

PRAKTIKUM 5

FUNGSI MATEMATIKA DAN FUNGSI STRING

1. Fungsi Matematika

Terdapat beberapa fungsi yang dapat digunakan untuk melakukan perhitungan matematika dalam C++.

(1). Fungsi max dan min

Fungsi max : Menentukan bilangan terbesar diantara 2 bilangan. Sintak: max(x,y)

Fungsi min : Menentukan bilangan terkecil diantara 2 bilangan. Sintak: min(x,y)

Praktikum 4-1:

```
1 #include <iostream>
2 using namespace std;
3 float x,y;
4 int main(){
5     cout <<"Ketik Nilai X: "; cin>>x;
6     cout <<"Ketik Nilai Y: "; cin>>y;
7     cout << "Bilangan Terbesar adalah : "<<max(x, y)<<endl;
8     cout << "Bilangan Terkecil adalah : "<<min(x, y);
9     return 0;
10 }
```

(2). Fungsi Pembuatan dan Sisa Bagi

| Fungsi | Keterangan | Sintaks |
|--------|---------------------------|---------------|
| ceil | Pembulatan nilai ke atas | ceil(double) |
| floor | Pembulatan nilai ke bawah | floor(double) |
| fmod | Menghitung sisa bagi | fmod(x,y) |

Fungsi ceil, floor, fmod tersimpan dalam file header **math.h**, oleh karena itu jika akan digunakan dalam rancangan program, maka file harus math harus diikuti ke dalam program tersebut dengan cara menuliskan *preprocessor directive* berikut: **#include<math.h>**

Praktikum 4-2:

```
1 #include <iostream>
2 #include<math.h>
3 using namespace std;
4 float x,y;
5 int main(){
6     cout <<"Ketik Nilai X: "; cin>>x;
7     cout <<"Ketik Nilai Y: "; cin>>y;
8     cout << "Hasil Dari "<<x<<"/"<<y<< "="<<x/y<<endl;
9     cout << "Sisa Bagi Dari "<<x<<"/"<<y<< "="<<fmod(x,y)<<endl;
10    cout << "Pembulatan Ke Atas : "<<ceil(x/y)<<endl;
11    cout << "Pembulatan Ke Bawah : "<<floor(x/y);
12    return 0;
13 }
```

(3). Fungsi Pangkat dan Akar

| Fungsi | Keterangan | Sintaks |
|--------|--|----------|
| pow | Menghitung pangkat (x^y) | pow(x,y) |
| sqrt | Menghitung akar kuadrat (\sqrt{x}) | sqrt(x) |
| cbrt | Menghitung akar pangkat tiga ($\sqrt[3]{x}$) | cbrt(x) |

Preprocessor directive: **#include<math.h>**

Praktikum 4-3:

```
1  #include <iostream>
2  #include<math.h>
3  using namespace std;
4  float x,y;
5  int main(){
6      cout <<"Ketik Nilai X: "; cin>>x;
7      cout <<"Ketik Nilai Y: "; cin>>y;
8      cout << "Hasil Dari "<<x<<" pangkat "<<y<<"="<<pow(x,y)<<endl;
9      cout << "Akar Kuadrat Dari "<<x<<"="<<sqrt(x)<<endl;
10     cout << "Akar Kuadrat Dari "<<y<<"="<<sqrt(y)<<endl;
11     cout << "Akar Pangkat 3 Dari "<<x<<" = "<<cbt(x)<<endl;
12     cout << "Akar Pangkat 3 Dari "<<y<<" = "<<cbt(y)<<endl;
13     return 0;
14 }
```

(4). Fungsi Trigonometri

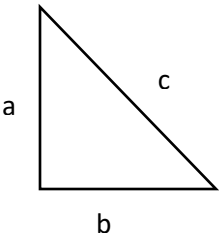
| Fungsi | Keterangan | Sintaks |
|--------|--------------------------|---------|
| cos | Untuk menghitung kosinus | cos(x) |
| sin | Untuk menghitung sinus | sin(x) |
| tan | Untuk menghitung tangen | tan(x) |

Preprocessor directive: **#include<math.h>**

Praktikum 4-4:

```
1  #include <iostream>
2  #include<math.h>
3  using namespace std;
4  float x,y;
5  int main(){
6      cout <<"Ketik Nilai Sudut X: "; cin>>x;
7      cout << "Cos ("<<y<<" = "<<cos(x)<<endl;
8      cout << "Sin ("<<y<<" = "<<sin(x)<<endl;
9      cout << "Tan ("<<y<<" = "<<tan(x)<<endl;
10     return 0;
11 }
```

Contoh Kasus: Menghitung panjang sisi miring suatu segita siku-siku jika nilai alas dan tinggi diketahui.



Analisis : Menghitung sisi miring segitia siku-siku dapat dihitung menggunakan rumus phytagoras $c^2= a^2 + b^2 \rightarrow c= \sqrt{a^2 + b^2}$

Praktikum 4-5:

```
1  #include <iostream>
2  #include<math.h>
3  using namespace std;
4  float a,b, c;
5  int main(){
6      cout <<"Ketik Nilai Tinggi (a) : "; cin>>a;
7      cout <<"Ketik Nilai Alas (b) : "; cin>>b;
8      c=sqrt(pow(a,2)+pow(b,2));
9      cout << "Panjang Sisi Miring (c) = "<<c;
10     return 0;
11 }
```

2. Fungsi String

Fungsi string digunakan untuk memanipulasi data berupa string. Fungsi string di C++ tersimpan dalam file **string.h**. Oleh karena itu setiap digunakan, maka harus ada preprocessor directive **#include<string.h>**.

(1). Menggabungkan String

Dua buah string dapat digabungkan dengan menggunakan operator +.

Praktikum 4-6:

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4      string namaDepan;
5      string namaBelakang;
6      string namaLengkap;
7
8  int main(){
9
10     cout<<"Ketik Nama Depan Anda: ";
11     cin>>namaDepan;
12     cout<<"Ketik Nama Belakang Anda: ";
13     cin>>namaBelakang;
14     namaLengkap=namaDepan+" "+namaBelakang;
15     cout <<"Nama Lengkap Anda: "<< namaLengkap;
16     return 0;
17 }
```

(2). Fungsi append()

Fungsi append() digunakan untuk menambahkan suatu string dibelakang string lainnya. Sintaks: variable.append(string yang mau ditambahkan);

Praktikum 4-7:

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4      string namaDepan;
5      string namaBelakang;
6      string namaLengkap;
7
8  int main(){
9
10     cout<<"Ketik Nama Depan Anda: ";
11     cin>>namaDepan;
12     cout<<"Ketik Nama Belakang Anda: ";
13     cin>>namaBelakang;
14     namaLengkap=namaDepan.append(" ");
15     namaLengkap=namaDepan.append(namaBelakang);
16     cout <<"Nama Lengkap Anda: "<< namaLengkap;
17     return 0;
18 }
```

(3). Fungsi length(), size()

Digunakan untuk menghitung panjang suatu string. Yang dimaksud dengan Panjang string adalah banyaknya karakter dalam suatu string. Menghitung panjang suatu string dapat digunakan salah satunya (fungsi length atau size).

Sintaks: variable_string.length() atau variable_string.size()

Praktikum 4-8:

```
1  //Menghitung Panjang String dengan fungsi Length()
2  #include <iostream>
3  #include <string>
4  using namespace std;
5      string kalimat;
6      int panjang;
7  int main(){
8      cout<<"Ketik Sebuah Kalimat: ";
9      getline(cin, kalimat);
10     panjang=kalimat.length();
11     cout<<"Panjang Kalimat "<<kalimat<<" = "<<panjang;
12     return 0;
13 }
```

Catatan: perintah `getline(cin,kalimat)` pada baris 9 digunakan untuk menginput string yang lebih dari satu kata dari keyboard. Jika menggunakan instruksi `cin>>variable` untuk menginput string yang lebih dari satu kata, maka kata yang kedua dan seterusnya tidak terbaca, oleh karena itu maka yang digunakan jika string yang dibaca lebih dari 1 kata adalah **`getline(cin, nama_variabel)`**

Praktikum 4-9:

```
1 //Menghitung Panjang String dengan fungsi size()
2 #include <iostream>
3 #include <string>
4 using namespace std;
5 string kalimat;
6 int panjang;
7 int main(){
8     cout<<"Ketik Sebuah Kalimat: ";
9     getline(cin, kalimat);
10    panjang=kalimat.size();
11    cout<<"Panjang Kalimat "<<kalimat<<" = "<<panjang;
12    return 0;
13 }
```