

Penjelasan Coding

1. Header dan Namespace :

```
cpp
#include <iostream>
using namespace std;
```

```
3
4  #include <iostream>
5  using namespace std;
```

Kode ini mengimpor pustaka `iostream`, yang digunakan untuk input dan output. `using namespace std;` memungkinkan kita untuk menggunakan elemen dalam namespace `std` tanpa perlu menuliskan `std::` setiap kali.

2. Fungsi `main()` :

```
cpp
int main()
{
```

```
6  int main()
7  {
```

Fungsi ini adalah titik awal eksekusi program.

3. Deklarasi Variabel :

```
cpp
int data, ganjil = 0, genap = 0;
```

```
int data, ganjil = 0, genap = 0;
```

- `data`: untuk menyimpan jumlah data yang akan dimasukkan oleh pengguna.
- `ganjil`: untuk menyimpan jumlah total bilangan ganjil (diinisialisasi dengan 0).
- `genap`: untuk menyimpan jumlah total bilangan genap (diinisialisasi dengan 0).

4. Input dari Pengguna :

```
cpp
cout << "Masukkan jumlah data: ";
cin >> data;
```

```
cout << "Masukkan jumlah data: "; cin >> data;
```

Baris ini meminta pengguna untuk memasukkan jumlah data yang ingin dihitung, dan menyimpannya ke dalam variabel `data`.

5. Perulangan :

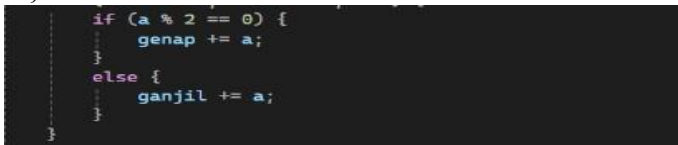
```
cpp
for (int a = 0; a <= data; a++)
```

```
for (int a = 0; a <= data; a++) {
```

Ini adalah perulangan for yang mulai dari 0 hingga nilai data. Namun, perlu dicatat bahwa perulangan ini akan menghitung hingga data, termasuk angka itu sendiri.

6. Pengkondisian untuk Menentukan Genap dan Ganjil :

```
cpp
if (a % 2 == 0) {
    genap += a;
}
else {
    ganjil += a;
}
```



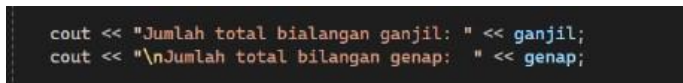
```
if (a % 2 == 0) {
    genap += a;
}
else {
    ganjil += a;
}
```

- Jika a adalah bilangan genap (hasil modulo 2 sama dengan 0), nilai a akan ditambahkan ke genap.

- Jika a adalah bilangan ganjil, nilai a akan ditambahkan ke ganjil.

7. Output Hasil :

```
cpp
cout << "Jumlah total bilangan ganjil: " << ganjil;
cout << "\nJumlah total bilangan genap: " << genap;
```



```
cout << "Jumlah total bilangan ganjil: " << ganjil;
cout << "\nJumlah total bilangan genap: " << genap;
```

Di sini, program mencetak total jumlah bilangan ganjil dan genap yang telah dihitung.

8. Mengakhiri Fungsi main() :

```
cpp
return 0;
```



```
return 0;
```

Mengindikasikan bahwa program selesai dieksekusi dengan sukses.

Kesimpulan

Program ini menghitung dan mencetak total jumlah bilangan ganjil dan genap dari 0 hingga nilai yang dimasukkan pengguna. Pastikan saat memasukkan data, pengguna memahami bahwa perulangan akan mencakup angka dari 0 hingga nilai yang dimasukkan.

