

NAMA : RICHARD EDGAR GONASSIS

KELAS : IF 03 – 03

NIM : 1203230084

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  // Fungsi untuk mengubah nilai kartu
6  int valueKartu(char kartu[]) {
7      // Jika kartu adalah ("J")
8      if (strcmp(kartu, "J") == 0) {
9          return 11; // Jika J, kembalikan 11
10     }
11     // Jika kartu adalah ("Q")
12     else if (strcmp(kartu, "Q") == 0) {
13         return 12; // Jika Q, kembalikan 12
14     }
15     // Jika kartu adalah ("K")
16     else if (strcmp(kartu, "K") == 0) {
17         return 13; // Jika K, kembalikan 13
18     }
19     //Jika kartu adalah "10"
20     else if (strcmp(kartu, "10") == 0) {
21         return 10; // Jika "10", kembalikan 10
22     }
23     // Selain kondisi diatas, konversi string menjadi integer
24     else {
25         return atoi(kartu); // Konversi string menjadi integer
26     }
27 }
28
29 // Fungsi untuk mengoutput tiap pertukaran
30 void pertukaran(int* kartu, int n, int petukar) {
31     printf("Pertukaran %d : ", petukar); // Cetak nomor pertukaran
32     for (int i = 0; i < n; i++) {
33         // Periksa nilai setiap kartu
34         if (kartu[i] == 11){
35             printf("J "); // Jika 11, cetak "J"
36         } else if (kartu[i] == 12){
37             printf("Q "); // Jika 12, cetak "Q"
38         } else if (kartu[i] == 13){
39             printf("K "); // Jika 13, cetak "K"
40         } else {
41             printf("%d ", kartu[i]); // Jika bukan maka, output nilai
42         }
43     }
44     printf("\n"); // Output baris kosong
45 }
46
```

```

1  int main() {
2      //Declare Variable
3      int n;
4      int tukar = 0;
5      scanf("%d", &n); // Input jumlah kartu
6      int kartu[100]; // Array untuk menyimpan nilai kartu
7      // Loop untuk membaca nilai setiap kartu
8      for (int i = 0; i < n; i++) {
9          char nilai[3]; // Array untuk menyimpan nilai kartu
10         scanf("%s", nilai); // Input nilai kartu
11         kartu[i] = valueKartu(nilai); // Konversi dan simpan nilai kartu
12     }
13     // Loop untuk melakukan pengurutan kartu dengan metode seleksi
14     for (int i = 0; i < n; i++){
15         int min = i; // Set variable min ke indeks i
16         // Loop untuk mencari nilai minimum kartu
17         for (int j = i + 1; j < n; j++){
18             if(kartu[j] < kartu[min]){
19                 min = j; // Perbarui indeks nilai minimum
20             }
21         }
22         // Jika nilai minimum bukan posisi awal, maka lakukan pertukaran
23         if (min != i){
24             int temp = kartu[i]; // Simpan nilai kartu i ke variabel temp
25             kartu[i] = kartu[min]; // Tukar nilai kartu i dengan nilai minimum
26             kartu[min] = temp; // Simpan nilai minimum di posisi i
27             tukar++; // Tambah jumlah pertukaran
28             pertukaran(kartu, n, tukar); // Panggil fungsi pertukaran untuk mencetak setiap pertukaran
29         }
30     }
31     printf("%d", tukar); // Cetak jumlah pertukaran
32     return 0;
33 }

```

OUTPUT :

<pre> 4 6 6 9 7 Pertukaran 1 : 6 6 7 9 1 </pre>	<pre> 8 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 5 </pre>
<pre> 5 3 2 8 7 4 Pertukaran 1 : 2 3 8 7 4 Pertukaran 2 : 2 3 4 7 8 2 </pre>	<pre> 6 10 J K Q 3 2 Pertukaran 1 : 2 J K Q 3 10 Pertukaran 2 : 2 3 K Q J 10 Pertukaran 3 : 2 3 10 Q J K Pertukaran 4 : 2 3 10 J Q K 4 </pre>

## 2. SOURCE CODE

```
1  #include <stdio.h>
2
3  void koboImaginaryChess(int i, int j, int size, int *papan) { //Fungsi untuk menandai setiap Langkah
4      // Mengecek setiap kemungkinan
5      int gerak[8][2] = {{-2, -1}, {-2, 1}, {2, -1}, {2, 1}, {-1, -2}, {-1, 2}, {1, -2}, {1, 2}};
6      for (int k = 0; k < 8; k++) { // Menandai setiap Langkah
7          int temp_i = i + gerak[k][0];
8          int temp_j = j + gerak[k][1];
9          // Mengecek apakah posisi baru berada dalam papan
10         if (temp_i >= 0 && temp_i < size && temp_j >= 0 && temp_j < size) {
11             *(papan + temp_i * size + temp_j) = 1;
12         }
13     }
14 }
15
16 int main() {
17     int i, j;
18     scanf("%d %d", &i, &j); // Input koordinat
19     int papan[8][8] = {0}; // Declare papan catur
20     koboImaginaryChess(i, j, 8, (int *)papan); // Panggil fungsi koboImaginaryChess
21
22     for (int baris = 0; baris < 8; baris++) { // Loop untuk mencetak papan catur yang telah ditandai
23         for (int kolom = 0; kolom < 8; kolom++) {
24             printf("%d ", papan[baris][kolom]); // Output nilai setiap posisi pada papan
25         }
26         printf("\n"); // Output baris berikutnya
27     }
28
29     return 0;
30 }
```

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
```

Process returned 0 (0x0) execution time : 22.032 s  
Press any key to continue.  
|

```
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
```

Process returned 0 (0x0) execution time : 1.925 s  
Press any key to continue.  
|