

Nama : muamar haikal fauzan

Nim : 1203230118

IF 03-02

## SOURCE CODE

1.

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

// Fungsi untuk mendapatkan nilai numerik dari kartu
int card_value(char card) {
    if (card >= '2' && card <= '9') {
        return card - '0'; // Konversi karakter ke integer
    } else if (card == 'J') {
        return 11;
    } else if (card == 'Q') {
        return 12;
    } else if (card == 'K') {
        return 13;
    } else {
        return 10;
    }
}

// Fungsi untuk menukar posisi dua kartu
void swap(char *x, char *y) {
    char temp = *x;
    *x = *y;
    *y = temp;
}

// Fungsi untuk melakukan pengurutan menggunakan bubble sort
int bubble_sort(char arr[], int n) {
    int i, j, steps = 0;
    for (i = 0; i < n-1; i++) {
        for (j = 0; j < n-i-1; j++) {
            if (card_value(arr[j]) > card_value(arr[j+1])) {
```

```

        swap(&arr[j], &arr[j+1]);
        steps++;
    }
}
}
return steps;
}

int main() {
    int n, steps;
    char cards[100];

    // Membaca jumlah kartu
    scanf("%d", &n);

    // Membaca nilai-nilai kartu
    getchar(); // Mengonsumsi karakter newline (\n) setelah membaca jumlah kartu
    fgets(cards, sizeof(cards), stdin);

    // Melakukan pengurutan dan mendapatkan jumlah langkah yang dibutuhkan
    steps = bubble_sort(cards, n);

    // Menampilkan jumlah langkah
    printf("%d\n", steps);

    return 0;
}

```

## 2.

```

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

// Fungsi untuk menentukan apakah suatu posisi valid di papan catur atau tidak
int isValidPosition(int x, int y) {
    return (x >= 0 && x < 8 && y >= 0 && y < 8);
}

```

```

// Fungsi untuk menandai posisi yang dapat dicapai oleh kuda
void markAccessiblePositions(int i, int j, int *chessBoard) {
    // Daftar langkah yang mungkin dilakukan oleh kuda
    int moves[8][2] = {
        {-2, -1}, {-2, 1}, {2, -1}, {2, 1},
        {-1, -2}, {-1, 2}, {1, -2}, {1, 2}
    };

    // Memeriksa setiap langkah yang mungkin
    for (int k = 0; k < 8; k++) {
        int x = i + moves[k][0];
        int y = j + moves[k][1];
        if (isValidPosition(x, y)) {
            chessBoard[x * 8 + y] = 1; // Menandai posisi yang dapat dicapai
        }
    }
}

```

```

// Fungsi untuk mencetak papan catur
void printChessBoard(int *chessBoard) {
    for (int i = 0; i < 8; i++) {
        for (int j = 0; j < 8; j++) {
            printf("%d ", chessBoard[i * 8 + j]);
        }
        printf("\n");
    }
}

```

```

// Fungsi utama
void kobolmaginaryChess(int i, int j, int *chessBoard) {
    // Mengosongkan papan catur
    for (int k = 0; k < 64; k++) {
        chessBoard[k] = 0;
    }

    // Menandai posisi yang dapat dicapai oleh kuda
    markAccessiblePositions(i, j, chessBoard);
}

```

```
}

int main() {
    int i, j;
    int chessBoard[64] = {0}; // Inisialisasi papan catur dengan nilai 0

    // Membaca posisi kuda
    scanf("%d %d", &i, &j);

    // Memeriksa dan menandai posisi yang dapat dicapai oleh kuda
    kobolmaginaryChess(i, j, chessBoard);

    // Mencetak papan catur dengan posisi yang dapat dicapai oleh kuda
    printChessBoard(chessBoard);

    return 0;
}
```

## HASIL OUTPUT

```
1 cd "/var/folders/Bd/pprh4gh53h13srgdjt_29f240000gn/T/" && gcc tempCodeRunnerFile.c -o tempCodeRunnerFile && "/var/folders/Bd/pprh4gh53h13srgdjt_29f240000gn/T/"tempCodeRunnerFile
4
6 6 9 7
1
muamarhaikal@haikals-Air T %
```

```
2. cd "/var/folders/Bd/pprh4gh53h13srgdjt_29f240000gn/T/" && gcc tempCodeRunnerFile.c -o tempCodeRunnerFile && "/var/folders/Bd/pprh4gh53h13srgdjt_29f240000gn/T/"tempCodeRunnerFile
4
6 6 9 7
1
muamarhaikal@haikals-Air T %
```

## PENJELASAN

### Code 1

#### 1. **card\_value(char card):**

- o Fungsi ini mengembalikan nilai numerik dari sebuah kartu.
- o Menggunakan nilai numerik untuk J, Q, K, dan 10 adalah 11, 12, 13, dan 10, sesuai aturan permainan kartu.

	<ul style="list-style-type: none"> <li>o Mengonversi karakter angka menjadi nilai integer dengan mengurangi karakter '0' dari nilai karakter tersebut.</li> </ul>
2.	<b>swap(char *x, char *y):</b> <ul style="list-style-type: none"> <li>o Fungsi ini bertujuan untuk menukar posisi dua kartu dalam array.</li> <li>o Menggunakan pointer untuk menukar nilai dua variabel.</li> </ul>
3.	<b>bubble_sort(char arr[], int n):</b> <ul style="list-style-type: none"> <li>o Fungsi ini menggunakan algoritma bubble sort untuk mengurutkan array kartu.</li> <li>o Membandingkan setiap elemen berturut-turut dalam array dan menukar posisinya jika diperlukan.</li> <li>o Menghitung jumlah langkah yang diperlukan untuk mengurutkan array.</li> </ul>
4.	<b>main():</b> <ul style="list-style-type: none"> <li>o Fungsi utama program.</li> <li>o Membaca jumlah kartu dan nilai-nilai kartu dari input.</li> <li>o Memanggil fungsi bubble_sort untuk mengurutkan kartu-kartu tersebut.</li> <li>o Menampilkan jumlah langkah yang diperlukan untuk mengurutkan kartu-kartu tersebut.</li> </ul>

## Code 2

1.	<b>isValidPosition(int x, int y):</b> <ul style="list-style-type: none"> <li>o Fungsi ini mengembalikan nilai 1 jika posisi (x, y) berada dalam batas papan catur (8x8), dan 0 jika tidak.</li> <li>o Digunakan untuk memastikan bahwa kuda tidak akan melangkah keluar dari papan catur.</li> </ul>
2.	<b>markAccessiblePositions(int i, int j, int *chessBoard):</b> <ul style="list-style-type: none"> <li>o Fungsi ini menandai posisi-posisi yang dapat dicapai oleh kuda dalam satu langkah.</li> <li>o Menerima posisi awal kuda (i, j) dan array yang mewakili papan catur.</li> <li>o Menggunakan daftar langkah yang mungkin dilakukan oleh kuda untuk memeriksa setiap posisi yang dapat dicapai.</li> <li>o Menandai posisi-posisi yang valid dengan nilai 1 pada papan catur.</li> </ul>
3.	<b>printChessBoard(int *chessBoard):</b> <ul style="list-style-type: none"> <li>o Fungsi ini mencetak papan catur yang telah ditandai dengan posisi-posisi yang dapat dicapai oleh kuda.</li> <li>o Menggunakan array yang mewakili papan catur untuk mencetak setiap elemen.</li> <li>o Angka 1 menunjukkan posisi yang dapat dicapai oleh kuda, sedangkan angka 0 menunjukkan posisi yang tidak dapat dicapai.</li> </ul>
4.	<b>kobolmaginaryChess(int i, int j, int *chessBoard):</b>

- Fungsi utama untuk melakukan simulasi permainan catur.
- Menginisialisasi papan catur dengan nilai 0.
- Memanggil fungsi markAccessiblePositions untuk menandai posisi yang dapat dicapai oleh kuda.

5. **main():**

- Fungsi utama program.
- Membaca posisi awal kuda dari input.
- Memanggil fungsi kobolmaginaryChess untuk melakukan simulasi.
- Mencetak papan catur hasil simulasi.

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