

LAPORAN PRAKTIKUM

NAMA : AISYAH

KELAS : IF-03-03

NIM : 1203230015

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 dan penjelasan

```
1 #include <stdio.h> //berfungsi untuk standar input output
2 #include <stdlib.h> //berfungsi untuk mengalokasikan blok memori secara dinamis
3 #include <string.h> //berfungsi untuk fungsi-fungsi yang berkaitan dengan string
4
5 // Fungsi untuk mendapatkan nilai angka dari kartu
6 int getCardValue(char kartu) { //berfungsi untuk menerima 1 parameter yaitu 'card' dan mengembalikan sebuah nilai
7     if (kartu == 'J') return 11; // jika karakter nya adalah 'J' maka fungsi akan mengembalikan nilai 11
8     else if (kartu == 'Q') return 12; //jika karakter nya adalah 'Q' maka fungsi akan mengembalikan nilai 12
9     // else if (kartu == 'K') return 13; // jika karakter nya adalah 'K' maka fungsi akan mengembalikan nilai 13
10    else if (kartu == '1') return 10; //jika karakter nya adalah '1' maka fungsi akan mengembalikan nilai 10
11    else return (int)(kartu - '0'); //jika karakter nya adalah angka 2-9 maka nilai
12 }
13
14 // Fungsi untuk menampilkan urutan kartu
15 void printCards(char *cards, int length) { //berfungsi untuk menerima dua parameter yg merupakan array karakter(string) yg akan dicetak
16     for (int i = 0; i < length; i++) { //perulangan yg akan berjalan dimana 'i' akan bertambah 1 setiap iterasi
17         printf("%c ", cards[i]); //berfungsi untuk mencetak setiap iterasi berupa karakter yg terdapat pada indeks 'i' dari array 'cards'
18     }
19     printf("\n"); //berfungsi untuk mencetak karakter baris baru setelah perulangan selesai, sehingga output selanjutnya akan dimulai dari baris baru
20 }
21
22 // Fungsi untuk mengurutkan kartu
23 int sortCards(char *cards, int length) { //berfungsi untuk mengembalikan jumlah pertukaran yang dilakukan
24     int swaps = 0; //deklarasi variabel untuk menghitung jumlah pertukaran yg dilakukan
25     for (int i = 0; i < length - 1; i++) { //perulangan pertama untuk mengiterasi setiap kartu kecuali yg terakhir
26         int min_idx = i; //deklarasi variabel yg merupakan indeks kartu terkecil dlm sisa array
27         for (int j = i + 1; j < length; j++) { //perulangan kedua untuk mencari indeks kartu terkecil diantara kartu yg blm ter urutkan
28             // konversi kartu ke nilai angka untuk membandingkan
29             if (getCardValue(cards[j]) < getCardValue(cards[min_idx])) { //setiap iterasi akan dibandingkan dgn nilai angka kartu, jika lebih kecil diupdate dgn nilai j
30                 min_idx = j;
31             }
32         }
33         if (min_idx != i) //untuk memeriksa apakah indeks kartu terkecil
34         {
35             char temp = cards[i]; //deklarasi variabel sementara untuk menyimpan nilai kartu pada indeks 'i'
36             cards[i] = cards[min_idx]; //berfungsi untuk kartu dgn nilai terkecil yg akan dipindahkan ke posisi yg benar
37             cards[min_idx] = temp; //variabel sementara untuk menyimpan indeks 'i' yg akan dipindahkan ke tempat yg benar
38             swaps++; //jumlah pertukaran akan ditambah 1
39
40             printf("Pertukaran %d : ", swaps); //untuk mencetak pesan dan kartu beserta nomor pertukarannya
41             printCards(cards, length); //berfungsi untuk memanggil fungsi dan mencetak urutan kartu setelah pertukaran
42         }
43     }
44     return swaps; //untuk mengembalikan jumlah total pertukaran yg dilakukan
45 }
46
47 int main() { // untuk membuka atau memulai membuat program dan fungsi utama
48     int n; //deklarasi variabel yg akan menyimpan jumlah kartu yg akan dimasukkan user
49     printf("Masukkan jumlah kartu: "); //berfungsi untuk mencetak pesan dan inputan user
50     scanf("%d", &n); //untuk membaca inputan dari user
51
52     char cards[n]; //deklarasi array variabel yg digunakan untuk menyimpan nilai dari setiap kartu yg diinput oleh user
53
54     printf("Masukkan nilai kartu : "); //berfungsi untuk mencetak pesan dan inputan user
55     for (int i = 0; i < n; i++) { //perulangan untuk memasukkan nilai dari setiap kartu dlm array
56         scanf("%c", &cards[i]); //untuk membaca inputan dari user
57     }
58
59     int swaps = sortCards(cards, n); //berfungsi untuk mengurutkan kartu yg telah dimasukkan oleh user
60
61     printf("\nJumlah minimal langkah pertukaran kartu: %d\n", swaps); //berfungsi untuk mencetak pesan dan mengurutkan kartu yg diinput oleh user
62
63     free(cards); //untuk membebaskan memori yg dialokasikan untuk array 'cards'
64     return 0; // berfungsi untuk mengakhiri fungsi dan program
65 }
66 }
```

2. Output

```
PS C:\Users\istiyono\SEMESTER 2.C> .\soal1.exe
Masukkan jumlah kartu: 5
Masukkan nilai kartu : 3 2 8 7 4
Pertukaran 1 : 2 3 8 7 4
Pertukaran 2 : 2 3 4 7 8

Jumlah minimal langkah pertukaran kartu: 2
PS C:\Users\istiyono\SEMESTER 2.C> █
```

```
PS C:\Users\istiyono\SEMESTER 2.C> .\soal1.exe
Masukkan jumlah kartu: 8
Masukkan nilai kartu : 9 4 2 J K 8 4 Q
Pertukaran 1 : 2 4 9 J K 8 4 Q
Pertukaran 2 : 2 4 4 J K 8 9 Q
Pertukaran 3 : 2 4 4 8 K J 9 Q
Pertukaran 4 : 2 4 4 8 9 J K Q
Pertukaran 5 : 2 4 4 8 9 J Q K

Jumlah minimal langkah pertukaran: 5
PS C:\Users\istiyono\SEMESTER 2.C> █
```

3. Source code & penjelasan

```
1 #include <stdio.h> //berfungsi untuk standar input output
2
3 // Fungsi untuk mengecek apakah posisi (a, b) valid pada papan catur 8x8
4 int isValid(int a, int b) { //deklarasi variabel a dan b berupa integer
5     return (a >= 0 && a < 8 && b >= 0 && b < 8); //berfungsi untuk mengembalikan nilai 1 jika posisi tsb berada dalam batas papan catur(0-7)
6     //dan akan mengembalikan nilai 0 jika berada diluar batas papan catur
7 }
8
9 // Fungsi untuk menandai semua langkah yang mungkin akan dilakukan oleh kuda pada papan catur
10 void koboImaginaryChess(int i, int j, int size, int *papanCatur) {
11     int switchA[] = {2, 1, -1, -2, -2, -1, 1, 2}; //array untuk menyimpan perubahan baris yg mungkin terjadi dlm setiap langkah kuda
12     int switchB[] = {1, 2, 2, 1, -1, -2, -2, -1}; //array untuk menyimpan perubahan kolom yg mungkin terjadi dlm setiap langkah kuda
13
14     for (int k = 0; k < 8; k++) { //perulangan akan berjalan sebanyak 8 kali sesuai dgn jumlah langkah yg mungkin dilakukan oleh bidak kuda
15         int nextA = i + switchA[k]; //berfungsi untuk menghitung posisi baris berikutnya yg akan dicapai oleh bidak kuda
16         int nextB = j + switchB[k]; //berfungsi untuk menghitung posisi kolom berikutnya yg akan dicapai oleh bidak kuda
17         if (isValid(nextA, nextB)) { //untuk mengecek apakah posisi '(nextA, nextB)' valid di dlm papan catur
18             *(papanCatur + nextA * size + nextB) = 1; // Menandai langkah kuda dengan nilai 1
19         }
20     }
21 }
22
23 int main() { // untuk membuka atau memulai membuat program dan fungsi utama
24     int i, j; //deklarasi variabel bertipe data integer yg akan dipakai untuk menyimpan posisi awal bidak kuda
25     scanf("%d %d", &i, &j); //untuk membaca inputan dari user dan disimpan di dalam variabel i dan j
26
27     int size = 8; // deklarasi variabel untuk menentukan Ukuran papan catur (8x8)
28     int papanCatur[size][size]; //deklarasi Array 2D untuk papan catur
29
30     // Inisialisasi papan catur dengan nilai awal 0
31     for (int a = 0; a < size; a++) { //perulangan awal untuk baris pertama yg akan mengiterasi baris pada papan catur
32         for (int b = 0; b < size; b++) { //perulangan awal untuk kolom pertama yg akan mengiterasi baris pada papan catur
33             papanCatur[a][b] = 0; //berfungsi untuk mengatur nilai setiap tempat pada papan catur menjadi 0
34         }
35     }
36
37     koboImaginaryChess(i, j, size, (int *)papanCatur); // Memanggil fungsi untuk menandai langkah kuda
38
39     // Menampilkan papan catur setelah langkah kuda ditandai
40     for (int a = 0; a < size; a++) { //perulangan awal untuk baris pertama yg akan mengiterasi baris pada papan catur
41         for (int b = 0; b < size; b++) { //perulangan awal untuk kolom pertama yg akan mengiterasi baris pada papan catur
42             printf("%d ", papanCatur[a][b]); //untuk mencetak nilai dari setiap tempat pada papan catur
43         }
44         printf("\n"); //berfungsi untuk membuat baris baru setelah mengecek semua kolom dlm satu baris pada papan catur
45     }
46
47     return 0; // berfungsi untuk mengakhiri fungsi dan program
48 }
```

4. Output

```
PS C:\Users\istiyono\SEMESTER 2.C> .\soal2.exe
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 C:\Users\istiyono\SEMESTER 2.C> █
```

```
PS C:\Users\istiyono\SEMESTER 2.C> .\soal2.exe
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 C:\Users\istiyono\SEMESTER 2.C> █
```