NAMA : Yonathan Hari Dharmawan

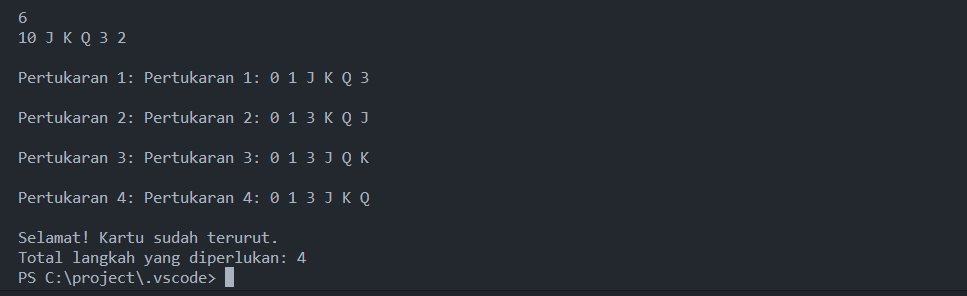
NIM : 1203230050

Kelas : IF 03-03

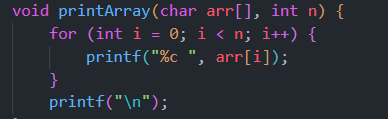
|  |  |  |
| --- | --- | --- |
| Komponen Penilian | YA | TIDAK |
| Soal 1 sesuai dengan ouput yang di inginkan |  |  |
| Soal 2 sesuai dengan ouput yang di inginkan |  |  |

1. Source Code 1
2. #include <stdio.h>
3. #include <stdlib.h>
4. // Fungsi untuk mencetak array
5. void printArray(char arr[], int n) {
6. for (int i = 0; i < n; i++) {
7. printf("%c ", arr[i]);
8. }
9. printf("\n");
10. }
11. // Fungsi untuk menghitung jumlah langkah minimum
12. int minLangkah(char arr[], int n) {
13. int langkah = 0;
14. for (int i = 0; i < n - 1; i++) {
15. for (int j = i + 1; j < n; j++) {
16. if (arr[i] > arr[j]) {
17. char temp = arr[i];
18. arr[i] = arr[j];
19. arr[j] = temp;
20. langkah++;
21. printf("\nPertukaran %d: ", langkah);
22. if (langkah == 1)
23. printf("Pertukaran 1: ");
24. else if (langkah == 2)
25. printf("Pertukaran 2: ");
26. else if (langkah == 3)
27. printf("Pertukaran 3: ");
28. else if (langkah == 4)
29. printf("Pertukaran 4: ");
30. else
31. printf("pertukaran 5:");
32. printArray(arr, n); // Tampilkan urutan kartu setiap terjadi pertukaran
33. }
34. }
35. }
36. return langkah;
37. }
38. int main() {
39. int n;
40. scanf("%d", &n);
41. char arr[n];
42. for (int i = 0; i < n; i++) {
43. scanf(" %c", &arr[i]);
44. }
45. int langkah = minLangkah(arr, n);
46. printf("\nSelamat! Kartu sudah terurut.\n");
47. printf("Total langkah yang diperlukan: %d\n", langkah);
48. return 0;
49. }

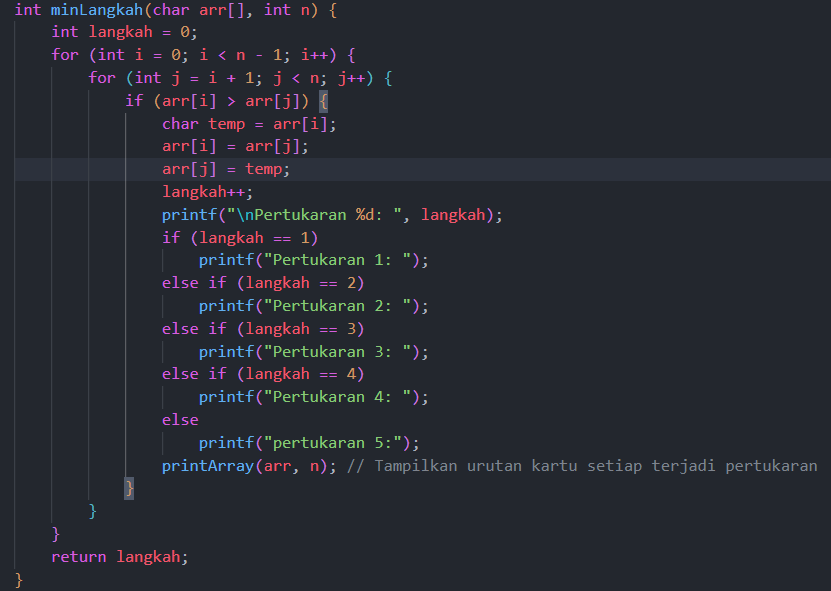
2. Outputnya



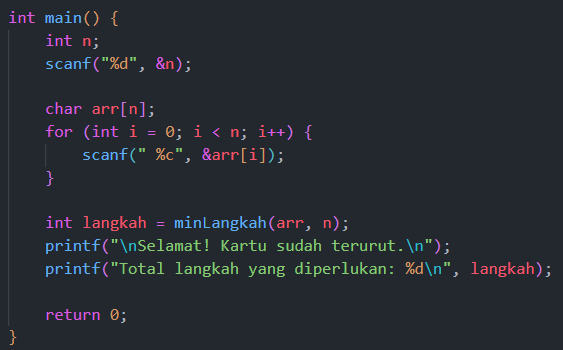
3. Penjelasan



* Melakukan iterasi melalui setiap elemen dalam array dan mencetaknya ke layar dengan format "elemen[index] " (dipisahkan oleh spasi), kemudian mencetak baris baru setelah selesai.



* Mendeklarasikan variabel langkah yang akan digunakan untuk menghitung jumlah pertukaran elemen yang dilakukan dalam algoritma Bubble Sort.
* Menggunakan dua loop bersarang untuk membandingkan setiap pasangan elemen dalam array.
* Jika pasangan elemen tidak dalam urutan yang benar, mereka ditukar, dan langkah (jumlah pertukaran) ditingkatkan. Setiap kali terjadi pertukaran, langkah dan urutan array saat itu dicetak ke layar.



* Mengambil input dari pengguna berupa jumlah elemen dalam array.
* Membuat array arr dengan ukuran yang ditentukan pengguna dan mengisi elemen-elemennya dengan input pengguna.
* Memanggil fungsi minLangkah untuk mengurutkan array dan menghitung jumlah langkah yang diperlukan.
* Mencetak pesan bahwa array telah diurutkan dan mencetak jumlah total langkah yang diperlukan.
* Memgakhiri program dengan sukses

2. Source code 2

#include <stdio.h>

int main() {

    int i, j;

    scanf("%d %d", &i, &j);

    // Inisialisasi dan cetak papan catur

    for (int row = 0; row < 8; row++) {

        for (int col = 0; col < 8; col++) {

            if ((row - i) \* (row - i) + (col - j) \* (col - j) == 5)

                printf("1 ");

            else

                printf("0 ");

        }

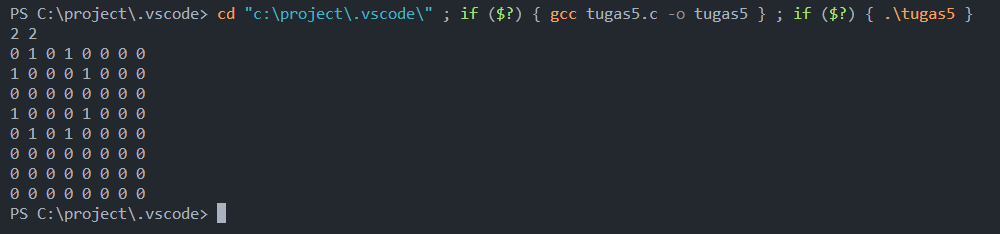
        printf("\n");

    }

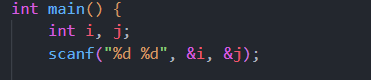
    return 0;

}

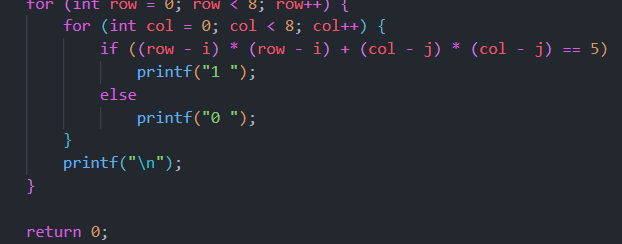
2. Outputnya



3. penjelasannya



* Berfungsi untuk mengambil inputan posisi kuda



* Menggunakan nested loop untuk mengiterasi setiap sel di papan catur 8x8.
* Pada setiap iterasi, menghitung jarak kuadrat antara titik (i, j) dan sel saat ini menggunakan rumus jarak Euclidean, yaitu (row - i) \* (row - i) + (col - j) \* (col - j).
* Jika jarak kuadrat tersebut sama dengan 5, maka cetak "1 " (dengan spasi) yang menandakan bahwa sel tersebut berada dalam jarak 5 dari titik (i, j). Jika tidak, cetak "0 ".
* Setelah mencetak satu baris penuh, cetak baris baru untuk beralih ke baris berikutnya.
* Dan program telah berakhir dengan sukses