queue:
1. BLACKPINK-BORN PINK-Typa Girl
2. BLACKPINK-THE ALBUM-How You Like That
3. BLACKPINK-BORN PINK-Pink Venom
4. BLACKPINK-THE ALBUM-How You Like That

```

Gambar 6.6.1 Queue Song

```

Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> QUEUE REMOVE 2;
Berhasil menghapus lagu dari queueJalankan command HELP untuk melihat daftar commands yang tersedia.
>>> STATUS;
>> STATUS;
Now Playing :
Arctic Monkeys-AM-R U Mine?
Queue:
1. BLACKPINK-BORN PINK-Typa Girl
2. BLACKPINK-BORN PINK-Pink Venom

```

Gambar 6.6.2 Queue Remove

```

Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> QUEUE SWAP 1 2;
Berhasil menukar urutan laguJalankan command HELP untuk melihat daftar commands yang tersedia.
>>> STATUS;
>> STATUS;
Now Playing :
Arctic Monkeys-AM-R U Mine?
Queue:
1. BLACKPINK-BORN PINK-Pink Venom
2. BLACKPINK-BORN PINK-Typa Girl
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>>

```

Gambar 6.6.3 Queue Swap

6.7 Song

Tes ini dilakukan untuk menguji command song. Command song ini terdiri dari “songNext” dan “songPrevious”. SongNext melakukan Push RiwayatLagu dan juga dequeue UrutanLagu. Sedangkan SongPrevious melakukan Enqueue ke awal dari UrutanLagu kemudian melakukan Pop ke RiwayatLagu.

```

>> STATUS;
Now Playing :
Arctic Monkeys-AM-R U Mine?
Queue:
1. BLACKPINK-BORN PINK-Typa Girl
2. BLACKPINK-THE ALBUM-How You Like That
3. BLACKPINK-BORN PINK-Pink Venom
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> SONG NEXT;
Memutar lagu selanjutnyaTypa Girl oleh BLACKPINKJalankan command HELP un
tuk melihat daftar commands yang tersedia.
>>> SONG PREV;
memutar lagu sebelumnyaR U Mine? oleh Arctic MonkeysJalankan command HEL
P untuk melihat daftar commands yang tersedia.
>>>

```

Gambar 6.7 Song Next dan Song Prev

6.8 Playlist

Tes ini dilakukan untuk menguji command playlist. Command ini meliputi fungsi-fungsi dasar untuk mengelola playlist seperti membuat playlist baru (CREATE), menambah lagu (ADD), menukar lagu (SWAP), menghapus lagu (REMOVE), dan menghapus seluruh playlist (DELETE). PLAYLIST CREATE membuat playlist kosong baru yang boleh memiliki nama yang sama dengan yang ada. PLAYLIST ADD menambahkan lagu tertentu atau semua lagu dalam album ke playlist yang sudah ada, kecuali lagu tersebut sudah ada di dalamnya. PLAYLIST SWAP menukar posisi dua lagu dalam sebuah playlist, sementara PLAYLIST

REMOVE menghapus lagu berdasarkan urutannya dalam playlist. Terakhir, PLAYLIST DELETE menghapus sebuah playlist yang ada. Semua command ini memberikan pesan error jika terdapat masukan yang tidak valid dari pengguna. Namun Sayangnya fungsi ini tidak berjalan sesuai yang diharapkan.

6.9 Status

Tes ini dilakukan untuk menguji Status. Setelah perintah dijalankan, akan keluar lagu yang sedang dimainkan, urutan lagu serta apakah urutan lagu itu berasal dari playlist

```
Save file berhasil dibaca. WayangWave berhasil dijalankan.
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> STATUS;
>> STATUS;
Now Playing :
Arctic Monkeys-AM-R U Mine?
Queue:
1. BLACKPINK-BORN PINK-Typa Girl
2. BLACKPINK-THE ALBUM-How You Like That
3. BLACKPINK-BORN PINK-Pink Venom
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> █
```

Gambar 6.9 Status

6.10 Save

Save menyimpan current state ke dalam sebuah filename.txt dan disimpan ke folder save

```
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> SAVE ping.txt;
Save berhasil dilakukan, Yeay!
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> █
```

Gambar 6.10 Save

6.11 <Invalid Command>

Ketika memasukkan command yang tidak sesuai dengan spesifikasi, akan keluar teks error. Namun, sebelumnya dipanggil system clear

```
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> TUBES ALSTRUKDAT;
Command tidak bisa dieksekusi!
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> █
```

Gambar 6.11 Invalid Comment

6.12 Quit

Tes ini dilakukan untuk menguji command “QUIT” yang akan mengeluarkan user dari program WayangWave.

```
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> QUIT;

Kamu keluar dari WayangWave.
Dadah ^_^/
PS D:\alstrukdat\ini tubes yaallah bisa pls\TubesAlstrukdatKel2\src>
```

Gambar 6.3.1. Tampilan QUIT saat belum memulai WayangWave

Adapun prosedur quit ketika ingin keluar dari program saat sudah memulai WayangWave atau melakukan START. Terdapat pilihan apakah ingin menyimpan data sesi atau tidak, jika iya maka user akan diminta untuk menginput <filename.txt> dan akan memanggil command save lalu keluar dari program.

```
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> QUIT;
Apakah kamu ingin menyimpan data sesi sekarang? (Y/N): Y;

Silahkan input nama file untuk disimpan: halo.txt;

Kamu keluar dari WayangWave.
Dadah ^_^/
```

Gambar 6.3.2. Tampilan QUIT saat ingin menyimpan data sesi

```
Jalankan command HELP untuk melihat daftar commands yang tersedia.
>>> QUIT;
Apakah kamu ingin menyimpan data sesi sekarang? (Y/N): N;

Kamu keluar dari WayangWave.
Dadah ^_^/
PS D:\alstrukdat\ini tubes yaallah bisa pls\TubesAlstrukdatKel2\src>
```

Gambar 6.3.3. Tampilan QUIT saat tidak ingin menyimpan data sesi

7 Test Script

No.	Fitur yang Dites	Tujuan Testing	Langkah-Langkah Testing	Input Data Test	Hasil yang Diharapkan	Hasil yang Keluar
1	Fitur START	Memeriksa apakah WayangWave	Memasukkan command START pada main page awal	Data Test 1	WayangWave berhasil dimulai dan menampilkan message bahwa file	Sesuai yang diharapkan

		dapat dijalankan			konfigurasi dapat dibaca	
2	Fitur LOAD	Memeriksa apakah dapat membaca file sesuai dengan inputan yang terdapat di folder save	Memasukkan command LOAD <filename.txt> pada main page untuk file yang berada di folder save	Data Test 2	Pemutar lagu WayangWave berhasil dilanjutkan sesuai dengan kondisi saat save	Sesuai yang diharapkan
3	Fitur QUIT	Memeriksa apakah dapat keluar dari program	Memasukkan command QUIT untuk keluar dari program	Data Test 12	Jika memasukkan command quit pada main page awal, maka akan keluar dari program utama. Jika memasukkannya saat program sudah berjalan akan diberikan opsi untuk save terlebih dahulu atau tidak.	Sesuai yang diharapkan
4	Fitur PLAY	Memeriksa apakah dapat memutar lagu atau playlist yang diinginkan user	Memasukkan command PLAY SONG untuk memutar lagu dan PLAY PLAYLIST untuk memutar playlist	Data Test 5	Saat melakukan PLAY SONG queue dan riwayat lagu akan menjadi kosong. Lalu, di PLAY PLAYLIST current song akan menjadi lagu pada urutan pertama playlist.	Sesuai yang diharapkan
5	Fitur LIST	Memeriksa apakah dapat menampilkan list playlist yang ada, list penyanyi, list album dari penyanyi, dan list lagu yang ada di album	Memasukkan command LIST DEFAULT untuk melihat list penyanyi, album, dan lagu yang ada dan LIST PLAYLIST untuk menampilkan playlist yang ada pada pengguna	Data Test 4	Jika memasukkan command LIST DEFAULT akan menampilkan list penyanyi dan terdapat konfirmasi apakah ingin melihat album/lagu. Lalu, LIST PLAYLIST menampilkan playlist yang ada	Sesuai yang diharapkan
6	Fitur HELP	Memeriksa apakah dapat menampilkan daftar command	Memasukkan command HELP untuk menampilkan daftar command	Data Test 3	Jika memasukkan HELP akan menampilkan daftar command yang mungkin untuk dieksekusi dengan deskripsinya.	Sesuai yang diharapkan
7	Fitur SONG	Memeriksa apakah navigasi lagu	Memasukkan command SONG untuk	Data Test 7	Jika memasukkan SONG NEXT akan memutar lagu yang	

		dapat dijalankan	melakukan next song dan previous song		berada di dalam queue. Lalu, SONG PREVIOUS akan memutar lagu yang terakhir kali diputar	
8	Fitur PLAYLIST	Memeriksa apakah bisa melakukan CREATE, ADD, SWAP, REMOVE, DELETE sebuah playlist	Memasukkan command PLAYLIST CREATE untuk membuat playlist baru, command PLAYLIST ADD untuk menambahkan lagu pada suatu playlist, PLAYLIST SWAP untuk menukar urutan lagu, PLAYLIST REMOVE untuk menghapus lagu dari playlist, PLAYLIST DELETE untuk menghapus playlist	Data Test 8	Jika memasukkan command PLAYLIST CREATE akan membuat playlist baru, command PLAYLIST ADD akan menambahkan lagu pada suatu playlist, PLAYLIST SWAP akan menukar urutan lagu, PLAYLIST REMOVE akan menghapus lagu dari playlist, PLAYLIST DELETE akan menghapus playlist	Kurang sesuai harapan hanya terdapat fungsi playlist delete.
9	Fitur QUEUE	Memeriksa apakah dapat memanipulasi urutan lagu	Memasukkan command QUEUE SONG untuk menambahkan lagu ke dalam queue, QUEUE PLAYLIST untuk menambahkan lagu yang ada dalam playlist ke dalam queue, QUEUE SWAP untuk menukar lagu, QUEUE REMOVE untuk menghapus lagu dari queue, QUEUE CLEAR untuk mengosongkan queue	Data Test 6	Jika memasukkan command QUEUE SONG akan menambahkan lagu ke dalam queue, QUEUE PLAYLIST akan menambahkan lagu yang ada dalam playlist ke dalam queue, QUEUE SWAP akan menukar lagu, QUEUE REMOVE akan menghapus lagu dari queue, QUEUE CLEAR akan mengosongkan queue	Hanya kurang dalam mengidentifikasi queue playlist
10	Fitur STATUS	Memeriksa lagu apa yang sedang dimainkan	Memasukkan command STATUS untuk menampilkan lagu yang sedang dimainkan beserta Queue song yang ada dan dari playlist mana lagu itu diputar	Data Test 9	Menampilkan lagu yang sedang dimainkan beserta Queue song yang ada dan dari playlist mana lagu itu diputar	Hanya kurang dalam mengidentifikasi apakah queue lagu berada pada playlist

11	Fitur SAVE	Memeriksa apakah dapat menyimpan state terbaru	Memasukkan command SAVE <filename.txt>	Data Test 10	Menyimpan state aplikasi terbaru ke dalam suatu file dan penyimpanan dilakukan di folder save.	Sesuai yang diharapkan
12	Fitur <Invalid Comment>	Memeriksa apakah akan keluar teks error saat command tidak sesuai	Memasukkan command yang tidak sesuai fitur atau tidak valid	Data Test 11	Menampilkan teks error	Sesuai yang diharapkan

8 Pembagian Kerja dalam Kelompok

NIM	Nama	Deskripsi Tugas
18222104	Adinda Khairunnisa I.	<ol style="list-style-type: none"> 1. Membuat ADT baru maps. 2. Membuat prosedur queue. 3. Membuat invalid command untuk queue. 4. Mengerjakan laporan bagian cover, ringkasan, program utama, deskripsi tugas besar, tabel pembagian kerja. 5. Membuat hiasan/interface untuk program. 6. Membuat Makefile.
18222108	M. Daffa Kusuma A.	<ol style="list-style-type: none"> 1. Membuat prosedur song dan playlist. 2. Membuat invalid command untuk song dan playlist. 3. Menambahkan dan update ADT yang digunakan. 4. Membantu menambahkan deskripsi data test Song dan Playlist 5. Membantu membuat
18222131	Samuel Franciscus T.	<ol style="list-style-type: none"> 1. Narahubung asisten dengan kelompok. 2. Membuat prosedur list, play & help. 3. Membuat invalid command untuk list, play & help. 4. Menghias & menata output. 5. Mengerjakan laporan bagian notulen rapat dan log activity. 6. Menambahkan ADT MapPlaylist
18222136	M. Faishal Firdaus	<ol style="list-style-type: none"> 1. Membuat prosedur status dan save.

		<ol style="list-style-type: none"> 2. Membuat invalid command untuk status dan save. 3. Membantu membuat Song 4. Membantu dokumen bagian test dan penjelasan spesifikasi tambahan 5. Membantu membuat driver ADT
18222138	Khayla Belva A.	<ol style="list-style-type: none"> 1. Membuat prosedur start, load, dan quit. 2. Membuat struktur main dan fungsi tambahan.c 3. Membuat invalid command untuk start, load & quit. 4. Membuat driver ADT. 5. Mengerjakan laporan bagian struktur data (ADT), fitur data test dan tes script.

9 Lampiran

9.1 Deskripsi Tugas Besar

Latar Belakang

Bandung, kota yang indah nan elok. Cuaca sejuk yang menyambut pagi dan suasana romantis yang menutup hari. Lagu sunda didengar di perempatan bawah jembatan selagi menunggu lampu merah. Banyak cerita yang diukir disini, penuh suka duka yang berasal dari semangat yang tak kunjung patah. Tapi cerita ini bukan tentang Bandung itu... Bondowoso dengan nama depan Bandung merupakan seorang remaja masa mataram kuno, terlahir (katanya) dari lagu dari boombox kesayangannya. Ia memiliki sahabat kasat mata yang biasa ia ajak berbicara. Hal ini disebabkan oleh ia beradaptasi karena tidak memiliki teman sejak kecil. Orang tua Bondowoso sudah berusaha membantunya mencari teman sejak kecil. Tetapi dikarenakan dia memiliki fisik seperti pekerja kasar di umur 30 sedangkan ia baru saja lulus SMP, tidak ada yang berani untuk berteman dengan dia.

Suatu hari, Bondowoso jatuh cinta kepada Roro, si teteh geulis yang hanya suka dengan lagu hip-hop terbaru dari walkman-nya. Bondowoso yang memiliki sense musik yang sangat berkelas pada masanya, merasakan bahwa Roro belum melihat seluruh dunia musik yang luas.

Bondowoso telah menyanyikan beragam jenis lagu yang disusunnya. tetapi Roro tidak ingin melepaskan headset-nya yang terhubung ke walkman-nya. Hingga suatu hari saat Roro sedang tidur dan Bondowoso mendapatkan akses penuh di walkman Roro (karena Roro menggunakan password lemah). Bondowoso tidak ingin berniat jahat, tetapi seluruh teman kasat matanya mendukung Bondowoso untuk memanfaatkan situasi. Salah satu teman kasat mata dari Bondowoso dulunya merupakan pendiri startup produksi roda

batu pada masanya berkata, “Bandung, berikan aku kekuatan Anak Intern AI. Saya akan buat kamu aplikasi WayangWave yang dapat meluluhkan hati Roro”. Bandung, seperti investor pemula mengikuti perkataannya dan mengerahkan 5 teman kasat mata lainnya untuk membangun aplikasi yang dapat menggantikan perangkat lunak di walkman Roro.

Kalian sebagai kumpulan makhluk kasat mata yang disuruh, harus memberikan terbaik untuk teman baik kalian.

Spesifikasi Umum

Buatlah sebuah aplikasi simulasi berbasis CLI (command-line interface). Sistem ini dibuat dalam bahasa C dengan menggunakan struktur data yang sudah kalian pelajari di mata kuliah ini. Kalian boleh menggunakan (atau memodifikasi) struktur data yang sudah kalian buat untuk praktikum pada tugas besar ini. Daftar ADT yang wajib digunakan dapat dilihat pada bagian Daftar ADT. Library yang boleh digunakan hanya `stdio.h`, `stdlib.h`, `time.h`, dan `math.h`.

System Mechanic

1. About The System

WayangWave merupakan sebuah aplikasi yang bisa mensimulasikan service pemutaran musik. WayangWave ini memiliki beberapa fitur utama, yaitu:

1. Memutar lagu
2. Menampilkan daftar lagu
3. Membuat dan menghapus playlist
4. Mengatur urutan dimainkannya lagu
5. Menampilkan status dari aplikasi

2. Main Menu

Ketika program pertama kali dijalankan, WayangWave akan memperlihatkan main menu yang berisi welcome page dan beberapa command yaitu START, LOAD, dan juga HELP. Setelah itu, main menu akan menerima masukan berupa command yang akan dijelaskan pada bagian berikutnya.

3. Command

Pemain dapat memasukkan command-command berikut:

1. START

START merupakan salah satu command yang dimasukkan pertama kali dalam WayangWave. Setelah menekan Enter, dibaca file konfigurasi default yang berisi daftar penyanyi serta album yang dimiliki.

2. LOAD <filename>

LOAD merupakan salah satu command yang dimasukkan pertama kali dalam WayangWave. Command ini memiliki satu argumen yaitu filename yang merepresentasikan suatu save file yang ingin dibuka. File didapatkan dari folder save. Setelah menekan Enter, akan dibaca save file <filename> yang berisi list penyanyi, album, dan lagu yang bisa diputar.

3. LIST

a. LIST DEFAULT

Command LIST DEFAULT digunakan untuk melihat list penyanyi yang ada. Selanjutnya dapat memilih untuk melihat album dari penyanyi yang dipilih. Kemudian melihat lagu yang ada dari album yang dipilih. Terdapat konfirmasi apakah ingin melihat album/lagu.

b. LIST PLAYLIST

Command LIST PLAYLIST digunakan untuk menampilkan playlist yang ada pada pengguna.

4. PLAY

a. PLAY SONG

Command PLAY SONG digunakan untuk memainkan lagu berdasarkan masukan nama penyanyi, nama album, dan id lagu. Ketika command ini berhasil dieksekusi, queue dan riwayat lagu akan menjadi kosong.

b. PLAY PLAYLIST

Command PLAY PLAYLIST digunakan untuk memainkan lagu berdasarkan id playlist. Ketika command ini berhasil dieksekusi, current song akan menjadi lagu pada urutan pertama playlist dan queue akan berisi semua lagu yang ada dalam playlist yang akan dimainkan dan isi riwayat lagu sama dengan queue, tetapi dengan urutan yang di-reverse.

5. QUEUE

a. QUEUE SONG

Command QUEUE SONG digunakan untuk menambahkan lagu ke dalam queue. Command ini menerima input lagu berdasarkan nama penyanyi, nama album, dan id dari lagu yang ingin dimasukkan ke dalam queue.

b. QUEUE PLAYLIST

Command QUEUE PLAYLIST digunakan untuk menambahkan lagu yang ada dalam playlist ke dalam queue. Command ini menerima input dari id playlist yang ingin dimasukkan ke dalam queue.

c. QUEUE SWAP < x y >

Command QUEUE SWAP digunakan untuk menukar lagu pada urutan ke x dan juga urutan ke y.

- d. **QUEUE REMOVE <id>**
Command **QUEUE REMOVE** digunakan untuk menghapus lagu dari queue. Command ini menerima input berupa urutan lagu (id) yang ingin dihapus dari queue.
- e. **QUEUE CLEAR**
Command **QUEUE CLEAR** digunakan untuk mengosongkan queue.

6. SONG

- a. **SONG NEXT**
Command **SONG NEXT** digunakan untuk memutar lagu yang berada di dalam queue. Lagu yang sedang diputar kemudian ditambah ke dalam daftar riwayat putar lagu. Jika queue kosong, yang diputar adalah lagu yang sedang diputar.
- b. **SONG PREVIOUS**
Command **SONG PREVIOUS** digunakan untuk memutar lagu yang terakhir kali diputar. Lagu yang sedang diputar kemudian ditambah ke dalam queue dengan urutan pertama. Jika daftar riwayat lagu kosong, yang diputar adalah lagu yang sedang diputar.

7. PLAYLIST

- a. **PLAYLIST CREATE**
Command **PLAYLIST CREATE** digunakan untuk membuat playlist baru dan ditambahkan pada daftar playlist pengguna. Keadaan awal playlist adalah kosong. Nama playlist dapat sama dengan playlist yang sudah ada.
- b. **PLAYLIST ADD**
Command **PLAYLIST ADD** digunakan untuk menambahkan lagu pada suatu playlist yang telah ada sebelumnya pada daftar playlist pengguna. Pada defaultnya, command ini hanya dapat menambahkan satu spesifik lagu atau semua lagu yang ada pada album kepada suatu existing playlist.

Apabila lagu atau lagu - lagu di dalam album yang ingin ditambahkan sudah ada di dalam suatu playlist pengguna maka lagu - lagu yang ditambahkan adalah yang belum ada di playlist pengguna.

Tampilkan pesan error apabila masukkan pengguna tidak valid pada setiap permintaan masukkan.

- c. **PLAYLIST SWAP <id> <x y>**
Command **PLAYLIST SWAP** digunakan untuk menukar lagu pada urutan ke x dan juga urutan ke y di playlist dengan urutan ke id.
- d. **PLAYLIST REMOVE <id> <n>**

Command PLAYLIST REMOVE digunakan untuk menghapus lagu dengan urutan n pada playlist dengan index id.

e. PLAYLIST DELETE

Command PLAYLIST DELETE digunakan untuk melakukan penghapusan suatu existing playlist dalam daftar playlist pengguna. Tampilkan pesan error apabila masukkan pengguna tidak valid pada setiap permintaan masukkan.

8. STATUS

STATUS merupakan command yang digunakan untuk menampilkan lagu yang sedang dimainkan beserta Queue song yang ada dan dari playlist mana lagu itu diputar.

9. SAVE <filename>

SAVE merupakan command yang digunakan untuk menyimpan state aplikasi terbaru ke dalam suatu file. Command SAVE memiliki satu argumen yang merepresentasikan nama file yang akan disimpan. Penyimpanan dilakukan pada folder save.

10. QUIT

QUIT merupakan command yang digunakan untuk keluar dari aplikasi WayangWave.

11. HELP

HELP merupakan command yang digunakan menampilkan daftar command yang mungkin untuk dieksekusi dengan deskripsinya.

12. <INVALID COMMAND>

Command-command selain yang disebutkan di atas dinyatakan akan tidak valid dan hanya akan mengeluarkan teks error.

9.2 Notulen Rapat

Form Asistensi Tugas Besar
IF2111/Algoritma dan Struktur Data STI
Sem. 1 2023/2024






No. Kelompok/Kelas : 2 / K-03
Nama Kelompok : Gak butuh imba
Anggota Kelompok (Nama/NIM) : 1. Muhammad Faishal Firdaus/18222136
2. Adinda Khairunnisa Indraputri/18222104
3. Muhammad Daffa Kusuma Atmaja/18222108

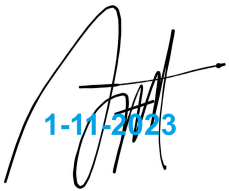
STEI- ITB	IF2111-TB-03-02	Halaman 24 dari 29 halaman
Template dokumen ini dan informasi yang dimilikinya adalah milik Sekolah Teknik Elektro dan Informatika ITB dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Sekolah Teknik Elektro dan Informatika ITB.		

4. Khayla Belva Annandira/18222138
5. Samuel Franciscus Togar H/18222131






Asisten Pembimbing : Aufa Fauqi Ardhiqi


Asistensi I

Tanggal : 1 November 2023	Catatan Asistensi: Mesin kalimat tidak perlu, untuk start pakai adt mesin kata dan mesin karakter saja, baca per line. buat load include adt set, array, map simple nya buat nyimpen lagu, album, dan penyanyi nya. Program memakai file config yang berisi dummy data. Untuk menunjukkan output dalam command list, bisa dengan mengakses key dan value (ADT set & map)
Tempat : Google Meet (online)	
Kehadiran Anggota Kelompok:	
1 NIM: 18222104 Tanda tangan: 	
2 NIM: 18222108 Tanda Tangan: 	
3 NIM: 18222131 Tanda Tangan: 	
4 NIM: 18222108 Tanda Tangan: 	
5 NIM: 18222138 Tanda Tangan: 	

	Tanda Tangan Asisten:  1-11-2023

Asistensi II

Tanggal : 15 November 2023	Catatan Asistensi: <ol style="list-style-type: none"> 1. Membahas fungsi Queue, khususnya masalah error pada saat menjalankan CurrentWord. Solusi sementara adalah dengan melakukan output (printf) di kata yang mengalami error. 2. Bagaimana cara menyinkronkan/menggabungkan hasil kerja kita jika kita mengerjakan fungsi/ADT nya masing-masing? jawaban: caranya adalah dengan menentukan dulu alur kerja (flow) dari program kita itu seperti apa, lalu setelah itu kita bisa menyambungkan serta menyusun fungsi-fungsi/ADT yang sudah kita kerjakan secara berurutan. 3. Lakukan inisialisasi pada prgram dengan membuat set berisi judul lagu di fungsi START lalu dimasukkan ke map album.
Tempat : Google Meet (online)	
Kehadiran Anggota Kelompok:	
<p>1 NIM: 18222104 Tanda tangan:</p> 	
<p>2 NIM: 18222108 Tanda Tangan:</p> 	
<p>3 NIM: 18222131 Tanda Tangan:</p> 	
<p>4 NIM: 18222108 Tanda Tangan:</p> 	
<p>5 NIM: 18222138 Tanda Tangan:</p> 	

	Tanda Tangan Asisten:  18-11-2023

9.3 Log Activity Anggota Kelompok

No	Waktu	Keterangan
1.	27 Oktober 2023	Pembagian tugas
2.	31 Oktober 2023	Pembuatan repository
3.	1 November 2023	Asistensi I
4.	1 - 15 November 2023	Pengerjaan Program dan Laporan
5.	15 November 2023	Asistensi II
6.	23 November 2023	Finalisasi Program
7.	24 November 2023	Finalisasi Laporan