Nama : Gardha Dananjaya

NIM : 1203230105

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

SOAL LATIHAN 1

SS Full Code

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

// Fungsi untuk mengkonversi nilai kartu ke angka
int nilaiKartu(char kartu[]) {
    if (strcmp(kartu, "J") == 0) return 11;
    if (strcmp(kartu, "Q") == 0) return 12;
    if (strcmp(kartu, "K") == 0) return 13;
    return atoi(kartu); // Mengkonversi string ke integer
}

// Fungsi utama
int main() {

    int banyak_angka;
    printf("Masukkan Banyaknya angka: ");
    scanf("%d", &banyak_angka); // Membaca jumlah kartu

    char kartu[banyak_angka][3]; // Array untuk menyimpan nilai kartu
    for(int i = 0; i < banyak_angka; i++) {
        scanf("%s", kartu[i]); // Membaca nilai kartu
    }

    // Mengurutkan kartu menggunakan Selection Sort
    int langkahPertukaran = 0;
    for(int i = 0; i < banyak_angka - 1; i++) {
        int minIndex = i;
        for(int j = i+1; j < banyak_angka; j++) {
            if(nilaiKartu(kartu[j]) < nilaiKartu(kartu[minIndex])) {
                minIndex = j;
            }
        }
        if(minIndex != i) {
            // Pertukaran posisi kartu
            char temp[3];
            strcpy(temp, kartu[i]);
            strcpy(kartu[i], kartu[minIndex]);
            strcpy(kartu[minIndex], temp);
            langkahPertukaran++;

            printf("\nPertukaran %d: ", langkahPertukaran);
            for (int L = 0; L < banyak_angka; L++) {
                printf("%s ", kartu[L]);
            }
        }
    }

    printf("\nJumlah langkah pertukaran: %d\n", langkahPertukaran);
    return 0;
}
```

SS dan Penjelasan

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

// Fungsi untuk mengkonversi nilai kartu ke angka
int nilaiKartu(char kartu[]) {
    if (strcmp(kartu, "J") == 0) return 11;
    if (strcmp(kartu, "Q") == 0) return 12;
    if (strcmp(kartu, "K") == 0) return 13;
    return atoi(kartu); // Mengkonversi string ke integer
}

int main() {

    int banyak_angka;   // deklarasi variabel
    printf("Masukkan Banyaknya angka: ");
    scanf("%d", &banyak_angka); // Membaca jumlah kartu

    char kartu[banyak_angka][3]; // Array untuk menyimpan nilai kartu
    for(int i = 0; i < banyak_angka; i++) {
        scanf("%s", kartu[i]); // Membaca nilai kartu
    }

    // Mengurutkan kartu menggunakan Selection Sort
    int langkahPertukaran = 0;      // variabel untuk menampung langkah pertukaran
    for(int i = 0; i < banyak_angka - 1; i++) {
        int minIndex = i;   // variabel yang menyimpan indeks terkecil
        for(int j = i+1; j < banyak_angka; j++) {
            if(nilaiKartu(kartu[j]) < nilaiKartu(kartu[minIndex])) {        // untuk cek jika nilai kartu j < minIndex maka
                minIndex = j;   // minIndex menjadi = j
            }
        }
        if(minIndex != i) { // jika angka terkecil tidak = i
            char temp[3];           // variabel sementara untuk menyimpan kartu
            strcpy(temp, kartu[i]); // copy kartu ke variabel sementara
            strcpy(kartu[i], kartu[minIndex]);  // copy kartu terkecil ke kartu [i]
            strcpy(kartu[minIndex], temp);  // copy kartu dari variabel temp ke kartu terkecil
            langkahPertukaran++;
```

```c
int main() {
    }

    // Mengurutkan kartu menggunakan Selection Sort
    int langkahPertukaran = 0;      // variabel untuk menampung langkah pertukaran
    for(int i = 0; i < banyak_angka - 1; i++) {
        int minIndex = i;   // variabel yang menyimpan indeks terkecil
        for(int j = i+1; j < banyak_angka; j++) {
            if(nilaiKartu(kartu[j]) < nilaiKartu(kartu[minIndex])) {        // untuk cek jika nilai kartu j < minIndex maka
                minIndex = j;   // minIndex menjadi = j
            }
        }
        if(minIndex != i) { // jika angka terkecil tidak = i
            char temp[3];           // variabel sementara untuk menyimpan kartu
            strcpy(temp, kartu[i]); // copy kartu ke variabel sementara
            strcpy(kartu[i], kartu[minIndex]);  // copy kartu terkecil ke kartu [i]
            strcpy(kartu[minIndex], temp);  // copy kartu dari variabel temp ke kartu terkecil
            langkahPertukaran++;

            printf("\nPertukaran %d: ", langkahPertukaran);
            for (int L = 0; L < banyak_angka; L++) {
                printf("%s ", kartu[L]);        // mencetak semua hasil kartu yang telah ditukar
            }
        }
    }

    printf("\nJumlah langkah pertukaran: %d\n", langkahPertukaran);
    return 0;
}
```

SS Output

```
PS C:\VsCode\C 2\output> & .\'OTH 1.exe'
Masukkan Banyaknya angka: 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
Jumlah langkah pertukaran: 5
PS C:\VsCode\C 2\output>
```
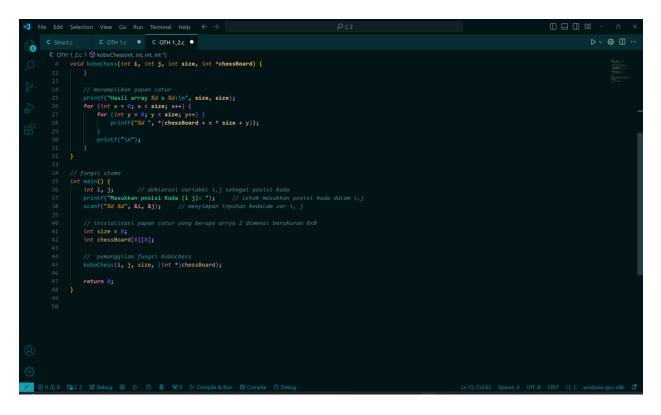
LATIHAN SOAL 2

SS Full Code

```c
#include <stdio.h>

void koboChess(int i, int j, int size, int *chessBoard) {
    for (int x = 0; x < size; x++) {
        for (int y = 0; y < size; y++) {
            *(chessBoard + x * size + y) = 0;
        }
    }

    int moves[8][2] = {{-2, -1}, {-1, -2}, {1, -2}, {2, -1},
                        {2, 1},   {1, 2},   {-1, 2}, {-2, 1}};

    for (int k = 0; k < 8; k++) {
        int new_i = i + moves[k][0];
        int new_j = j + moves[k][1];
        if (new_i >= 0 && new_i < size && new_j >= 0 && new_j < size) {
            *(chessBoard + new_i * size + new_j) = 1;
        }
    }

    printf("Hasil array %d x %d:\n", size, size);
    for (int x = 0; x < size; x++) {
        for (int y = 0; y < size; y++) {
            printf("%d ", *(chessBoard + x * size + y));
        }
        printf("\n");
    }
}

int main() {
    int i, j;
    printf("Masukkan posisi Kuda (i j): ");
    scanf("%d %d", &i, &j);

    int size = 8;
    int chessBoard[8][8];

    koboChess(i, j, size, (int *)chessBoard);

    return 0;
}
```

SS dan Penjelasan


```c
#include <stdio.h>

// Inisialisasi papan catur
void koboChess(int i, int j, int size, int *chessBoard) {
    for (int x = 0; x < size; x++) {
        for (int y = 0; y < size; y++) {
            *(chessBoard + x * size + y) = 0;
        }
    }

    // mendefinisikan semua gerakan yang bisa dilakukan kuda
    int moves[8][2] = {{-2, -1}, {-1, -2}, {1, -2}, {2, -1},
                       {2, 1},   {1, 2},   {-1, 2}, {-2, 1}};

    // cek setiap gerakan kuda jika benar maka posisi di papan diubah menjadi 1
    for (int k = 0; k < 8; k++) {
        int new_i = i + moves[k][0];
        int new_j = j + moves[k][1];
        if (new_i >= 0 && new_i < size && new_j >= 0 && new_j < size) {
            *(chessBoard + new_i * size + new_j) = 1;
        }
    }

    // menampilkan papan catur
    printf("Hasil array %d x %d:\n", size, size);
    for (int x = 0; x < size; x++) {
        for (int y = 0; y < size; y++) {
            printf("%d ", *(chessBoard + x * size + y));
        }
        printf("\n");
    }
}

// fungsi utama
int main() {
    int i, j;        // deklarasi variabel i,j sebagai posisi kuda
    printf("Masukkan posisi Kuda (i j): ");    // cetak masukkan posisi kuda dalam i,j
    scanf("%d %d", &i, &j);    // menyimpan inputan kedalam var i, j
```



```c
void koboChess(int i, int j, int size, int *chessBoard) {
    }

    // menampilkan papan catur
    printf("Hasil array %d x %d:\n", size, size);
    for (int x = 0; x < size; x++) {
        for (int y = 0; y < size; y++) {
            printf("%d ", *(chessBoard + x * size + y));
        }
        printf("\n");
    }
}

// fungsi utama
int main() {
    int i, j;        // deklarasi variabel i,j sebagai posisi kuda
    printf("Masukkan posisi Kuda (i j): ");    // cetak masukkan posisi kuda dalam i,j
    scanf("%d %d", &i, &j);    // menyimpan inputan kedalam var i, j

    // inisialisasi papan catur yang berupa arrya 2 dimensi berukuran 8x8
    int size = 8;
    int chessBoard[8][8];

    //  pemanggilan fungsi kobochess
    koboChess(i, j, size, (int *)chessBoard);

    return 0;
}
```

SS Output

```
PS C:\VsCode\C 2\output> cd 'c:\VsCode\C 2\output'
PS C:\VsCode\C 2\output> & .\'OTH 1_2.exe'
Masukkan posisi Kuda (i j): 2 2
Hasil array 8 x 8:
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:\VsCode\C 2\output> & .\'OTH 1_2.exe'
Masukkan posisi Kuda (i j): 3 7
Hasil array 8 x 8:
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:\VsCode\C 2\output> []
```