

Soal Latihan Prak ALPRO – Array, Pointer, dan Fungsi

Nama : Muhammad Arifbillah Kamil

NIM : 1203230028

Kelas : IF-03-01

1. Soal Pertama

```
2. #include <stdio.h>
3. #include <string.h>
4.
5. void swap(int* a, int* b) {
6.     int temp = *a;
7.     *a = *b;
8.     *b = temp;
9. }
10.
11. void printCard(int number) {
12.     if (number == 1) printf("1 ");
13.     else if (number == 2) printf("2 ");
14.     else if (number == 3) printf("3 ");
15.     else if (number == 4) printf("4 ");
16.     else if (number == 5) printf("5 ");
17.     else if (number == 6) printf("6 ");
18.     else if (number == 7) printf("7 ");
19.     else if (number == 8) printf("8 ");
20.     else if (number == 9) printf("9 ");
21.     else if (number == 10) printf("10 ");
22.     else if (number == 11) printf("J ");
23.     else if (number == 12) printf("Q ");
24.     else if (number == 13) printf("K ");
25.     else printf("%d ", number);
26. }
27.
28. void sortCards(int n, int *arr) {
29.     int swaps = 0;
30.
31.     for (int i = 0; i < n - 1; i++) {
32.         for (int j = 0; j < n - i - 1; j++) {
33.             if (arr[j] > arr[j + 1]) {
34.                 swap(&arr[j], &arr[j + 1]);
```

```

35.         swaps++;
36.
37.         printf("Langkah %d: Pertukaran antara kartu %d (%c) dan
kartu %d (%c)\n", swaps, j+1, arr[j], j+2, arr[j+1]);
38.
39.         printf("Kartu setelah pertukaran %d: ", swaps);
40.         for (int k = 0; k < n; k++) {
41.             printCard(arr[k]);
42.         }
43.         printf("\n");
44.     }
45. }
46. }
47.
48.     printf("Jumlah pertukaran yang dilakukan : %d\n", swaps);
49. }
50.
51. int main() {
52.     int n;
53.     printf("Masukkan jumlah kartu : ");
54.     scanf("%d", &n);
55.
56.     int arr[n];
57.     char temp[3];
58.     printf("Masukkan angka kartu (1-10, J/Q/K) : ");
59.     for (int i = 0; i < n; i++) {
60.         scanf("%s", temp);
61.         if (strcmp(temp, "J") == 0) {
62.             arr[i] = 11;
63.         } else if (strcmp(temp, "Q") == 0) {
64.             arr[i] = 12;
65.         } else if (strcmp(temp, "K") == 0) {
66.             arr[i] = 13;
67.         } else {
68.             sscanf(temp, "%d", &arr[i]);
69.         }
70.     }
71.
72.     sortCards(n, arr);
73.
74.     return 0;
75. }

```

Output :

```

Masukkan jumlah kartu : 5
Masukkan angka kartu (1-10, J/Q/K) : 5 J Q K 3
Langkah 1: Pertukaran antara kartu 4 (♥) dan kartu 5 (
Kartu setelah pertukaran 1: 5 J Q 3 K
Langkah 2: Pertukaran antara kartu 3 (♥) dan kartu 4 (
)
Kartu setelah pertukaran 2: 5 J 3 Q K
Langkah 3: Pertukaran antara kartu 2 (♥) dan kartu 3 (
)
Kartu setelah pertukaran 3: 5 3 J Q K
Langkah 4: Pertukaran antara kartu 1 (♥) dan kartu 2 (♣)
Kartu setelah pertukaran 4: 3 5 J Q K
Jumlah pertukaran yang dilakukan : 4

```

Penjelasan :

1. **swap**: Fungsi ini menukar nilai dua variabel integer. Ini menggunakan pointer untuk mengubah nilai asli variabel yang dilewatkan ke fungsi.
2. **printCard**: Fungsi ini mencetak representasi teks dari nomor kartu. Untuk angka 1 hingga 10, ia mencetak angka tersebut. Untuk 11, 12, dan 13, ia mencetak 'J', 'Q', dan 'K' yang mewakili Jack, Queen, dan King.
3. **sortCards**: Fungsi ini mengurutkan array kartu menggunakan algoritma bubble sort. Selama proses pengurutan, fungsi ini juga mencetak setiap langkah pertukaran yang dilakukan dan kondisi array setelah pertukaran.
4. **main**: Fungsi utama ini meminta pengguna untuk memasukkan jumlah kartu dan nilai-nilai kartu. Nilai-nilai ini bisa berupa angka 1-10 atau huruf 'J', 'Q', 'K' yang mewakili kartu bergambar. Setelah nilai-nilai dimasukkan, fungsi sortCards dipanggil untuk mengurutkan kartu dan mencetak proses pengurutan.

2. Soal Kedua

```
#include <stdio.h>

void koboImaginaryChess(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}, {-2, 1}, {2, -1}, {2, 1}, {-1, -2}, {-1, 2}, {1,
-2}, {1, 2}};

```

```

        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;
            }
        }
    }

int main() {
    int i, j;
    int chessBoard[8][8];

    printf("Masukkan nilai i dan j (0 <= i, j < 8): ");
    scanf("%d %d", &i, &j);

    koboImaginaryChess(i, j, 8, &chessBoard[0][0]);

    for (int x = 0; x < 8; x++) {
        for (int y = 0; y < 8; y++) {
            printf("%d ", chessBoard[x][y]);
        }
        printf("\n");
    }

    return 0;
}

```

Output :

```

Masukkan nilai i dan j (0 <= i, j < 8): 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\diarno\Documents\VSS\Kuliah\Semester2\TUGAS>
lcatur } ; if ($?) { .\soalcatatur }
Masukkan nilai i dan j (0 <= i, j < 8): 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

```

Penjelasan :

1. **Fungsi** kobolmaginaryChess:

- Menerima empat parameter: i dan j untuk posisi awal kuda pada papan catur, size untuk ukuran papan catur, dan pointer ke array chessBoard yang merepresentasikan papan catur.
- Menginisialisasi seluruh papan catur dengan nilai 0, yang berarti tidak ada kuda di posisi tersebut.
- Mendefinisikan array moves yang berisi 8 kemungkinan gerakan kuda dalam catur.
- Melakukan iterasi melalui setiap gerakan yang mungkin. Jika gerakan tersebut valid (tidak keluar dari papan), maka posisi tersebut di papan catur ditandai dengan 1, yang menunjukkan bahwa kuda dapat bergerak ke posisi tersebut.

2. **Fungsi** main:

- Mendeklarasikan variabel i, j, dan array dua dimensi chessBoard yang merepresentasikan papan catur.
- Meminta pengguna untuk memasukkan nilai i dan j, yang merupakan koordinat awal kuda pada papan catur.
- Memanggil fungsi kobolmaginaryChess dengan koordinat yang dimasukkan pengguna dan pointer ke array chessBoard.
- Setelah fungsi kobolmaginaryChess dipanggil, program mencetak papan catur yang telah diperbarui dengan menunjukkan semua posisi yang dapat dicapai oleh kuda dari posisi awalnya.

Letakkan tabel di bawah ini pada halaman pertama laporan untuk mempermudah dalam proses pengecekan!

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