

Nama : Philif Diamond Song
Kelas : TI-25-KA
NPM : 252310024

Jelaskan perbedaan If-Else dan Switch Case!

➤ Pernyataan IF- ELSE

Pernyataan If mempunyai pengertian, “Jika kondisi bernilai benar, maka perintah-1 akan dikerjakan dan jika tidak memenuhi syarat maka akan mengerjakan perintah-2”. Perintah-1 dan perintah-2 dapat berupa sebuah pernyataan tunggal, pernyataan majemuk atau pernyataan kosong. IF-ELSE Majemuk digunakan untuk membuat percabangan dengan 3 atau lebih kemungkinan yang akan terjadi. IF ELSE di gunakan untuk mengeksekusi program berdasarkan dua kondisi yang ada. Jika kondisi bernilai benar, maka perintah-1 akan di kerjakan dan jika tidak memenuhi syarat maka akan mengerjakan perintah-2.

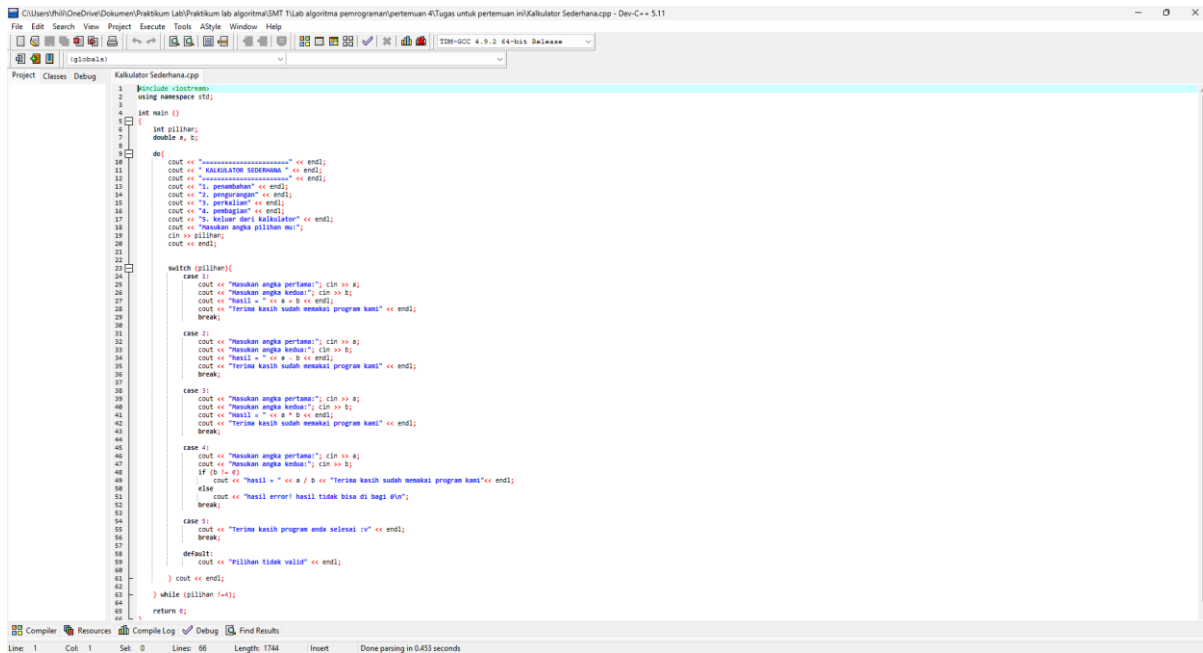
- Nested if merupakan pernyataan if berada di dalam pernyataan if yang lainnya, dan digunakan apabila ingin membuat *function* IF di dalam IF.
- If-else bertingkat sebenarnya serupa dengan nested if, keuntungan penggunaan if-else bertingkat di banding dengan nested if adalah penggunaan bentuk penulisan yang lebih sederhana.

➤ Pernyataan Switch-Case

Bentuk dari Switch-Case merupakan pernyataan yang di rancang khusus untuk menangani pengambilan keputusan yang melibatkan sejumlah atau banyak alternatif. Pernyataan Switch-case ini memiliki kegunaan sama seperti if – else bertingkat, tetapi penggunaanya untuk memeriksa data yang bersifat karakter atau integer.

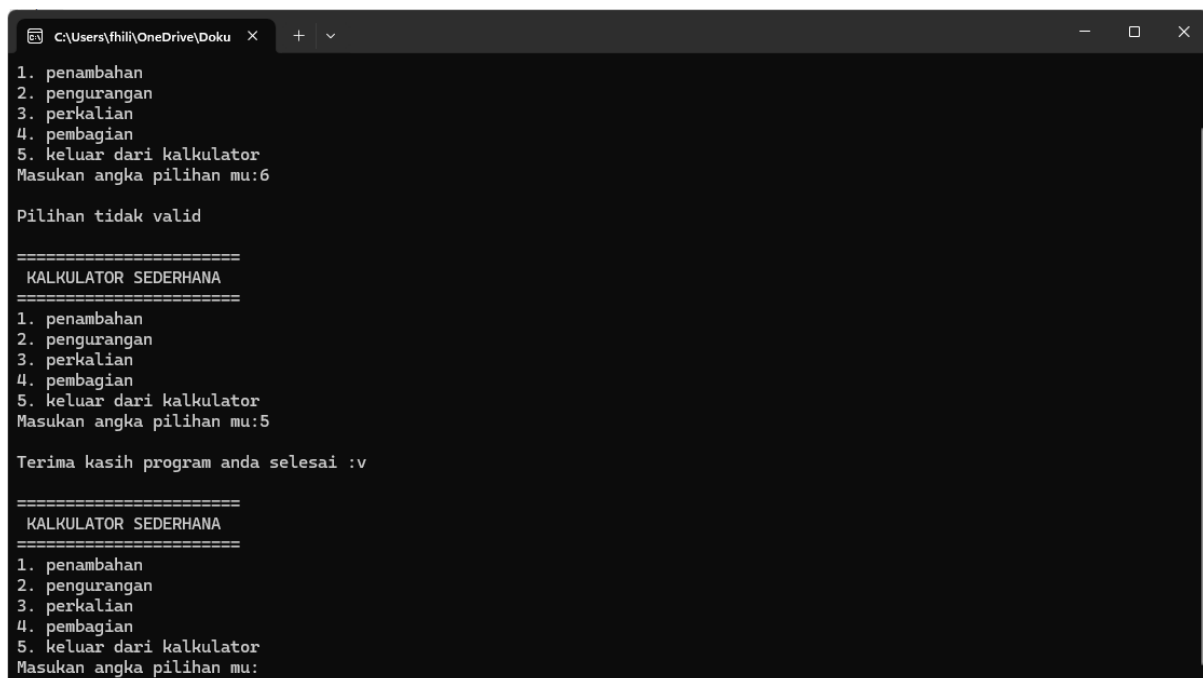
- Switch Case digunakan untuk percabangan kode program dimana terdapat perbandingan isi sebuah variable dengan beberapa nilai. SWITCH CASE digunakan jika hanya ada satu variable yang di cek apakah sudah memenuhi value tertentu atau belum.

Kakulator sederhana:



```
1 #include <iostream>
2 using namespace std;
3
4 int main ()
5 {
6     int pilihan;
7     double a, b;
8
9     do{
10         cout << "=====Kalkulator Sederhana=====" << endl;
11         cout << "KALKULATOR SEDERHANA" << endl;
12         cout << "1. penambahan" << endl;
13         cout << "2. pengurangan" << endl;
14         cout << "3. perkalian" << endl;
15         cout << "4. pembagian" << endl;
16         cout << "5. keluar dari kalkulator" << endl;
17         cout << "Masukan angka pilihan mu:" << endl;
18         cin >> pilihan;
19         cout << endl;
20     } while (pilihan != 5);
21
22     switch (pilihan){
23         case 1:
24             cout << "Masukan angka pertama:" << endl;
25             cin >> a;
26             cout << "Masukan angka kedua:" << endl;
27             cin >> b;
28             cout << "Hasil = " << a + b << endl;
29             cout << "Terima kasih sudah memakai program kami" << endl;
30             break;
31         case 2:
32             cout << "Masukan angka pertama:" << endl;
33             cin >> a;
34             cout << "Masukan angka kedua:" << endl;
35             cin >> b;
36             cout << "Hasil = " << a - b << endl;
37             cout << "Terima kasih sudah memakai program kami" << endl;
38             break;
39         case 3:
40             cout << "Masukan angka pertama:" << endl;
41             cin >> a;
42             cout << "Masukan angka kedua:" << endl;
43             cin >> b;
44             cout << "Hasil = " << a * b << endl;
45             cout << "Terima kasih sudah memakai program kami" << endl;
46             break;
47         case 4:
48             cout << "Masukan angka pertama:" << endl;
49             cin >> a;
50             cout << "Masukan angka kedua:" << endl;
51             cin >> b;
52             if (b != 0)
53                 cout << "Hasil = " << a / b << endl;
54             else
55                 cout << "Hasil error! Hasil tidak bisa di bagi 0!" << endl;
56             cout << "Terima kasih sudah memakai program kami" << endl;
57             break;
58         case 5:
59             cout << "Terima kasih program anda selesai :v" << endl;
60             break;
61         default:
62             cout << "Pilihan tidak valid" << endl;
63     }
64     cout << endl;
65 }
66
67 return 0;
68 }
```

Program kalkulator ketika di jalankan:



```
C:\Users\fhili\OneDrive\Doku x + v
1. penambahan
2. pengurangan
3. perkalian
4. pembagian
5. keluar dari kalkulator
Masukan angka pilihan mu:6

Pilihan tidak valid

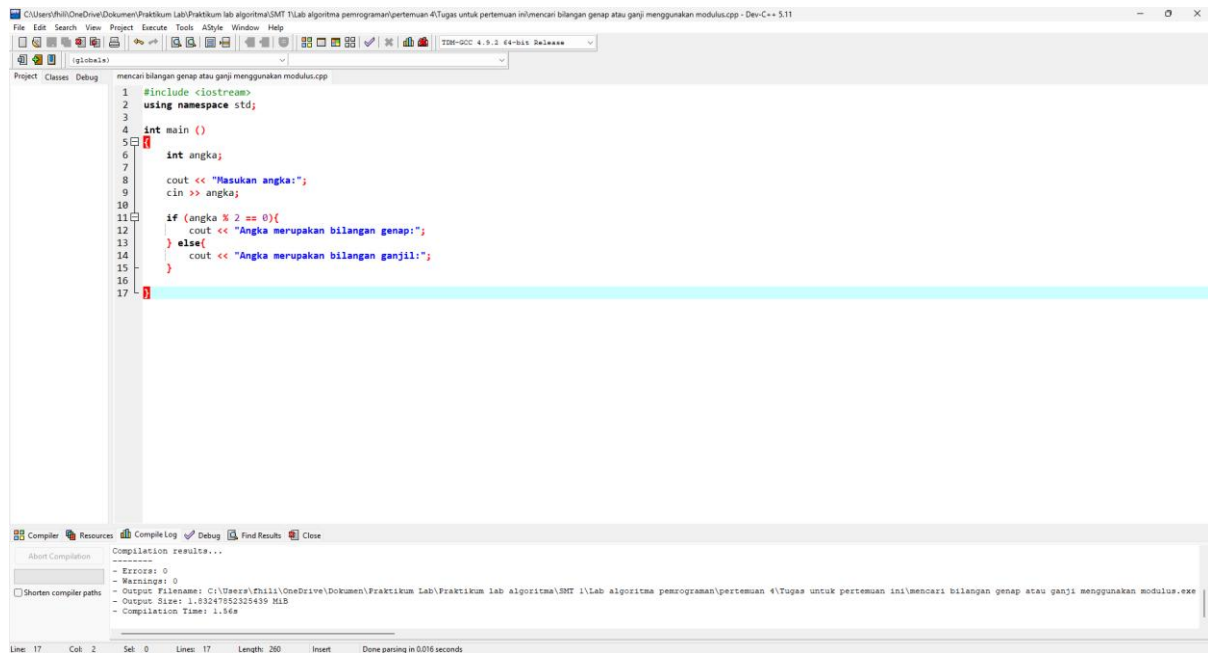
=====
KALKULATOR SEDERHANA
=====
1. penambahan
2. pengurangan
3. perkalian
4. pembagian
5. keluar dari kalkulator
Masukan angka pilihan mu:5

Terima kasih program anda selesai :v

=====
KALKULATOR SEDERHANA
=====
1. penambahan
2. pengurangan
3. perkalian
4. pembagian
5. keluar dari kalkulator
Masukan angka pilihan mu:
```

Membuat program mencari bilangan genap dan ganjil:

Ini programnya:



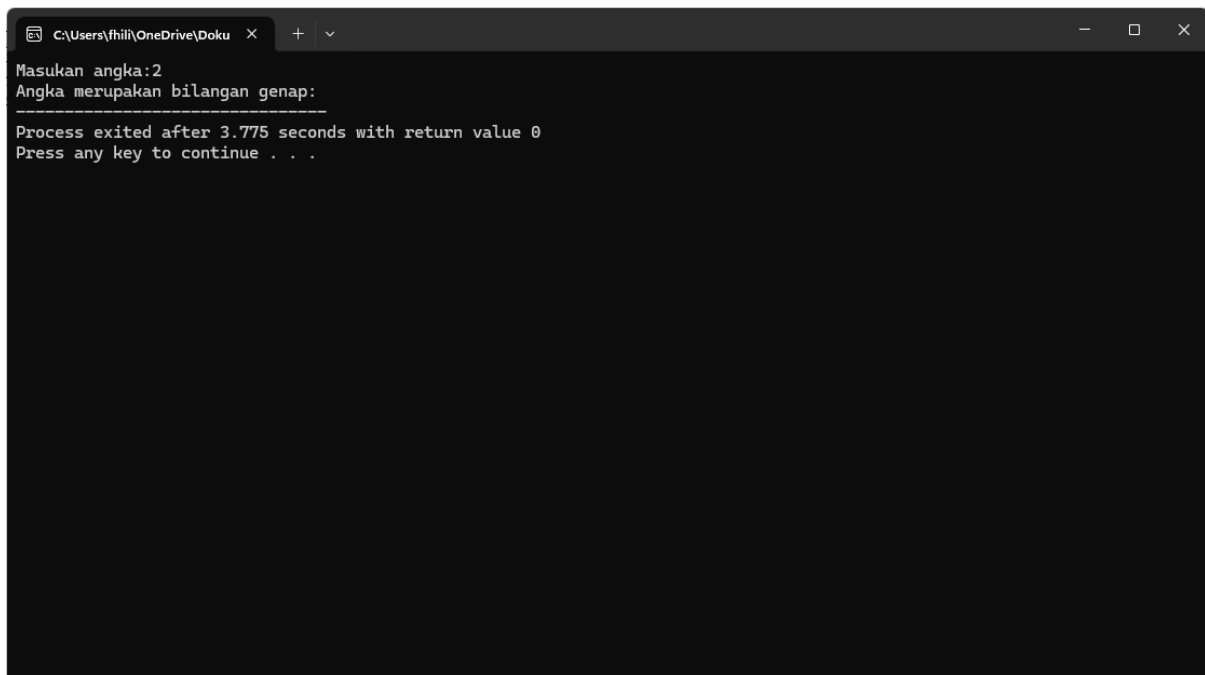
The screenshot shows a C++ IDE with a project named "mencari bilangan genap atau ganjil menggunakan modulus.cpp". The code is as follows:

```
1 #include <iostream>
2 using namespace std;
3
4 int main ()
5 {
6     int angka;
7
8     cout << "Masukan angka:";
9     cin >> angka;
10
11     if (angka % 2 == 0){
12         cout << "Angka merupakan bilangan genap:";
13     } else{
14         cout << "Angka merupakan bilangan ganjil:";
15     }
16 }
17
```

The bottom panel shows the compilation results:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\thill\OneDrive\Documents\Praktikum Lab\Praktikum lab algoritma\SMK 1\Lab algoritma pemrograman\pertemuan 4\Tugas untuk pertemuan ini\mencari bilangan genap atau ganjil menggunakan modulus.exe
- Output Size: 1.3324755232439 KB
- Compilation Time: 1.56s
```

Ini programnya ketika di jalankan bagian mencari genap:



The screenshot shows a terminal window with the following output:

```
C:\Users\thill\OneDrive\Doku x + v
Masukan angka:2
Angka merupakan bilangan genap:
-----
Process exited after 3.775 seconds with return value 0
Press any key to continue . . .
```

Ini programnya ketika di jalankan bagian mencari ganjil:

```
C:\Users\fhili\OneDrive\Doku  X  +  v  -  □  X
Masukan angka:5
Angka merupakan bilangan ganjil:
-----
Process exited after 3.13 seconds with return value 0
Press any key to continue . . .
```

Ini adalah Flowchartnya untuk yang bagian ganjil dan genap:

