	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Komentar, Identifier, dan Variabel	
	No. LST/TE/EKA5208/02	Revisi : 01	Tgl : 20 Februari 2017

#### A. KOMPETENSI

1. Mampu membuat komentar dalam kode program
2. Memahami konstanta, variabel dalam C++
3. Mampu membuat inialisasi nilai ke dalam variabel

#### B. ALAT DAN BAHAN

1. PC/ Laptop
2. Flowchart application
3. C++ (atau bahasa pemrograman lain yang dikuasai)
4. Labsheet

#### C. KAJIAN TEORI

Setiap program yang menggunakan nilai pasti membutuhkan variabel yang menampung nilai tersebut. Nilai yang digunakan dalam program dapat berupa numerik, karakter, atau teks. Karena nilai tersebut akan disimpan dalam memori komputer oleh karena itu diperlukan tipe data untuk memberi tahu compiler jenis data apa yang akan disimpan dalam memori komputer.

Contoh program sederhana yang tidak memerlukan nilai sehingga tidak perlu memiliki variabel sebagai berikut:

Contoh program 1:

```
#include <iostream>

using namespace std;

int main()
{
    cout<<"Teks ini digunakan sebagai contoh"<<endl;
    cout<<2+9;

    return 0;
}
```


Keterangan:

Program 1 di atas tidak memiliki variabel karena nilai langsung ditampilkan di layar monitor. Perintah yang digunakan untuk menampilkan teks atau bilangan ke layar adalah `cout<<`. Perintah tersebut dikenal karena menggunakan namespace `std`. Namespace tersimpan di dalam file header `<iostream>` sehingga file header perlu didaftarkan dengan menggunakan perintah `#include`. Perintah lain, `endl` digunakan untuk membuat baris baru.

#### Komentar program

Komentar diperlukan untuk menjelaskan algoritma dan keterangan-keterangan yang diperlukan dalam program sehingga kode-kode program yang rumit akan lebih mudah dipahami oleh pembaca.

Komentar merupakan bagian yang tidak ikut dibaca pada saat proses kompilasi atau dengan kata lain komentar tidak mempengaruhi jalannya program.

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Komentar, Identifier, dan Variabel	
	No. LST/TE/EKA5208/02	Revisi : 01	Tgl : 20 Februari 2017

Dalam menuliskan komentar 1 baris diperlukan tanda // sehingga teks komentar tidak terbaca waktu proses kompilasi. Contoh:

```
Int x; // mendeklarasikan variabel berupa bilangan
```

Dalam menulis komentar tidak boleh disisipkan, contoh:

```
Int // mendeklarasikan variabel berupa bilangan x;
```

Sedangkan untuk menulis komentar yang panjang/ lebih dari 1 baris digunakan tanda /\* ... \*/

### Jenis identifier

Identifier merupakan suatu pengenalan atau pengidentifikasi yang dideklarasikan agar compiler dapat mengenalinya. Identifier dapat berupa variabel, konstanta, fungsi, kelas, template, maupun namespace.

Identifier yang berperan sebagai variabel dan konstanta berfungsi untuk menampung nilai yang akan digunakan dalam program. Identifikasi ini mempermudah proses penanganan data atau nilai.

Contoh program 2:

```
#include <iostream>

using namespace std;

int main()
{
    char Teks[20];
    int X;
    cout<<"Masukkan sebuah kata : "; cin>>Teks;
    cout<<"Masukkan sebuah angka : "; cin>>X;
    cout<<Teks<<endl; // bisa ditulis dengan cout<<X<<'\n';
    cout<<X;

    return 0;
}
```

Program 2 di atas memiliki 2 identifier yaitu Teks dan X.

Saat program dijalankan, identifier digunakan untuk menyimpan nilai-nilai yang dimasukkan melalui keyboard. Dalam C++, proses penyimpanan digunakan perintah cin>>.

Hal-hal yang perlu diperhatikan dalam membuat identifier, yaitu:

1. Bahasa C++ bersifat case sensitive
2. Identifier tidak boleh angka atau karakter yang diawali dengan angka.


Contoh:

```
Long 2000; // identifier yang salah karena berupa angka
Long 2X; // identifier yang salah karena karakter diawali dengan angka
Long X2; // identifier benar karena karakternya tidak didahului dengan angka
```

3. Identifier tidak boleh mengandung spasi

Contoh:

```
Int Bilangan Ganjil; //salah karena ada spasi
Int Bilangan_Ganjil; //benar
Int BilanganGanjil; //benar
```

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Komentar, Identifier, dan Variabel	
	No. LST/TE/EKA5208/02	Revisi : 01	Tgl : 20 Februari 2017

```
Int _BilanganGanjil; //benar
```

4. Identifier tidak boleh menggunakan karakter-karakter simbol(#, @, ?,!, \$, dll)

Contoh:

```
Long !satu; //salah
Long dua@; //salah
Long #tiga; // salah
```

5. Identifier tidak boleh menggunakan kata kunci (keyword) yang ada pada C++

Contoh:

```
Long break; //salah karena menggunakan kata kunci break
Long return; //salah karena menggunakan kata kunci return
```

6. Hindari penggunaan nama identifier yang sama

### Konstanta

Adalah jenis identifier yang bersifat konstan atau tetap, srtingnya nilai dari konstanta di dalam program tidak dapat diubah. Dengan mendefinisikan konstanta yang bersifat global, maka konstanta dapat digunakan di setiap bagian program.

Contoh:

Nilai phi, kecepatan cahaya, dll.

Ada 2 cara membuat konstanta dalam C++, yaitu:

1. Menggunakan preprocessor directive #define

Contoh program 3:

```
#include <iostream>

using namespace std;

int main() {
    int A[5];
    for (int C=0; C<5; C++) {
        // mengisi nilai ke dalam A[C]
        A[C] = C * 10;
        // menampilkan nilai A[C]
        cout<<A[C]<<endl;
    }

    return 0;
}
```

Keterangan:

Pada program 3 di atas, terdapat 2 bilangan 5. Apabila bilangan 5 itu ingin diganti dengan bilangan lain maka, apabila di dalam suatu program terdapat lebih dari 10 bilangan 5 maka akan sangat merepotkan. Oleh karena itu diperlukan suatu konstanta untuk mewakili bilangan 5 tersebut. Apabila kita ingin mengganti bilangan 5, maka cukup konstantanya saja yang diganti.

Contoh program 4:

```
#include <iostream>

using namespace std;

#define MAX 5
```



FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA

LABSHEET ALGORITMA DAN STRUKTUR DATA

Semester 1

Komentar, Identifier, dan Variabel

No.  
LST/TE/EKA5208/02

Revisi : 01

Tgl : 20 Februari  
2017

Hal 4 dari 12

```
int main() {
    int A[MAX];

    // MAX = 10; // SALAH,
    // nilai konstanta tidak dapat ubah

    for (int C=0; C<MAX; C++) {
        // mengisi nilai ke dalam A[C]
        A[C] = C * 10;
        // menampilkan nilai A[C]
        cout<<A[C]<<endl;
    }

    return 0;
}
```

**Keterangan:**

Penggunaan #define tidak menggunakan tanda titik koma;

2. Menggunakan kata kunci `const`.

Bentuk umum:

`Const tipe_data nama_konstanta = nilai_tetapan;`

Contoh pendeklarasian konstanta-konstanta sebagai berikut:

```
Const double phi = 3.14;
Const int NILAI_MAX = 100;
Const char MyChar = 'A';
```

**Contoh program 5:**

```
#include <iostream>

using namespace std;

const int MAX = 5;

int main() {
    int A[MAX];
    for (int C=0; C<MAX; C++) {
        // mengisi nilai ke dalam A[C]
        A[C] = C * 10;
        // menampilkan nilai A[C]
        cout<<A[C]<<endl;
    }


    return 0;
}
```

**Keterangan:**

Konstanta `MAX` didefinisikan dengan menggunakan kata kunci `const`.

Penggunaan kata kunci `const` harus diakhiri dengan tanda titik koma (;).

**Variabel:**

	<b>FAKULTAS TEKNIK</b> <b>UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Komentar, Identifier, dan Variabel	
	No. LST/TE/EKA5208/02	Revisi : 01	Tgl : 20 Februari 2017

sebuah identifier yang mempunyai nilai dinamis, yang berarti nilai variabel bisa diubah sesuai kebutuhan dalam program.

Pendeklarasian variabel dalam C++:

tipe\_data nama\_variabel;

contoh:

```
int A;
```

contoh di atas dideklarasikan sebuah variabel bertipe int dengan nama A. dengan cara tersebut, variabel ini dapat digunakan untuk menampung nilai-nilai berupa bilangan bulat (sesuai dengan rentang yang ada pada tipe data `int`).

Apabila ada beberapa variabel dengan tipe data yang sama maka dapat penulisannya sebagai berikut:

Contoh:

```
Int A, B, C;
```

### Inisialisasi Variabel:

Pengertian:

Inisialisasi merupakan proses pengisian nilai awal (nilai default) ke dalam suatu variabel.

Bentuk umum:

```
tipe_data nama_variabel = nilai_awal;
```

atau

```
tipe_data nama_variabel1 = nilai_awal1, nama_variabel2 =  
nilai_awal2, ....;
```

contoh:

`int A=9;` → merupakan inisialisasi variabel A dengan nilai 9, apabila ingin melakukan inisialisasi terhadap lebih dari 1 variabel maka:

```
int A=9, B=10, =8;
```

catt: inisialisasi nilai tidak harus dilakukan untuk semua variabel

contoh program 6:

```
#include <iostream>

using namespace std;

int main() {
    int X;

    // Menampilkan nilai X sebelum
    // dilakukan pengisian nilai (assignment)
    cout<<"Nilai X sebelum assignment: "<<X<<endl;
```



FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA

LABSHEET ALGORITMA DAN STRUKTUR DATA

Semester 1

Komentar, Identifier, dan Variabel

No.  
LST/TE/EKA5208/02

Revisi : 01

Tgl : 20 Februari  
2017

Hal 6 dari 12

```
// mengisi nilai 10 ke dalam variabel X
X = 10;
// Menampilkan nilai X setelah
// dilakukan pengisian nilai (assignment)
cout<<"Nilai X sesudah assignment: "<<X<<endl;

return 0;
}
```

#### Hasil program 6:

```
Nilai X sebelum assignment: 0
Nilai X sesudah assignment: 10
```

#### Keterangan:

Sebelum dilakukan inisialisasi nilai X = 0 tetapi setelah dilakukan inisialisasi nilai X menjadi 10

#### Contoh program 7:

```
#include <iostream>

using namespace std;

int main() {
    // Inisialisasi X dengan nilai 5
    int X = 5;

    // Menampilkan nilai X sebelum
    // dilakukan pengisian nilai (assignment)
    cout<<"Nilai X sebelum assignment: "<<X<<endl;

    // mengisi nilai 10 ke dalam variabel X
    X = 10;
    // Menampilkan nilai X setelah
    // dilakukan pengisian nilai (assignment)
    cout<<"Nilai X sesudah assignment: "<<X<<endl;

    return 0;
}
```

#### Hasil dari program 7:


```
Nilai X sebelum assignment: 5
Nilai X sesudah assignment: 10
```

### VARIABEL GLOBAL – VARIABEL LOKAL

#### Variabel Global:

Variabel ini digunakan apabila pembuat program menginginkan variabel yang dibuat dikenali oleh semua lingkungan dalam program yang dibuat.

Program dalam bahasa C++ selalu terdapat fungsi utama dengan nama **main ()**. Apabila variabel dideklarasikan di luar fungsi **main**, maka secara otomatis *compiler* akan menganggap variabel tersebut sebagai variabel global.

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Komentar, Identifier, dan Variabel	
	No. LST/TE/EKA5208/02	Revisi : 01	Tgl : 20 Februari 2017

Contoh program yang menunjukkan variabel global sebagai berikut.

**Program 8:**

```
#include <iostream>

using namespace std;

int A;    // Variabel A adalah variabel global
         // karena dideklarasikan di luar fungsi

// Membuat fungsi test()
void test() {
    // Mengisikan (assign) nilai ke dalam variabel A
    A = 20;
    cout<<"Nilai A di dalam fungsi test(): "<<A<<endl;
}

// Membuat fungsi main() atau fungsi utama
int main() {
    // Mengisikan (assign) nilai ke dalam variabel A
    A = 10;
    cout<<"Nilai A di dalam fungsi main(): "<<A<<endl;

    // Memanggil fungsi test()
    test();

    return 0;
}
```

**Hasil dari program 8:**

```
Nilai A di dalam fungsi main(): 10
Nilai A di dalam fungsi test(): 20
```

**Keterangan:**

Melalui kode program di atas variabel A dikenal oleh semua fungsi yang ada di dalam program, yaitu test () dan main ().

**Variabel lokal:**

Variabel lokal adalah variabel yang hanya dikenal oleh satu fungsi saja.


**Contoh program 9:**

```
#include <iostream>

using namespace std;

// Membuat fungsi test()
void test() {
    int A;    // A bersifat lokal
             // dan hanya dikenal oleh fungsi test()
    A = 20;
    cout<<"Nilai A di dalam fungsi test(): "<<A<<endl;
}

// Membuat fungsi main() atau fungsi utama
int main() {
```

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Komentar, Identifier, dan Variabel	
	No. LST/TE/EKA5208/02	Revisi : 01	Tgl : 20 Februari 2017

```

// A tidak dapat digunakan oleh fungsi main()
// A = 10; // SALAH
// cout<<"Nilai A di dalam fungsi main(): "
// <<A<<endl; // SALAH

// Memanggil fungsi test()
test();

return 0;
}

```

Hasil dari program 9:

```

Nilai A di dalam fungsi test(): 20

```

Keterangan:

Pada kode program 9 variabel A dideklarasikan untuk fungsi `test ()`. Oleh karena itu variabel A merupakan variabel lokal yang hanya dapat diakses oleh fungsi `test ()` sedangkan fungsi `main ()` tidak bisa mengakses variabel A tersebut.

## JENIS-JENIS VARIABEL DI DALAM C++

### 1. VARIABEL AUTO

Variabel auto merupakan variabel normal yang dideklarasikan di dalam lingkup atau blok program tertentu. Variabel ini juga termasuk variabel lokal. Pada saat pendeklarasian, secara eksplisit dapat ditambah kata `auto` saat menuliskan variabel.

Contoh:

```

{
    int a;
    auto int b;
    ....
}

```

Keterangan:

a dan b merupakan variabel lokal yang hanya dikenal di dalam blok program bersangkutan.

### 2. VARIABEL STATIS

Variabel statis merupakan variabel yang menempati ruang memori komputer secara permanen. Artinya nilai terakhir dari variabel ini akan terus disimpan.

Dalam C++, untuk menyatakan variabel statis digunakan kata kunci `static`.

Bentuk umum:

```

Static tipe_data nama_variabel;

```

Contoh:


```

static int a;
static char c;

```

variabel statis dapat diperankan sebagai variabel global maupun lokal.



	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Komentar, Identifier, dan Variabel	
	No. LST/TE/EKA5208/02	Revisi : 01	Tgl : 20 Februari 2017

### 3. VARIABEL STATIS LOKAL

Variabel statis lokal diterapkan di dalam suatu fungsi, sehingga nama variabel ini hanya akan dikenali di dalam fungsi tersebut saja. Dan seperti halnya variabel statis, nilai terakhir yang dihasilkan akan terus disimpan.

Contoh program 10:

```
#include <iostream>

using namespace std;

// Membuat fungsi dengan nama contoh
int contoh() {
    int A = 0; // Mendeklarasikan variabel biasa
    A = A + 10;
    return A;
}

int main() {
    int x, y, z; // Mendeklarasikan variabel x, y, dan z
    x = contoh(); // Memanggil fungsi contoh()
                // untuk yang pertama kali
    y = contoh(); // Memanggil fungsi contoh()
                // untuk yang kedua kali
    z = contoh(); // Memanggil fungsi contoh()
                // untuk yang ketiga kali

    // Menampilkan nilai
    // yang telah diproses di dalam fungsi
    cout<<"Nilai pada pemanggilan pertama : "<<x<<endl;
    cout<<"Nilai pada pemanggilan kedua : "<<y<<endl;
    cout<<"Nilai pada pemanggilan ketiga : "<<z<<endl;

    return 0;
}
```

Hasil program 10:

```
Nilai pada pemanggilan pertama : 10
Nilai pada pemanggilan kedua : 10
Nilai pada pemanggilan ketiga : 10
```

Keterangan:

Pada kode di atas, variabel A merupakan variabel normal (non-statis). Oleh karena itu setiap pemanggilan fungsi `contoh ()` akan memberikan nilai yang sama.

Tetapi apabila variabel A diubah menjadi variabel statis seperti pada program 11 di bawah maka hasil yang didapatkan akan berbeda.

Program 11:

```
#include <iostream>

using namespace std;

// Membuat fungsi dengan nama contoh
int contoh() {
    static int A = 0; // Mendeklarasikan variabel statis
    A = A + 10;
```



FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA

LABSHEET ALGORITMA DAN STRUKTUR DATA

Semester 1

Komentar, Identifier, dan Variabel

No.  
LST/TE/EKA5208/02

Revisi : 01

Tgl : 20 Februari  
2017

Hal 10 dari 12

```
return A;
}

int main() {
    int x, y, z; // Mendeklarasikan variabel x, y, dan z
    x = contoh(); // Memanggil fungsi contoh()
                // untuk yang pertama kali
    y = contoh(); // Memanggil fungsi contoh()
                // untuk yang kedua kali
    z = contoh(); // Memanggil fungsi contoh()
                // untuk yang ketiga kali

    // Menampilkan nilai
    // yang telah diproses di dalam fungsi
    cout<<"Nilai pada pemanggilan pertama : "<<x<<endl;
    cout<<"Nilai pada pemanggilan kedua : "<<y<<endl;
    cout<<"Nilai pada pemanggilan ketiga : "<<z<<endl;

    return 0;
}
```

Hasil:

```
Nilai pada pemanggilan pertama : 10
Nilai pada pemanggilan kedua : 20
Nilai pada pemanggilan ketiga : 30
```

Keterangan:

Hasil program di atas memberikan nilai yang berbeda karena, pada pemanggilan fungsi yang pertama kali nilai variabel A = 0 (sesuai dengan inisialisasi. Fungsi akan menambahkan nilai tersebut dengan 10, amaka nilai yang dikembalikan fungsi contoh adalah 10. Karena variabel A adalah variabel statis maka saat pemanggilan fungsi yang kedua variabel A menyimpan nilai 10 dan bukan 0 lagi. Oleh karena itu pemanggilan fungsi yang kedua tentu akan menambahkan nilai A dengan nilai 10, sehingga nilai yang dihasilkan adalah 20, demikian seterusnya.

#### 4. VARIABEL STATIS GLOBAL

Variabel statis global digunakan apabila variabel statis lokal tidak dapat digunakan.

#### 5. VARIABEL REGISTER

Variabel register akan tersimpan di register cpu. Dalam C++variabel register hanya dapat diisi oleh tipe data char, int, dan pointer saja serta hanya boleh dideklarasikan sebagai variabel lokal ataupun parameter dari sebuah fungsi.

Bentuk umum:

```
register tipe_data nama_variabel;
```

conoh program 12:

```
#include <iostream>
```

```
using namespace std;
```

```
// Membuat fungsi untuk menghitung M pangkat e
int Hitung(register int M, register int e) {
```



```
    register int temp;
    temp = 1;
    for ( ; e; e--) {
        temp = temp * M;
    }
    return temp;
}

// Fungsi utama
int main() {
    int X;
    X = Hitung(5, 3); // 5 pangkat 3
    cout<<X<<endl;

    return 0;
}
```

Hasil:

125

## 6. VARIABEL EKSTERNAL

Variabel eksternal adalah variabel global yang ada atau sudah dideklarasikan di dalam file lain. Variabel ini biasa digunakan apabila program yang ditulis berjumlah lebih dari satu file. Kata kunci yang digunakan untuk mendeklarasikan variabel ini adalah `extern` yang letak penulisannya di depan deklarasi variabel yang bersangkutan.

Contoh:

Program 12:

File pertama: `main.cpp`

```
#include <iostream>

int a;

// mendeklarasikan fungsi eksternal
extern void tulisNilai();

int main() {
    a = 99;
    tulisNilai();

    return 0;
}
```


File kedua: `tulis.cpp`

```
#include <iostream>

using namespace std;

// mendeklarasikan variabel eksternal
extern int a;

void tulisNilai(void)
{
```

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Komentar, Identifier, dan Variabel	
	No. LST/TE/EKA5208/02	Revisi : 01	Tgl : 20 Februari 2017

```
cout<<"Nilai a: "<<a<<endl;
}
```

Keterangan:

Pada kedua contoh di atas, pada file kedua dideklarasikan variabel eksternal a. var a merupakan variabel yang sudah dideklarasikan di file pertama. Selain variabel eksternal, dalam file pertama digunakan fungsi `tulisNilai ()` yang sebenarnya terdapat dalam file kedua. Oleh karena itu perlu dideklarasikan fungsi eksternal di dalam file pertama dengan menggunakan kata kunci `extern`.

Untuk melakukan kompilasi dan eksekusi kedua program di atas dalam dev C++ perlu diikuti langkah-langkah sebagai berikut:

1. Buka Dev C++, pilih menu file – new – project
2. Pilih tab Basic, empty project pilih opsi C++ project
3. Pada code editor yang tersedia, tuliskan kode untuk file pertama dan simpan sebagai `main.cpp`
4. Pilih menu project – new file, lalu tuliskan kode file kedua dan simpan sebagai `tulis.cpp`
5. Pilih menu execute – compile untuk melakukan kompilasi project (semua file akan dikompilasi)
6. Pilih menu execute – run untuk menjalankan program

Hasil yang diperoleh setelah menjalankan dua program di atas adalah sbb:

```
Nilai a: 99
```

#### D. LANGKAH KERJA

1. Berdoalah sebelum memulai pekerjaan.
2. Baca dan pahami labsheet yang diberikan.
3. Kerjakan contoh-contoh program di atas .
4. Apabila mengalami kesulitan atau pekerjaan telah selesai, konsultasikan pada dosen pengampu.
5. unggah hasil pekerjaan anda di bestmart pada mata kuliah algoritma dan struktur data.

#### E. TUGAS

1. Pelajari dan pahami setiap kode program di atas.
2. Buatlah pseudocode dari setiap kode program di atas.

#### F. DAFTAR PUSTAKA

Kajian pustaka diambil dari buku:

**Budi Raharjo. 2015. Pemrograman C++ Mudah dan Cepat Menjadi Master C++. Bandung: Penerbit Informatika.**