

## C 言語：三目並べ (TicTacToe)

○×工業高校 機械工学科 2 年

2023 年 10 月 11 日

## 1 ゲームの概要

プログラムを実行すると、盤面が表示され、×の石を置く場所を指定するよう促されます。

画面上に示された番号を入力すると、その番号のスロットに×の石が置かれた盤面が表示され、次の手番の○に、石を置く場所を指定するように促されます。

手番を交互に変えながらゲームは進み、縦、横、斜めの何れかに、先に一行に自分の石を並べた方が勝ちとなります。

既に石の置かれているスロット番号を指定することはできません。また、スロット番号として 0 ～ 8 以外の数値を指定することもできません。

**スタート！ [Tic Tac Toe]**

```
/---|---|---\  
| 0 | 1 | 2 |  
|---|---|---|  
| 3 | 4 | 5 |  
|---|---|---|  
| 6 | 7 | 8 |  
\---|---|---/
```

'X' さんの turn です

石を置く場所 0 ～ 8 を指定して下さい : 4

```
/---|---|---\  
| 0 | 1 | 2 |  
|---|---|---|  
| 3 | X | 5 |  
|---|---|---|  
| 6 | 7 | 8 |  
\---|---|---/
```

'O' さんの turn です

石を置く場所 0 ～ 8 を指定して下さい : 2

```
/---|---|---\  
| 0 | 1 | 0 |  
|---|---|---|  
| 3 | X | 5 |  
|---|---|---|  
| 6 | 7 | 8 |  
\---|---|---/
```

'X' さんの turn です

石を置く場所 0 ～ 8 を指定して下さい :

## 2 主処理 (main)

```
1  int main(int argc, char *argv){
2      /* 先手後手を決定 */
3      int turn = BATSU, winner, num;
4      if (1 < argc){
5          if (!strcmp(argv[1], "-r"))
6              turn = MARU;
7      }
8      printf("スタート!_Tic_Tac_Toe\n");
9      do{
10         printBoard();          /* ①盤面の表示 */
11         num = slotNum(turn); /* ②手を入力 */
12         board[num] = turn;     /* ③手を盤面に配置 */
13         turn = switchTurn(turn); /* ④手番の交代 */
14         winner = checkWinner(); /* ⑤勝敗の判定 */
15     } while (winner == NEXT);
16     /* 対戦結果の表示 */
17     printBoard();
18     result(winner);
19     return 0;
20 }
```

## 3 盤面の表示 (printBoard)

```
1  void printBoard() {
2      char bd[9];
3      int i;
4      for (i = 0; i < 9; i++) {
5          if (board[i] == MARU) bd[i] = 'O';
6          else if (board[i] == BATSU) bd[i] = 'X';
7          else bd[i] = 'O' + i;
8      }
9      printf("\n/---|---|---\\n");
10     printf("|_%c_|_%c_|_%c_|_n", bd[0], bd[1], bd[2]);
11     printf("|---|---|---|_n");
12     printf("|_%c_|_%c_|_%c_|_n", bd[3], bd[4], bd[5]);
13     printf("|---|---|---|_n");
14     printf("|_%c_|_%c_|_%c_|_n", bd[6], bd[7], bd[8]);
15     printf("\\---|---|---/_n");
16 }
```

## 4 手番の交代 (switchTurn)

```
1  int switchTurn(int turn) {
2      if (turn == BATSU) return MARU;
3      return BATSU;
4  }
```

## 5 入力 (slotNum)

```
1  int slotNum(int turn) {
2      int num;
3      char *fig = "";
4      if (turn==MARU) fig = "'O'";
5      else if (turn==BATSU) fig = "'X'";
6      do {
7          printf("\n%sさんのturnです\n石を置く場所 0~8を指定して下さい
8              :", fig);
9          //while (getchar() != '\n'); /* 標準入力バッファのクリア */
10         scanf("%d", &num);
11         if (!(0 <= num && num < 9)) {
12             printf("再指定: 0~8を指定して下さい");
13             continue;
14         }
15         if (board[num] != num) {
16             printf("再指定: そこには既に石が置かれています\n");
17             continue;
18         }
19         break;
20     } while (1);
21     return num;
22 }
```

## 6 判定 (checkWinner)

```
1  int lineSum(int n1, int n2, int n3) {
2      return board[n1] + board[n2] + board[n3];
3  }
4  int checkWinner() {
5      int i, line = 0;
6      for (i = 0; i < 8; i++) {
7          switch (i) {
8              case 0: line = lineSum(0, 1, 2); break;
9              case 1: line = lineSum(3, 4, 5); break;
10             case 2: line = lineSum(6, 7, 8); break;
11             case 3: line = lineSum(0, 3, 6); break;
12             case 4: line = lineSum(1, 4, 7); break;
13             case 5: line = lineSum(2, 5, 8); break;
14             case 6: line = lineSum(0, 4, 8); break;
15             case 7: line = lineSum(2, 4, 6); break;
16         }
17         if (line == 3 * MARU) return MARU;
18         else if (line == 3 * BATSU) return BATSU;
19     }
20     for (i = 0; i < 9; i++){
21         if (0 <= board[i] && board[i] < 9) return NEXT;
22     }
23     return DRAW;
24 }
```

## 7 結果表示 (result)

```
1 void result(int winner) {
2     printf("\n");
3     switch (winner) {
4         case DRAW: printf("引き分け\t"); break;
5         case MARU: printf("'0'の勝ち\t"); break;
6         case BATSU: printf("'X'の勝ち\t"); break;
7     }
8     printf("またね!\n");
9 }
```

## 8 各種宣言など

これは冒頭に記述する

```
1  #include <stdio.h>
2  #include <string.h>
3
4  // function prototypes
5  /*
6  int lineSum(int, int, int);
7  int switchTurn(int);
8  void printBoard();
9  int slotNum(int);
10 int checkWinner();
11 void result(int);
12 */
13
14 static int board[] = {0, 1, 2, 3, 4, 5, 6, 7, 8};
15 #define MARU 10
16 #define BATSU -10
17 #define DRAW 100
18 #define NEXT 200
```

## 9 プログラムの全体

ソースコード 1 Tic Tac Toe(三目並べ)

```
1  #include <stdio.h>
2  #include <string.h>
3
4  // function prototypes
5  /*
6  int lineSum(int, int, int);
7  int switchTurn(int);
8  void printBoard();
9  int slotNum(int);
10 int checkWinner();
11 void result(int);
12 */
13
14 static int board[] = {0, 1, 2, 3, 4, 5, 6, 7, 8};
15 #define MARU 10
16 #define BATSU -10
17 #define DRAW 100
18 #define NEXT 200
19 /* ===== */
20 int switchTurn(int turn) {
21     if ( turn == BATSU ) return MARU;
22     return BATSU;
23 }
24 /* ===== */
25 void printBoard() {
26     char bd[9];
27     int i;
28     for (i = 0; i < 9; i++) {
29         if ( board[i] == MARU ) bd[i] = 'O';
30         else if ( board[i] == BATSU ) bd[i] = 'X';
31         else bd[i] = '0' + i;
32     }
33     printf("\n/---|---|---\\n");
34     printf("|_|_|_|_|_|_|_|_|", bd[0], bd[1], bd[2]);
35     printf("|---|---|---|n");
36     printf("|_|_|_|_|_|_|_|_|", bd[3], bd[4], bd[5]);
37     printf("|---|---|---|n");
38     printf("|_|_|_|_|_|_|_|_|", bd[6], bd[7], bd[8]);
39     printf("\\---|---|---/n");
40 }
41 /* ===== */
42 void result(int winner) {
43     printf("\n");
44     switch (winner) {
45     case DRAW: printf("引き分け\\t"); break;
46     case MARU: printf("'O'の勝ち\\t"); break;
47     case BATSU: printf("'X'の勝ち\\t"); break;
48     }
49     printf("またね!\\n");
50 }
51 /* ===== */
```

```

52 int slotNum(int turn) {
53     int num;
54     char *fig = "";
55     if ( turn == MARU ) fig = "'0'";
56     else if ( turn == BATSU ) fig = "'X'";
57     do {
58         printf("\n%sさんのturnです\n石を置く場所 0~8を指定して下さい:",
59             fig);
60         //while (getchar() != '\n'); /* 標準入力バッファのクリア */
61         scanf("%d", &num);
62         if (!(0 <= num && num < 9)) {
63             printf("再指定: 0~8を指定して下さい");
64             continue;
65         }
66         if (board[num] != num) {
67             printf("再指定: そこには既に石が置かれています\n");
68             continue;
69         }
70         break;
71     } while (1);
72     return num;
73 }
74 /* ===== */
75 int lineSum(int n1, int n2, int n3) {
76     return board[n1] + board[n2] + board[n3];
77 }
78 /* ===== */
79 int checkWinner() {
80     int i, line = 0;
81     for (i = 0; i < 8; i++) {
82         switch (i) {
83             case 0: line = lineSum(0, 1, 2); break;
84             case 1: line = lineSum(3, 4, 5); break;
85             case 2: line = lineSum(6, 7, 8); break;
86             case 3: line = lineSum(0, 3, 6); break;
87             case 4: line = lineSum(1, 4, 7); break;
88             case 5: line = lineSum(2, 5, 8); break;
89             case 6: line = lineSum(0, 4, 8); break;
90             case 7: line = lineSum(2, 4, 6); break;
91         }
92         if (line == 3 * MARU) return MARU;
93         else if (line == 3 * BATSU) return BATSU;
94     }
95     for (i = 0; i < 9; i++){
96         if (0 <= board[i] && board[i] < 9) return NEXT;
97     }
98     return DRAW;
99 }
100 /* ===== */
101 int main(int argc, char *argv[]){
102     /* 先手後手を決定 */
103     int turn = BATSU, winner, num;
104     if (1 < argc){
105         if (!strcmp(argv[1], "-r"))
106             turn = MARU;
107     }
108     printf("スタート! [Tic Tac Toe]\n");
109     do{

```

```
109     printBoard();          /* ① 盤面の表示 */
110     num = slotNum(turn);    /* ② 手を入力 */
111     board[num] = turn;      /* ③ 手を盤面に配置 */
112     turn = switchTurn(turn); /* ④ 手番の交代 */
113     winner = checkWinner(); /* ⑤ 勝敗の判定 */
114 } while (winner == NEXT);
115 /* 対戦結果の表示 */
116 printBoard();
117 result(winner);
118 return 0;
119 }
```

## 参考文献

- [1] 田中賢一郎、「ゲームで学ぶ Java Script 入門」、インプレス