

Nama : Ahmad Wahyudi

NIM : 1203230116

Kelas : IF 03-02

Tugas algoritma dan struktur data

Code 1

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

int card_value(char card) {
    if (card >= '2' && card <= '9') {
        return card - '0';
    } else if (card == 'J') {
        return 11;
    } else if (card == 'Q') {
        return 12;
    } else if (card == 'K') {
        return 13;
    }
    return -1;
}

int min_swaps_to_sort(char cards[], int n) {
    int swaps = 0;
    for (int i = 0; i < n - 1; i++) {
        int min_idx = i;
        for (int j = i + 1; j < n; j++) {
            if (card_value(cards[j]) < card_value(cards[min_idx])) {
                min_idx = j;
            }
        }
        if (min_idx != i) {
            char temp = cards[i];
            cards[i] = cards[min_idx];
            cards[min_idx] = temp;
            swaps++;
            printf("Pertukaran %d: %s\n", swaps, cards);
        }
    }
    return swaps;
}

int main() {
    int n;
    scanf("%d", &n);
    char cards[n+1];
    scanf(" %[^\\n]", cards);
```

```

    int min_swaps = min_swaps_to_sort(cards, n);
    printf("Jumlah minimal langkah pertukaran: %d\n", min_swaps);

    return 0;
}

```

Code 2

```

#include <stdio.h>

// Fungsi untuk mengecek apakah posisi (x, y) valid pada papan catur 8x8
int isValid(int x, int y) {
    return (x >= 0 && x < 8 && y >= 0 && y < 8);
}

// Fungsi untuk menandai semua langkah yang mungkin dilakukan oleh kuda pada
papan catur
void koboImaginaryChess(int i, int j, int size, int *chessBoard) {
    // 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++) {
        int nextX = i + movesX[k];
        int nextY = j + movesY[k];
        if (isValid(nextX, nextY)) {
            *(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++) {
        for (int y = 0; y < size; y++) {
            chessBoard[x][y] = 0;
        }
    }
}

```

```
    }  
}  
  
    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;  
}
```

Tugas no 1

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4
5  int card_value(char card) {
6      if (card >= '2' && card <= '9') {
7          return card - '0';
8      } else if (card == 'J') {
9          return 11;
10     } else if (card == 'Q') {
11         return 12;
12     } else if (card == 'K') {
13         return 13;
14     }
15     return -1;
16 }
17
18 int min_swaps_to_sort(char cards[], int n) {
19     int swaps = 0;
20     for (int i = 0; i < n - 1; i++) {
21         int min_idx = i;
22         for (int j = i + 1; j < n; j++) {
23             if (card_value(cards[j]) < card_value(cards[min_idx])) {
24                 min_idx = j;
25             }
26         }
27         if (min_idx != i) {
28             char temp = cards[i];
29             cards[i] = cards[min_idx];
30             cards[min_idx] = temp;
31             swaps++;
32             printf("Pertukaran %d: %s\n", swaps, cards);
33         }
34     }
35     return swaps;
36 }
37
38 int main() {
39     int n;
40     scanf("%d", &n);
41     char cards[n+1];
42     scanf(" %[^\\n]", cards);
43
44     int min_swaps = min_swaps_to_sort(cards, n);
45     printf("Jumlah minimal langkah pertukaran: %d\\n", min_swaps);
46
47     return 0;
48 }
49
```

Tugas no 2

```
1  #include <stdio.h>
2
3  // Fungsi untuk mengecek apakah posisi (x, y) valid pada papan catur 8x8
4  int isValid(int x, int y) {
5      return (x >= 0 && x < 8 && y >= 0 && y < 8);
6  }
7
8  // Fungsi untuk menandai semua langkah yang mungkin dilakukan oleh kuda pada papan catur
9  void koboImaginaryChess(int i, int j, int size, int *chessBoard) {
10     // Langkah-langkah yang mungkin dilakukan oleh kuda
11     int movesX[] = {2, 1, -1, -2, -2, -1, 1, 2};
12     int movesY[] = {1, 2, 2, 1, -1, -2, -2, -1};
13
14     // Menandai setiap langkah yang mungkin dilakukan oleh kuda
15     for (int k = 0; k < 8; k++) {
16         int nextX = i + movesX[k];
17         int nextY = j + movesY[k];
18         if (isValid(nextX, nextY)) {
19             *(chessBoard + nextX * size + nextY) = 1; // Menandai langkah kuda dengan nilai 1
20         }
21     }
22 }
23
24 int main() {
25     int i, j;
26     scanf("%d %d", &i, &j); // Membaca input posisi i dan j
27
28     int size = 8; // Ukuran papan catur
29     int chessBoard[size][size]; // Array 2D untuk papan catur
30
31     // Inisialisasi papan catur dengan nilai awal 0
32     for (int x = 0; x < size; x++) {
33         for (int y = 0; y < size; y++) {
34             chessBoard[x][y] = 0;
35         }
36     }
37
38     koboImaginaryChess(i, j, size, (int *)chessBoard); // Memanggil fungsi untuk menandai langkah kuda
39
40     // Menampilkan papan catur setelah langkah kuda ditandai
41     for (int x = 0; x < size; x++) {
42         for (int y = 0; y < size; y++) {
43             printf("%d ", chessBoard[x][y]);
44         }
45         printf("\n");
46     }
47
48     return 0;
49 }
```

Penjelasan

Tugas 1

Program ini berfungsi untuk menghitung jumlah langkah untuk mengurutkan kartu, pada permainan kartu

- card_value berfungsi untuk menghitung nilai dari sebuah kartu.
- min_swaps_to_short berfungsi untuk menghitung jumlah minimal langkah pertukaran untuk mengurutkan sebuah kartu

```
PS C:\Users\Ahmad Wahyudi> cd "d:\program\Praktikum\" ;  
8  
9 4 2 J K 8 4 Q  
Pertukaran 1: 94 2 J K 8 4 Q  
Pertukaran 2: 492 J K 8 4 Q  
Pertukaran 3: 924J K 8 4 Q  
Pertukaran 4: 24J9K 8 4 Q  
Pertukaran 5: 249JK 8 4 Q  
Jumlah minimal langkah pertukaran: 5  
PS D:\program\Praktikum>
```

Tugas 2

Program ini berfungsi untuk mengetahui posisi mana saja yang bisa dicapai oleh bidak kuda pada permainan catur berukuran 8x8

- isValid berfungsi untuk mengecek posisi (x,y) yang valid pada papan catur
- koboImaginaryChess berfungsi untuk menandai semua yang mungkin bisa dilakukan oleh bidak catur
- main berfungsi untuk membaca posisi, ukuran dan menginisialisasikan papan catur

```
PS C:\Users\Ahmad Wahyudi> cd "d:\program\Praktikum\"  
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 D:\program\Praktikum>
```