

# TIC-TAC-TOE

**Tugas Besar Dasar - Dasar Pemrograman**

*Dikumpulkan untuk memenuhi sebagian persyaratan kelulusan mata kuliah Dasar-Dasar  
pemrograman*



**Kelompok A8 :**

**Difa Moch Fadillah – 201524004**

**Muhammad Azhar Alauddin – 201524013**

**Program Studi D4 Teknik Informatika  
Jurusan Teknik Komputer dan Informatika  
Politeknik Negeri Bandung  
2020/2021**

# BAB 1

## “Deskripsi Aplikasi”

### 1. Penjelasan Umum



Gambar 1 tic-tac-toe game

Game ini merupakan permainan klasik yang bernama “*Tic-tac-toe*” yang terdiri dari 2 komponen utama yaitu *board* (papan permainan) dan juga player, baik itu *bot* maupun *player* yang ditandai dengan tanda ‘X’ dan ‘O’. Dalam permainan ini, *board* yang bisa dimainkan berbentuk *grid* dengan ukuran 3x3, 5x5, dan 7x7. *Player 1* dapat menentukan akan melawan *bot* atau *player 2*. Jika *player 1* memilih *bot* sebagai lawan, maka *player 1* diwajibkan untuk memilih tingkat kesulitan dari *bot* itu sendiri. Setelah itu, *player 1* harus meng-inputkan *username*. Dan jika *player 1* memilih *player 2* sebagai lawan, maka *player 1* dan *player 2* harus meng-inputkan *username* masing-masing. Setiap turn/giliran, baik itu dari *player 1* maupun *player 2*, terdapat timer yang menunjukkan batas waktu untuk memindahkan tanda ‘X’ atau ‘O’ selama 10 detik. Jika *player* menandai dengan lebih dari 10 detik, maka giliran akan berpindah ke *player* lainnya atau ke *bot*. Selain itu juga, terdapat fitur *highscore* yang menyimpan jumlah kemenangan setiap permainan melawan *bot*, sehingga setelah menang melawan *bot*, *player* bisa melihat riwayat kemenangannya pada fitur *highscore* ini. Adapun format isi dari *highscore* ini adalah nomor sebagai jumlah dari riwayat kemenangan, time sebagai waktu mainnya, size sebagai ukuran board yang dimainkan, difficulty sebagai level kesulitan dari komputer dan winner sebagai nama dari *player* yang menang melawan *bot*.

“*Tic-tac-toe*” ini juga dilengkapi dengan menu *help* yang berfungsi untuk menampilkan aturan dan cara bermain dari *game* ini sehingga *player* yang baru pertama kali bermain *game* ini tidak kebingungan untuk cara bermainnya. Selain itu juga

terdapat menu *color theme* yang berisi pilihan warna dari game ini yang dapat dipilih oleh *player* sesuai dengan keinginan. Dengan adanya fitur ini, akan menambah desain tampilan dari *game* ini semakin menarik dan juga lebih atraktif. Adapun warna yang dipilih merupakan warna untuk tampilan text dan juga background dari permainan. *Player* juga bisa melihat identitas *developer* atau *creator* dari *game* ini dengan memilih menu *credits* dan juga bisa langsung keluar dari permainan dengan memilih menu *exit*. Dan setiap *player* sudah menuju ke salah satu pilihan di main menu, *player* juga bisa kembali dengan menekan apapun pada keyboard.

## 2. Skenario

### 2.1. Tampilan Main Menu

Pada saat *player* membuka permainan, maka tampilan yang pertama muncul adalah tampilan main menu. Pada tampilan pertama ini, terdapat beberapa pilihan yang harus dipilih oleh *player* sesuai dengan keinginan. Adapun rincian dari pilihan-pilihan yang terdapat di main menu sebagai berikut :

- a. *start-game* untuk memulai permainan
- b. *help* untuk menampilkan aturan dan cara bermain
- c. *color-theme* untuk memilih warna tampilan permainan ini
- d. *credits* untuk menampilkan identitas *developer* atau *creator* dari permainan ini
- e. *highscore* untuk menampilkan riwayat kemenangan *player* melawan bot
- f. *exit* untuk keluar dari permainan



Gambar 2 sketsa tampilan main menu

### 2.2. Tampilan Memilih Ukuran Grid

Setelah *player* memilih *start-game* pada main menu, maka *player* akan ditampilkan 3 pilihan ukuran grid yaitu 3x3, 5x5, dan 7x7. Adapun sketsa dari tampilan ini sebagai berikut



Gambar 3 sketsa tampilan memilih ukuran grid

### 2.3. Tampilan Memilih Lawan

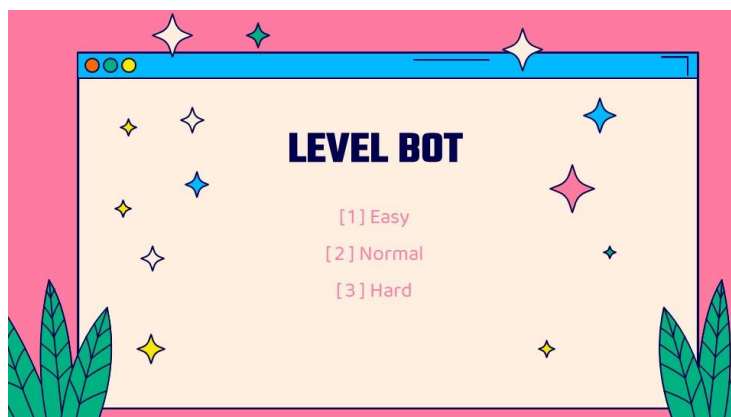
Setelah player memilih ukuran grid, maka player akan ditampilkan pilihan lawan apakah akan melawan bot atau player 2, Adapun sketsa dari tampilan ini sebagai berikut



Gambar 4 sketsa tampilan memilih lawan

### 2.4. Tampilan Memilih Level Kesulitan Komputer

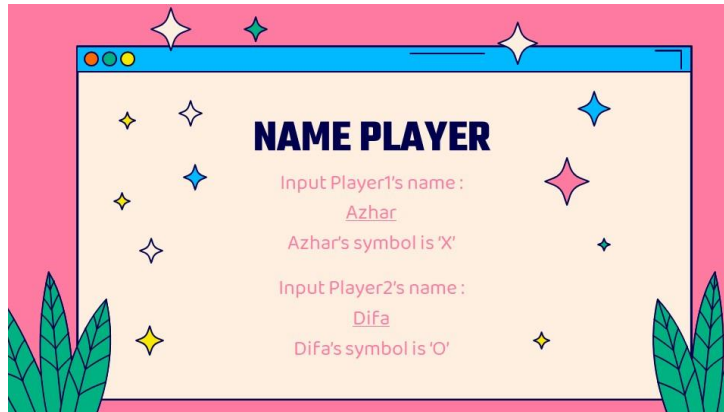
Setelah player memilih lawan, maka player akan ditampilkan tampilan pemilihan level kesulitan dari komputer. Level kesulitan dari bot nya itu sendiri terdapat 3 yaitu, easy, normal, dan hard. Adapun sketsa dari tampilan ini sebagai berikut



Gambar 5 sketsa tampilan memilih level komputer

## 2.5. Tampilan Meng-inputkan Username Player

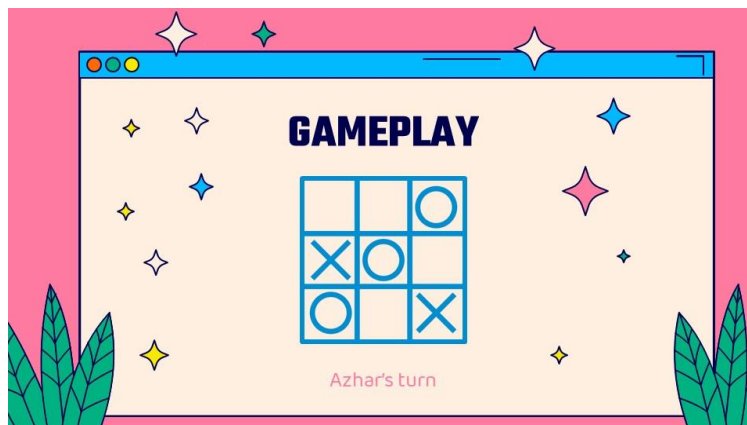
Setelah player memilih level kesulitan komputer, maka player akan ditampilkan tampilan pengisian nama player. Jika melawan bot, maka hanya player 1 yang harus menginputkan username dan jika melawan player 2, maka player 1 dan player 2 harus meng-inputkan username. Setelah masing-masing player meng-inputkan username, maka ditampilkan juga tanda dari masing-masing player/bot. Adapun sketsa dari tampilan ini sebagai berikut



Gambar 6 sketsa tampilan input nama player

## 2.7. Tampilan Utama Permainan (Cara bermain Tic-tac-toe)

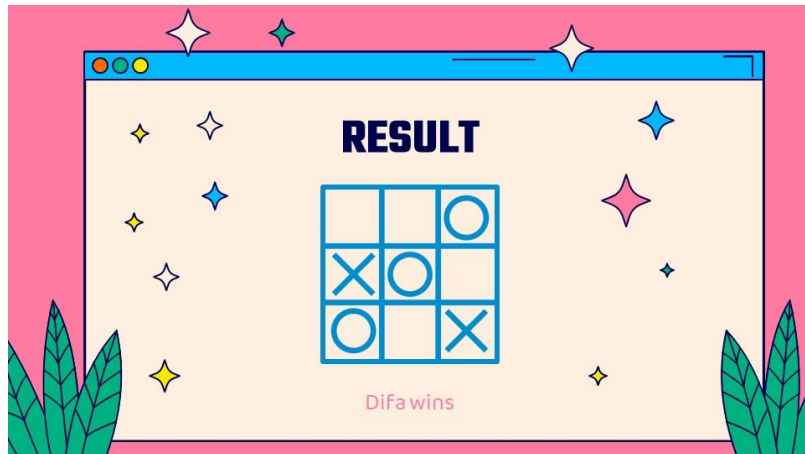
Selanjutnya, player akan ditampilkan dengan gameplay atau tampilan utama permainan. Pada bagian inilah, bot dan player bermain dengan memasukkan koordinat tandanya masing-masing. Dan disini juga terdapat keterangan turn/giliran player secara bergantian.



Gambar 7 sketsa tampilan permainan utama

## 2.8. Tampilan Hasil Permainan

Setelah permainan selesai, maka player akan ditampilkan dengan tampilan hasil permainan. Pada tampilan ini, terdapat nama player1/player2 atau komputer sebagai pemenangnya. Adapun sketsanya sebagai berikut.



Gambar 8 sketsa tampilan hasil permainan

## 2.9. Tampilan Menu Help

Pada main menu, terdapat berbagai pilihan, salah satunya menu help. Pada menu ini ditampilkan cara bermain dan juga penjelasan dari setiap menu yang ada pada permainan ini dengan bahasa Indonesia maupun bahasa Inggris. Adapun sketsanya sebagai berikut.



Gambar 9 sketsa menu help

## 2.9. Tampilan Color Theme

Terdapat juga menu color theme pada main menu yang menampilkan pilihan warna beserta kodenya. Warna yang dipilih bisa untuk warna teks maupun warna background. Adapun sketsanya sebagai berikut.



Gambar 10 sketsa menu help

## 2.9. Tampilan Credit

Ada juga menu credit pada main menu yang menampilkan identitas developer dari permainan ini. Adapun sketsanya sebagai berikut.



Gambar 11 menu credit

## 2.9. Tampilan Highscore

Pada main menu, terdapat juga menu highscore yang menampilkan riwayat kemenangan player dalam melawan bot. Adapun sketsanya sebagai berikut.



Gambar 12 sketsa menu highscore

Seluruh sketsa skenario bisa dicek di :

[https://drive.google.com/drive/folders/1rJ1dUGtM1Q13ePEDBy6vf1BwWXQc1ND\\_?usp=sharing](https://drive.google.com/drive/folders/1rJ1dUGtM1Q13ePEDBy6vf1BwWXQc1ND_?usp=sharing)

### 3. Tingkat Kesulitan

*Difficulty level* atau tingkat kesulitan dari bot pada permainan ini terdapat 3 tingkat yaitu easy, normal, dan hard. Dari masing-masing level ini, player dapat memilih 1 level komputer yang akan menjadi lawannya.

### 4. Aturan Bermain

- Player1 mendapatkan selalu mendapatkan first move dan juga tanda 'X'. Sedangkan bot/player 2 selalu mendapatkan tanda 'O' dan mulai menandai dari giliran yang kedua
- Setiap giliran/turn, peng-inputan tanda 'X' ataupun 'O' untuk player1 maupun player2 akan diwaktu sebanyak 10 detik. Jika lebih dari itu, player1 ataupun player2 tidak bisa meng-input tandanya masing-masing
- Setelah player1 menempatkan tanda 'X' yang pertama, maka giliran bot/player2 untuk menempatkan tanda O secara acak. Begitu seterusnya secara bergantian.
- Diantara player1, player2 maupun bot, yang pertama kali membentuk 3 tandanya berturut-turut pada grid 3x3, 4 tandanya berturut-turut pada grid 5x5, dan 5 tandanya berturut-turut pada grid 7x7 adalah pemenang
- Setiap kemenangan player melawan bot, riwayat kemenangannya akan tersimpan pada menu highscore yang bisa dilihat setelah permainan selesai

### 5. Jumlah Player

Dalam game ini hanya terdapat 2 mode permainan yaitu single player dan juga multi player. Pada single player, pemain akan melawan bot secara bergantian, sedangkan pada multi player, pemain akan melawan player 2 secara bergantian pula



## 6. Karakteristik Computer

Pergerakan dari masing-masing bot, terdapat beberapa karakteristik berbeda bergantung pada tingkat kesulitan dan ukuran grid yang dipilih. Berikut tabel ilustrasi karakteristik yang dibuat dimana setiap nomor menunjukkan prioritas/urutan :

Level/Ukuran	Grid 3x3	Grid 5x5	Grid 7x7
<b>Easy</b>	<ol style="list-style-type: none"><li>1. Blocking untuk 2 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li><li>2. Mengisi random di sekitar tanda 'X'</li><li>3. Mengisi random yang masih kosong</li></ol>	<ol style="list-style-type: none"><li>1. Mengisi random di sekitar tanda 'X'</li><li>2. Mengisi random yang masih kosong</li></ol>	<ol style="list-style-type: none"><li>1. Blocking untuk 4 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li><li>2. Mengisi random di sekitar tanda 'X'</li><li>3. Mengisi random yang masih kosong</li></ol>
<b>Normal</b>	<ol style="list-style-type: none"><li>1. Blocking untuk 2 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li><li>2. Pemantauan pergerakan pertama dari lawan</li><li>3. Mengisi random di sekitar tanda 'X'</li><li>4. Mengisi random yang masih kosong</li></ol>	<ol style="list-style-type: none"><li>1. Blocking untuk 3 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li><li>2. Mengisi random di sekitar tanda 'X'</li><li>3. Mengisi random yang masih kosong</li></ol>	<ol style="list-style-type: none"><li>1. Blocking untuk 4 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li><li>2. Blocking untuk 3 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li><li>3. Mengisi random di sekitar tanda 'X'</li><li>4. Mengisi random yang masih kosong</li></ol>

<b>Hard</b>	<ol style="list-style-type: none"> <li>1. Blocking untuk 2 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li> <li>2. Pemantauan pergerakan pertama dari lawan</li> <li>3. Pemantauan pergerakan kedua dari lawan</li> <li>4. Mengisi random di sekitar tanda 'X'</li> <li>5. Mengisi random yang masih kosong</li> </ol>	<ol style="list-style-type: none"> <li>1. Blocking untuk 3 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li> <li>2. Blocking untuk 2 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li> <li>3. Mengisi random di sekitar tanda 'X'</li> <li>4. Mengisi random yang masih kosong</li> </ol>	<ol style="list-style-type: none"> <li>1. Blocking untuk 4 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li> <li>2. Blocking untuk 3 tanda 'X' berturut-turut secara baris, kolom dan diagonal</li> <li>3. Blocking untuk 2 tanda 'X' berturut-turut secara baris dan kolom saja</li> <li>4. Mengisi random di sekitar tanda 'X'</li> <li>5. Mengisi random yang masih kosong</li> </ol>
<b>Keterangan :</b> Pengurutan nomor diatas diurutkan berdasarkan prioritas dengan nomor 1 paling priortias			

## 6. Inisialisasi Program

- a. Matrix[8][8] => Seluruh isi array diisi dengan ' '.
- b. Player => 1 // sebagai penanda giliran pada setiap giliran
- c. p1, p2 => Undefined // sebagai nama player pada permainan
- d. done => ' ' // sebagai penanda lanjut atau berhentinya permainan

## BAB 2

### “Desain dan Perancangan”

#### 1. Kebutuhan Data (struktur data)

##### a. Data Player

Data ini bertujuan untuk menampung segala kebutuhan yang dibutuhkan pada permainan seperti ukuran grid, namaPlayer, levelComputer, dan waktu. Data ini memiliki tipe struct yang di dalamnya memiliki 4 field yang terdiri dari grid sebagai penampung pilihan ukuran grid, namePlayer sebagai penampung nama player, levelComputer sebagai penampung pilihan level kesulitan bot, dan time sebagai penampung waktu untuk highscore.

Struktur Data :

```
typedef struct data{  
    int grid; //menampung pilihan ukuran grid  
    char namePlayer[255]; //menampung nama player  
    int levelComputer; //menampung pilihan level kesulitan bot  
    char time[100]; //menampung waktu untuk highscore  
};
```

##### b. Data Papan Permainan

Data ini bertujuan untuk menampilkan papan permainan dengan tiga pilihan ukuran yaitu 3x3, 5x5, dan 7x7. Adapun papan permainan dibentuk dengan menggunakan tabel sehingga tampilan permainan akan terlihat lebih rapih. Data ini terdiri dari 1 data bertipe array 2 dimensi bertipe karakter dengan ukuran 8x8 dan 1 data bertipe integer sebagai pilihan user.

Struktur Data :

```
char matrix[8][8]; //menampilkan papan permainan  
int grid; //menampung pilihan ukuran papan permainan
```

##### c. Data Mode Permainan

Data ini bertujuan untuk menampung pilihan mode permainan. Player bisa memilih akan melawan player lagi atau komputer. Data ini terdiri dari 1 data bertipe integer sebagai penampung pilihan user untuk mode permainan.

Struktur Data :

```
int against; //menampung pilihan mode permainan
```

##### d. Data Level Computer

Data ini bertujuan untuk menampung pilihan level komputer oleh player. Data ini terdiri dari 1 data bertipe integer sebagai penampung pilihan user untuk level komputer.

**Struktur Data :**

```
int levelComputer; //menampung pilihan level komputer
```

#### **e. Data Pergerakan Posisi Player/Bot**

Data ini bertujuan untuk menampung posisi koordinat pergerakan permainan yang kemudian posisi ini menjadi index untuk array 2 dimensi dari papan permainan. Data ini terdiri dari 2 data bertipe integer sebagai posisi koordinat pergerakan permainan.

**Struktur Data :**

```
int x; //menampung pilihan posisi koordinat-x
```

```
int y; //menampung pilihan posisi koordinat-y
```

#### **f. Data ColorTheme**

Data ini bertujuan untuk pilihan kode warna permainan yang dipilih player. Data ini terdiri dari 1 data bertipe integer sebagai kode warna permainan.

**Struktur Data :**

```
int kodeWarna; //menampung pilihan kode warna permainan
```

#### **g. Data Pilihan Main Menu**

Data ini bertujuan untuk menampung pilihan yang ada di prosedur mainMenu. Data ini terdiri dari 1 data bertipe integer sebagai pilihan yang ada di prosedur mainMenu dan .

**Struktur Data :**

```
int choiceMainMenu; //menampung pilihan mainMenu
```

#### **h. Data Lanjut atau Berakhirnya Permainan**

Data ini bertujuan untuk menampung hasil keluaran dari function check. Data ini terdiri dari 1 data bertipe character dengan nilai keluaran 'X' maka player 1 menang, nilai keluaran 'O' maka player 2 atau komputer yang menang, dan nilai keluaran ' ' maka permainan tetapi lanjut

**Struktur Data :**

```
char done; //menampung nilai keluaran dari function check
```

#### **i. Data Giliran Player**

Data ini bertujuan untuk menampung giliran player apakah giliran player 1 atau player 2. Data ini terdiri dari 1 data bertipe integer dengan player = 1, maka giliran player 1, dan player = 2, maka giliran player 2

**Struktur Data :**

```
int player; //menampung giliran player
```

**j. Data Highscore**

Data ini bertujuan untuk menampung hal terkait yang ada pada menu highscore. Data ini memiliki tipe struct yang di dalamnya memiliki 4 field yang terdiri dari grid sebagai penampung pilihan ukuran grid, namePlayer sebagai penampung nama player, levelComputer sebagai penampung pilihan level kesulitan bot, dan time sebagai penampung waktu untuk highscore. Masing-masing field adalah penampung dari struct dataPemain sebelumnya untuk ditampilkan di menu highscore.

**Struktur Data :**

```
typedef struct highscore{  
    int grid; //menampung pilihan ukuran grid  
    char namePlayer[255]; //menampung nama player  
    int levelComputer; //menampung pilihan level kesulitan bot  
    char time[100]; //menampung waktu untuk highscore  
}Record;
```

**k. Data Pemilihan Bahasa Pada Menu Help**

Data ini bertujuan untuk pilihan bahasa dalam menu help. Data ini terdiri dari hanya 1 data yang bertipe integer. Player dapat memilih 2 pilihan, pilihan pertama untuk bahasa Indonesia dan pilihan kedua untuk bahasa inggris

**Struktur Data :**

```
int chooseLanguage; //menampung pilihan bahasa pada menu help
```

**2. Modul program**

Bagian ini menjelaskan semua modul yang digunakan dalam program TicTacToe. Modul-modul tersebut dapat berupa function atau prosedur. Tabel ini adalah uraian mengenai modul-modul yang terkait :

No	Nama Modul	Jenis	Keterangan	PJ
----	------------	-------	------------	----

1	main	Function	Program utama yang menjalankan keseluruhan program	Difa Mochammad Fadillah
2	mainMenu	Prosedur	Menampilkan pilihan Start, Help, Color Theme, Credit, Highscore, dan Exit.	M Azhar Alauddin
3	menuHelp	Prosedur	Menampilkan cara bermain, aturan bermain dan deskripsi singkat permainan	M Azhar Alauddin
4	showColorTheme	Prosedur	Menampilkan warna background dan teks aplikasi.	M Azhar Alauddin
5	chooseColorTheme	Prosedur	Memilih warna background dan teks aplikasi.	M Azhar Alauddin
6	credit	Prosedur	Menampilkan identitas developer dan creator	M Azhar Alauddin
7	introduction	Prosedur	Menampilkan introduction atau header	M Azhar Alauddin
8	disp_matrix_3X3	Prosedur	Menampilkan board game berukuran 3x3.	M Azhar Alauddin
9	disp_matrix_5X5	Prosedur	Menampilkan board game berukuran 5x5.	M Azhar Alauddin
10	disp_matrix_7X7	Prosedur	Menampilkan board game berukuran 7x7.	M Azhar Alauddin
11	init_matrix	Prosedur	Menginisialisasi matrix pada grid dengan ' '.	M Azhar Alauddin
12	player1_move	Prosedur	Penempatan tanda 'X' untuk player 1 pada setiap turn	M Azhar Alauddin
13	player2_move	Prosedur	Penempatan tanda 'O' untuk player 1 pada setiap turn	M Azhar Alauddin
14	computer_1_move_3X3	Prosedur	Penempatan tanda 'O' untuk komputer dengan	M Azhar Alauddin

			easy level pada ukuran 3x3	
15	computer_2_move_3X3	Prosedur	Penempatan tanda 'O' untuk komputer dengan normal level pada ukuran 3x3	M Azhar Alauddin
16	computer_3_move_3X3	Prosedur	Penempatan tanda 'O' untuk komputer dengan hard level pada ukuran 3x3	M Azhar Alauddin
17	computer_1_move_5X5	Prosedur	Penempatan tanda 'O' untuk komputer dengan easy level pada ukuran 5x5	M Azhar Alauddin
18	computer_2_move_5X5	Prosedur	Penempatan tanda 'O' untuk komputer dengan normal level pada ukuran 5x5	M Azhar Alauddin
19	computer_3_move_5X5	Prosedur	Penempatan tanda 'O' untuk komputer dengan hard level pada ukuran 5x5	M Azhar Alauddin
20	computer_1_move_7X7	Prosedur	Penempatan tanda 'O' untuk komputer dengan easy level pada ukuran 7x7	M Azhar Alauddin
21	computer_2_move_7X7	Prosedur	Penempatan tanda 'O' untuk komputer dengan normal level pada ukuran 7x7	M Azhar Alauddin
22	computer_3_move_7X7	Prosedur	Penempatan tanda 'O' untuk komputer dengan hard level pada ukuran 7x7	M Azhar Alauddin
23	check_3X3	Function	Menentukan kondisi Win/ Lose/ Draw pada ukuran 3x3 baik itu secara baris, kolom maupun diagonal	M Azhar Alauddin
24	check_5X5	Function	Menentukan kondisi Win/ Lose/ Draw pada	M Azhar Alauddin

			ukuran 5x5 baik itu secara baris, kolom maupun diagonal	
25	check_7X7	Function	Menentukan kondisi Win/ Lose/ Draw pada ukuran 7x7 baik itu secara baris, kolom maupun diagonal	M Azhar Alauddin
26	readHighscore	Function	Menampilkan file riwayat kemenangan player dalam melawan bot	M Azhar Alauddin
27	saveHighscore	Function	Menyimpan riwayat kemenangan dalam file	M Azhar Alauddin

### 3. Spesifikasi Modul

Function main()	
Deskripsi	Program utama yang menjalankan program dari awal hingga akhir
Initial State	Program belum berjalan
Final State	Keluar dari program
Modul Dipanggil	mainMenu, menuHelp, showColorTheme, chooseColorTheme, credit, introduction, disp_matrix_3X3, disp_matrix_5X5, disp_matrix_7X7, init_matrix, player1_move, player2_move, computer_1_move_3X3, computer_2_move_3X3, computer_3_move_3X3, computer_1_move_5X5, computer_2_move_5X5, computer_3_move_5X5, computer_1_move_7X7, computer_2_move_7X7, computer_3_move_7X7, check_3X3, check_5X5, check_7X7, readHighscore, saveHighscore
Author	Difa Mochammad Fadillah
Kamus Data	i : integer player : integer against : integer choiceMainMenu : integer



	kodeWarna : integer chooseLanguage : integer done : character p1[8] : Array of character [1..255] p2[8] : Array of character [1..255]
<b>Algoritma</b>	
lakukan tampilkan mainMenu jika choiceMainMenu = 1 maka mulai permainan jika choiceMainMenu = 2 maka menuHelp( ) jika choiceMainMenu = 3 maka chooseColorTheme( ) showColorTheme( ) jika choiceMainMenu = 4 maka credit( ) jika choiceMainMenu = 5 maka readHighscore( ) return 0	

<b>Prosedur mainMenu()</b>	
Deskripsi	Menampilkan pilihan Start, Help, Color Theme, Credit, Highscore, dan Exit
Initial State	Belum ada yang ditampilkan
Final State	Sudah ditampilkan pilihan Start, Help, Color Theme, Credit, Highscore, Exit
Modul Pemanggil	main
Author	Muhammad Azhar Alauddin
<b>Algoritma</b>	
lakukan tampilkan pilihan main menu masukkan pilihan main menu	

<b>Prosedur menuHelp( )</b>
-----------------------------

Deskripsi	Menampilkan Deskripsi singkat terkait dengan permainan Tic Tac Toe, cara bermain, dan aturan bermain
Initial State	Belum ada yang ditampilkan
Final State	Sudah ditampilkan Deskripsi singkat terkait dengan permainan Tic Tac Toe, cara bermain, dan aturan bermain
Modul Pemanggil	main
Author	Muhammad Azhar Alauddin
Algoritma	
lakukan tampilkan deskripsi singkat permainan tampilkan cara bermain tampilkan aturan bermain	

<b>Prosedur chooseColorTheme( )</b>	
Deskripsi	Memilih warna background dan teks aplikasi.
Initial State	Belum ada yang ditampilkan
Final State	Sudah ditampilkan pilihan warna background dan warna teks sesuai dengan pilihan yang diinput
Modul Pemanggil	main
Author	Muhammad Azhar Alauddin
<b>Algoritma</b>	
lakukan tampilkan pilihan warna background tampilkan pilihan warna text masukkan kode warna	

<b>Prosedur showColorTheme( )</b>	
Deskripsi	Menampilkan warna background dan warna teks
Initial State	Belum ada yang ditampilkan
Final State	Sudah ditampilkan warna background dan warna teks sesuai dengan pilihan yang diinput
Modul Pemanggil	main
Author	Muhammad Azhar Alauddin
Parameter input	kode warna

<b>Prosedur credit( )</b>	
Deskripsi	Menampilkan identitas developer dan creator dari permainan ini
Initial State	Belum ada yang ditampilkan
Final State	Sudah ditampilkan identitas developer dan creator dari permainan ini
Modul Pemanggil	main
Author	Muhammad Azhar Alauddin

<b>Prosedur introduction( )</b>	
Deskripsi	Menampilkan introduction atau header
Initial State	Belum ada yang ditampilkan
Final State	Sudah ditampilkan introduction atau header
Modul Pemanggil	main
Author	Muhammad Azhar Alauddin

<b>Prosedur disp_matrix_3X3( )</b>	
Deskripsi	Menampilkan board game berukuran 3x3.
Initial State	Belum ada yang ditampilkan
Final State	Sudah ditampilkan board game berukuran 3x3.
Modul Pemanggil	main
Author	Muhammad Azhar Alauddin

<b>Prosedur disp_matrix_5X5( )</b>	
Deskripsi	Menampilkan board game berukuran 3x3.
Initial State	Belum ada yang ditampilkan
Final State	Sudah ditampilkan board game berukuran 3x3.
Modul Pemanggil	main
Author	Muhammad Azhar Alauddin

<b>Prosedur disp_matrix_7X7( )</b>	
Deskripsi	Menampilkan board game berukuran 7x7.
Initial State	Belum ada yang ditampilkan
Final State	Sudah ditampilkan board game berukuran 7x7.
Modul Pemanggil	main
Author	Muhammad Azhar Alauddin

<b>Prosedur init_matrix( )</b>	
Deskripsi	Menginisialisasi matrix pada grid dengan ‘ ‘
Initial State	Isi keseluruhan matrix belum diisi apapun
Final State	Isi keseluruhan matrix sudah diisi ‘ ‘
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin

<b>Prosedur player1_move( )</b>	
Deskripsi	Penempatan tanda ‘X’ untuk player 1 pada setiap turn. Pada penempatan ini, player 1 diberikan waktu selama 10 detik. Jika lebih dari itu, maka player 1 tetap bisa mengisi tanda tetapi tidak akan tersimpan pada board game
Initial State	Belum ada penempatan tanda ‘X’ pada turn tertentu
Final State	Sudah ada penempatan tanda ‘X’ pada turn tertentu
Modul Pemanggil	main
Kamus Data	waktuAwal : integer waktuAkhir : integer x : integer y : integer
Author	Muhammad Azhar Alauddin

Algoritma
<p>lakukan</p> <p>    waktuAwal = clock( )</p> <p>    waktuAkhir = clock( )</p> <p>    masukkan titik koordinat x dan y untuk pergerakan</p> <p>    jika waktuAkhir-waktuAwal &gt;= 10000 <b>maka</b></p> <p>        kembali ke modul pemanggil</p> <p>    jika matrix[--x][--y] tidak sama dengan spasi <b>maka</b></p> <p>        player1_move( )</p> <p>    selain itu <b>maka</b></p> <p>        matrix[x][y] = 'X'</p>

Prosedur player2_move( )	
Deskripsi	Penempatan tanda 'O' untuk player 2 pada setiap turn. Pada penempatan ini, player 2 diberikan waktu selama 10 detik. Jika lebih dari itu, maka player 2 tetap bisa mengisi tanda tetapi tidak akan tersimpan pada board game
Initial State	Belum ada penempatan tanda 'O' pada turn tertentu
Final State	Sudah ada penempatan tanda 'O' pada turn tertentu
Modul Pemanggil	main
Kamus Data	waktuAwal : integer waktuAkhir : integer x : integer y : integer
Author	Muhammad Azhar Alauddin
Algoritma	
<p>lakukan</p> <p>    waktuAwal = clock( )</p> <p>    waktuAkhir = clock( )</p> <p>    masukkan titik koordinat x dan y untuk pergerakan</p> <p>    jika waktuAkhir-waktuAwal &gt;= 10000 <b>maka</b></p> <p>        kembali ke modul pemanggil</p> <p>    jika matrix[--x][--y] tidak sama dengan spasi <b>maka</b></p> <p>        player1_move( )</p> <p>    selain itu <b>maka</b></p>	

$\text{matrix}[x][y] = 'O'$

#### Prosedur computer\_move\_1\_3X3

Deskripsi	Penempatan tanda 'O' untuk computer pada setiap turn pada grid 3x3 dengan easy level
Initial State	Belum ada penempatan tanda 'O' pada turn tertentu
Final State	Sudah ada penempatan tanda 'O' pada turn tertentu
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin
Algoritma	
<b>lakukan</b> jika tanda 'X' sudah 2 kali berturut-turut <b>maka</b> blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal selain itu <b>maka</b> Isi secara acak di sekitar tanda 'X' yang sudah ada selain itu <b>maka</b> Isi secara acak yang masih kosong	

#### Prosedur computer\_move\_1\_5X5

Deskripsi	Penempatan tanda 'O' untuk computer pada setiap turn pada grid 5x5 dengan easy level
Initial State	Belum ada penempatan tanda 'O' pada turn tertentu
Final State	Sudah ada penempatan tanda 'O' pada turn tertentu
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin
Algoritma	
<b>lakukan</b> jika dalam board ada tanda 'X' <b>maka</b>	

Isi secara acak di sekitar tanda 'X' yang sudah ada  
selain itu **maka**  
Isi secara acak yang masih kosong

#### Prosedur computer\_move\_1\_7X7

Deskripsi	Penempatan tanda 'O' untuk computer pada setiap turn pada grid 7x7 dengan easy level
Initial State	Belum ada penempatan tanda 'O' pada turn tertentu
Final State	Sudah ada penempatan tanda 'O' pada turn tertentu
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin

#### Algoritma

##### **lakukan**

jika tanda 'X' sudah 4 kali berturut-turut **maka**  
blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal  
selain itu **maka**  
Isi secara acak di sekitar tanda 'X' yang sudah ada  
selain itu **maka**  
Isi secara acak yang masih kosong

#### Prosedur computer\_move\_2\_3x3

Deskripsi	Penempatan tanda 'O' untuk computer pada setiap turn pada grid 3x3 dengan normal level
Initial State	Belum ada penempatan tanda 'O' pada turn tertentu
Final State	Sudah ada penempatan tanda 'O' pada turn tertentu
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin

#### Algoritma

##### **lakukan**

jika tanda 'X' sudah 2 kali berturut-turut **maka**



blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal selain itu **maka**  
 pantau pergerakan pertama dari player 1 selain itu **maka**  
 Isi secara acak di sekitar tanda 'X' yang sudah ada selain itu **maka**  
 Isi secara acak yang masih kosong

#### Prosedur computer\_move\_2\_5x5

Deskripsi	Penempatan tanda 'O' untuk computer pada setiap turn pada grid 5x5 dengan normal level
Initial State	Belum ada penempatan tanda 'O' pada turn tertentu
Final State	Sudah ada penempatan tanda 'O' pada turn tertentu
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin
Algoritma	
<b>lakukan</b> jika tanda 'X' sudah 3 kali berturut-turut <b>maka</b> blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal selain itu <b>maka</b> Isi secara acak di sekitar tanda 'X' yang sudah ada selain itu <b>maka</b> Isi secara acak yang masih kosong	

#### Prosedur computer\_move\_2\_7x7

Deskripsi	Penempatan tanda 'O' untuk computer pada setiap turn pada grid 7x7 dengan normal level
Initial State	Belum ada penempatan tanda 'O' pada turn tertentu
Final State	Sudah ada penempatan tanda 'O' pada turn tertentu
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin

Algoritma
<p><b>lakukan</b></p> <p>    jika tanda 'X' sudah 4 kali berturut-turut <b>maka</b></p> <p>        blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal</p> <p>    jika tanda 'X' sudah 3 kali berturut-turut <b>maka</b></p> <p>        blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal</p> <p>    selain itu <b>maka</b></p> <p>        Isi secara acak di sekitar tanda 'X' yang sudah ada</p> <p>    selain itu <b>maka</b></p> <p>        Isi secara acak yang masih kosong</p>

Prosedur computer_move_3_3x3	
Deskripsi	Penempatan tanda 'O' untuk computer pada setiap turn pada grid 3x3 dengan hard level
Initial State	Belum ada penempatan tanda 'O' pada turn tertentu
Final State	Sudah ada penempatan tanda 'O' pada turn tertentu
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin
Algoritma	
<p><b>lakukan</b></p> <p>    jika tanda 'X' sudah 2 kali berturut-turut <b>maka</b></p> <p>        blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal</p> <p>    selain itu <b>maka</b></p> <p>        pantau pergerakan pertama dari player 1</p> <p>    selain itu <b>maka</b></p> <p>        pantau pergerakan kedua dari player 1</p> <p>    selain itu <b>maka</b></p> <p>        Isi secara acak di sekitar tanda 'X' yang sudah ada</p> <p>    selain itu <b>maka</b></p> <p>        Isi secara acak yang masih kosong</p>	

Prosedur computer_move_3_5x5	
Deskripsi	Penempatan tanda 'O' untuk computer pada setiap turn pada grid 5x5 dengan hard level

Initial State	Belum ada penempatan tanda 'O' pada turn tertentu
Final State	Sudah ada penempatan tanda 'O' pada turn tertentu
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin
Algoritma	
<b>lakukan</b> jika tanda 'X' sudah 3 kali berturut-turut <b>maka</b> blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal jika tanda 'X' sudah 2 kali berturut-turut <b>maka</b> blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal selain itu <b>maka</b> Isi secara acak di sekitar tanda 'X' yang sudah ada selain itu <b>maka</b> Isi secara acak yang masih kosong	

Prosedur computer_move_3_7x7	
Deskripsi	Penempatan tanda 'O' untuk computer pada setiap turn pada grid 7x7 dengan hard level
Initial State	Belum ada penempatan tanda 'O' pada turn tertentu
Final State	Sudah ada penempatan tanda 'O' pada turn tertentu
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin
Algoritma	
<b>lakukan</b> jika tanda 'X' sudah 4 kali berturut-turut <b>maka</b> blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal jika tanda 'X' sudah 3 kali berturut-turut <b>maka</b> blocking pergerakan tersebut baik itu secara baris, kolom ataupun diagonal jika tanda 'X' sudah 2 kali berturut-turut <b>maka</b> blocking pergerakan tersebut baik itu secara baris ataupun kolom selain itu <b>maka</b> Isi secara acak di sekitar tanda 'X' yang sudah ada selain itu <b>maka</b>	

Isi secara acak yang masih kosong

Function check_3X3( )	
Deskripsi	Menentukan kondisi Win/ Lose/ Draw pada ukuran 3x3 baik itu secara baris, kolom maupun diagonal
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin
Algoritma	
<b>lakukan</b> jika tanda 'X' sudah 3 kali berturut-turut baik itu secara baris, kolom ataupun diagonal <b>maka</b> kembalikan 'X' ke modul pemanggil jika tanda 'O' sudah 3 kali berturut-turut baik itu secara baris, kolom ataupun diagonal <b>maka</b> kembalikan 'O' ke modul pemanggil	

Function check_5X5( )	
Deskripsi	Menentukan kondisi Win/ Lose/ Draw pada ukuran 5x5 baik itu secara baris, kolom maupun diagonal
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin
Algoritma	
<b>lakukan</b> jika tanda 'X' sudah 4 kali berturut-turut baik itu secara baris, kolom ataupun diagonal <b>maka</b> kembalikan 'X' ke modul pemanggil jika tanda 'O' sudah 4 kali berturut-turut baik itu secara baris, kolom ataupun diagonal <b>maka</b> kembalikan 'O' ke modul pemanggil	

Function check_7X7( )	
Deskripsi	Menentukan kondisi Win/ Lose/ Draw pada ukuran 7x7 baik itu secara baris, kolom maupun diagonal
Modul Pemanggil	main
Kamus Data	i : integer j : integer
Author	Muhammad Azhar Alauddin
Algoritma	
<p><b>lakukan</b>          jika tanda 'X' sudah 5 kali berturut-turut baik itu secara baris, kolom ataupun diagonal <b>maka</b>              kembalikan 'X' ke modul pemanggil          jika tanda 'O' sudah 5 kali berturut-turut baik itu secara baris, kolom ataupun diagonal <b>maka</b>              kembalikan 'O' ke modul pemanggil</p>	

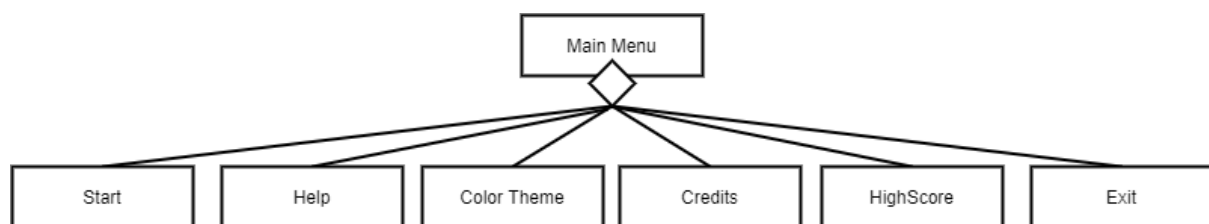
Prosedur saveHighscore( )	
Deskripsi	Menyimpan riwayat kemenangan melawan bot dalam file
Initial State	Riwayat kemenangan belum tersimpan dalam file
Final State	Riwayat kemenangan sudah tersimpan dalam file
Modul Pemanggil	main
Kamus Data	
Author	Muhammad Azhar Alauddin (credit : Rangga Yudha Yudistira)
Algoritma	

Prosedur readHighscore( )	
Deskripsi	Menampilkan file riwayat kemenangan player dalam melawan bot
Initial State	Riwayat kemenangan belum ditampilkan di implementasi program

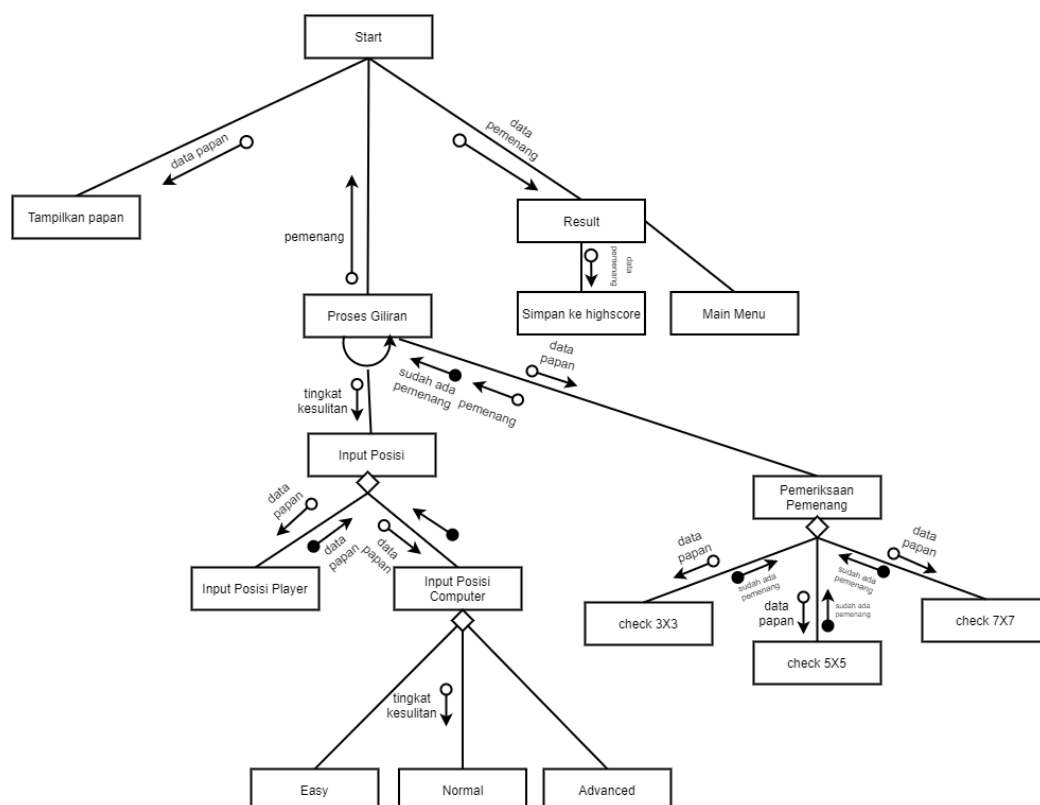
Final State	Riwayat kemenangan sudah ditampilkan di implementasi program
Modul Pemanggil	main
Kamus Data	
Author	Muhammad Azhar Alauddin (credit : Ranga Yudha Yudistira)
Algoritma	

### 3. Structure Chart

Berikut kami ilustrasikan interaksi modul yang sudah kami buat dalam bentuk struktur chart



Gambar 13 structure chart prosedur main menu



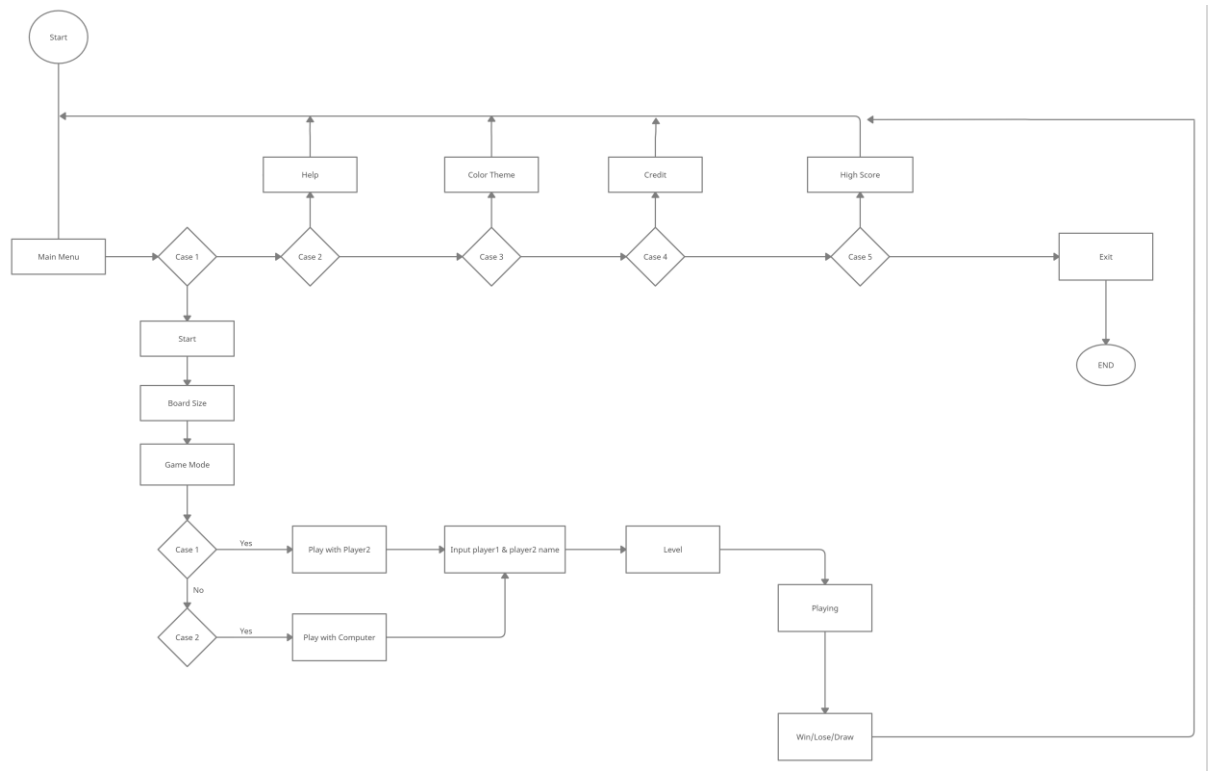
Gambar 14 structure chart gameplay

Selengkapnya :

[https://drive.google.com/drive/folders/1m\\_GFMMMSKfY-eKOILIVprkVNHhuUm-mQa?usp=sharing](https://drive.google.com/drive/folders/1m_GFMMMSKfY-eKOILIVprkVNHhuUm-mQa?usp=sharing)

#### 4. Flowchart Program Utama

Berikut kami ilustrasikan alur program utama yang sudah kami buat dalam bentuk flow chart (alur diagram).



Gambar 15 flow chart program utama

Selengkapnya :

<https://drive.google.com/file/d/127sdFnGZ4WdYJh0mmxy5PBnLBjBLhNKN/view?usp=sharing>

## BAB IV : HASIL AKHIR PROGRAM

### 1. Link github

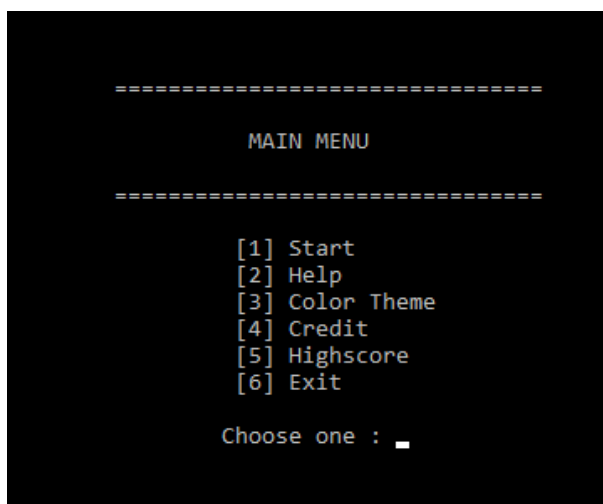
<https://github.com/zharmedia/Tugas-Besar-Tic-Tac-Toe.git>

### 2. Interface dan Data Testing

- Pre-Game

- a) Tampilan Menu Utama

Pada tampilan main menu, player akan ditampilkan dengan 5 pilihan yaitu start, help, colortheme, credit, highscore, dan exit. Jika player memilih 1, maka player akan ditampilkan ke gameplay utama seperti yang ada pada gambar 27, jika player memilih 2 maka player akan ditampilkan ke menu help seperti yang ada pada gambar 21, jika player memilih 3 maka player akan ditampilkan ke colortheme seperti yang ada pada gambar 24, jika player memilih 4 maka player akan ditampilkan ke credit seperti yang ada pada gambar 25, jika player memilih 5 maka player akan ditampilkan ke highscore seperti yang ada pada gambar 26, dan jika player memilih 6 maka player akan keluar dari permainan. Sedangkan, jika player memilih angka di luar dari pilihan tersebut maka player akan diminta dan terus diminta untuk memilih pilihan yang tersedia (OKE)



Gambar 16 Hasil implementasi tampilan utama



#### **b) Tampilan Memilih Ukuran Grid**

Setelah player memilih 1 pada menu utama, maka player akan ditampilkan pada tampilan untuk memilih ukuran grid dimana disini player bisa memilih 3 untuk ukuran grid 3x3, memilih 5 untuk grid ukuran 5x5, dan memilih 7 untuk grid ukuran 7x7. (OKE)

```
*** TIC TAC TOE GAME ***

## Halo! Selamat datang di game ini
## Berikut ada beberapa hal yang harus anda penuhi

Board Game
Press 3 and Enter for [3X3]
Press 5 and Enter for [5X5]
Press 7 and Enter for [7X7]

Choose one :
```

Gambar 17 Hasil implementasi tampilan memilih ukuran grid

#### **c) Tampilan Memilih Lawan**

Setelah memilih ukuran grid, maka player akan ditampilkan pada tampilan mode permainan atau dengan siapa player akan main. Disini terdapat 2 pilihan, yaitu 1 untuk bermain bersama bot dan 2 untuk bermain bersama player 2. (OKE)

```
*** TIC TAC TOE GAME ***

## Halo! Selamat datang di game ini
## Berikut ada beberapa hal yang harus anda penuhi

Whom do you wanna play?
Press 1 and Enter for playing with computer
Press 2 and Enter for playing with player2

Choose one : _
```

Gambar 18 Hasil implementasi tampilan memilih lawan

#### **d) Tampilan Menginput Nama Player**

Setelah menginput mode permainan, player akan ditampilkan pada tampilan yang mana player harus menginput namanya dengan maksimal 8 huruf. Dan jika player sebelumnya memilih player 2 sebagai lawan, maka player 2 pun harus menginput namanya. Nama disini diperuntukkan ketika player mulai bermain dan juga setelah bermain. (OKE)

```
*** TIC TAC TOE GAME ***

## Halo! Selamat datang di game ini
## Berikut ada beberapa hal yang harus anda penuhi

Enter player's name : Azhar
```

Gambar 19 Hasil implementasi tampilan input nama player

#### e) Tampilan Memilih Level Kesulitan

Setelah player menginput nama, maka player harus memilih level kesulitan dari komputer. Disini terdapat 3 level kesulitan, yaitu easy, normal dan hard. Jika player memilih 1 maka player akan melawan komputer dengan level easy. Jika player memilih 2 maka player akan melawan komputer dengan level normal. Dan jika player memilih 3 maka player akan memilih komputer dengan level hard.(OKE)

```
*** TIC TAC TOE GAME ***

## Halo! Selamat datang di game ini
## Berikut ada beberapa hal yang harus anda penuhi

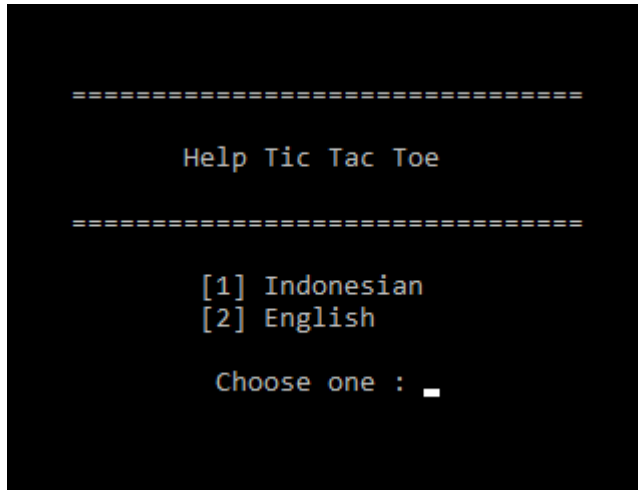
Enter level komputer :
[1] Easy
[2] Normal
[3] Hard

Choose one : 3
```

Gambar 20 Hasil implementasi tampilan level kesulitan

#### f) Tampilan Help 1.0

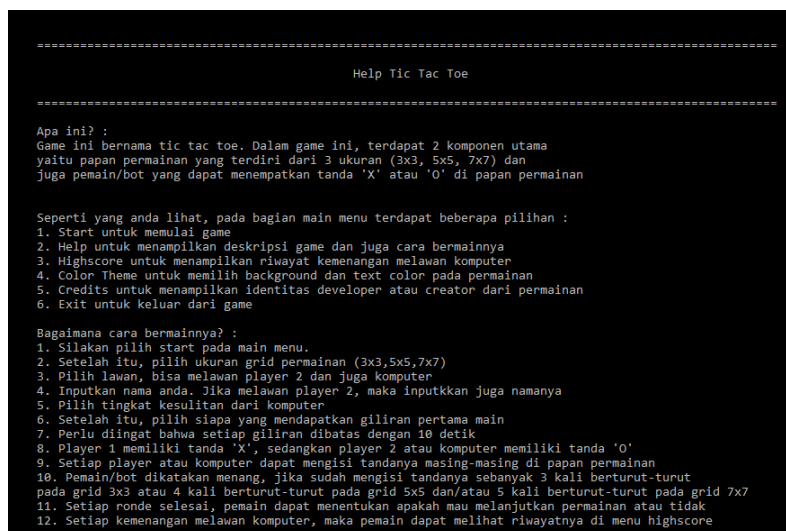
Jika player memilih 2 pada main menu, maka player akan ditampilkan pada tampilan menu help dimana pada menu help ini terdapat penjelasan terkait dengan cara bermain, aturan bermain dan juga deskripsi singkat. Dan pada tampilan ini, terdapat 2 bahasa, jika player memilih 1 maka player akan ditampilkan penjelasan terkait cara bermain, aturan bermain dan juga deskripsi singkat dalam bahasa Indonesia, dan jika player memilih 2 maka player akan ditampilkan penjelasan terkait cara bermain, aturan bermain dan juga deskripsi singkat dalam bahasa Inggris. (OKE)



Gambar 21 Hasil implementasi tampilan Help 1.0 (Memilih bahasa)

### g) Tampilan Help 1.1

Berikut tampilan penjelasan terkait cara bermain, aturan bermain dan juga deskripsi singkat dalam bahasa Indonesia. Tampilan ini akan muncul jika player memilih 1 pada menu help sebelumnya. (OKE)



Gambar 22 Hasil implementasi tampilan Help 1.1

### h) Tampilan Help 1.2

Berikut tampilan penjelasan terkait cara bermain, aturan bermain dan juga deskripsi singkat dalam bahasa Inggris. Tampilan ini akan muncul jika player memilih 1 pada menu help sebelumnya. (OKE)

```
=====
Help Tic Tac Toe
=====

What is this? :
So, this game is called tic-tac-toe. In this game, there's a board game in the form of grid (3x3, 5x5, 7x7)
and two players, X and O, who take turns marking spaces ' ' in the board game. The player who takes
the turn can mark in the row, column or diagonal side in the board game.

Also, as you can see in main menu, there's a couple of choices : ' ' :
1. Start to start the game.
2. Help to display game descriptions, rules and how to play.
3. Highscore to display the history of the player's victory against the computer.
4. Color Theme to select the display color of this game.
5. Credits for displaying the identity of the developer or creator of this game.
6. Exit to exit the game.

How to play this game? :
1. Select Start.
2. Select the size of the game board (3x3, 5x5, 7x7).
3. Choose who you want to play with (Player/Computer).
4. Enter the player name.
5. Choose a difficulty level.
6. Choose who will take the first turn.
7. Each turn is limited to 10 seconds.
8. You are 'X', your friend (or the computer in this case) is 'O'.
Players take turns putting their marks in empty squares.
9. The first player to get all of her marks in a row (up, down, across, or diagonally) is the winner.
10. When all squares are full, the game is over. If no player has won, the game ends in a tie

Press Esc on your keyboard to main menu.
```

Gambar 23 Hasil implementasi tampilan Help 1.2

## i) Tampilan Color Theme

Jika player memilih 3 pada main menu, maka player akan ditampilkan pada pilihan warna berikut ini, dimana player bisa menentukan warna dari teks maupun background sesuai dengan keinginan. Sebagai contoh, player ingin memilih warna biru sebagai background dan warna merah cerah sebagai teks, maka player bisa menginputkan dengan kode warna 25. (OKE)

```
D:\02 ZHAR'S PROJECT\UTK POLBAN\DDP\Ani Rahmani (Praktik)\TUGAS\Tubes\A8_TicTacToe\TIC TAC TOE.exe
=====
Color Theme
=====

Aturan Perwarna :
[1] Pemain dapat mengatur background color dan juga text color
[2] Untuk mengatur background color, pemain harus menginput kode antara
0-9 dengan rincian warna sebagai berikut :
1 = White
2 = Blue
3 = Gray
[3] Untuk mengatur text color, pemain harus menginput kode antara A-F dengan
rincian warna sebagai berikut :
4 = Light Purple
5 = Light Red
6 = Light Aqua

Contoh :
Pemain ingin mengubah background color menjadi White dan text color menjadi Light Purple
Maka kode yang diinput adalah '14'

Input Kode Warna : 25

Press Esc on your keyboard to main menu.
```

Gambar 24 Hasil implementasi tampilan Color Theme

#### j) Tampilan Credits

Jika pada main menu player memilih 4, maka player akan ditampilkan pada tampilan credit yang berupa identitas developer maupun creator dari permainan ini seperti pada gambar berikut. (OKE)

```
* Game's Name: Tic-Tac-Toe
* Developer 1: Muhammad Azhar Alauddin D4TI-1A-201524013
* Developer 2: Difa Mochammad Fadillah D4TI-1A-201524006
* Date: 29 January 2021
* Comment: I hope you guys can enjoy this game ^_^

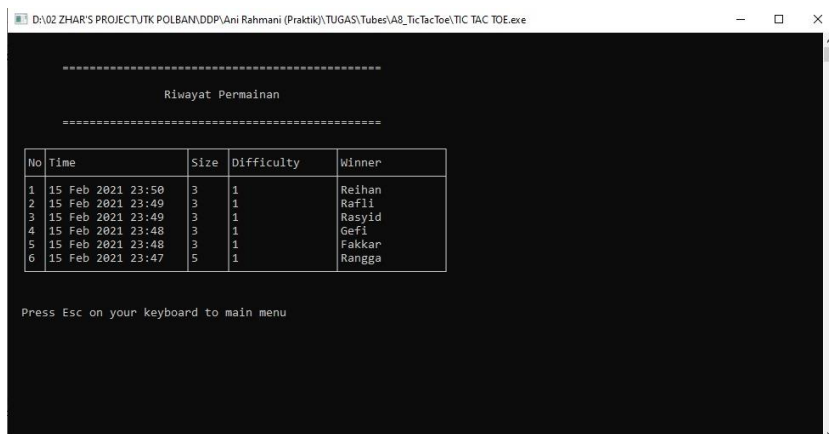
Stay in touch! Thanks a lot...

Press Esc on your keyboard to main menu
```

Gambar 25 Hasil implementasi tampilan Credits

#### k) Tampilan High Score

Jika player memilih 5 pada main menu, maka player akan ditampilkan dengan tampilan highscore seperti gambar di bawa dimana disini akan ditampilkan waktu permainan berlangsung, ukuran grid, level komputer, dan nama player yang memenangkan permainan melawan komputer. (OKE)



The screenshot shows a window titled "Riwayat Permainan" (Game History) with a table containing 6 rows of game data. The table has columns for No, Time, Size, Difficulty, and Winner. Below the table, it says "Press Esc on your keyboard to main menu".

No	Time	Size	Difficulty	Winner
1	15 Feb 2021 23:50	3	1	Reihan
2	15 Feb 2021 23:49	3	1	Rafli
3	15 Feb 2021 23:49	3	1	Rasyid
4	15 Feb 2021 23:48	3	1	Gefi
5	15 Feb 2021 23:48	3	1	Fakkar
6	15 Feb 2021 23:47	5	1	Rangga

Gambar 26 Hasil implementasi tampilan High Score

- **GamePlay**

- a) **Tampilan Game Play**

Berikut tampilan gameplay atau permainan utama, dimana pada tampilan ini player maupun computer menginputkan tandanya masing-masing. Pada tampilan ini, seperti nama player dan ukuran player itu ditentukan oleh pilihan sebelumnya. Seperti contoh disini diinputkan nama “Azhar” sebagai player dan 7x7 sebagai ukuran gridnya. (OKE)

```
Player1(X)
Player2/Computer(O)

Azhar as player 1
Player1's turn

(x,y)  1  2  3  4  5  6  7
1  [ X |  |  |  |  |  |  ]
2  [ O |  |  |  |  |  |  ]
3  [  |  |  | O | X |  |  ]
4  [  |  | O | X | O |  |  ]
5  [  |  |  | X |  |  |  ]
6  [  |  |  | O |  |  |  ]
7  [  |  |  |  |  | O | X ]

Enter x and y co-ordinates of your move: _
```

Gambar 27 Hasil implementasi tampilan Gameplay

- **End-Game**

- a) **Tampilan Result**

Berikut tampilan hasil permainan setelah kemudian player maupun komputer menempatkan tandanya sesuai kondisi berakhir. Pada gambar di bawah ditampilkan player 1 memenangkan permainan dengan berhasil menempatkan 3 tanda berturut-turut. (OKE)

```
Player1(X)
Player2/Computer(O)

Azhar as player 1
Player1's turn

      1   2   3
1  [ X | O | X ]
2  [ O | X | O ]
3  [ X |   | O ]

Player1 wins

Press Esc on your keyboard to play again
```

Gambar 28 Hasil implementasi tampilan Result

- b) **Tampilan Exit**

Jika player memilih 6 pada main menu, maka player akan keluar dari game seperti yang ditampilkan pada tampilan berikut. Dan jika player menekan “Esc”, maka permainan akan tertutup sendirinya.(OKE)

```
Press Esc on your keyboard to exit this game
.....
Process exited after 2.10 seconds with return value 0
Press any key to continue . . .
```

Gambar 29 Hasil implementasi tampilan Result

### 3. Petunjuk penggunaan

- bahasa Indonesia

Apa ini?

Game ini bernama tic tac toe. Dalam game ini, terdapat 2 komponen utama yaitu papan permainan yang terdiri dari 3 ukuran (3x3, 5x5, 7x7) dan juga player/bot yang dapat menempatkan tanda 'X' atau 'O' di papan permainan

Seperti yang anda lihat, pada bagian main menu terdapat beberapa pilihan :

1. Start untuk memulai game
2. Help untuk menampilkan deskripsi game dan juga cara bermainnya
3. Highscore untuk menampilkan riwayat kemenangan melawan komputer
4. Color Theme untuk memilih background dan text color pada permainan
5. Credits untuk menampilkan identitas developer atau creator dari permainan
6. Exit untuk keluar dari game

Bagaimana cara bermainnya?

1. Silakan pilih start pada main menu
2. Setelah itu, pilih ukuran grid permainan (3x3, 5x5, 7x7)
3. Pilih lawan, bisa melawan player 2 dan juga komputer
4. Inputkan nama anda. Jika melawan player 2, maka inputkan juga namanya
5. Pilih tingkat kesulitan dari komputer
6. Setelah itu, pilih siapa yang mendapatkan giliran pertama main
7. Perlu diingat bahwa setiap giliran dibatas dengan 10 detik
8. Player 1 memiliki tanda 'X', sedangkan player 2 atau komputer memiliki tanda 'O'
9. Setiap player atau komputer dapat mengisi tandanya masing-masing di papan permainan
10. Player/bot dikatakan menang, jika sudah mengisi tandanya sebanyak 3 kali berturut-turut pada grid 3x3 atau 4 kali berturut-turut pada grid 5x5 dan/atau 5 kali berturut-turut pada grid 7x7
11. Setiap ronde selesai, player dapat menentukan apakah mau melanjutkan permainan atau tidak
12. Setiap kemenangan melawan komputer, maka player dapat melihat riwayatnya di menu highscore



- bahasa Inggris

What is this?

So, this game is called tic-tac-toe. In this game, there's a board game in the form of grid (3x3, 5x5, 7x7) and two players, X and O, who take turns marking spaces ' ' in the board game. The player who takes the turn can mark in the row, column or diagonal side in the board game.

Also, as you can see in main menu, there's a couple of choices :

1. Start to start the game.
2. Help to display game descriptions, rules and how to play.
3. Highscore to display the history of the player's victory against the computer.
4. Color Theme to select the display color of this game.
5. Credits for displaying the identity of the developer or creator of this game.
6. Exit to exit the game.

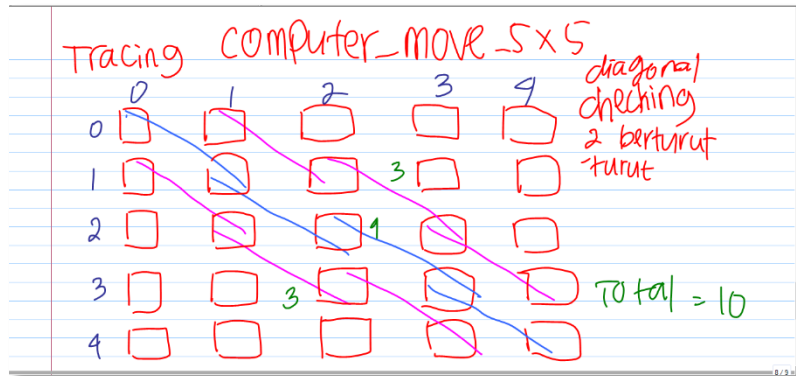
How to play this game? :

1. Select Start.
2. Select the size of the game board (3x3, 5x5, 7x7).
3. Choose who you want to play with (Player/Computer).
4. Enter the player name.
5. Choose a difficulty level.
6. Choose who will take the first turn.
7. Each turn is limited to 10 seconds.
8. You are 'X', your friend (or the computer in this case) is 'O'. Players take turns putting their marks in empty squares.
9. The first player to get all of her marks in a row (up, down, across, or diagonally) is the winner.
10. When all squares are full, the game is over. If no player has won, the game ends in a tie.

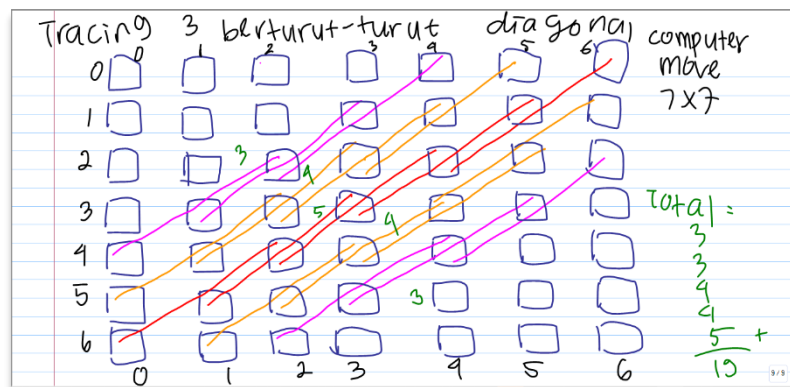
## BAB IV : PENUTUP

### 1. Dokumentasi

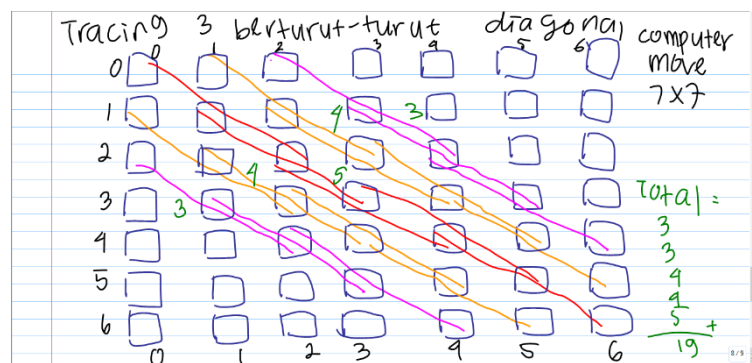
Berikut kami tampilkan beberapa hasil tracing dari pergerakan komputer dalam masing-masing ukuran grid nya sebelum kemudian mengimplementasikannya.



Gambar 30 tracing computer\_move\_5x5



Gambar 31 tracing computer\_move\_7x7



Gambar 32 tracing computer\_move\_7x7

Selengkapnya:

[https://drive.google.com/drive/folders/1DfFFqLDxrD4pDa\\_JJPK8fS9Gj9AuI3W7?usp=sharing](https://drive.google.com/drive/folders/1DfFFqLDxrD4pDa_JJPK8fS9Gj9AuI3W7?usp=sharing)

## 2. Timeline Pengerjaan

Pada awalnya, kami merencanakan dan juga merancang segala hal terkait dengan permainan apa yang akan dibuat secara umum. Setelah dirancang, maka kami mulai menentukan kebutuhan apa saja yang kira-kira dibutuhkan di implementasinya. Setelah ditentukan apa saja yang dibutuhkan, maka kami lanjut untuk menentukan modul-modul apa saja yang harus ada baik itu berupa prosedur ataupun function.

Setelah kemudian modulnya ditentukan, maka kami melanjutkan untuk mulai terjun ke implementasi disertai dengan datacase dan juga tracing. Seiring berjalannya waktu, kami juga saling me-review bersama, konsultasi dan juga mengoreksi apabila ada kesalahan. Setelah dirasa semuanya hampir atau bahkan selesai, maka kami mulai merancang untuk laporan dan juga bahan presentasi kami.

Berikut timeline singkat kami yang menggambarkan alur pengerjaan dari tugas besar ini :

- a. Akhir Desember 2020 : Penerimaan tugas besar
- b. Awal Januari 2021 : Merancang dan merencanakan permainan yang akan dibuat
- c. Pertengahan Januari 2021 : Mulai membuat program permainan
- d. Akhir Januari 2021 : Konsultasi dan perbaikan konsep dan juga program permainan
- e. 1 - 15 Februari 2021 : Finalisasi program permainan dan juga konsultasi
- f. 15 - 16 Februari 2021 : Testing program dan menyelesaikan laporan dan juga bahan untuk presentasi

## 3. Ringkasan Kontribusi Setiap Anggota

M Azhar Alauddin	Difa Mochammad Fadillah
Membuat keseluruhan modul, kecuali modul main	Mencari kebutuhan referensi permainan tic-tac-toe dari yang sudah ada di internet dan lainnya
Pada laporan mengerjakan bab 2, structure	Pada laporan mengerjakan bab 1, bab 3,

chart, dan sketsa skenario	flow chart, dan hasil akhir program
Me-review segala pengerjaan yang sudah dibuat	Membuat PPT untuk keperluan presentasi
Mengerjakan tracing dan data testing dari setiap implementasi program yang sudah dibuat	Membuat modul main pada implementasi program

#### 4. Lesson Learned

M Azhar Alauddin	Difa Mochammad Fadillah
<p>Bagi saya pribadi dalam seluruh rangkaian pengerjaan tugas besar ini baik itu pengerjaan skenario, implementasi maupun laporan, pelajaran besar yang bisa diambil adalah sesuatu hal besar pasti dimulai dari proses dan langkah yang kecil yang disertai dengan keseriusan dan konsisten. Awalnya, program ini terasa berat dan panjang, terutama ketika mengerjakan implementasi program untuk pergerakan komputer pada grid 7x7.</p> <p>Tetapi karena saya yakin bahwa hal itu akan selesai bila saya mau memulai dan juga melanjutkan sedikit demi sedikit mulai dari tracing dan juga membuat implementasi programnya sehingga saya terus lanjutkan setiap ada waktu dan akhirnya semua programnya selesai dengan lebih dari 5400 baris. Dari sini, saya mengerti betul bahwa hal yang paling utama adalah kita harus memulai dan konsisten.</p>	<ol style="list-style-type: none"> <li>1. Memahami pentingnya kerangka sebelum mengimplentasikan,</li> <li>2. Memahami <i>structure chart</i></li> <li>3. Memahami <i>source code</i> yang terstruktur.</li> <li>4. <i>Requirement</i> sangat penting, dimana di sana terletak informasi dan batasan - batasan.</li> <li>5. Lebih memahami penggunaan <i>function</i> dengan definisi, deklarasi, parameter.</li> </ol>