



LET'S PLAY

THE TIC TAC TOE GAME

BY KARMA TEAM
PRG 5

KARMA

Kelompok mAhasiswa stRess kiMiA

ANGGOTA KELOMPOK 5 PRG

FITRA 22/504395/PA/21696

BAYU 22/494032/PA/21239

EKASATRIA /22/498517/PA/21526

CORNELIA HAPPY
22/496781/PA/21362

MIFTAHUL FADILAH
22/502544/PA/21556

RATIH CINTIA
22/504640/PA/21702

LATAR BELAKANG MASALAH

- Apa masalah yang perlu dipecahkan?

Mahasiswa jenuh dan merasa bosan dengan laprak dan tugas yang serius

- Mengapa ini menjadi masalah?
Jenuh, bosan, dan bikin stres
- Di mana masalahnya diamati?
(lokasi, produk)
di Diri sendiri

- Siapa yang terkena dampaknya?
Diri sendiri dan teman-teman

- Kapan masalah pertama kali diamati?

Di suatu matkul, ada aja yang ngerjain laprak matkul lain wkwk



HOW TO PLAY

Dalam game ini hasil permainan yang didapat berupa menang, kalah, atau seri. Hasil permainan dinyatakan menang jika simbol dari salah satu pemain telah tersusun membentuk garis diagonal, vertikal, atau horizontal, sedangkan pemain yang lain dianggap kalah. Hasil permainan dinyatakan seri jika simbol dari kedua pemain tidak ada yang tersusun membentuk garis diagonal, vertikal, atau horizontal, serta semua bidak papan permainan telah terisi. Kemungkinan terbaik untuk permainan ini hanyalah seri.

Game Tic Tac Toe merupakan permainan klasik berjenis permainan papan (board-game) dengan papan permainan berukuran 6x6

Dalam game Tic Tac Toe terdapat 36 buah ruang (6x6) berbentuk kotak yang bersekat (bidak). Game ini menggunakan dua simbol pemain yaitu X atau O. Game ini dimulai dengan mengisi salah satu bentuk simbol pada salah satu bidak, hingga tiga buah simbol yang berbentuk sama tersusun membentuk garis diagonal, vertikal, atau horizontal.

HOW TO PLAY

Besar papan permainan berukuran 6x6

Untuk memberikan langkah, setiap pemain harus mengisi bidak dengan simbol pemain masing-masing, biasanya X atau O.

Bidak yang sudah terisi tidak bisa ditimpa oleh langkah berikutnya

Tujuan dari game ini adalah untuk mendapatkan deret dengan tiga simbol yang sama secara horizontal, vertikal atau diagonal

Pemain dalam permainan ini dibatasi untuk satu lawan satu, dilakukan bergiliran antara pemain pertama dan pemain kedua.

Setiap pemain hanya mempunyai satu kali kesempatan pada setiap giliran

Langkah yang sudah diambil tidak dapat dibatalkan atau diganti dengan langkah yang lain.

pemain yang berhasil menyusun deret akan mendapatkan poin tambahan

TAMPILAN



```
=====
mmmmmmmm " mmmmmmmmm mmmmmmmmm
# mmm mmm # mm # mm # mm # mm #
# # # " " # " " " # " " # " " #
# # # " " " # " " " # " " " # "
# mm#mm "#mm" # "mm"# "#mm" # "#m#" "#mm"
=====
```

Masukkan nama player1 : Narto
Masukkan nama player2 : Narti

```
(X) Baya 3
(0) Bayu 2
=====
```

OUR WINNER IS ...

(X) Baya!!!
Point : 3

Mau main lagi ? (y/n) :

Board

\ - - -
\ \ 0 0
\ \
\ / /
/ /
/ /

Visual

ca			cd				
da	db		dd				
ea	eb	ec			ef		
fa	fb			fe	ff		

(0) Bayu giliran main, silakan input koordinat :

MAIN.CPP

```
22 }  
23  
24 ▼ int main() {  
25 ▼ do{  
26     iFace.title();  
27     ticTacToe ttt(  
28         iFace.inputString("Masukkan nama player1 : "),  
29         iFace.inputString("Masukkan nama player2 : "),  
30         6  
31     );  
32  
33     while(!ttt.isFull()){  
34         iFace.clearScreen();  
35  
36         iFace.showScore(ttt);  
37         iFace.showBoard(ttt);  
38  
39         iFace.newLine(2);  
40  
41         iFace.inputCoordinate(ttt);  
42     }  
43     iFace.clearScreen();  
44     iFace.gameOver(ttt);  
45  
46 }while(mainLagi());  
47  
48 iFace.clearScreen();  
49 }  
50
```

```
=====  
mmmmmmmm " mmmmmmmmm mmmmmmmmm  
# mmm mmm # # " # " # " # "# "# "# "#  
# # # # " # " # " # " # " # "# "# "# "#  
# mm#mm "#mm" # "mm"# "#mm" # "#m#" "#mm"  
=====  
Masukkan nama player1 : Narto  
Masukkan nama player2 : Narti█  
  
▼ bool mainLagi(){  
    ▼ while(true){  
        cout << "Mau main lagi ? (y/n) : ";  
        char c;  
        cin >> c;  
        if(c == 'y'){  
            return true;  
        }else if(c== 'n'){  
            return false;  
        }  
    }  
}
```

MAIN.CPP

```
22 }  
23  
24 ▼ int main() {  
25 ▼ do{  
26     iFace.title();  
27     ticTacToe ttt(  
28         iFace.inputString("Masukkan nama player1 : "),  
29         iFace.inputString("Masukkan nama player2 : "),  
30         6  
31     );  
32  
33 ▼     while(!ttt.isFull()){  
34         iFace.clearScreen();  
35  
36         iFace.showScore(ttt);  
37         iFace.showBoard(ttt);  
38  
39         iFace.newLine(2);  
40  
41         iFace.inputCoordinate(ttt);  
42     }  
43     iFace.clearScreen();  
44     iFace.gameOver(ttt);  
45  
46 }while(mainLagi());  
47  
48 iFace.clearScreen();  
49 }  
50
```

=====
(X) Baya 3
(0) Bayu 2
=====

Board								
	\		-		-			
	\		0	0				
		\						
			\		/		/	
				/	/			
			/		/			

Visual								
	ca				cd			
	da	db			dd			
	ea	eb	ec			ef		
	fa	fb			fe	ff		

(0) Bayu giliran main, silakan input koordinat : █



MAIN.CPP

```
22 }  
23  
24 ▼ int main() {  
25 ▼ do{  
26     iFace.title();  
27     ticTacToe ttt(  
28         iFace.inputString("Masukkan nama player1 : "),  
29         iFace.inputString("Masukkan nama player2 : "),  
30         6  
31     );  
32  
33 ▼     while(!ttt.isFull()){  
34         iFace.clearScreen();  
35  
36         iFace.showScore(ttt);  
37         iFace.showBoard(ttt);  
38  
39         iFace.newLine(2);  
40  
41         iFace.inputCoordinate(ttt);  
42     }  
43     iFace.clearScreen();  
44     iFace.gameOver(ttt);  
45  
46 }while(mainLagi());  
47  
48 iFace.clearScreen();  
49 }  
50
```

OUR WINNER IS ...

(X) Baya!!!
Point : 3

Mau main lagi ? (y/n) : █

INTERFACE.H

```

> ...
1 #include<iostream>
2
3 using namespace std;
4
5 class Interface{
6 public:
7
8 void clearScreen(){
9 #ifdef WINDOWS
10     system("cls");
11 #else
12     // Assume POSIX
13     system("clear");
14 #endif
15 }
16
17 void title(){
18     this->newLine(2);
19     this->divider('=');
20     this->newLine();
21     cout << "mmmmmm \\" mmmmmmm mmmmmm\n";
22     cout << " #  mm   mm   #  mm   mm   #  mm   mm\n";
23     cout << " #   #   # \\" #   #   # \\" #   #   # \\" #\n";
24     cout << " #   #   #   m\\\"#   #   #   #   # \\"#\\\"#\n";
25     cout << " #   mm \\"#mm\" #   \\\"mm\"#   \\\"mm\"#   \\\"mm\"# \\"#mm\"#\n";
26     this->newLine();
27     this->divider('=');
28     this->newLine(2);
29 }
30
31 string inputString(string message){
Line 1: Col 19
    History ⌂
}

```

```

... 31    string inputString(string message){
32        cout << message;
33        string s;
34        cin >> s;
35        return s;
36    }
37
38 void divider(char style, int width=63, bool newLine=true){
39     if(width<1){
40         cout << "Invalid divider() usage: width must be 1 above";
41     }else{
42         for(int i=0; i<width; i++){
43             cout << style;
44         }
45         if(newLine) cout << endl;
46     }
47 }
48
49 void newLine(int lineCount=1){
50     if(lineCount<1){
51         cout << "Invalid newLine() usage: lineCount must be 1 above";
52     }else{
53         for(int i=0; i<lineCount; i++){
54             cout << "\n";
55         }
56     }
57 }
58
59 void showBoard(ticTacToe& game, string divider = " | ", int distance=10){
60
61     this->divider('-', 3 + 4*game.getDimension(), false);
62     this->divider(' ', distance, false);
63     this->divider('-', 3 + 5*game.getDimension());
64     cout << " Board";
65     this->divider(' ', 4*game.getDimension()+distance-2, false);
66     cout << "Visual" << endl;
67     this->divider('-', 3 + 4*game.getDimension(), false);
68     this->divider(' ', distance, false);
69     this->divider('-', 3 + 5*game.getDimension());
70
71
72 for(int i=0; i<game.getDimension(); i++){
73     cout << divider;
74     for(int j=0; j<game.getDimension(); j++){
75         cout << game.getCell(i,j);
76         cout << divider;
77     }
78     this->divider(' ', distance, false);
79     cout << divider;
80     for(int j=0; j<game.getDimension(); j++){
81         (game.getCell(i, j)==' ') ? cout << char(97+i) << char(97+j): cout << " ";
82         cout << divider;
83     }
84
85
86     this->newLine();
87     this->divider('-', 3 + 4*game.getDimension(), false);
88     this->divider(' ', distance, false);
89     this->divider('-', 3 + 5*game.getDimension());
Line 1: Col 19
    History ⌂
}

```

```

... 93 void showScore(ticTacToe& game){
94     this->divider('=');
95     cout << game.getPlayer1().name << "\t\t" << game.getPlayer1().score << endl;
96     cout << game.getPlayer2().name << "\t\t" << game.getPlayer2().score << endl;
97     this->divider('=');
98     this->newLine(2);
99 }
100
101 void inputCoordinate(ticTacToe& game){
102     string coordinate;
103     char val;
104     if(game.getXPlaying()){
105         coordinate = this->inputString(game.getPlayer1().name + " gitiran main, sila \\\n";
106         game.setXPlaying(false);
107         val = 'X';
108     }else{
109         coordinate = this->inputString(game.getPlayer2().name + " gitiran main, sila \\\n";
110         game.setXPlaying(true);
111         val = 'O';
112     }
113     int row = coordinate.at(0)-97;
114     int column = coordinate.at(1)-97;
115     if(game.getCell(row, column) == game.getInitValue()){
116         game.setCell(row, column, val);
117     }else{
118         game.getIsXPlaying()?game.setIsXPlaying(false):game.setIsXPlaying(true);
119     }
120     game.checkLine(row, column);
121 }
122
123 void gameOver(ticTacToe game){
124     newLine(5);
125     cout << "OUR WINNER IS ...";
126     divider('=');
127     newLine(2);
128     cout << game.getWinnerName() << "!!!!" << endl;
129     cout << "Point : " << game.getWinnerScore() << endl;;
130     newLine(2);
131     divider('=');
132     newLine(5);
133 }
134
135
Line 1: Col 19
    History ⌂
}

```

TICTACTOE.H

```

> ticTacToe > ...
5▼ class ticTacToe {
6 private:
7     int dimension;
8
9     char board[100][100];
10    char isXplaying = true;
11    char initValue;
12▼ struct player {
13     string name;
14     int score;
15 };
16    player player1, player2, winner;
17
18 public:
19▼ ticToe(string p1, string p2, int dimension = 3, char initialValue = ' ') {
20     this->player1 = {"(X) " + p1, 0};
21     this->player2 = {"(O) " + p2, 0};
22     this->dimension = dimension;
23     this->resetBoard(initialValue);
24     this->initValue = initialValue;
25 }
26
27▼ void resetBoard(char initialValue) {
28    for (int i = 0; i < dimension; i++) {
29        for (int j = 0; j < dimension; j++) {
30            board[i][j] = initialValue;
31        }
32    }
33 }
34 ▾ interactive / ...
35 93
36 //plus diagonal
37 for(int i=0; i<3; i++){
38     if(
39         this->board[row - 2 + i][column + 2 + i] == nowPlaying &&
40         this->board[row - 1 + i][column + 1 + i] == nowPlaying &&
41         this->board[row      + i][column      + i] == nowPlaying
42     ) {
43         this->board[row - 2 + i][column + 2 + i] = '/';
44         this->board[row - 1 + i][column + 1 + i] = '/';
45         this->board[row      + i][column      + i] = '/';
46         (this->isXplaying)?this->player2.score++:this->player1.score++;
47     }
48 }
49 //minus diagonal
50 for(int i=0; i<3; i++){
51     if(
52         this->board[row - 2 + i][column - 2 + i] == nowPlaying &&
53         this->board[row - 1 + i][column - 1 + i] == nowPlaying &&
54         this->board[row      + i][column      + i] == nowPlaying
55     ) {
56         this->board[row - 2 + i][column - 2 + i] = '＼＼';
57         this->board[row - 1 + i][column - 1 + i] = '＼＼';
58         this->board[row      + i][column      + i] = '＼＼';
59         (this->isXplaying)?this->player2.score++:this->player1.score++;
60     }
61 }
62 }catch (...) {}
63 // update score
64 if (player1.score > player2.score) {
65     winner = player1;
66 }
67 else if (player1.score < player2.score) {
68     winner = player2;
69 }
70 else {
71     winner = {"seri", player1.score};
72 }
73
74 bool getXPlaying() { return isXplaying; }
75
76 void setXPlaying(bool value) { this->isXplaying = value; }
77
78 char getInitValue() { return this->initValue; }
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137

```

```

char getCell(int row, int column) { return this->board[row][c
▼
void setCell(int row, int column, char value) {
    this->board[row][column] = value;
}
char getDimension() { return this->dimension; }
bool getIsXPlaying() { return this->isXplaying; }
void setIsXPlaying(bool value) { this->isXplaying = value; }

player getPlayer1() { return player1; }

player getPlayer2() { return player2; }

void setPlayer1Score(int score) { this->player1.score = score }
void setPlayer2Score(int score) { this->player2.score = score }

string getWinnerName() { return this->winner.name; }

int getWinnerScore() { return this->winner.score; }

bool isFull() {
    for (int i = 0; i < this->dimension)
        for (int j = 0; j < this->dimension)
            if (board[i][j] == initialValue) {
                return false;
            }
    }
    return true;
}

```

```

65
66▼ void checkLine(int row, int column) {
67    char nowPlaying = (this->isXplaying) ? '0' : 'X';
68    try {
69        // check vertical
70        for (int i = 0; i < 3; i++) {
71            if (this->board[row - 2 + i][column] == nowPlaying &&
72                this->board[row - 1 + i][column] == nowPlaying &&
73                this->board[row      + i][column] == nowPlaying)
74                this->board[row - 2 + i][column] = '|';
75                this->board[row - 1 + i][column] = '|';
76                this->board[row      + i][column] = '|';
77                // (this->isXplaying)?this->player2.score++:this->player1.score++;
78        }
79    }
80    //check horizontal
81    for(int i=0; i<3; i++){
82        if(
83            this->board[row][column - 2 + i] == nowPlaying &&
84            this->board[row][column - 1 + i] == nowPlaying &&
85            this->board[row][column      + i] == nowPlaying
86        ) {
87            this->board[row][column - 2 + i] = '-';
88            this->board[row][column - 1 + i] = '-';
89            this->board[row][column      + i] = '-';
90            (this->isXplaying)?this->player2.score++:this->player1.score++;
91        }
92    }
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137

```

RECAPT

interface.h

```
clearScreen()  
title()  
inputString()  
divider()  
newLine()  
showBoard()  
showScore()  
inputCoordinate()  
gameOver()
```

main.cpp

```
mainLagi()  
main()
```

ticTacToe.h

```
dimension  
board[][]  
inXplaying  
initValue  
ticTacToe()  
resetBoard()  
getCell()  
setCell()  
getDimension()  
setDimension()  
getIsXPlaying()  
setIsXPlaying()  
getPlayer1()  
getPlayer2()  
setPlayer1Score()  
setPlayer2Score()  
getWinnerName()  
getWinnerScore  
isFull()  
checkLine()  
getInitValue()
```



USAI SUDAH PRESENTASI KAMI

Seperti kisah cintamu yang sudah
usai walau belum dimulai

