

ALGORITMA DAN PEMROGRAMAN

4 SKS



Al-Khuwarizmi

TEKNIK PERANGKAT LUNAK
FT - UNPAM

Pertemuan	Pokok Bahasan	Sub Pokok Bahasan	Tujuan Instruksional	Metode	Waktu
1 - 2	Pengantar, Elemen-elemen Dasar Bahasa Pemrograman Prosedural	Penjelasan perkuliahan	Mahasiswa dapat memahami definisi algoritma, cara penulisan algoritma, tipe data dalam bahasa C++, dan cara pemberian nama variabel	Ceramah, diskusi	200 mnt
		Pengertian Algoritma dan Program, bahasa pemrograman, Sifat algoritma			
		Notasi Algoritmik : untaian kalimat deskriptif, diagram alir, pseudo -code			
		Struktur Bahasa C++ : Preprocessor, deklarasi, tubuh program, baris komentar			
		Contoh sintak C++ untuk input dan output standar			
		Identifier dan aturan pemberian nama, keyword, variabel, tipe data dan literal			
3 - 4	Operator dan ekspresi, File I/O	Operator : aritmetika, penambahan/pengurangan, penugasan, perbandingan, logika, bit, penugasan	Mahasiswa dapat memahami operator, File I/O, dan format I/O	Ceramah, diskusi	200 mnt
		Type-Casting			
		File I/O secara sekuensial dalam C			
		Format keluaran C++: lebar, pemenuh, rata kiri dan kanan, presisi, format sains, basis			
		Contoh-contoh pemecahan masalah			
5 - 6	Runtunan, Pemilihan	Algoritma runtunan dan Pemilihan	Mahasiswa mengerti algoritma runtunan dan pemilihan dan dapat mengaplikasikannya dalam suatu permasalahan	Ceramah, diskusi	200 mnt
		Operator perbandingan, Operator logika, Pernyataan if, else, switch			
		Contoh-contoh pemecahan masalah			
7 - 8	Pengulangan	Struktur pengulangan	Mahasiswa mengerti algoritma pengulangan dan dapat mengaplikasikannya dalam suatu permasalahan	Ceramah, diskusi	200 mnt
		Penggunaan for, do, while, continue			
		Contoh-contoh pemecahan masalah			
9 - 10	Array	Array berdimensi satu, banyak	Mahasiswa mengerti dan dapat menggunakan array dalam suatu permasalahan	Ceramah, diskusi	200 mnt
		Operasi pada array			
		Array karakter			
		Contoh-contoh pemecahan masalah			
11 - 12	Studi Kasus	Mahasiswa secara berkelompok menyelesaikan suatu permasalahan	Mahasiswa dapat mengaplikasikan bahasan-bahasan yang telah dipelajari		200 mnt
UTS					
13 - 14	Pointer	Pengertian	Mahasiswa mengerti penggunaan pointer dan dapat mengaplikasikannya dalam suatu masalah	Ceramah, diskusi	200 mnt
		Deklarasi, operasi dasar pada pointer			
		Pointer dan Array, Pointer dan String, Array dari pointer, Pointer menunjuk ke Pointer			
		Pengalokasian memori secara dinamis, Referensi			
		Contoh-contoh pemecahan masalah			
15 - 16	Fungsi	Kegunaan fungsi, struktur fungsi, fungsi tanpa nilai balik	Mahasiswa mengerti penggunaan fungsi dan dapat mengaplikasikannya dalam suatu permasalahan	Ceramah, diskusi	200 mnt
		Lingkup variabel : Otomatis, eksternal, statis			
		Fungsi dengan argumen			
		Melewatkan argumen array, pointer ke fungsi			
		Fungsi bernilai balik, fungsi rekursif			
		Contoh-contoh pemecahan masalah			

17 - 18	Pencarian	Persoalan Pencarian : memeriksa keberadaan data, hasil pencarian berupa index array, hasil pencarian berupa boolean	Mahasiswa mengerti algoritma pencarian dan dapat menggunakannya dalam suatu permasalahan	Ceramah, diskusi	200 mnt
		Pencarian Beruntun(sequential searching), Pencarian bagi dua			
		Contoh-contoh pemecahan masalah			
19 - 20	Pengurutan	Definisi	Mahasiswa mengerti algoritma pengurutan dan dapat menggunakannya dalam suatu permasalahan	Ceramah, diskusi	200 mnt
		Metoda Pengurutan Apung(Bubble Sort)			
		Metode Pengurutan Seleksi(Selection Sort)			
		Contoh-contoh pemecahan masalah			
21 - 22	Pengurutan	Pengurutan Sisipan(Insertion Sort)	Mahasiswa mengerti algoritma pengurutan dan dapat menggunakannya dalam suatu permasalahan	Ceramah, diskusi	200 mnt
		Pengurutan Shell(Shell Sort)			
		Contoh-contoh pemecahan masalah			
23 - 24	C++ Stream I/O Studi Kasus	Akses file secara sequensial dan random, Mahasiswa secara berkelompok menyelesaikan suatu permasalahan	Mahasiswa dapat mengaplikasikan bahasan-bahasan yang telah dipelajari		200 mnt
UAS					

Referensi :

1. Abdul kadir, Panduan Pemrograman Visual C++, Penerbit Andi Yogyakarta
2. Rinaldi Munir, Algoritma dan Pemrograman, Edisi-3, Informatika Bandung
3. M. Sjukani, Algoritma dan Struktur Data dengan C, C++, dan Java, Mitra Wacana Media
4. H.M. Deitel, C How To Program, 3rd, Prentice Hall
5. Bacaan lain dari Internet

MODUL 1 - 2

PENGERTIAN DASAR

Algoritma :

- ❖ Rangkaian langkah-langkah/alur pikiran untuk menyelesaikan suatu masalah yang sifatnya logis dan sistimatis

Sifat-sifat Algoritma :

- ✓ Mudah dibaca/jelas maksudnya (tiap langkah jelas uraiannya)
- ✓ Berawal dan berakhir
- ✓ Tidak harus ada data masukan yang dimasukkan dari luar
- ✓ Paling tidak ada satu keluaran
- ✓ Tiap instruksi harus efektif

Bahasa pemrograman :

- ❖ Bahasa komputer yang digunakan dalam penulisan program

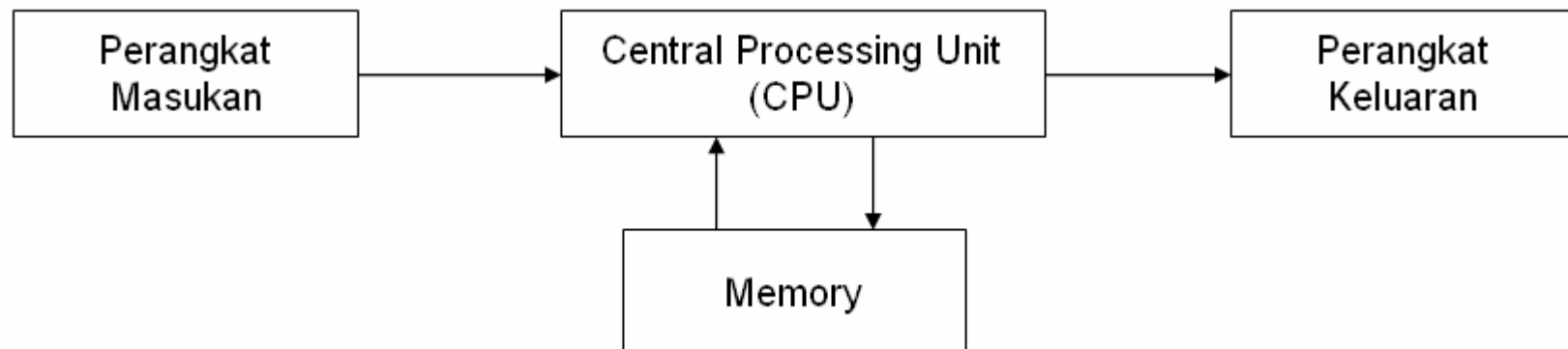
Pemrogram/Programmer :

- ❖ Orang yang membuat program komputer

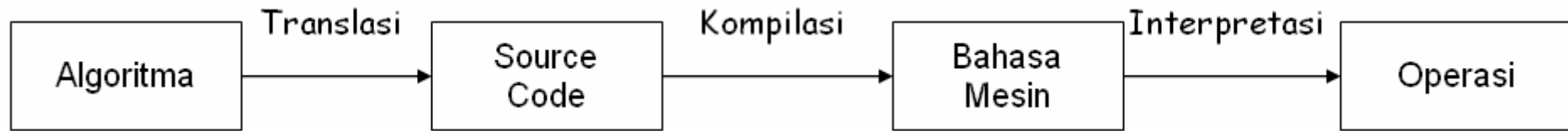
Pemrograman :

- ❖ Kegiatan merancang atau menulis program komputer

PERANGKAT UTAMA KOMPUTER



TAHAP PELAKSANAAN PROGRAM OLEH KOMPUTER



STRUKTUR DASAR ALGORITMA

Algoritma berupa langkah-langkah penyelesaian suatu masalah. Langkah-langkah tersebut dapat berupa :

1. Runtunan (sequence)
2. Pemilihan (Selection)
3. Pengulangan (repetition)

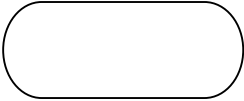
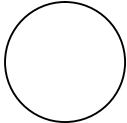

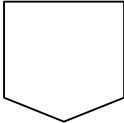


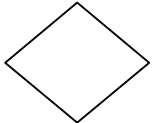

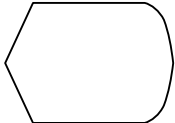



NOTASI ALGORITMIK

Notasi Algoritmik bukan notasi bahasa pemrograman sehingga siapapun dapat membuat notasi algoritmik yang berbeda. Namun demikian ketaatan atas notasi perlu diperhatikan untuk menghindari kekeliruan.

Beberapa notasi yang digunakan untuk menulis algoritma :

- I. Untaian kalimat deskriptif
Setiap langkah dinyatakan dengan bahasa yang gamblang/jelas
- II. Menggunakan diagram alir (flow chart)
- III. Menggunakan pseudo-code
Pseudo : semu, tidak sebenarnya, pura-pura; adalah notasi yang menyerupai notasi bahasa pemrograman tingkat tinggi

SIMBOL-SIMBOL DIAGRAM ALIR

Simbol	Maksud	Simbol	Maksud
	Terminal (START, END)		Titik sambungan pada halaman yang sama
	Input/Output (READ, WRITE)		Titik konektor yang berada pada halaman lain
	Proses (menyatakan assignment statement)		Call (Memanggil subprogram)
	Decision (YES, NO)		Dokumen
	Display		Stored Data
	Alur proses		Preparation (Pemberian nilai awal suatu variabel)

PSEUDO-CODE

Pernyataan	Notasi algoritmik	Maksud
Penulisan	write(x)	Nilai x dicetak di piranti keluaran
	write(x,y)	Nilai x dan y dicetak di piranti keluaran
	write("Hello")	Text Hello dicetak di piranti keluaran
Pembacaan	read(a)	Baca nilai a
	read(a,b)	Baca nilai a,b
Penugasan	bil ← x	Isikan nilai variabel x kedalam variabel bil

Teks algoritma (pseudo-code) terdiri dari :

- Head(Judul) : memberikan nama pada algoritma; umumnya nama sudah dapat memberi gambaran pada prosedur penyelesaian masalah atau masalah yang akan diselesaikan
- Deklarasi : menyatakan jenis dari setiap elemen data (variabel) yang akan digunakan dalam algoritma.
- Deskripsi : merupakan inti prosedur penyelesaian masalah; meliputi pernyataan/operasi, fungsi, penjelasan, dll.

CONTOH ALGORITMA

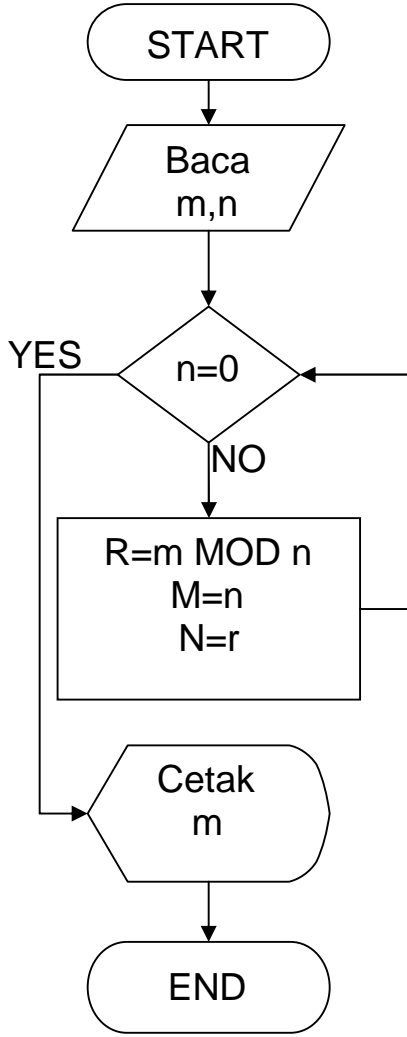
I. Untaian kalimat deskriptif

ALGORITMA Euclidean

Diberikan dua buah bilangan bulat positif m dan n ($m \geq n$). Algoritma Euclidean mencari pembagi bersama terbesar dari kedua bilangan tersebut, yaitu bilangan positif terbesar yang habis membagi m dan n

DESKRIPSI :

1. Jika $n = 0$ maka
 m adalah jawabannya;
 Stop.
 Tetapi jika $n \neq 0$,
 Lanjutkan ke langkah 2.
2. Bagilah m dengan n dan misalkan r adalah sisanya
3. Ganti nilai m dengan nilai n , nilai n dengan nilai r , lalu ulang kembali ke langkah 1.

II. Flow Chart	III. Pseudo-code
 <pre> graph TD Start([START]) --> Read[/Baca m,n/] Read --> Decision{n=0} Decision -- YES --> Print{{Cetak m}} Print --> End([END]) Decision -- NO --> Process[R=m MOD n M=n N=r] Process --> Decision </pre>	<p><u>ALGORITMA</u> Eucledian Program mencari pbt, m dan n bil bulat positif</p> <p><u>DEKLARASI</u> :</p> <p>m, n : <u>integer</u> {bil yg akan dicari pbt-nya} r : <u>integer</u> {sis hasil bagi}</p> <p><u>DESKRIPSI</u> :</p> <p><u>read</u>(m,n) {m ≥ n} <u>while</u> n ≠ 0 do r ← m <u>MOD</u> n {hitung sisa hasil bagi} m ← n n ← r <u>endwhile</u> {kondisi selesai pengulangan: n=0, maka pbt=m} <u>write</u>(m)</p>

STRUKTUR BAHASA C++

Struktur Bahasa C++ terdiri dari :

- Compiler directive (preprocessor)
- Deklarasi (variabel, fungsi)
- Tubuh program
- Baris komentar

```

/*Program pertama, menghitung luas segi empat*/ } baris komentar
//hello.cpp
#include<iostream.h> ← preprocessor
void main(void)
{
    int panjang, lebar, luas; ← deklarasi
    panjang=5;
    lebar=3;
    luas=panjang*lebar;
    cout<<"Menghitung Luas Segi Empat"<<endl;
    cout<<"Luas = "<<luas<<endl;
}
    
```

ELEMEN-ELEMEN DASAR C++

- ❖ Identifier/pengenal : suatu nama yang dipakai dalam pemrograman, umumnya diberikan kepada nama :
 - ✓ Variabel : suatu lokasi dalam memori komputer yang digunakan untuk menyimpan nilai
 - ✓ Konstanta bernama
 - ✓ Tipe data
 - ✓ Fungsi
 - ✓ Obyek
- ❖ Tipe data dasar
- ❖ Data Obyek
- ❖ Ekspresi aritmetika
- ❖ Statemen kontrol
- ❖ Fungsi

Ketentuan pemberian nama :

- Terdiri dari huruf A sampai Z, atau a sampai z, atau underscore (_), atau 0 sampai 9
- Karakter pertama suatu identifier harus huruf atau underscore
- Tidak boleh sama dengan kata kunci (keyword) dalam bahasa C++
- Tidak boleh ada spasi
- *Case sensitive* : hurup kapital dan hurup kecil diperlakukan beda

Karakter :

Elemen terkecil pada pemrograman C++ adalah karakter. Karakter dapat berupa :

- Huruf (A sampai Z, a sampai z)
- Angka (0 sampai 9)
- Simbol (*, !, dll)
- Kode kontrol (misal *newline*)

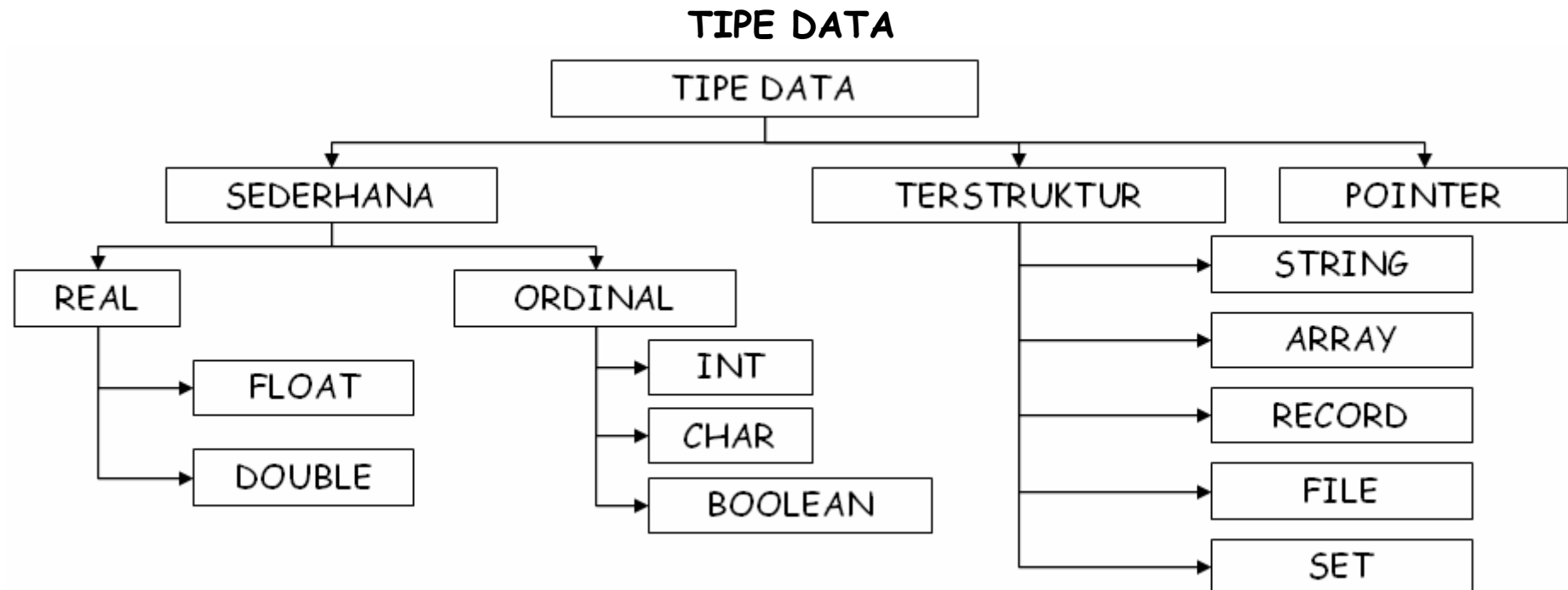
Keywords :

asm, auto, break, case, char, class, const, continue, default, delete, do, double, else, enum, extern, float, for, friends, goto, if, inline, int, long, new, operator, private, protected, public, register, return, short, signed, sizeof, static, struct, switch, template, this, typedef, union, unsigned, virtual, void, volatile, while.

Kode escape :

kode karakter yang penulisannya diawali dengan simbol \.

Kode	Ket.	Kode	Ket.	Kode	Ket.
\0	Karakter Null	\n	New line	\'	Petik tunggal
\a	Karakter audio (bel)	\r	Cariage return	\"	Petik ganda
\b	Backspace	\t	Tab	\xdd	Hexadesimal (dd:0 sd FF)
\f	Formfeed	\v	Tab vertikal	\\	Menampilkan \



TIPE DATA DASAR

Tipe	Ukuran
char, unsigned char, signed char	1 byte
short, unsigned short	2 byte
int, unsigned int	4 byte
long, unsigned long	4 byte
float	4 byte
double	8 byte
long double	8 byte

JANGKAUAN NILAI

Tipe	Nilai
char	-128 sd 127
unsigned char	0 sd 255
short	-32.768 sd 32.767
unsigned short	0 sd 65.535
long	-2.147.483.648 sd 2.147.483.647
unsigned long	0 sd 4.294.967.295
int	-2.147.483.648 sd 2.147.483.647
unsigned int	0 sd 4.294.967.295
float	$1,2 \cdot 10^{-38}$ sd $3,4 \cdot 10^{38}$
double	$2,2 \cdot 10^{-308}$ sd $1,8 \cdot 10^{308}$

Literal Karakter

Literal Karakter digunakan untuk menyatakan sebuah karakter. Karakter ditulis dalam tanda petik tunggal.

Contoh :

- 'a' : karakter huruf a
- '#' : karakter simbol #
- '8' : karakter angka 8
- '\n' : karakter *newline*

Literal String

String berarti deretan atau kombinasi sejumlah karakter. Literal String ditulis dalam tanda petik ganda.

Contoh :

"Rudy Hartono"

"4 x 4 = 16"

"" : string kosong

"abc\ndef" : antara abc dan def terdapat *newline*

"abc\tdef" : antara abc dan def terdapat tab

Konstanta Bernama

Contoh : `const double PI=3.14;`

menyatakan bahwa PI adalah literal bertipe double dengan nilai 3,14. kata kunci `const` menyatakan bahwa PI tak dapat diubah setelah didefinisikan.

Variabel

Menyatakan suatu lokasi dalam memori komputer yang digunakan untuk menyimpan suatu nilai dan nilai yang ada didalamnya bisa diubah.

Deklarasi variabel :

`int a;`

`int a, b;`

`int a=30;`

`cin>>usia;`

Pernyataan

Pernyataan adalah suatu instruksi lengkap yang berdiri sendiri dan ditunjukan untuk melaksanakan suatu tugas tertentu. Sebuah pernyataan selalu diakhiri oleh tanda *titik koma*.

Contoh :

`luas=panjang*lebar;`

`const double PI=3.14;`

TIPE DATA dan MEMBACA-MENCETAK NILAI KONSTANTA

I. Instruksi Baca - Tulis dalam bahasa C/C++

Cetak ke layar monitor	Baca dari keyboard	
printf("format",var) puts("string") cout<<var	gets(var) var=getche() var=getchar()	var=getch() scanf("format",&var)

II. Mencetak sebuah kalimat (literal String) pada program C.

Kode program	Tercetak di monitor
<pre>//mencetak_string.cpp #include<stdio.h> void main(void) { printf("Jakarta\n"); puts("Jakarta"); printf("%s","Jakarta\n"); }</pre>	<p>Jakarta</p> <p>Jakarta</p> <p>Jakarta</p>

III. Mencetak sebuah karakter pada program C.

Sebuah karakter disimpan dalam satu BYTE (8 bit)

```
printf("%c",data);
```


data	Hasil tercetak	Keterangan
'AB'	A	
'a'	a	
65	A	kode ASCII 65 adalah karakter A
97	a	kode ASCII 97 adalah karakter a
49	1	kode ASCII 49 adalah karakter 1
50	2	kode ASCII 50 adalah karakter 2
66-1	A	kode ASCII 65 adalah karakter A
35*2	F	kode ASCII 70 adalah karakter F
41)	kode ASCII 41 adalah karakter)
297)	kode ASCII 41 (297-256) adalah karakter)
172	$\frac{1}{4}$	kode ASCII 172 adalah karakter $\frac{1}{4}$
171	$\frac{1}{2}$	kode ASCII 171 adalah karakter $\frac{1}{2}$

Catatan :

Yang dimaksud dengan karakter diatas adalah karakter ASCII (American Standard Code for Information Interchange)

Perhatikan program berikut :

Statement	Tercetak	Keterangan
printf("%i",'A')	65	Nilai ASCII dari karakter 'A' adalah 65
printf("%i",'a')	97	Nilai ASCII dari karakter 'a' adalah 97
printf("%d",'A')	65	Nilai ASCII dari karakter 'A' adalah 65
printf("%d",'a')	97	Nilai ASCII dari karakter 'a' adalah 97

IV. Mencetak karakter escape pada program C.

Kode Program	Hasil
<pre>//mencetak_escape.cpp #include<stdio.h> void main(void) { printf("Jakarta\nBali\n"); printf("Jakarta\bBali\n"); printf("Jakarta\rBali\n"); printf("Jakarta\\Bali\n"); printf("Jakarta\'Bali\n"); printf("Jakarta\t\tBali\n"); printf("Jakarta\"Bali\n"); }</pre>	<pre>Jakarta Bali JakartaBali Balirta Jakarta\Bali Jakarta'Bali Jakarta Bali Jakarta"Bali</pre>

V. Mencetak konstanta numerik pada program C

```
printf("format",var);
```

Format yang dapat digunakan :

Kode format	Hasil yang tercetak
%i atau %d	Numerik integer bertanda
%u	Numerik integer tak bertanda
%li atau %ld	Numerik long integer bertanda

%lu	Numerik unsigned long integer
%f	Numerik floating point
%e	Numerik floating point
%le atau %IE	Numerik floating point double precission
%Le atau %LE	Numerik floating point long double precission
%c	Karakter ASCII
%x atau %X	Hexadesimal Integer
%o	Octal Integer

Kode program	Hasil	Kode program	Hasil
<pre>//cetak_numerik1.cpp #include<stdio.h> void main(void) { short a,b,c,d; //-32768 s/d 32767 a=-32768; b=-32769; c=32767; d=32768; printf("a = %i\n",a); printf("b = %i\n",b); printf("c = %i\n",c); printf("d = %i\n",d); }</pre>	<pre>a = -32768 b = 32767 c = 32767 d = -32768</pre>	<pre>//cetak_numerik2.cpp #include<stdio.h> void main(void) { unsigned short a,b,c,d; //0 s/d 65535 a=0; b=-1; c=65535; d=65536; printf("a = %u\n",a); printf("b = %u\n",b); printf("c = %u\n",c); printf("d = %u\n",d); }</pre>	<pre>a = 0 b = 65535 c = 65535 d = 0</pre>

short :

0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	=	32767
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	=	-32768
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	=	-32767
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	=	-1

Kode program	Hasil	Kode program	Hasil
<pre>//cetak_float1.cpp #include<stdio.h> void main(void) { float a,b,c,d; a=25; b=25.25; c=25.123456; d=25.123456789; printf("a = %f\n",a); printf("b = %f\n",b); printf("c = %f\n",c); printf("d = %f\n",d); printf("\n"); printf("a = %7.2f\n",a); printf("b = %7.3f\n",b); printf("c = %7.0f\n",c); printf("d = %7.4f\n",d); }</pre>	<pre>25.000000 25.250000 25.123455 25.123457 25.00 25.250 25 25.1235</pre>	<pre>//cetak_float1.cpp #include<stdio.h> void main(void) { float a,b,c,d; a=25; b=25.25; c=25.123456; d=25.123456789; printf("a = %e\n",a); printf("b = %E\n",b); printf("c = %e\n",c); printf("d = %E\n",d); printf("\n"); printf("a = %7e\n",a); printf("b = %5e\n",b); printf("c = %7.0E\n",c); printf("d = %10.2E\n",d); }</pre>	<pre>2.500000e+001 2.525000E+001 2.512346e+001 2.512346E+001 2.500000e+001 2.525000e+001 3E+001 2.51E+001</pre>

VI. Menginput data melalui keyboard pada program C

<pre>//baca_keyboard1.cpp #include<stdio.h> void main(void) { char S[6]; printf("Inputkan String : "); gets(S); printf("%s\n",S); }</pre>	<pre>//baca_keyboard2.cpp #include<stdio.h> #include<conio.h> void main(void) { char C; printf("Inputkan sebuah karakter : "); C=getche(); printf("\n%c\n",C); }</pre>
<pre>//baca_keyboard3.cpp #include<stdio.h> void main(void) { char C; printf("Masukan sebuah karakter : "); C=getchar(); printf("%c\n",C); }</pre>	<pre>//baca_keyboard4.cpp #include<stdio.h> #include<conio.h> void main(void) { char C; printf("Masukan sebuah karakter : "); C=getch(); printf("\n%c\n",C); }</pre>

```
//baca_keyboard5.cpp
```

```
#include<stdio.h>
```

```
void main(void)
```

```
{
```

```
    char C,D;
```

```
    printf("Inputkan dua karakter : ");
```

```
    scanf("%c",&C);
```

```
    scanf("%c",&D);
```

```
    printf("%c %c\n",C,D);
```

```
}
```

```
//baca_keyboard6.cpp
```

```
#include<stdio.h>
```

```
void main(void)
```

```
{
```

```
    int C,D,E;
```

```
    printf("Inputkan bilangan ke-1 : ");
```

```
    scanf("%i",&C);
```

```
    printf("Inputkan bilangan ke-2 : ");
```

```
    scanf("%i",&D);
```

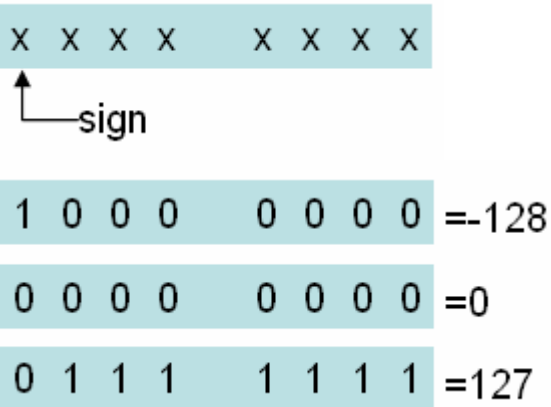
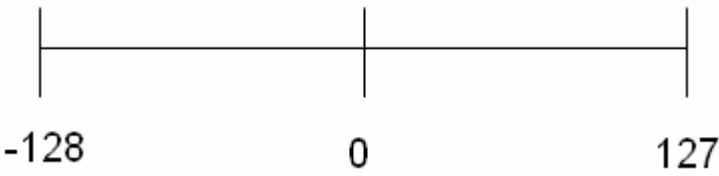
```
    E=C*D;
```

```
    printf("%i * %i = %i\n",C,D,E);
```

```
}
```

MASALAH TIPE DATA

1. Char

	<p>Bila isinya diperlakukan sebagai nilai numerik, maka bit sign :</p> <p>0 : berarti nilai positif</p> <p>1 : berarti nilai negatif</p> <p>Nilai yang dapat ditampung berkisar dari</p> 
---	---


<pre>//char1.cpp #include<iostream.h> void main(void) { char a,b,c,d; a=127; b=-128; c=128; d=-129; cout<<(int)a<<endl; cout<<(int)b<<endl; cout<<(int)c<<endl; cout<<(int)d<<endl; }</pre>	<p>Hasilnya :</p> <p>127</p> <p>-128</p> <p>-128</p> <p>127</p>	<p>Analisis :</p> <p>a = 127, masih dalam jangkauannya</p> <p>b = -128, masih dalam jangkauannya</p> <p>c = 128, diluar jangkauannya.</p> <p>128 = 1000 0000, karena dideklarasikan sebagai char maka bit MSB memiliki nilai negatif yaitu -128</p> <p>d = -129, diluar jangkauannya</p> <p>129 = 1000 0001</p> <p>-129 = 0111 1111, karena dideklarasikan sebagai char maka nilai biner tersebut sama dengan 127</p>
---	---	---

2. unsigned char

<div> <div>x x x x x x x x</div> <div>1 1 1 1 1 1 1 1 =255</div> <div>0 0 0 0 0 0 0 0 =0</div> </div>	<p>Bila isinya diperlakukan sebagai nilai numerik, maka nilai yang dapat ditampung berkisar antara 0 sampai dengan 255</p>
--	--

<pre>//char2.cpp #include<iostream.h> void main(void) { unsigned char a,b,c,d; a=0; b=255; c=-1; d=256; cout<<(int)a<<endl; cout<<(int)b<<endl; cout<<(int)c<<endl; cout<<(int)d<<endl; }</pre>	<p>Hasilnya :</p> <pre>0 255 255 0</pre>	<p>Analisis :</p> <p>a = 0, masih dalam jangkauannya</p> <p>b = 255, masih dalam jangkauannya</p> <p>c = -1, diluar jangkauannya 1 = 0000 0001 -1 = 1111 1111 sama dengan 255</p> <p>d = 256, diluar jangkauannya 256 = 1 0000 0000, karena hanya 8 bit maka bit paling kiri diabaikan sehingga nilainya menjadi 0</p>
---	--	--

3. short

<p>x x x x x x x x x x x x x x x x</p> <p>↑ sign</p> <p>1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</p> <p>-32.768</p> <p>0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>32.767</p>	<p>bit sign :</p> <p>0 : berarti nilai positif</p> <p>1 : berarti nilai negatif</p> <p>Nilai yang dapat ditampung berkisar dari</p>  <p>-32.768 0 32.767</p>
--	---

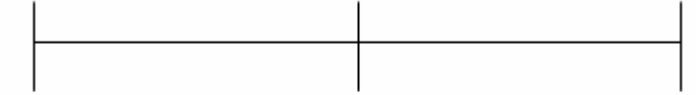
<pre>//short1.cpp #include<iostream.h> void main(void) { short a,b,c,d; a=-32768 ; b=32767; c=-32769; d=32769; cout<<a<<endl; cout<<b<<endl; cout<<c<<endl; cout<<d<<endl; }</pre>	<p>Hasilnya :</p> <p>-32768</p> <p>32767</p> <p>32767</p> <p>-32768</p>	<p>Jelaskan!!!!!!</p>
--	---	-----------------------

4. unsigned short

<div> <div>x x x x x x x x x x x x x x</div> <div>0 0 0 0 0 0 0 0 0 0 0 0 0 0</div> <div>0</div> </div> <div> <div>1 1 1 1 1 1 1 1 1 1 1 1 1 1</div> <div>65.535</div> </div>	<p>nilai yang dapat ditampung berkisar antara 0 sampai dengan 65.535</p>
---	--

<pre>//short2.cpp #include<iostream.h> void main(void) { unsigned short a,b,c,d; a=0; b=65535; c=-1; d=65536; cout<<a<<endl; cout<<b<<endl; cout<<c<<endl; cout<<d<<endl; }</pre>	<p>Hasilnya :</p> <pre>0 65535 65535 0</pre>	<p>Jelaskan!!!!!!!!!!!!!!!!!!!!</p>
---	--	-------------------------------------

5. int

<p>x x x x x x x x x x x x x x x x</p> <p>↑ sign</p> <p>1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</p> <p>-2.147.483.648</p> <p>0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>2.147.483.647</p>	<p>bit sign :</p> <p>0 : berarti nilai positif</p> <p>1 : berarti nilai negatif</p> <p>Nilai yang dapat ditampung berkisar dari</p> <p>  </p> <p>-2.147.483.648 0 2.147.483.647</p>
--	--

<pre>//int1.cpp #include<iostream.h> void main(void) { int a,b,c,d; a=-2147483648; b=2147483647; c=-2147483649; d=2147483648; cout<<a<<endl; cout<<b<<endl; cout<<c<<endl; cout<<d<<endl; }</pre>	<p>Hasilnya :</p> <p>-2.147.483.648</p> <p>2.147.483.647</p> <p>2.147.483.647</p> <p>-2.147.483.648</p>	<p>Jelaskan!!!!!!!!!!</p>
---	---	---------------------------

6. unsigned int

<div> <div>x x x x x x x x x x</div> <div>0 0 0 0 0 0 0 0 0 0 0 0</div> <div>0</div> <div>1 1 1 1 1 1 1 1 1 1 1 1</div> <div>4.294.967.295</div> </div>	<p>nilai yang dapat ditampung berkisar antara 0 sampai dengan 4.294.967.295</p>
---	---

<pre>//int2.cpp #include<iostream.h> void main(void) { unsigned int a,b,c,d; a=0; b=4294967295; c=-1; d=4294967296; cout<<a<<endl; cout<<b<<endl; cout<<c<<endl; cout<<d<<endl; }</pre>	<p>Hasilnya :</p> <pre>0 4294967295 4294967295 0</pre>	<p>Jelaskan!!!!!!!!!!!!!!</p>
---	--	-------------------------------

TUGAS

1. Buat algoritma (dalam bentuk flow chart dan pseudo-code) dan program bahasa C untuk menghitung luas lingkaran
2. Buat algoritma (dalam bentuk flow chart dan pseudo-code) dan program bahasa C untuk menghitung nilai-nilai x dari persamaan

$$ax^2 + bx + c = 0$$

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

inputnya a,b,c

3. Jika diketahui nilai A=5, B=2, T=4. Berapa isi A, B, dan T jika dikenai instruksi sbb :
a. T=A b. T=T+A c. T=T+A*B d. T=A%B e. T=B%A f. T=A-B/2
4. Jika diketahui A=5, B=2, berapa isi A dan B jika dikenai instruksi :

T=A

A=B

B=T