

ASD OTH

NAMA : FARANDIO ALKHALID

NIM : 1203230081

PRODI : INFORMATIKA

KELAS : IF0301

Komponen Penilaian	Ya	Tidak
Soal 1 sesuai dengan output yang diinginkan		
Soal 2 sesuai dengan output yang diinginkan		
Bonus soal 1 dikerjakan		

1. Source Code :

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4
5  int minSwaps(int arr[], int n) {
6      int swaps = 0;
7
8      for (int i = 0; i < n - 1; i++) {
9          int minIndex = i;
10
11         for (int j = i + 1; j < n; j++) {
12             if (arr[j] < arr[minIndex])
13                 minIndex = j;
14         }
15
16         if (minIndex != i) {
17             swaps++;
18
19             int temp = arr[i];
20             arr[i] = arr[minIndex];
21             arr[minIndex] = temp;
22
23             printf("Pengurutan ke-%d:\n", swaps);
24             printf("Setelah pertukaran: ");
25             for (int k = 0; k < n; k++) {
26                 if (arr[k] >= 11 && arr[k] <= 13) {
27                     char card;
28                     switch (arr[k]) {
29                         case 11:
30                             card = 'J';
31                             break;
32                         case 12:
33                             card = 'Q';
34                             break;
35                         case 13:
36                             card = 'K';
37                             break;
38                     }
39                     printf("%c ", card);
40                 } else {
41                     printf("%d ", arr[k]);
42                 }
43             }
44             printf("\n\n");
45         }
46     }
47
48     return swaps;
49 }
50
51 int main() {
52     int n;
53     printf("Masukkan jumlah kartu: ");
54     scanf("%d", &n);
55
56     if (n <= 0 || n > 100) {
57         printf("Jumlah kartu tidak valid. Program berhenti.\n");
58         return 1;
59     }
60
61     int arr[n];
62     char str[3 * n];
63
64     printf("Masukkan urutan kartu (pisahkan dengan spasi): ");
65     getchar();
66     fgets(str, sizeof(str), stdin);
67
68     char *token = strtok(str, " ");
69     for (int i = 0; i < n && token != NULL; i++) {
70         if (token[0] >= '2' && token[0] <= '9')
71             arr[i] = atoi(token);
72         else
73             arr[i] = (token[0] == 'J') ? 11 : (token[0] == 'Q') ? 12 : (token[0] == 'K') ? 13 : 0;
74
75         token = strtok(NULL, " ");
76     }
77
78     if (token != NULL) {
79         printf("Jumlah kartu tidak sesuai. Program berhenti.\n");
80         return 1;
81     }
82
83     int swaps = minSwaps(arr, n);
84     printf("Jumlah minimum swaps yang diperlukan untuk mengurutkan kartu: %d\n", swaps);
85
86     return 0;
87 }
```

Output :

```
● Masukkan jumlah kartu: 8
○ Masukkan urutan kartu (pisahkan dengan spasi): 9 4 2 J K 8 4 Q
Pengurutan ke-1:
Setelah pertukaran: 2 4 9 J K 8 4 Q

Pengurutan ke-2:
Setelah pertukaran: 2 4 4 J K 8 9 Q

Pengurutan ke-3:
Setelah pertukaran: 2 4 4 8 K J 9 Q

Pengurutan ke-4:
Setelah pertukaran: 2 4 4 8 9 J K Q

Pengurutan ke-5:
Setelah pertukaran: 2 4 4 8 9 J Q K

Jumlah minimum swaps yang diperlukan untuk mengurutkan kartu: 5
PS E:\Documents\ITTS\SEMESTER 2\ALPRO\0TH ASDOS>
```

Penjelasan :

Pada fungsi minSwaps adalah untuk ngitung berapa kali kita harus tukeran angka biar bisa urutin arraynya. Buat urutinnya, dia pakai algoritma Selection Sort. Nah, setiap kali tukeran, dia juga bakal nunjukin langkah-langkahnya satu per satu. Kalo angkanya itu 11, 12, atau 13, bakal ditampilkan sebagai 'J', 'Q', sama 'K'.

2. Source Code :

```
1  #include <stdio.h>
2
3  void koboImaginaryChess(int i, int j, int chessBoard[8][8]) {
4      int moves[8][2] = {{-2, -1}, {-1, -2}, {1, -2}, {2, -1},
5                          {2, 1}, {1, 2}, {-1, 2}, {-2, 1}};
6
7      for (int k = 0; k < 8; k++) {
8          int newX = i + moves[k][0];
9          int newY = j + moves[k][1];
10         if (newX >= 0 && newX < 8 && newY >= 0 && newY < 8) {
11             chessBoard[newX][newY] = 1;
12         }
13     }
14 }
15
16 void printChessBoard(int chessBoard[8][8]) {
17     for (int i = 0; i < 8; i++) {
18         for (int j = 0; j < 8; j++) {
19             printf("%d ", chessBoard[i][j]);
20         }
21         printf("\n");
22     }
23 }
24
25 int main() {
26     int i, j;
27     scanf("%d %d", &i, &j);
28     int chessBoard[8][8] = {{0}};
29     koboImaginaryChess(i, j, chessBoard);
30     printChessBoard(chessBoard);
31     return 0;
32 }
33
```

Output :

```
2 2
0 1 0 1 0 0 0 0
1 0 0 0 1 0 0 0
0 0 0 0 0 0 0 0
1 0 0 0 1 0 0 0
0 1 0 1 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
PS E:\Documents\ITTS\SEMESTER 2\ALPRO\OTH ASDOS> cd "e:\D
"; if ($?) { gcc tempCodeRunnerFile.c -o tempCodeRunnerF
3 7
0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0
0 0 0 0 0 1 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0
0 0 0 0 0 0 1 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
PS E:\Documents\ITTS\SEMESTER 2\ALPRO\OTH ASDOS>
```

Penjelasan :

Mengimplementasikan permainan catur imajinasi "Kobo Imaginary Chess". Fungsi `koboImaginaryChess` menandai sel-sel yang dapat diakses oleh sebuah bidak pada papan catur 8x8, sedangkan fungsi `printChessBoard` mencetak papan catur yang telah dimodifikasi. Dengan memasukkan koordinat posisi awal bidak melalui input, program menampilkan papan catur dengan sel-sel yang dapat diakses oleh bidak tersebut.