

Nama : Josefania Tirsa Putri Immanuely
Kelas : IF 03-02
NIM : 1203230012

1. KARTU

a) Source Code

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

// Fungsi untuk mendapatkan nilai angka dari kartu
int getCardValue(char card) { //mengembalikan nilai numerik dari kartu tersebut
    if (card == 'J') return 11; //menggunakan karakter J untuk mengembaliakn nilai 11
    else if (card == 'Q') return 12; //menggunakan karakter Q untuk mengembalikan nilai 12
    else if (card == 'K') return 13; //menggunakan karakter K untuk mengembalikan nilai 13
    else if (card == '1') return 10; //menggunakan karakter 1 untuk mengembalikan nilai 10
    else return (int)(card - '0'); //menutup fungsi dari getCardValue
}

// Fungsi untuk menampilkan urutan kartu
void printCards(char *cards, int length) {
    for (int i = 0; i < length; i++) { //melakukan literasi pada setiap elemen dalam karakter 'cards'
        printf("%c ", cards[i]); //untuk mencetak karakter
    }
    printf("\n");
}

//fungsi untuk mengurutkan kartu
int sortCards(char *cards, int length) { //mengurutkan dalam urutan non-menurun
    int swaps = 0; //menghitung jumlah swap yang dilakukan
    for (int i = 0; i < length - 1; i++) {
        int min_idx = i; // untuk menyimpan indeks elemen yang memiliki nilai minimum
        for (int j = i; j < length; j++) {
            if (getCardValue(cards[j]) < getCardValue(cards[min_idx])) { //mengonversi kartu ke
                nilai angka untuk membandingkan
                min_idx = j;
            }
        }
        if (min_idx != i)
        {
            char temp = cards[i]; //menyimpan elemen dari indeks 'i' sebelum pertukaran
```

```

        cards[i] = cards[min_idx]; //menyimpan nilai dari elemen pada indeks 'min_idx' ke indeks
        'i'
        cards[min_idx] = temp; //untuk menyimpan nilai dari elemen pada indeks 'i' ke indeks
        'min_idx'
        swaps++;

        printf("Pertukaran %d : ", swaps); //untuk menghitung jumlah swap yang dilakukan
        printCards(cards, length);
    }

}

return swaps; //setelah elemen diurutkan, fungsi mengembalikan nilai 'swaps'
}

int main() { //mengembalikan nilai 0
    int n;
    printf("Masukkan jumlah kartu: "); //untuk menginputkan jumlah kartu
    scanf("%d", &n); //mengambil input dari pengguna dan menyimpan nilai tersebut dalam
    variabel 'n'

    char cards[n]; //membuat array karakter dengan panjang yang sama dengan nilai yang
    diinputkan pengguna

    printf("Masukkan nilai kartu (1-10, J, Q, K): ");
    for (int i = 0; i < n; i++) { //melakukan iterasi pada setiap elemen dalam array karakter
        scanf(" %c", &cards[i]); //mengambil input dari pengguna dan menyimpan nilai tersebut
        dalam elemen array karakter
    }

    int swaps = sortCards(cards, n);

    printf("\nJumlah minimal langkah pertukaran: %d\n", swaps); //mengembalikan jumlah
    langkah pertukaran yang dilakukan

    free(cards); //membebaskan memori yang digunakan oleh array karakter
    return 0;
}

```

b) Ouput

```
PS C:\Users\LENOVO\Documents\coba> cd "c:\Users\LENOVO\Documents\coba"
Masukkan jumlah kartu: 4
Masukkan nilai kartu (1-10, J, Q, K): 6 6 9 7
Pertukaran 1 : 6 6 7 9

Jumlah minimal langkah pertukaran: 1
PS C:\Users\LENOVO\Documents\coba> cd "c:\Users\LENOVO\Documents\coba"
Masukkan jumlah kartu: 5
Masukkan nilai kartu (1-10, J, Q, K): 3 2 8 7 4
Pertukaran 1 : 2 3 8 7 4
Pertukaran 2 : 2 3 4 7 8

Jumlah minimal langkah pertukaran: 2
PS C:\Users\LENOVO\Documents\coba> cd "c:\Users\LENOVO\Documents\coba"
Masukkan jumlah kartu: 6
Masukkan nilai kartu (1-10, J, Q, K): 10 J K Q 3 2
Pertukaran 1 : 0 1 J K Q 3
Pertukaran 2 : 0 3 J K Q 1
Pertukaran 3 : 0 3 1 K Q J
Pertukaran 4 : 0 3 1 J Q K

Jumlah minimal langkah pertukaran: 4
PS C:\Users\LENOVO\Documents\coba> cd "c:\Users\LENOVO\Documents\coba"
Masukkan jumlah kartu: 8
Masukkan nilai kartu (1-10, J, Q, K): 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
```

2. CATUR

a) Source Code

```
#include <stdio.h>

//fungsi untuk mengecek apakah posisi (x, y) valid pada papan catur 8x8
int isValid(int x, int y) { //mengembalikan nilai true jika angka x dan y berada di dalam batasan 0
sampai dengan 7
    return (x >= 0 && x < 8 && y >= 0 && y < 8);
}

void koboImaginaryChess(int i, int j, int size, int *chessBoard) { //menyimpan langkah-langkah
yang mungkin dilakukan oleh kuda pada papan catur.
    //langkah-langkah yang mungkin dilakukan oleh kuda
    int movesX[] = {2, 1, -1, -2, -2, -1, 1, 2};
    int movesY[] = {1, 2, 2, 1, -1, -2, -2, -1};

    //menandai setiap langkah yang mungkin dilakukan oleh kuda
    for (int k = 0; k < 8; k++) { //untuk melakukan iterasi pada setiap elemen dalam array 'movesX'
dan 'movesY'
        int nextX = i + movesX[k];
        int nextY = j + movesY[k];
```

```

        if (isValid(nextX, nextY)) { //memeriksa apakah langkah yang diambil dari array 'movesX'
dan 'movesY' adalah valid.
            *(chessBoard + nextX * size + nextY) = 1; // menandai langkah kuda dengan nilai 1
        }
    }
}

int main() {
    int i, j;
    scanf("%d %d", &i, &j); //membaca input posisi i dan j

    int size = 8; //ukuran papan catur
    int chessBoard[size][size]; //array 2D untuk papan catur

    //inisialisasi papan catur dengan nilai awal 0
    for (int x = 0; x < size; x++) { //perulangan akan terus berjalan selama nilai 'x' kurang dari
variabel size, setiap iterasi 'x' akan ditambahkan 1
        for (int y = 0; y < size; y++) { //perulangan akan terus berjalan selama nilai 'y' kurang dari
variabel size, setiap iterasi 'y' akan ditambahkan 1
            chessBoard[x][y] = 0; //menginisialisasi nilai 0 ke elemen pada posisi '[x]''[y]' dalam array
2D 'chessBoard'
        }
    }

    koboImaginaryChess(i, j, size, (int *)chessBoard); //memanggil fungsi untuk menandai langkah
kuda

    //menampilkan papan catur setelah langkah kuda ditandai
    for (int x = 0; x < size; x++) {
        for (int y = 0; y < size; y++) {
            printf("%d ", chessBoard[x][y]);
        }
        printf("\n");
    }

    return 0;
}

```

b) Ouput

```
PS C:\Users\LENOVO\Documents\coba> cd "c
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\LENOVO\Documents\coba> cd "c
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\LENOVO\Documents\coba> |
```

KOMPONEN NILAI	YA	TIDAK
Soal 1 sesuai dengan output yang diinginkan		
Soal 2 sesuai dengan output yang diinginkan		
Bonus soal 1 dikerjakan		