	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

#### A. KOMPETENSI

1. Mampu memahami jenis-jenis dan penggunaan tipe dasar di dalam C++
2. Mendefinisikan tipe data bentukan (array, struktur, dan enumerasi)
3. Memahami null terminated string

#### B. ALAT DAN BAHAN

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

#### C. KAJIAN TEORI

Tipe data berfungsi untuk merepresentasikan jenis dari nilai yang terdapat dalam program. Pada saat mendeklarasikan suatu variabel perlu disebutkan tipe data dari variabel tersebut. Kesalahan dalam menyebutkan tipe data akan menyebabkan program yang dibuat tidak dapat dijalankan. Walaupun program dapat dijalankan, maka program akan memberikan nilai yang tidak akurat seperti yang diharapkan.

Dalam C++, tipe data dikelompokkan menjadi 2 yaitu:

1. Tipe data dasar
2. Tipe data bentukan


##### Macam-macam tipe data dasar yaitu:

1. Tipe bilangan bulat

Digunakan untuk merepresentasikan data numerik yang berupa bilangan bulat, yaitu bilangan yang tidak memiliki angka dibelakang koma. Tipe data yang termasuk dalam kategori ini dapat dilihat pada tabel 1 di bawah.

Tabel 1. Tipe data bilangan bulat

Tipe Data	Ukuran (dalam bit)	Rentang
int	16 atau 32	-32.768 sampai 32.767 atau -2.147.483.648 sampai 2.147.483.647
Unsigned int	16 atau 32	0 sampai 65.535 atau 0 sampai 4.294.967.295
Signed int	16 atau 32	-32.768 sampai 32.767 atau -2.147.483.648 sampai 2.147.483.647
Short int	16	-32.768 sampai 32.767
Unsigned short int	16	0 sampai 65.535
Signed short int	16	-32.768 sampai 32.767
Long int	32	-2.147.483.648 sampai 2.147.483.647
Signed long int	32	-2.147.483.648 sampai 2.147.483.647
Unsigned long int	32	0 sampai 4.294.967.295

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

### Contoh program 1:

```
#include <iostream>

using namespace std;

int main() {
    // Mendeklarasikan variabel x dengan tipe data int
    int x;

    // Melakukan assigment terhadap variabel x
    x = 3;
    cout<<"Nilai x: "<<x;

    return 0;
}
```

### Hasil:

```
Nilai x: 3
```

## 2. Tipe bilangan Riil

Tipe yang merepresentasikan data-data yang mengandung angka di belakang koma. Tipe data yang ada dalam kategori ini ditunjukkan oleh tabel 2 di bawah.

Tabel 2. Tipe data bilangan Riil

Tipe data	Ukuran dalam bit)	Rentang	Presisi
Float	32	1.22E-38 sampai 3.4E+38	6 digit
double	64	2.3E-308 sampai 1.7E+308	15 digit
Long double	80	3.4E-4932 sampai 1.1E+4932	19 digit

### Contoh program 2:

```
#include <iostream>

using namespace std;

int main() {
    // Mendeklarasikan variabel y
    // dengan tipe data double
    double y;


    // Melakukan assigment terhadap variabel y
    y = 222.134;
    cout<<"Nilai y: "<<y;

    return 0;
}
```

### Hasil:

```
Nilai y: 222.134
```

## 3. Tipe logika

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

Tipe ini merepresentasikan data-data yang hanya mengandung dua buah nilai yaitu nilai logika (boolean).

Dalam C++ standar, tipe ini dinyatakan dalam tipe `bool`, yang memiliki nilai berupa `true` (benar) atau `false` (salah).

Contoh program 3:

```
#include <iostream>

using namespace std;

int main() {
    // Mendeklarasikan variabel y
    // dengan tipe data bool
    bool benar;

    // Meminta input dari user
    int a, b;
    cout<<"Masukkan nilai a: "; cin>>a;
    cout<<"Masukkan nilai b: "; cin>>b;

    // Melakukan assignment terhadap variabel benar
    benar = a > b;

    if (benar == true) { // bisa ditulis: if (benar) {
        cout<<"a lebih besar dari b";
    } else {
        cout<<"a lebih kecil dari atau sama dengan b";
    }

    return 0;
}
```

Hasil:

```
Masukkan nilai a: 3
Masukkan nilai b: 6
a lebih kecil dari atau sama dengan b
```

#### 4. Tipe karakter


Tipe ini digunakan untuk merepresentasikan data bertipe karakter. Dalam C++, tipe karakter dinyatakan dalam `char`. Untuk karakter *unicode (wide character)* dinyatakan dalam `wchar_t`.

Tipe data karakter dapat ditunjukkan dalam tabel 3 berikut:

Tabel 3. Tipe data karakter

Tipe data	Uuran (dalam bit)	Rentang
Char	8	-128 sampai 127 atau 0 sampai 255
Unsigned char	8	0 sampai 255
Signed char	8	-128 sampai 127
Wchar_t	16 atau 32	0 sampai 65.535

Dalam C++, tipe karakter selalu diapit oleh petik tunggal. Apabila karakter diapit oleh petik ganda maka akan dianggap sebagai string.

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

#### Contoh program 4:

```
#include <iostream>

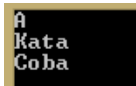
using namespace std;

int main() {
    // Mendeklarasikan variabel Karakter
    // dan mengisinya dengan nilai 'A'
    char Karakter = 'A';

    // Mendeklarasikan variabel Teks
    // dan mengisinya dengan nilai "Kata"
    char* Teks = (char*) "Kata";
    char TEKS[10] = "Coba";
    cout<<Karakter<<endl;
    cout<<Teks<<endl;
    cout<<TEKS<<endl;

    return 0;
}
```

#### Hasil:



#### Tipe data bentukan:

Tipe data bentukan adalah tipe data yang dibuat sendiri sesuai kebutuhan dalam program yang dibuat.

Macam-macam tipe data bentukan::

##### 1. Tipe struktur


Struktur adalah tipe data bentukan yang menyimpan lebih dari satu variabel bertipe sama maupun berbeda.

Bentuk umum pendeklarasian tipe data struktur dalam C++:

```
Struc nama_struktur {
    tipe_data variabel1;
    tipe_data variabel2;
    ...
};
```

Dalam pendeklarasian struktur harus diakhiri dengan tanda titik koma (;). Pada saat pengaksesan terhadap isi dari struktur digunakan operator titik (.).

Sebagai contoh, tipe data struktur dapat dilihat saat perlu menjabarkan informasi data diri dari siswa. Seperti yang sudah diketahui bahwa siswa pasti mempunyai nomor induk siswa

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

(NIS), nama, alamat, dll. Oleh karena itu informasi data diri siswa dapat diimplementasikan ke dalam tipe data struktur. Contoh implementasi dalam program adalah sebagai berikut:

```
//membuat tipe struktur dengan nama SISWA
Struc SISWA {
    char NIS [9];
    char Nama [25];
    char Alamat [20];
    char Kota [15];
};

//menggunakan tipe SISWA dalam mendeklarasikan variabel A
SISWA A;

//melakukan pengisian nilai terhadap variabel A
Strcpy (A.NIS, "135044018");
Strcpy (A>Nama, "Daniel Joseph");
Strcpy (A.Alatnat, "Palagan");
Strcpy (A.Kota, "Yogyakarta");
```

Contoh lengkap mengenai struktur dapat dilihat pada program 5 berikut:

```
#include <iostream>
#include <cstring>

using namespace std;

int main() {

    struct SISWA {
        char NIS[9];
        char Nama[25];
        char Alamat[20];
        char Kota[15];
    };

    SISWA A; // Mendeklarasikan variabel A yang bertipe SISWA

    strcpy(A.NIS, "175044018");
    strcpy(A>Nama, "Daniel Joseph");
    strcpy(A.Alatnat, "Palagan");
    strcpy(A.Kota, "Yogyakarta");


    // Menampilkan nilai yang diisikan ke layar
    cout<<A.NIS<<endl;
    cout<<A>Nama<<endl;
    cout<<A.Alatnat<<endl;
    cout<<A.Kota<<endl;

    return 0;
}
```

## 2. Tipe array

Digunakan untuk merepresentasikan kumpulan data sejenis atau data yang bertipe sama.

## 3. Tipe enumerasi

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

Tipe enumerasi adalah tipe data yang nilainya terbatas pada nilai-nilai yang telah didefinisikan saja. Tipe ini digunakan untuk membentuk tipe data yang nilainya bersifat pasti, contoh: jenis kelamin, nama hari, dll.

Kata kunci untuk mendefinisikan tipe enumerasi adalah `enum`.

Bentuk umum:

```
enum nama_tipe {nilai_1, nilai_2, ...};
```

contoh:

```
enum jenis_kelamin {pria, wanita};
```

```
enum hari {minggu, senin, selasa, rabu, kamis, jumat, sabtu};
```

contoh program 6:

```
#include <iostream>
#include <cstring>

using namespace std;

// Mendefinisikan tipe JENIS_KELAMIN
enum JENIS_KELAMIN {Pria, Wanita};

int main() {

    struct SISWA {
        char NIS[9];
        char Nama[25];
        JENIS_KELAMIN gender;
    } A;

    strcpy(A.NIS, "075022419");
    strcpy(A.Nama, "Hadassah");
    A.gender = Wanita;

    cout<<"NIS : "<<A.NIS<<endl;
    cout<<"Nama : "<<A.Nama<<endl;
    cout<<"Gender : "<<A.gender<<endl;

    return 0;
}
```

#### 4. Tipe string


Tipe ini untuk merepresentasikan data yang berupa teks.

Contoh:

```
char namadepan [15]; //string yang berupa array
char *namabelakang; //string yang berupa pointer
```

untuk mengisi data ke dalam string yang berupa array, dapat digunakan fungsi `strcpy ()`, seperti kode program berikut:

```
strcpy (namadepan, "Hadassah");
```

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

untuk string yang berupa pointer, dapat digunakan operator sama dengan (=) seperti kode berikut:

```
namabelakang = (char*) "Quenn Sugiarto";
```

(char \*) merupakan kode yang digunakan untuk melakukan typecasting (pemeranan tipe data) terhadap teks yang akan diisikan.

#### Contoh program 7:

```
#include <iostream>
#include <cstring>

using namespace std;

int main() {

    char namadepan[15];
    char *namabelakang;

    strcpy(namadepan, "Hadassah");
    namabelakang = (char *) "Queen Sugiarto";

    cout<<"Nama lengkap: "<<namadepan
    <<" "
    <<namabelakang;

    return 0;
}
```

### **Null Terminated String**

Merupakan string yang diakhiri oleh karakter null ('\0'). Dalam C++, fungsi-fungsi yang menggunakan string dideklarasikan dalam file header <cstring>, jika dalam bahasa C <string.h>.


String (dalam gaya bahasa C) merupakan kumpulan atau array (larik) dari tipe karakter dengan karakter terakhir berupa karakter null.

Contoh:

```
//mendeklarasikan string dengan lebar 9 karakter
Char NAMA [9];
//mengisi (menyalin) teks "Hadassah" ke variabel NAMA
Strcpy (NAMA, "Hadassah");
```

Apabila kode di atas diamati, sebenarnya variabel NAMA hanya diisi 8 karakter. Sisa ruang 1 karakter digunakan untuk menyimpan karakter null. Berikut adalah gambar yang akan mengilustrasikan konsep tersebut.

'H'	'a'	'd'	'a'	's'	's'	'a'	'h'	'\0'
NAMA [0]	NAMA [1]	NAMA [2]	NAMA [3]	NAMA [4]	NAMA [5]	NAMA [6]	NAMA [7]	NAMA [8]

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

seperti yang dilihat pada gambar di atas, masing-masing elemen karakter dalam string akan diambil berdasarkan indeksinya. Indeks array selalu dimulai dari 0 bukan 1. Oleh karena itu, setiap karakter dalam string NAMA dapat diakses dengan cara NAMA [0], NAMA [1] dst. Dalam mengisi nilai ke dalam variabel bertipe string, tidak perlu memasukkan karakter null secara manual karena C++ secara otomatis akan menambahkannya.

Lebih jelasnya dapat dilihat pada contoh program 8 berikut:

```
#include <iostream>
#include <cstring>

using namespace std;

int main() {

    // Mendeklarasikan variabel NAMA dengan tipe char dan
    // lebar 5 karakter
    char NAMA[5];

    // Mengisikan nilai ke dalam variabel NAMA
    strcpy(NAMA, "BIMO");

    cout<<"Karakter ke-0: "<<NAMA[0]<<endl;
    cout<<"Karakter ke-1: "<<NAMA[1]<<endl;
    cout<<"Karakter ke-2: "<<NAMA[2]<<endl;
    cout<<"Karakter ke-3: "<<NAMA[3]<<endl;
    cout<<"Karakter ke-4: "<<NAMA[4]<<endl;

    return 0;
}
```

### **Menyalin string**

Proses penyalinan string dapat dilakukan menggunakan fungsi strcpy ().

Bentuk umum:

Char \*strcpy (char S1, const char \*S2);

Fungsi ini akan menyalin string dari S2 ke S1. Berikut ini contoh program yang akan menunjukkan fungsi tersebut.

Contoh program 9:

```
#include <iostream>
#include <cstring>


using namespace std;

int main() {
    char S1[50];
    char* S2;

    // Melakukan pengisian nilai terhadap variabel S2
    S2 = (char*) "Ini adalah contoh penyalinan string";

    // Melakukan penyalinan string
    // dari variabel S2 ke variabel S1
```



	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

```
strcpy(S1, S2);

// Menampilkan nilai dari variabel S1
cout<<S1<<endl;

return 0;
}
```

Selain menggunakan fungsi `strcpy ()`, proses penyalinan string juga dapat dilakukan dengan fungsi `strncpy ()`. Kelebihan fungsi ini adalah dapat menentukan jumlah karakter yang akan disalin (tidak harus semua karakter).

Bentuk umum:

```
Char*strncpy(char* S1, const char* S2, size_t n);
```

Fungsi tersebut berguna untuk menyalin string sebanyak n karakter dari variabel S2 ke variabel S1. Berikut contohnya:

Program 10:

```
#include <iostream>
#include <cstring>

using namespace std;

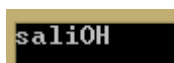
int main() {
    char S1[50] = "CONTOH";
    char* S2 = (char *) "salin string";

    // Melakukan penyalinan string
    // dari variabel S2 ke variabel S1
    strncpy(S1, S2, 4);

    // Menampilkan nilai dari variabel S1
    cout<<S1<<endl;

    return 0;
}
```

Hasil:



Keterangan:


Pada kode di atas, 4 karakter pertama dalam string S1 ditimpa oleh 4 karakter pertama string S2.

### **Menyambung String**

Dua buah string dapat digabung menggunakan fungsi `strcat ()`.

Bentuk umum:

```
Char* strcat (char* S1, const char* S2);
```

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

Fungsi tersebut akan menggabungkan string S2 ke posisi akhir string S1.

Contoh program 11 menunjukkan penggabungan string:

```
#include <iostream>
#include <cstring>

using namespace std;

int main() {

    char S1[50];
    char* S2;
    char* spasi = (char*) " ";

    // Melakukan pengisian nilai
    // terhadap variabel S1 dan S2
    strcpy(S1, "Teks");
    S2 = (char*) "Gabungan";

    // Menambahkan karakter spasi (' ')
    // ke dalam variabel S1
    strcat(S1, spasi);

    // Melakukan penggabungan string yang terdapat
    // pada variabel S1 dan S2
    strcat(S1, S2);

    // Menampilkan nilai dari variabel S1
    // setelah proses penggabungan
    cout<<S1<<endl;

    return 0;
}
```

Hasil:

**Teks Gabungan**

Proses penggabungan string juga dapat dilakukan dengan menggunakan fungsi `strncat()`.

Melalui fungsi ini jumlah karakter yang akan digabung dapat ditentukan sendiri.

Bentuk umum:


```
Char* strncat (char* S1, const char* S2, size_t n);
```

Fungsi tersebut berguna untuk menggabungkan n karakter dari variabel S2 ke posisi akhir variabel S1. Cara kerja fungsi `strncat()` dapat dilihat pada program 12 berikut:

```
#include <iostream>
#include <cstring>

using namespace std;

int main() {
    char S1[50];
    char* S2;
```

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

```

// Melakukan pengisian nilai
// terhadap variabel S1 dan S2
strcpy(S1, "Nama saya adalah ");
S2 = (char*) "Hadassah Queen";

// Menggabungkan 4 karakter
// dari variabel S2 ke variabel S1
strncat(S1, S2, 8);

// Menampilkan nilai dari variabel S1
// setelah dilakukan penggabungan
cout<<S1<<endl;

return 0;
}

```

Hasil:

Nama saya adalah Hadassah

### memperoleh panjang string

untuk memperoleh panjang dari suatu string dapat digunakan fungsi `strlen()`. Catt: karakter null tidak akan dihitung.

Bentuk umum:

```
size_t strlen(const char* S);
```

contoh program 13 berikut adalah contoh fungsi untuk mengembalikan panjang atau jumlah karakter dari string S.

```

#include <iostream>
#include <cstring>

using namespace std;

int main() {
    char* S1;
    char* S2;
    int panjang_S1, panjang_S2;


    // Melakukan pengisian nilai
    // terhadap variabel S1 dan S2
    S1 = (char*) "Hadassah";
    S2 = (char*) "Queen";

    panjang_S1 = strlen(S1);
    panjang_S2 = strlen(S2);

    // Menampilkan nilai dari
    // variabel panjang_S1 dan panjang_S2
    cout<<"Panjang S1 : "<<panjang_S1<<endl;
    cout<<"Panjang S2 : "<<panjang_S2<<endl;

    return 0;
}

```

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

Hasil:

```
Panjang S1 : 8
Panjang S2 : 5
```

### Melakukan input data string

Perintah standar untuk melakukan input data dalam C++ adalah `cin`. Tetapi ketika data yang dimasukkan bertipe string maka perlu diperhatikan saat data yang dimasukkan mengandung spasi. Karena karakter spasi yang ditemukan pertamakali dalam string akan dianggap sebagai karakter null.

#### Contoh program 14:

```
#include <iostream>

using namespace std;

int main() {

    char S[20];

    cout<<"Masukkan nama Anda: ";

    // Meminta input dari user dan menyimpan nilainya
    // ke dalam variabel S
    cin>>S;

    // Menampilkan kembali nama yang telah dimasukkan
    cout<<"Hai "<<S<<" , apa kabarmu?"<<endl;

    return 0;
}
```

Saat memasukkan input nama tanpa spasi tidak muncul masalah, tetapi ketika input nama yang dimasukkan menggunakan spasi maka nama yang diinput tidak sesuai dengan output. Oleh karena itu untuk mengatasi hal tersebut digunakan fungsi `getline ()`.

#### Bentuk umum:

```
&istream getline (char* buffer, int ukuran, char delimiter= '\n');
&istream getline (signed char* buffer, int ukuran, char delimiter= '\n');
&istream getline (unsigned char* buffer, int ukuran, char delimiter= '\n');
```

Dengan menggunakan fungsi `getline ()`, program di atas dapat diperbaiki seperti berikut:


#### Contoh program 15:

```
#include <iostream>

using namespace std;

int main() {

    char S[20];
```

	<b>FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA</b>		
	<b>LABSHEET ALGORITMA DAN STRUKTUR DATA</b>		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

```

cout<<"Masukkan nama Anda: ";

// Meminta input dari user
cin.getline(S, sizeof(S) - 1);

// Menampilkan nama yang telah dimasukkan
cout<<"Hai "<<S<<" , apa kabarmu?"<<endl;

return 0;
}

```

Atau dengan menggunakan string, program 15 di atas dapat dituliskan sebagai berikut:

#### Contoh program 16:

```

#include <iostream>
#include <string>

using namespace std;

int main() {

// mendeklarasikan string gaya C++
string S;

cout<<"Masukkan nama Anda: ";

// Meminta input dari user
getline(cin, S);

// Menampilkan nama yang telah dimasukkan
cout<<"Hai "<<S<<" , apa kabarmu?"<<endl;

return 0;
}

```

#### 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 algoritma dengan bahasa narasi, flowchart, dan pseudocode dari setiap kode program di atas.

#### F. DAFTAR PUSTAKA

Kajian pustaka diambil dari buku:

	FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA		
	LABSHEET ALGORITMA DAN STRUKTUR DATA		
	Semester 1	Tipe Data dalam C++	
	No. LST/TE/EKA5208/03	Revisi : 01	Tgl : 20 Februari 2017

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