



Modul Peer Group

Pertemuan 1:

Input dan Output

Disusun Oleh:

Tim PJ Peer Group Badan Khusus

Pengantar

Tipikal program C++ dapat terdiri dari daftar pernyataan yang terdiri dari variabel, konstanta, operator, fungsi, dll. Komputer dapat mengenalinya dengan nama yang mereka miliki. Sehingga, tidak ada nama variabel atau konstanta yang bernama sama, kecuali fungsi, namanya mungkin bisa sama, namun ada parameter yang membedakan.

C++ termasuk ke dalam golongan *case sensitive language*, yang artinya 'A' dan 'a' merupakan dua objek yang berbeda. Sama halnya dengan Area, area, dan AREA adalah 3 objek yang berbeda.

C++ juga merupakan *highly typed language* yang berarti data dikategorikan ke beberapa tipe yang berbeda. Contohnya, bilangan bulat membentuk kategori yang disebut integer. Jadi, ketika variabel yang akan dibuat hanya dapat menampung bilangan bulat, perlu mendeklarasikan int sebelum nama variabel. Int adalah tipe variable tersebut.

Komponen dari Program Sederhana C++

```
1. #include <iostream>
2.
3. int main() {
4.     std::cout<< "Hello, World!\n";
5.     return 0;
6. }
```

File Header

`#include <iostream>` adalah sebuah statement yang mengarahkan kompiler untuk menyertakan file header `<iostream>` (input/output stream) yang merupakan library untuk melakukan proses input dan output. Object yang sering digunakan dalam library tersebut adalah `cin` dan `cout`. C++ sendiri memiliki library bawaan lain yang termasuk ke dalam C++ Standard Library. Jika ingin memasukkan fungsi matematika seperti akar kuadrat, `sin`, `cos`, `tan`, `log`, dll file header `<cmath>` bisa disertakan dalam program.

Int main()

`Int main()` mengindikasikan awalan program. Dalam C++, tanda kurung () digunakan untuk mengidentifikasi suatu function/fungsi. Dalam program

tersebut, `main()` adalah sebuah fungsi. Setiap program C++ harus memiliki satu fungsi `main()`.

Pernyataan Keluaran/Output

Pernyataan: `std::cout << "Hello, World!";`. Fungsi dan class dalam C++ Standard Library didefinisikan didalam namespace `std`. Operator (`::`) adalah operator resolusi lingkup/cakupan. `Std::cout` berarti kita memanggil objek `cout` yang berada dalam namespace `std`. Objek `cout` menggunakan operator stream insertion (`<<`) yang akan memasukkan argumen yang berada di sisi kanan ke dalam stream output/keluaran.

Escape Sequence

Beberapa karakter yang didahului oleh backslash (`\`) seperti `"\n"` memiliki arti spesial terhadap kompiler. Backslash (`\`) merupakan karakter escape. Kombinasi `"/n"` merepresentasikan escape sequence yang mengarahkan kursor ke baris selanjutnya. Berikut karakter escape yang sering digunakan:

Karakter Escape	Deskripsi Aksi
<code>\n</code>	Pindah ke baris baru
<code>\t</code>	Tab horizontal
<code>\\</code>	Menampilkan karakter backslash
<code>\'</code>	Menampilkan karakter kutipan tunggal (<code>'</code>)
<code>\"</code>	Menampilkan karakter kutipan ganda (<code>"</code>)

Cin

Dalam program yang interaktif, pengguna perlu memasukkan beberapa data. Dalam program berikut kita menggunakan fungsi `cin` (input stream standar) untuk melakukan proses input. `Cin` digunakan bersamaan dengan (`>>`) yang disebut sebagai operator ekstraksi, yang diikuti dengan nama variabel dimana data akan disimpan dalam memori. Tipe pada saat meinputkan data harus sama dengan tipe variabel yang dideklarasikan.

```
1. #include <iostream>
2.
3. int main() {
4.     int sisi;
5.     std::cout << "Masukkan panjang sisi persegi: ";
6.     std::cin >> sisi;
7.     std::cout << "Luas persegi: " << sisi * sisi;
8.     return 0;
9. }
```

Namespace

Namespace adalah fitur ungguland alam bahasa C++. Fitur ini memungkinkan programmer untuk mengelompokkan satu set objek atau fungsi dengan satu nama. Semua kelas, fungsi, dan objek dalam C++ Standard Language didefinisikan di bawah namespace std. Kita dapat menyederhanakan penulisan std::cout dan std::cin dengan memasukkan statement berikut kedalam program

```
using namespace std;
```

Program untuk menghitung luas persegi diatas dapat simplifikasi dengan menggunakan namespace.

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     int sisi;
6.     cout << "Masukkan panjang sisi persegi: ";
7.     cin >> sisi;
8.     cout << "Luas persegi: " << sisi * sisi;
9.     return 0;
10. }
```

Fungsi Cin Getline

Ketika menggunakan cin dalam proses input suatu bilangan, mungkin tidak ditemukan sesuatu masalah. Tetapi, jika tipe input merupakan sebuah kalimat yang terdiri dari lebih satu kata, maka menggunakan cin sedikit tidak berguna. Cin hanya mengambil kata pertama sebelum ditemukan sebuah spasi.

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     char kalimat[30];
6.     cout << "Masukkan kalimat: ";
7.     cin >> kalimat;
8.     cout << "Kalimat: " << kalimat;
9.     return 0;
10. }
```

Output:

```
Masukkan kalimat: Ini Adalah Kalimat
Kalimat: Ini
```

Oleh karenanya kita membutuhkan `cin.getline` untuk menyelesaikan masalah ini. Ganti `cin` dalam program diatas dengan `cin.getline`:

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     char kalimat[30];
6.     cout << "Masukkan kalimat: ";
7.     cin.getline(kalimat, 30);
8.     cout << "Kalimat: " << kalimat;
9.     return 0;
10. }
```

Output:

```
Masukkan kalimat: Ini Adalah Kalimat
Kalimat: Ini Adalah Kalimat
```

Let's Coding

1. Buat program yang mengoutputkan data persis seperti dibawah (tanpa input), tanpa menggunakan namespace, dan menggunakan bantuan escape character:

```
Kegiatan      : Peer Group
Tempat        : Online
```

2. Modifikasi program diatas dengan mengganti escape character `""\n"` dengan `endl`
3. Buat program yang mengoutputkan huruf 'D' dengan gabungan karakter (`"`):

```
" " " " "
"      "
"      "
"      "
"      "
" " " " "
```

4. Buat program yang mengoutputkan segitiga sama kaki seperti dibawah

```
  *
 ***
*****
*****
```

5. Buat program yang mengimplementasikan input stream dengan input bilangan bulat dan mengoutputkan hasil pangkat 2 bilangan tersebut.

Hint: Program ini dibuat menggunakan header file cmath

```
Masukkan bilangan      : 5
Hasil Pangkat 2        : 25
```

6. Buat program yang meminta input nama, kelas, kelompok, dan alamat kepada user, lalu kemudian mengoutputkannya kembali

```
Nama      : Jamal Akhyer
Kelas    : E
Kelompok : 17
Alamat    : Kalifornia

Output:
Nama      : Jamal Akhyer
Kelas    : E
Kelompok : 17
Alamat    : Kalifornia
```

Hasil Coding

Problem 1

```
1. #include <iostream>
2.
3. int main() {
4.     std::cout << "Kegiatan\t: Peer Group\n";
5.     std::cout << "Tempat\t\t: Online";
6. }
```

Problem 2

```
1. #include <iostream>
2.
3. int main() {
4.     std::cout << "Kegiatan\t: Peer Group" << std::endl;
5.     std::cout << "Tempat\t\t: Online";
6. }
```

Problem 3

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     cout << "\\\"\\\"\\\"\\\"\\\"\\n";
6.     cout << "\\\"    \\\"\\n";
7.     cout << "\\\"    \\\"\\n";
8.     cout << "\\\"    \\\"\\n";
9.     cout << "\\\"\\\"\\\"\\\"\\\"\\n";
10. }
```

Problem 4

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     cout << "    *\n";
6.     cout << "   **\n";
7.     cout << "  ***\n";
8.     cout << " *****\n";
9. }
10.
```

Problem 5

```
1. #include <iostream>
2. #include <cmath>
3. using namespace std;
4.
5. int main() {
6.     int a;
7.     cout << "Masukkan bilangan\t: ";
8.     cin >> a;
```

```
9.     cout << "Hasil Pangkat 2\t\t: " << pow(a, 2);  
10. }
```

Problem 6

```
1. #include <iostream>  
2. using namespace std;  
3.  
4. int main() {  
5.     char nama[30], kelas[3], kelompok[3], alamat[50];  
6.     cout << "Nama\t\t: ";  
7.     cin.getline(nama, 30);  
8.     cout << "Kelas\t\t: ";  
9.     cin >> kelas;  
10.    cout << "Kelompok\t: ";  
11.    cin >> kelompok;  
12.    cout << "Alamat\t\t: ";  
13.    cin.ignore(1, '\n');  
14.    cin.getline(alamat, 50);  
15.  
16.    cout << "\nOutput:" << endl;  
17.    cout << "Nama\t\t: " << nama << endl;  
18.    cout << "Kelas\t\t: " << kelas << endl;  
19.    cout << "Kelompok\t: " << kelompok << endl;  
20.    cout << "Alamat\t\t: " << alamat << endl;  
21. }
```