

Rizkia Putra Raditya

1203230083

### Penjelasan soal 1

```
PS C:\Users\Molly> cd "d:\TUGAS PUTE\" ; if ($?) { gcc SOAL_OTH.c -o SOAL_OTH } ; if ($?) { .\SOAL_OTH }
Masukkan Jumlah Kartu: 5
Masukkan Nilai Kartu: 3 4 J K Q
Minimum pengacakan: 3
Setelah Pengacakan 1: 0 3 4 16 32
Setelah Pengacakan 2: 0 3 4 16 32
Setelah Pengacakan 3: 0 3 4 16 32
Setelah Pengacakan 4: 0 3 4 16 32
Setelah Pengacakan 5: 0 3 4 16 32
PS D:\TUGAS PUTE> cd "d:\TUGAS PUTE\" ; if ($?) { gcc SOAL_OTH.c -o SOAL_OTH } ; if ($?) { .\SOAL_OTH }
Masukkan Jumlah Kartu: 4
Masukkan Nilai Kartu: 6 6 9 5
Minimum pengacakan: 3
Setelah Pengacakan 1: 5 6 6 9
Setelah Pengacakan 2: 5 6 6 9
Setelah Pengacakan 3: 5 6 6 9
Setelah Pengacakan 4: 5 6 6 9
PS D:\TUGAS PUTE> cd "d:\TUGAS PUTE\" ; if ($?) { gcc SOAL_OTH.c -o SOAL_OTH } ; if ($?) { .\SOAL_OTH }
Masukkan Jumlah Kartu: 3
Masukkan Nilai Kartu: 1 2 3
Minimum pengacakan: 0
Setelah Pengacakan 1: 1 2 3
Setelah Pengacakan 2: 1 2 3
Setelah Pengacakan 3: 1 2 3
PS D:\TUGAS PUTE> 
```

Fungsi ini menghitung jumlah minimum pengacakan yang diperlukan untuk mengurutkan kartu secara menaik. Berikut uraiannya:

int swaps = 0; Inisialisasi variabel swaps ke 0 untuk melacak total jumlah pengacakan yang dilakukan.

int swapped = 1; Inisialisasi variabel swapped ke 1, dengan asumsi mungkin ada pengacakan yang diperlukan di iterasi pertama.

while (swapped): Loop ini terus berjalan selama swapped bernilai true, menunjukkan bahwa setidaknya satu pengacakan terjadi pada iterasi sebelumnya.

swapped = 0; Mengatur ulang swapped ke 0 sebelum setiap iterasi loop dalam, dengan asumsi tidak ada pengacakan yang mungkin terjadi pada iterasi ini.

for (int i = 0; i < n - 1; i++): Melakukan iterasi melalui array cards hingga elemen n-1 karena elemen terakhir sudah berada di posisi yang benar setelah pengurutan.

if (cards[i] > cards[i + 1]): Memeriksa apakah kartu saat ini (cards[i]) lebih besar daripada kartu berikutnya (cards[i + 1]).

swap(&cards[i], &cards[i + 1]); Jika kondisinya benar, memanggil fungsi swap untuk menukar nilai kartu saat ini dan berikutnya.

swaps++; Menambah penghitung swaps sebesar 1 untuk melacak pengacakan.

swapped = 1;; Mengatur swapped ke 1 lagi, menunjukkan bahwa pengacakan terjadi pada iterasi ini, sehingga loop luar akan terus berjalan.

return swaps;; Setelah loop selesai, fungsi mengembalikan nilai akhir dari swaps, yang mewakili jumlah minimum pengacakan yang diperlukan untuk pengurutan.

## Penjelasan soal 2

```
Masukkan posisi awal kuda (baris kolom): 3 7
Simulasi papan catur:
00000000
00000011
00000101
00000110
00000101
00000011
00000000
00000000
```

```
Masukkan posisi awal kuda (baris kolom): 2 2
Simulasi papan catur:
01110000
10101000
11011000
10101000
01110000
00000000
00000000
00000000
```

kobolmaginaryChess(i, j, size, chessBoard) fungsi ini mensimulasikan gerakan kuda pada papan catur. Parameter i dan j mewakili posisi awal kuda, size adalah ukuran papan catur, dan chessBoard adalah pointer ke array 2 dimensi yang menyimpan simulasi papan catur.

for (int di = -2; di <= 2; di++) { ... } loop ini iterasi melalui semua kemungkinan gerakan kuda yang bisa dilakukan. di dan dj mewakili perubahan posisi baris dan kolom dari posisi awal kuda.

if (abs(di) == abs(dj) || ... ) { continue; } kondisi ini untuk mengecek apakah gerakan kuda tersebut valid. Kuda hanya bisa bergerak berbentuk L, sehingga gerakan diagonal yang tidak membentuk L ( $\text{abs}(di) == \text{abs}(dj)$ ) dan gerakan yang keluar dari papan catur (posisi setelah bergerak kurang dari 0 atau lebih dari ukuran papan catur) diabaikan dengan continue.

chessBoard[(i + di) \* size + j + dj] = 1; setelah gerakan yang valid ditemukan, posisi yang dapat dicapai kuda ditandai dengan nilai 1 pada array chessBoard.

Pada fungsi main, program:

Mendeklarasikan ukuran papan catur (size) dan menginisialisasi array chessBoard dengan nilai 0 (menandakan posisi kosong).

Meminta input untuk posisi awal kuda (i dan j).

Memanggil fungsi kobilmaginaryChess untuk menghitung gerakan kuda yang mungkin dari posisi awal tersebut.

Mencetak hasil simulasi papan catur, dengan angka 1 menunjukkan posisi yang dapat dicapai oleh kuda.