

Nama: Ramadian Abid Permana

Kelas: IF 03-02

NIM: 1203230026

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

Code 1.

```
void swap(int *a, int *b) {  
    int temp = *a;  
    *a = *b;  
    *b = temp;  
}
```

Baris ini mendeklarasikan fungsi bernama `swap` yang mengambil dua pointer bilangan bulat (`int \*a` dan `int \*b`) sebagai parameter. Fungsi ini tidak mengembalikan nilai apa pun (`void`)

```
int main() {  
    int n;  
    scanf("%d", &n);  
  
    int cards[n];  
    for (int i = 0; i < n; i++) {  
        char c;  
        scanf(" %c", &c);  
        if (c >= '0' && c <= '9') {  
            cards[i] = c - '0';  
        } else {  
            cards[i] = 10 + (c - 'J');  
        }  
    }  
}
```

Meminta pengguna untuk memasukkan bilangan bulat `n` yang mewakili jumlah kartu. Fungsi `scanf` dengan penentu format `%d` membaca bilangan bulat dari input standar (keyboard) dan menyimpannya dalam variabel `n`

```
int swaps = 0;
for (int i = 0; i < n - 1; i++) {
    for (int j = i + 1; j < n; j++) {
        if (cards[i] > cards[j]) {
            swaps++;

            swap(&cards[i], &cards[j]);
        }
    }
}
```

Mengimplementasikan algoritma Bubble Sort di C untuk mengurutkan array bilangan bulat yang disimpan dalam array `cards`

```
printf("%d\n", swaps);

return 0;
}
```

Baris ini mencetak nilai variabel `swaps`, yang mewakili jumlah swap yang dilakukan selama proses penyortiran, diikuti dengan karakter baris baru (`\n`). `%d` adalah penentu format untuk mencetak bilangan bulat, dan `swaps` adalah variabel bilangan bulat yang nilainya akan dicetak.

Kembali



```
1  #include <stdio.h>
2
3  void swap(int *a, int *b) {
4      int temp = *a;
5      *a = *b;
6      *b = temp;
7  }
8
9  int main() {
10     int n;
11     scanf("%d", &n);
12
13     int cards[n];
14     for (int i = 0; i < n; i++) {
15         char c;
16         scanf(" %c", &c);
17         if (c >= '0' && c <= '9') {
18             cards[i] = c - '0';
19         } else {
20             cards[i] = 10 + (c - 'J');
21         }
22     }
23
24     int swaps = 0;
25     for (int i = 0; i < n - 1; i++) {
26         for (int j = i + 1; j < n; j++) {
27             if (cards[i] > cards[j]) {
28                 swaps++;
29
30                 swap(&cards[i], &cards[j]);
31             }
32         }
33     }
34
35     printf("%d\n", swaps);
36
37     return 0;
38 }
```

## Code 2

```
void koboImaginaryChess(int i, int j, int chessBoard[][8]){
    int moves[8][2] = {{-2, -1}, {-1, -2},{1, -2}, {2, -1}, {2, 1}, {1, 2}, {-1, 2},{-2, 1}};

    for(int move = 0; move < 8; move++){
        int newrow = i + moves[move][0];
        int newcol = j + moves[move][1];
        if(newrow >= 0 && newrow < 8 && newcol >= 0 && newcol < 8){
            chessBoard[newrow][newcol] = 1;
        }
    }
}
```

mendefinisikan fungsi bernama `koboImaginaryChess`, yang bertujuan untuk mensimulasikan gerakan pion dalam permainan catur.

```
int main () {

    int i, j ;
    scanf("%d %d", &i, &j);

    int chessBoard[8][8] = {0};

    koboImaginaryChess(i,j,chessBoard);

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

    return 0;
}
```

mengisi papan catur dengan angka-angka yang merepresentasikan jarak terdekat dari titik (i, j) ke setiap titik pada papan catur.

Kembali.



```
1  #include <stdio.h>
2
3  void koboImaginaryChess(int i, int j, int chessBoard[][8]){
4      int moves[8][2] = {{-2, -1}, {-1, -2}, {1, -2}, {2, -1}, {2, 1}, {1, 2}, {-1, 2}, {-2, 1}};
5
6      for(int move = 0; move < 8; move++){
7          int newrow = i + moves[move][0];
8          int newcol = j + moves[move][1];
9          if(newrow >= 0 && newrow < 8 && newcol >= 0 && newcol < 8){
10             chessBoard[newrow][newcol] = 1;
11         }
12     }
13 }
14 int main () {
15
16     int i, j ;
17     scanf("%d %d", &i, &j);
18
19     int chessBoard[8][8] = {0};
20
21     koboImaginaryChess(i,j,chessBoard);
22
23     for (int row = 0; row < 8; row++){
24         for(int col = 0; col < 8; col++){
25             printf("%d", chessBoard[row][col]);
26         }
27         printf("\n");
28     }
29
30     return 0;
31 }
```