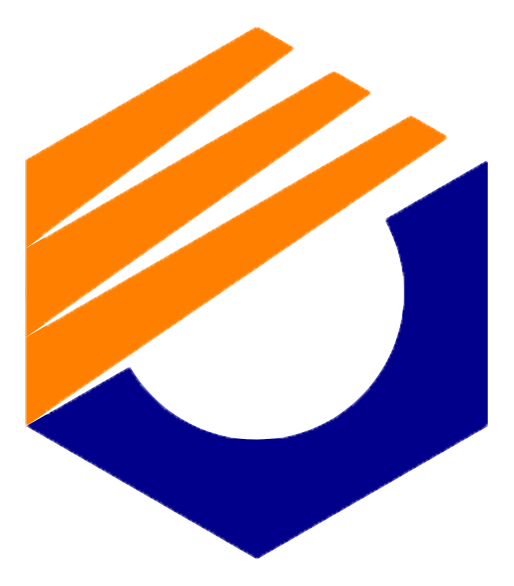
**LAPORAN TUGAS BESAR**

**Dasar – Dasar Pemograman**

**Permainan Tic Tac Toe**

****

Di susun Oleh :

**Dimas Kurniawan – 201524040**

**Muhamad Aryadipura Sasmita Atmadja – 201524054**

**Program Studi D-IV Teknik Informatika**

**Departemen Teknik Komputer dan Informatika**

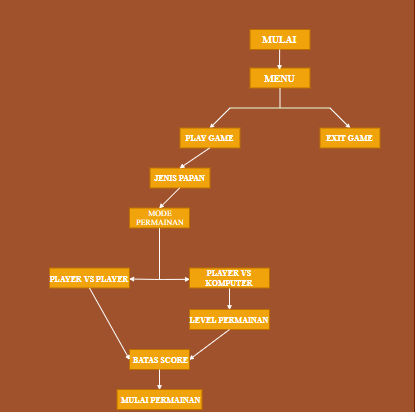
**Politeknik Negeri Bandung**

**2021**

**BAB 1**

1. **Deskripsi Aplikasi**
   1. **Definisi** Program yang akan kita buat adalah permainan Tic Tac Toe. Tic Tac Toe adalah permainan papan yang dimainkan oleh 2 orang atau dengan komputer dimana pemain bertujuan untuk memenangkan permainan dengan cara kedua pemain berusaha untuk membuat satu garis dengan karakter yang sama bisa secara vertikal, horizontal, atau diagonal.

Ada beberapa jenis papan dari permainan tic tac toe diantaranya papan berjenis 3x3, 4x4, dan 5x5. Permainan ini dapat dimainkan dengan 2 mode, yaitu player vs player dan player vs komputer. Untuk mode players vs komputer, terdapat level permainan terhadap tingkat kesulitan komputer yang dapat dipilih oleh pemain yaitu easy, medium, dan hard. Setiap pemain secara bergiliran mengisi kotak kosong yang tersedia dengan karakter yang sudah ditetapkan, dimana pemain pertama secara otomatis akan mendapatkan karakter ‘X’ dan pemain kedua/komputer dengan otomatis akan mendapatkan karakter ‘O’. Setiap pemain yang berhasil membuat garis dengan karakternya terlebih dahulu, berhasil memenangkan 1 ronde permainan dan akan mendapatkan 1 poin tambahan. Jika salah satu poin pemain sudah mencapai batas dari skor yang sudah ditentukan (high score) maka permainan berakhir/ game over dan pemain dengan poin yang paling banyaklah yang menjadi pemenangnya.

* 1. **Skenario** Ada beberapa tahapan yang perlu kita lakukan sebelum memulai permainan tic tac toe.

Saat permainan pertama kali menjalankan program pemain tic tac toe akan diarahkan pada tampilan main menu yang akan muncul.

Menu yang akan muncul berisikan pilihan – pilihan seperti berikut :

1. Play game, Melanjutkan ke tahapan memulai permainan
2. Exit game, Berhenti dan keluar dari game

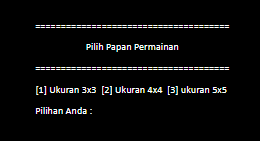


(Sketsa Dari tampilan menu game )

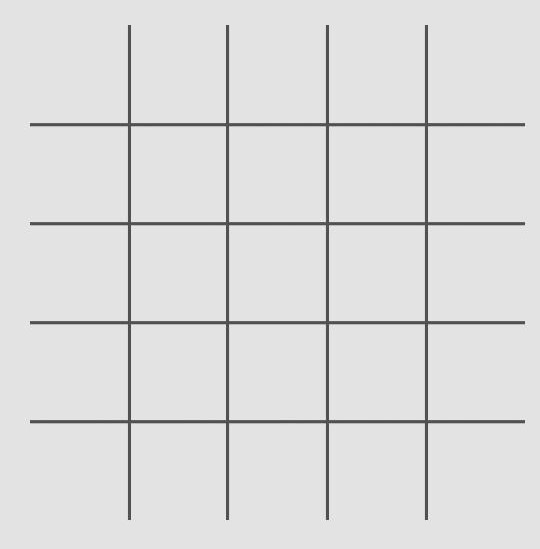
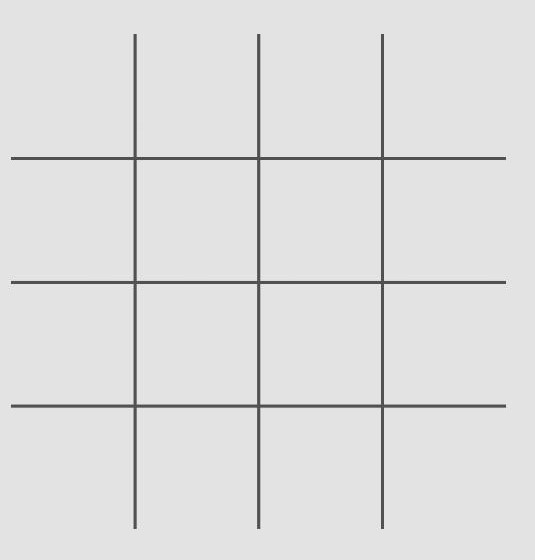
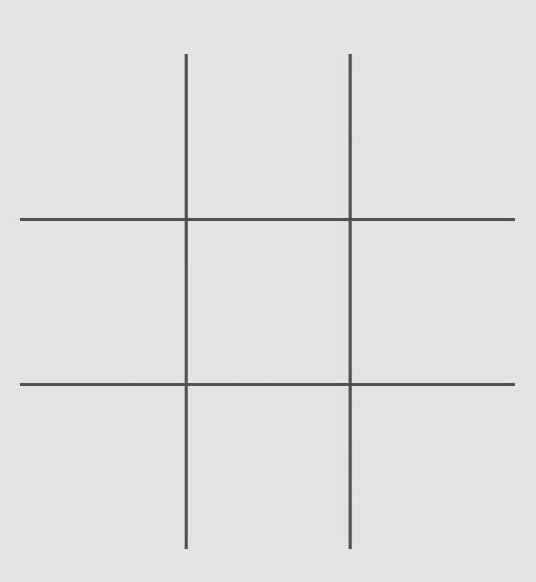
Jika pemain memilih *Play game* maka pemain akan di di bawa ke tahap selanjutnya dengan tampilan pemilihan papan permainan. Terdapat papan permainan yang berbeda, kalian dapat memilih papan permainan sesuai kemauan. Terdapat pilihan papan sebagai berikut :

1. Ukuran 3x3
2. Ukuran 4x4
3. Ukuran 5x5

Dibawah ini menunjukan tampilan yang akan muncul di halaman pemilih papan permainan.



Berikut merupakan ilustrasi dari tiap jenis papan :

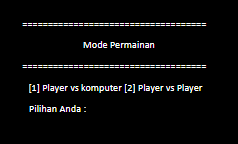


(papan 3x3) (papan 4x4) (papan 5x5)

Setelah Memilih papan permainannya kalian akan dibawa pada tampilan pemilihan mode permainan. Menu mode permainan terdapat dua pilihan yang bisa dipilih, yaitu :

1. Player vs komputer, untuk bermain melawan bot/komputer.
2. Player vs Player, untuk bermain melawan pemain lain.

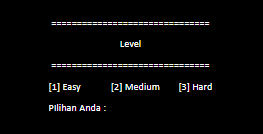
Dibawah Ini merupakan ilustrasi tampilan yang akan muncul dari menu mode permainan.



Setelah pemain memilih mode permainan, ada dua alur yang mungkin terjadi, jika pemain memilih mode permainan Player vs bot/computer maka akan dibawa ke tampilan dengan kesulitan yang berbeda. Ada 3 pilihan untuk level bermain dengan bot/komputer, yaitu :

1. Easy, maka kesulitan permainan akan relatif mudah, dimenangkan dimana komputer akan mengisi petak tepat disebelah kanan dari petak yang sudah player isi.
2. Medium, maka kesulitan relatif sedang, dimana komputer akan secara random mengisi kotak tanpa mempedulikan apa yang sudah diisi oleh player.
3. Hard, maka kesulitan permainan akan mencapai level teratas yang relatif sulit dimenangkan, dimana komputer berusaha untuk memenangkan permainan dengan cara berusaha untuk mengincar kotak tengah terlebih dahulu jika kosong, lalu akan menempatkan karakternya di petak lain yang akan membuat satu garis dengan simbol yang dimilikinya.

Dibawah Ini merupakan ilustrasi tampilan yang akan muncul dari menu level/ tingkat kesulitan

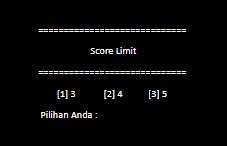


Tetapi, jika pemain memilih mode *player vs player* makan pemain tidak akan melewati tahap memilih mode permainan dan langsung ke tahap pemilihan batas skor.

Selanjutnya pemain akan dibawa ke tampilan pemilihan batas skor. Pemain dapat memilih batas skor yang ingin dicapai untuk memenangkan permainan dan mengkhiri permainan tersebut (game over). Terdapat 3 pilihan dalam pemilihan batas skor, yaitu :

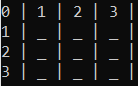
1. Batas skor = 3
2. Batas skor = 4
3. batas skor = 5

Dibawah Ini merupakan ilustrasi tampilan yang akan muncul dari menu batas skor.



Setelah pemain melakukan pemilihan batas skor, maka selanjutnya permainan tic tac toe dimulai.

ilustrasi untuk tampilan awal ketika permainan dimulai (anggapan jenis papan yang dipilih adalah papan 3x3) :



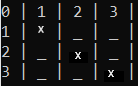
Pilihan Anda : (tempat menginput nomer petak dengan array 2 dimensi )

Player harus mengingputkan nomor petak untuk meletakan karakter di bidak yang diinginkan.

* Untuk mode permainan player vs player, pemain akan bergantian mengisi petak yang masih kosong dengan cara masing-masing pemain menginputkan nomer bidak sesuai letak bidak yang ingin mereka isi. Jika pemain menginputkan nomor yang sebelumnya telah terisi oleh karakter, maka karakter tidak dapat diisikan di petak tersebut dan pemain harus mengisi karakter di petak lain yang masih kosong.
* Untuk mode player vs komputer, pemain akan mengisi petak dengan menginputkan nomor petak yang masih kosong. Sedangkan komputer akan otomatis mengisi petak lainnya yang masih kosong dengan karakteristik komputer yang sudah ditentukan sebelumnya. Jika pemain menginputkan nomor yang sebelumnya telah terisi oleh komputer, maka karakter tidak dapat diisikan di petak tersebut dan pemain harus mengisi karakter di petak lain yang masih kosong.

Permainan ini menggunakan sistem ronde. Jika kedua pemain tidak ada yang memenangkan permainan dalam 1 ronde, maka papan akan direset dengan tidak ada tambahan poin untuk kedua pemain. Jika salah satu pemain memenangkan permainan, maka ia akan mendapatkan 1 poin tambahan. Poin ini dihitung dan bertambah terus 1 poin tiap kali pemain memenangkan permainan dan akan berakhir/ game over jika poin dari salah satu pemain sudah mencapai batas skor/ high score.

Berikut adalah salah satu kondisi untuk memenangkan permainan untuk mendapat 1 poin :



**1.3 Inisialisasi**

1. Skor awal setiap pemain yaitu 0.
2. Yang menjadi giliran pertama dalam mengisi kotak selalu player yang mendapat ‘X’ dan yang menjadi giliran kedua dalam mengisi kotak akan selalu mendapat ‘O’.
3. Jika player vs player, maka giliran pertama yang akan mendapat ‘X’ akan dipilih secara acak.
4. jika player vs komputer, maka giliran pertama yang akan mendapat ‘X’ adalah player sedangkan komputer akan mendapatkan ‘O’

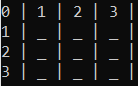
**1.4 Setting**

1. Pemain memilih papan permainan yang terdiri dari papan 3x3, 4x4, dan 5x5.
2. Pemain memilih mode permainan (player vs komputer/ player vs player).
3. Pemain memilih level/ tingkat kesulitan yaitu easy, medium, dan hard (jika memilih mode player vs komputer).
4. Pemain memilih batas skor yang ditentukan untuk mengakhiri permainan.

**1.5 Aturan Game**

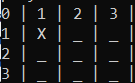
1. Waktu yang diberikan kepada pemain untuk mengisi petak adalah 15 detik. Jika tidak diisi dalam kurun waktu tersebut, maka pemain akan dilewati dan berlanjut ke pemain selanjutnya.
2. Pemenang dapat ditentukan jika salah satu dari pemain sudah berhasil membuat pola ‘X X X’ untuk pemain pertama dan ‘O O O’ untuk pemain kedua. Bisa secara vertikal, horizontal, maupun diagonal dan banyaknya mengikuti papan yang digunakan.
3. Papan 3x3 memerlukan 3 simbol dalam 1 deret pola untuk memenangkan permainan.
4. Papan 4x4 memerlukan 4 simbol dalam 1 deret pola untuk memenangkan permainan.
5. Papan 5x5 memerlukan 5 simbol dalam 1 deret pola untuk memenangkan permainan.

**1.6 Cara Bermain**



Pilihan Anda :

1. Pemain pertama mengisi ‘X’ pada kotak kosong yang tersedia.

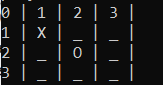


Pilihan Anda : 1 1

Saat pemain pertama memilih petak pemain harus mengisikan nomor petak dengan cara memasukan nomor 2 dimensi, contohnya 1 1 maka ‘X’ akan terisi di petak yang ada pada gambar. Jika petak sudah diisi maka akan berpindah ke giliran selanjutnya

1. Pemain kedua mengisi ‘O’ pada kotak lain yang masih kosong dan tersedia (nomor kotak yang belum terpilih).

* Jika player vs player, maka player kedua harus melakukan hal yang sama dengan pemain pertama.



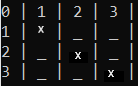
Pilihan Anda : 2 2

Karena player kedua menginput nomer 2 2, maka nomer 2 2 belum terisi dan akan terisi oleh karakter ‘O’ dari player 2

* Jika player vs komputer, maka komputer akan mengisi petak secara otomatis tanpa harus menginputkan nomor petak terlebih dahulu.

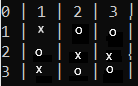
1. Pengulangan pengisian terjadi secara terus menerus hingga terjadi kondisi dimana petak sudah penuh dan tidak bisa lagi diisi oleh kedua pemain, atau juga ketika salah satu pemain berhasil memenuhi syarat untuk menang.
2. Pemenang didapatkan setelah salah satu pemain berhasil memenuhi syarat untuk menang. Pengecekan dilakukan setiap kali pemain/ komputer mengisi petak. Jika tidak ada yang memenuhi syarat untuk menang, maka permainan akan berakhir imbang dan tidak ada pemain yang mendapat poin.

Kondisi salah satu pemain menang (disini X) :



Jika sampai akhir tidak ada pemain yang memenuhi syarat tersebut dan petak sudah penuh/ tidak dapat diisi lagi oleh pemain, maka permainan pun berakhir imbang

Kondisi imbang :



**1.7 Kondisi Game Over**

Permainan dapat berakhir dengan menemukan siapa pemenangnya atau juga dapat berakhir dengan imbang, tetapi permainan tidak akan GAME OVER jika belum ada salah satu dari pemain yang mencapai batas skor yang telah ditentukan.

Game over dalam permainan ini terjadi jika salah satu skor dari pemain sudah mencapai batas skor yang sudah ditentukan. Setelah game over, user akan ditawarkan apakah ingin bermain kembali atau tidak.

**BAB 2**

**2. PERANCANGAN PROGRAM**

**2.1**  **Kebutuhan Data**

**2.1.1 Data Player**

{

Score, = score player

Status, = status player apakah bermain atau menunggu giliran

}

**2.1.2 Data Komputer**

{

Score, = score komputer

Status, = status komputer apakah bermain atau menunggu giliran

Level, = level komputer apakah easy, medium atau hard

}

**2.1. 3 Data Papan**

{

Kolom, = kolom dari papan

Baris, = baris dari papan

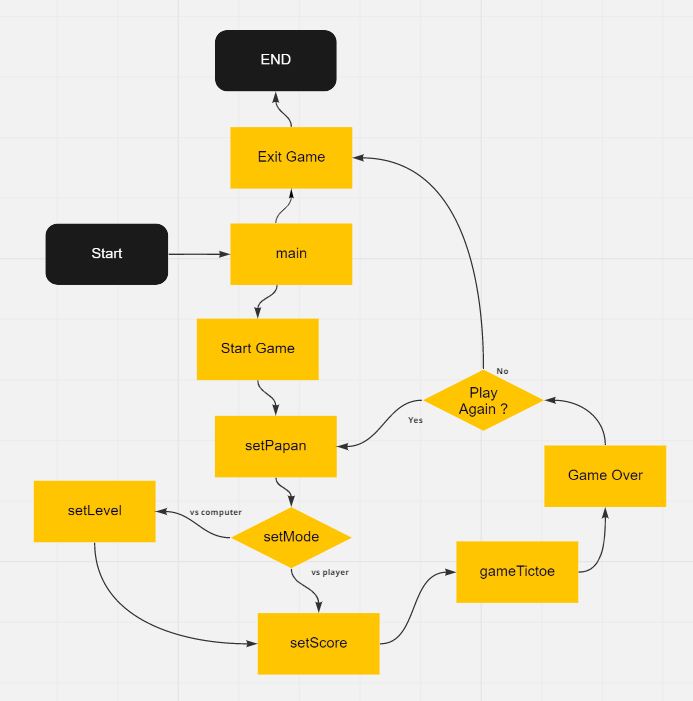
Status, = status apakah papan telah terisi atau belum

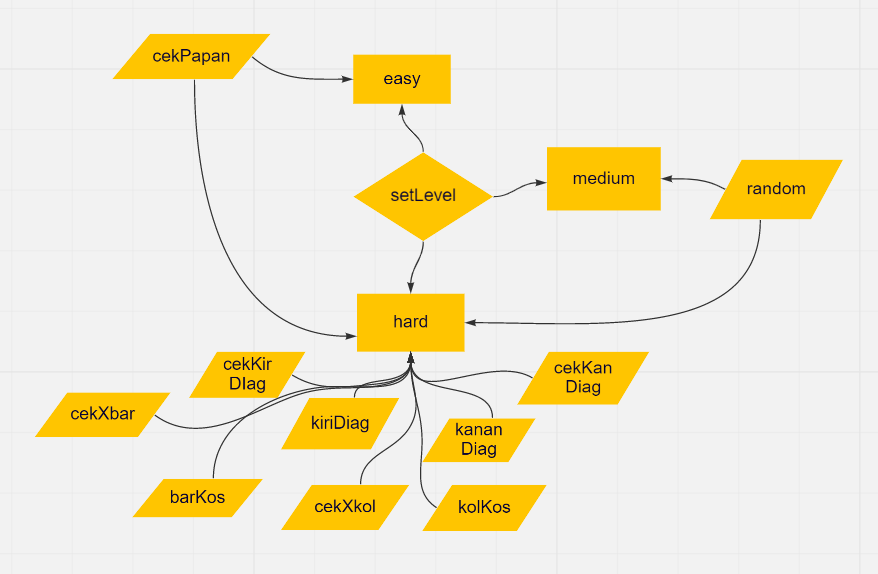
Waktu, = waktu dari setiap giliran

Matriks 2 dimensi

}

**2.2 Flowchart Program Utama**

****

****

**2.3** **Perancangan Proses**

**Modul Utama :**

|  |  |  |  |
| --- | --- | --- | --- |
| **NAMA MODUL** | **PJ** | **JENIS** | **KETERANGAN** |
| gameTictoe | M. A | Procedure | Untuk memulai permainan setelah sudah menetapkan pilihan |
| setPapan | D. K | Procedure | Untuk menampilkan menu pemilihan papan |
| setLevel | D. K | Procedure | Untuk menampilkan menu pemilihan level |
| setMode | D. K | Procedure | Untuk menampilkan menu pemilihan mode permainan |
| setScore | D. K | Procedure | Untuk menampilkan menu pemilihan limit dari score |
| turn1 | M. A | Procedure | Untuk inisialisasi awal score dan penambahan score pemain pada papan 3x3 dengan mode pvp |
| turn2 | M. A | Procedure | Untuk inisialisasi awal score dan penambahan score pemain pada papan 4x4 dengan mode pvp |
| turn3 | M. A | Procedure | Untuk inisialisasi awal score dan penambahan score pemain pada papan 5x5 dengan mode pvp |
| turn1com | M. A | Procedure | Untuk inisialisasi awal score dan penambahan score pemain pada papan 3x3 dengan mode pvcom |
| turn2com | M. A | Procedure | Untuk inisialisasi awal score dan penambahan score pemain pada papan 4x4 dengan mode pvcom |
| turn3com | M. A | Procedure | Untuk inisialisasi awal score dan penambahan score pemain pada papan 5x5 dengan mode pvcom |
| inputPlayer1 | M. A | Function | Untuk cara penginputan player 1 |
| inputPlayer2 | M. A | Function | Untuk cara penginputan player 2 |
| inputcom | M. A | Function | Untuk cara penginputan computer |
| dispPapan | D. K | Procedure | Untuk menampilkan papan |
| papan1 | D. K | Procedure | Untuk 3x3 |
| papan2 | D. K | Procedure | Untuk 4x4 |
| papan3 | D. K | Procedure | Untuk 5x5 |
| papan1com | D. K | Procedure | Untuk jika yang dipilih 3x3 dan melawan com |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| papan2com | D. K | | Procedure | | Untuk jika yang dipilih 4x4 dan melawan com | |
| papan3com | | D. K | | Procedure | | Untuk jika yang dipilih 4x4 dan melawan com |
| easy | | M. A | | Procedure | | Untuk karakteristik computer bila level yang dipilih easy |
| medium | | M. A | | Function | | Untuk karakteristik computer bila level yang dipilih medium |
| hard | | M. A | | Procedure | | Untuk karakteristik computer bila level yang dipilih hard |
| cekXbar | | D. K | | Procedure | | Untuk mengecek dan dihitung apakah X pada baris sudah ada 2 |
| cekXkol | | D. K | | Procedure | | Untuk mengecek dan dihitung apakah X pada kolom sudah ada 2 |
| barKos | | D. K | | Function | | Untuk mengecek apakah baris masih kosong |
| kolKos | | D. K | | Function | | Untuk mengecek apakah kolom masih kosong |
| cekKanDiag | | D. K | | Procedure | | Untuk mengecek dan dihitung apakah X pada diagonal kanan sudah ada 2 |
| cekKirDiag | | D. K | | Procedure | | Untuk mengecek dan dihitung apakah X pada diagonal kiri sudah ada 2 |
| kananDiag | | D. K | | Function | | Untuk mengecek apakah diagonal kanan masih kosong |
| kiriDiag | | D. K | | Function | | Untuk mengecek apakah diagonal kiri masih kosong |
| papan1ply | | M. A | | Procedure | | Untuk 3x3 dan permainan pvp |
| papan2ply | | M. A | | Procedure | | Untuk 4x4 dan permainan pvp |
| papan3ply | | M. A | | Procedure | | Untuk 5x5 dan permainan pvp |
| persiapanPapan | | M. A | | Procedure | | Untuk menampilkan desain papan ketika permainan dimulai |
| cekPapan | | M. A | | Function | | Untuk mengecek apakah papan masih kosong |
|  |  |  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| cekPenuh | M. A | Procedure | Untuk mengecek apakah papan sudah penuh |
| cekBarisJadi1 | M. A | Procedure | Untuk mengecek pada 3x3 apakah baris sudah jadi |
| cekKolomJadi1 | M. A | Procedure | Untuk mengecek pada 3x3 apakah kolom sudah jadi |
| cekDiagonalJadi1 | M. A | Procedure | Untuk mengecek pada 3x3 apakah diagonal sudah jadi |
| cekBarisJadi2 | M. A | Procedure | Untuk mengecek pada 4x4 apakah baris sudah jadi |
| cekKolomJadi2 | M. A | Procedure | Untuk mengecek pada 4x4 apakah kolom sudah jadi |
| cekDiagonalJadi2 | M. A | Procedure | Untuk mengecek pada 4x4 apakah diagonal sudah jadi |
| cekBarisJadi3 | M. A | Procedure | Untuk mengecek pada 5x5 apakah baris sudah jadi |
| cekKolomJadi3 | M. A | Procedure | Untuk mengecek pada 5x5 apakah kolom sudah jadi |
| cekDiagonalJadi3 | M. A | Procedure | Untuk mengecek pada 5x5 apakah daigonal sudah jadi |
| starttime | D. K | Procedure | keterangan waktu mengisi |
| endtime | D. K | Procedure | keterangan waktu mengisi |

KET : M. A = M. Aryadipura D.K = Dimas Kurniawan

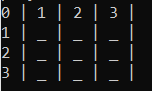
**Spesifikasi Modul :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama Modul** | | setMode() | | |
| **Deskripsi** | | Modul untuk memilih mode permainan yang akan dimainkan | | |
| **I. S** | | Tempat untuk menginput masih kosong | | |
| **F. S** | | Tempat untuk menginput sudah terisi untuk memanggil modul berikutnya | | |
| **Modul Pemanggil** | | setPapan() | | |
| **Modul yang dipanggil** | | setLevel(), setScore() | | |
| **PJ** | | Dimas Kurniawan | | |
| **Tanggal** | |  | | |
| **Parameter Input** | |  | | |
| **Parameter output** | |  | | |
| **Algoritma** | | void setMode()  {  printf("\n==================================================");  printf("\n ");  printf("\n Mode permainan");  printf("\n ");  printf("\n==================================================");  printf("\n [1] Player vs komputer [2] Player vs player");  printf("\n Pilihan anda : "); scanf("%d", &r);  system("cls");  switch(r)  {  case 1:  printf("player vs komp \n");  setLevel();  system("cls");  break;  case 2:  printf("player vs player \n");  setScore();  system("cls");  break;  default:  printf("WRONG INPUT NUMBER.\n");  system("pause");  setMode();  }  } | | |
| **Data test** | | | | |
| **NO** | **Nama case data** | | **Data input** | **Hasil yang diharapkan** |
| 1 | Jika memilih 1 | |  | Masuk ke modul setLevel() |
| 2 | Jika memilih 2 | |  | Masuk ke modul setScore() |
|  |  |  |  |  |

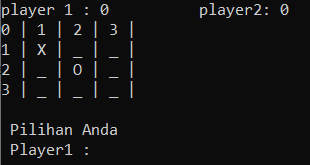
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama Modul** | | gameTictoe() | | |
| **Deskripsi** | | Modul untuk memulai permainan setelah sudah menetapkan pilihan | | |
| **I. S** | | permainan belum dapat dimulai | | |
| **F. S** | | Memainkan permainan sesuai dengan yang sudah dipilih | | |
| **Modul Pemanggil** | | setScore() | | |
| **Modul yang dipanggil** | | papan1(), papan2(), papan3 | | |
| **PJ** | | Muhamad Aryadipura | | |
| **Tanggal** | |  | | |
| **Parameter Input** | |  | | |
| **Parameter Output** | |  | | |
| **Algoritma** | | void gameTictoe(){  switch(q){  case 1:  papan1();  break;  case 2:  papan2();  break;  case 3:  papan3();  break;    }  } | | |
| **Data test** | | | | |
| **NO** | **Nama case data** | | **Data input** | **Hasil yang diharapkan** |
| 1 | Jika 3x3 | |  | Permainan dimulai dengan papan 3x3 dan bermain dengan ketentuan yang sudah dipilih |
| 2 | Jika 4x4 | |  | Permainan dimulai dengan papan 3x3 dan bermain dengan ketentuan yang sudah dipilih |
| 3 | Jika 5x5 | |  | Permainan dimulai dengan papan 3x3 dan bermain dengan ketentuan yang sudah dipilih |
|  |  |  |  |  |

**2. 4 Interface/ tampilan**

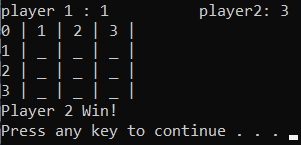
* Papan :

****

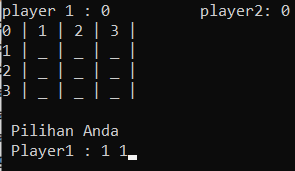
* Awal permainan :



* Game Over :



* Giliran bermain :



( Dilihat dari perintah inputan apakah player 1/2 )

**BAB 3**

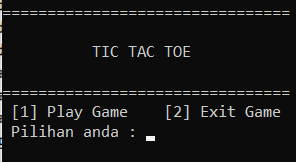
**3. Hasil Akhir Program**

**3. 1 Link Github**

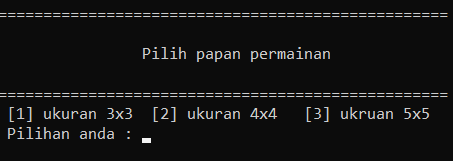
https://github.com/maskur17/Tubes-Tic-Tac-Toe

**3. 2 Screenshot Data Case**

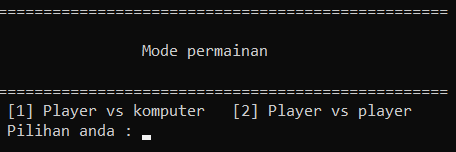
* Menu :

****

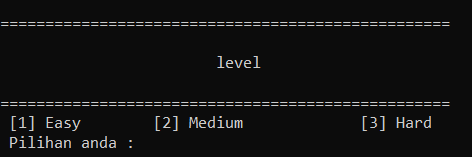
* Pilihan papan :



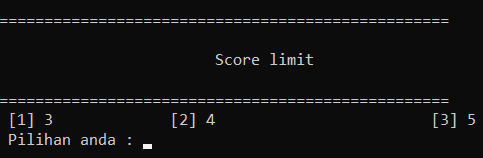
* Pilihan mode :



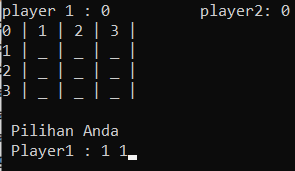
* Pilihan level :



* Pilihan limit score :



* Permainan



**3.3 Petunjuk Penggunaan**

1. Player akan diminta menginput pilihan sesuai yang ada pada menu.
2. Player akan diminta menginput nomor dengan ketentuan matriks 2 dimensi untuk mengisi papan saat permainan dimulai. Contoh : 1 1
3. Jika ingin keluar dari permainan, maka tekan tombol ESC atau klik manual pada pilihan yang tersedia.

**BAB 4**

**4. Penutup**

**4.1 Timeline**

Diawal waktu, kami mendiskusikan apa yang akan kami buat. Setelah disepakati, program yang akan kami buat adalah tictactoe, kami mulai merancang program kami. Kami memulai dari membuat laporan terlebih dahulu dan benar- benar hanya menggambarkan bagaimana game kita nanti akan dibuat. setelah menyelesaikan BAB 1 dari laporan tentang deskripsi aplikasi, kami baru memulai coding dari baris 1 sesuai gambaran dari laporan yang sudah dibuat. Awalnya kami berencana membuat program dengan menggunakan modul yang sedikit. Namun pada akhirnya kami memutuskan untuk membuat program yang penuh dengan modul. Saat sudah menyelesaikan codingan dan program dapat dijalankan, kami memulai mengerjakan laporan BAB 2 hingga selesai. laporan dari BAB 2 ini kita kerjakan sesuai dengan apa yang ada pada program kami.

**4.2 Kontribusi**

Kami telah menyepakati untuk membuat program dan laporan secara bersama- sama dan tidak membagi-bagi tugas terlalu banyak. Jadi tidak ada pembagian tugas dan hampir setiap bagian dikerjakan bersama.

**4.3 Lesson learned**

Banyak sekali pengalaman dan temuan pembelajaran yang didapat selama proses pengerjaan tugas besar membuat program game TicTacToe ini. Hal yang semula nya belum bisa kita lakukan menjadi bisa kita lakukan, misalnya proses tracing. Dalam pengerjaan tugas besar ini, membuat kita sering kali melakukan tracing karena harus mengecek secara keseluruhan ketika ada kesalahan atau bug dalam program yang kita buat sehingga melatih kami dalam melakukan proses tracing. Dalam pengerjaan tugas besar ini, kami juga dapat pelajaran yang suatu saat akan kami ingat dalam membuat program, yaitu selalu menambahkan komentar di setiap bagian penting/ bagian yang sedikit membuat pusing supaya mudah diingat. Contoh kasus kami dalam pengerjaan tubes ini yaitu kita kami tidak menambahkan komentar saat pembuatan modul, sehingga kami terkadang lupa salah satu fungsi dari modul tersebut atau kenapa modul tersebut dibuat, dll.