BAB III
TIPE DATA, VARIABEL DAN OPERASI

Tipe Data

Komputer bisa diartikan sebagai alat untuk menghitung. Untuk melakukan proses perhitungan tersebut, maka dibutuhkan data yang akan diproses. Tipe data ada beberapa jenis, yaitu

a. Tipe data karakter
b. Tipe data bilangan bulat
c. Tipe data bilangan pecahan

Jika kita membutuhkan suatu tipe data yang baru yang tidak terdapat pada tipe data standar, maka kita dapat membuatkan tipe data baru dengan menggunakan perintah struct. Perintah struct akan dijelaskan pada bab selanjutnya.

Tipe Data Karakter

Untuk tipe data karakter kita gunakan perintah char.

Contoh

```
char karakter;
char kar1,kar2,kar3;
char kar4='A';
char kar5=65;
```

Tipe data ini mempunyai jangkauan dari 0 sampai 255 atau karakter ASCII ke 0 sampai karakter ASCII 255. Tipe data karakter bisa ditampilkan sebagai suatu karakter atau sebagai suatu bilangan. Hal ini tergantung dari bagaimana penulisannya apakah dianggap sebagai karakter atau sebagai bilangan. Untuk menuliskan isi dari data bertipe char adalah dengan menggunakan printf dengan format penulisannya menggunakan tanda %c kalau ingin ditampilkan sebagai suatu karakter atau dengan %i jika ingin ditampilkan sebagai suatu angka.

Pemberian nilai kepada suatu karakter digunakan perintah sebagai berikut :

```
karakter='A';
```

Atau

```
karakter=65;
```

Kedua cara tersebut menghasilkan suatu efek yang sama yaitu memberikan nilai 65 atau karakter A ke variabel karakter. Kalau pengisian variable ingin menggunakan karakter maka karakter yang akan dimasukan harus diapit dengan tanda apostrof.

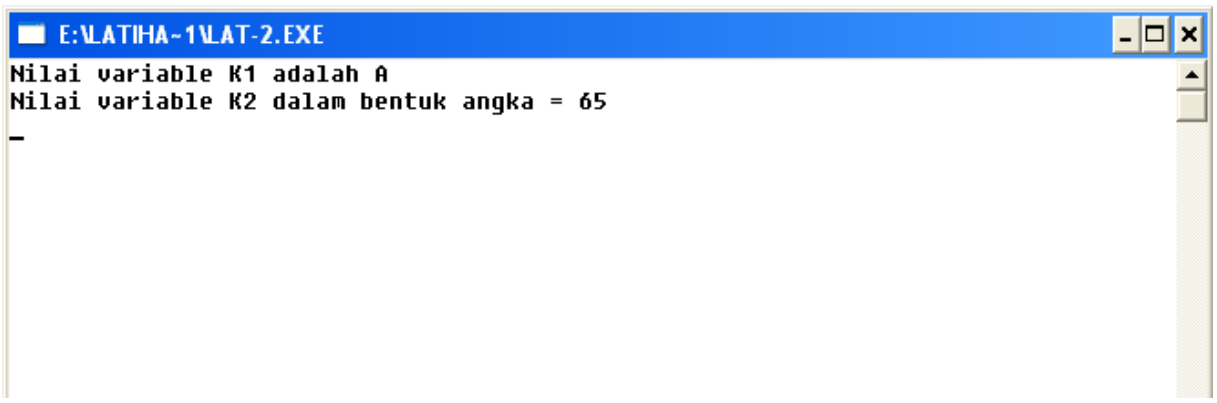
Untuk melihat nilai yang ada dalam suatu variable yang bertipe char gunakan perintah berikut :

```
printf("Karakter dilihat dalam bentuk karakter = %c.\n",karakter);
printf("Karakter dilihat dalam bentuk angka = %d.\n",karakter);
```

Contoh Program

```
//Program Ke-2 Nama File : Lat-2.CPP
#include <stdio.h>
#include <conio.h>
main()
    {
        char k1,k2;
        k1='A';
        k2=k1;
        printf("Nilai variable K1 adalah %c\n",k1);
        printf("Nilai variable K2 dalam bentuk angka = %d\n",k2);
        getch();
        return 0;
    }
```

Hasil Eksekusi dari program diatas adalah



Keterangan program Lat-2.cpp

Perintah “char k1,k2;” pada baris 6 berarti program memesan 2 buah tempat di memori untuk menyimpan data bertipe karakter dengan nama k1 dan k2.

Perintah “k1=’A’;” pada baris 7 adalah perintah untuk memasukan nilai karakter A kapital ke dalam variable k1 sehingga untuk baris berikutnya k1 berisi karakter A kapital atau angka 65.

Perintah “k2=k1;” pada baris 8 berarti bahwa nilai k2 diisi dari nilai k1 sehingga isi k2 sama dengan isi variable di k1.

Perintah printf pada baris 9 berarti perintah penulisan ke layar sesuai dengan format “Nilai variable k1 adalah %c \n”. Karakter %c tidak dicetak sebagai %c tetapi akan diganti dari variable yang sesuai dengan urutannya yaitu k1 dalam bentuk karakter. Perintah printf pada baris 10 cara kerjanya sama dengan perintah printf pada baris 9 bedanya hanya tanda %d berasal ditulis berdasarkan isi variable k2 yang dicetak dalam bentuk angka bukan karakter. Tanda %d digunakan untuk format pencetakan data dalam bentuk bilangan bulat.

Perintah getch() digunakan untuk menunggu sampai pengguna menekan sembarang karaker.

Perintah return digunakan untuk memberikan nilai kembalian dari fungsi main().

Tipe Data Bilangan Bulat

Ada beberapa tipe data standar yang digunakan untuk data bilangan bulat.

Tipe Data	Memori	Format	Jangkauan Nilai
int	2 byte	%d/%i	-32.768 s/d 32.767
unsigned int	2 byte	%u	0 s/d 65.535
char	1 byte	%d/%i	-128 s/d 127
unsigned char	1 byte	%u	0 s/d 255
unsigned long	4 byte	%lu	0 s/d 4.294.967.295
long	4 byte	%ld/%li	-2.147.483.648 s/d 2.147.483.647

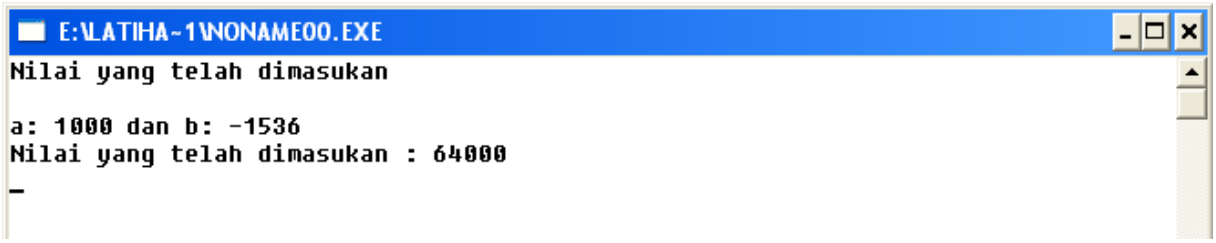
Tipe-tipe data yang ada dalam table tersebut khusus untuk data yang nilai bilangannya bulat. Cara pendeklarasian tipe data ini sama seperti pendeklarasian lainnya, yaitu :

```
int a;
unsigned int b;
char c;
long d;
```

Contoh Program :

```
#include <stdio.h>
#include <conio.h>
main()
{
    int a=1000,b=64000;
    unsigned int c=64000;
    printf("Nilai yang telah dimasukan\n \na: %i dan b: %i\n",a,b);
    printf("Nilai yang telah dimasukan : %u\n",c);
    getch();
    return 0;
}
```

Hasil Eksekusi dari program diatas adalah



Tipe Data Bilangan Pecahan

Tipe data bilangan pecahan terdiri dari beberapa jenis, yaitu :

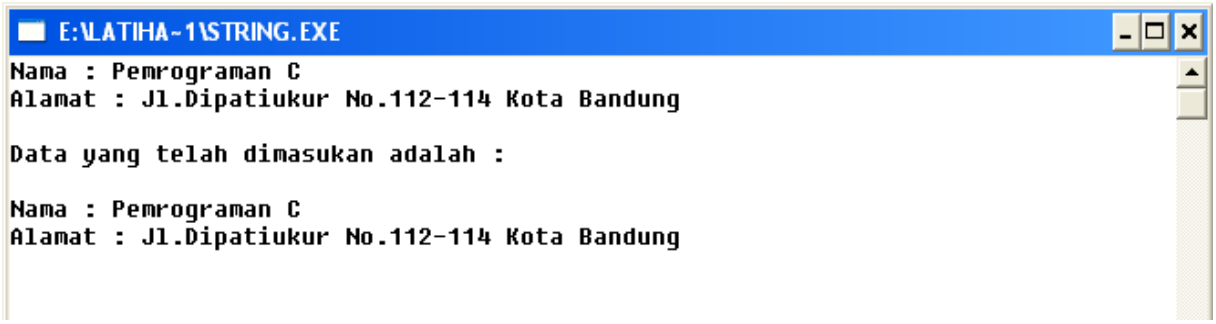
Tipe Data	Memori	Format	Jangkauan Nilai
float	4 byte	%f	3.4*(10^-38) - 3.4*(10^+38
double	8 byte	%f	1.7*(10^-308) - 1.7*(10^+308)
long double	10 byte	%lf	3.4*(10^-4932) - 1.1*(10^+4932)

Tipe Data String

Dalam pemrograman C, untuk variabel yang menampung data string tidak ada perintah khusus, karena dalam bahasa C, string adalah sebuah array karakter atau sebuah pointer ke sebuah variabel char.

Contoh Program :

```
#include <stdio.h>
#include <conio.h>
main()
{
    char nama[50]; //deklarasi dengan cara array
    char alamat[50];
    printf("Nama : ");gets(nama);
    printf("Alamat : ");gets(alamat);
    printf("\nData yang telah dimasukan adalah : \n \n");
    printf("Nama : %s\nAlamat : %s\n",nama,alamat);
    getch();
    return 0;
}
```



Note	*Pemilihan tipe data harus hati-hati. Pertimbangkan jangkauanyang dimiliki oleh tipe data yang dipilih. Kesalahan dalam memilih tipe data akan menimbulkan suatu hasil yang tidak diperkirakan.*

Contoh :

```
int a=32000;
int b=769;
int c;
c=a+b;
printf("%i + %i = %i\n",a,b,c);
```

Jika program tersebut dijalankan, maka akan menghasilkan output seperti berikut :

32000 + 769 = -32767

Hal tersebut terjadi karena jangkuan nilai c sudah melebihi jangkauan nilai untuk sebuah tipe data int. Bila suatu variable telah melebihi jangkauan nilainya maka nilai variable tersebut akan berputar menjadi nilai minimalnya dan jika nilainya kurang dari minimal jangkauan nilainya maka variable tersebut akan terisi oleh bilangan maksimal tipe tersebut.

Variabel dan Konstanta

Aturan penulisan pengenal untuk sebuah variabel, konstanta atau fungsi yang didefinisikan oleh pemrogram adalah sebagai berikut :

a. Variabel harus diawali dengan huruf (A..Z, a..z) atau karakter garis bawah (_)
b. Selanjutnya dapat berupa huruf, digit (0..9) atau karakter garis bawah atau tanda dollar ($)
c. Panjang nama variabel boleh lebih dari 31 karakter, tetapi hanya 31 karakter pertama yang akan dianggap berarti
d. Variabel tidak boleh menggunakan nama yang tergolong sebagai kata-kata cadangan (reserved words) seperti int, if, while dan sebagainya.

Deklarasi Variable yang akan digunakan dalam program haruslah dideklarasikan terlebih dahulu → pengertian deklarasi disini berarti memesan memori dan menentukan jenis data yang bisa disimpan didalamnya.

Contoh :

Int var_bulat1;

Float var_pecahan1, var_pecahan2;

Konstanta menyatakan nilai yang tetap. Penulisan konstanta mempunyai aturan tersendiri, sesuai dengan tipe masing-masing.

a. Konstanta karakter misalnya ditulis dengan diawali dan diakhiri dengan tanda petik tunggal. Contohnya : 'A' dan '@'.
b. Konstanta integer ditulis dengan angka (tanpa tanda petik) tanpa mengandung pemisah ribuan dan tak mengandung bagian pecahan. Contoh : -1 dan 32767.
c. Konstanta real (float dan double) bisa mengandung pecahan (dengan tanda berupa titik) dan nilainya bisa ditulis dalam bentuk eksponensial (menggunakan tanda e). Contoh : 27.5f (untuk tipe float) atau 27.5 (untuk tipe double) dan 2.1e+5 (maksudnya 2,1 x 10^5).
d. Konstanta string merupakan deretan karakter yang diawali dan diakhiri dengan tanda petik-ganda (“). Contoh : “Pemrograman C”.
e. Contoh :

   #define Phi 3.14

   Atau const float Phi = 3.14

Operator – Operator Perhitungan

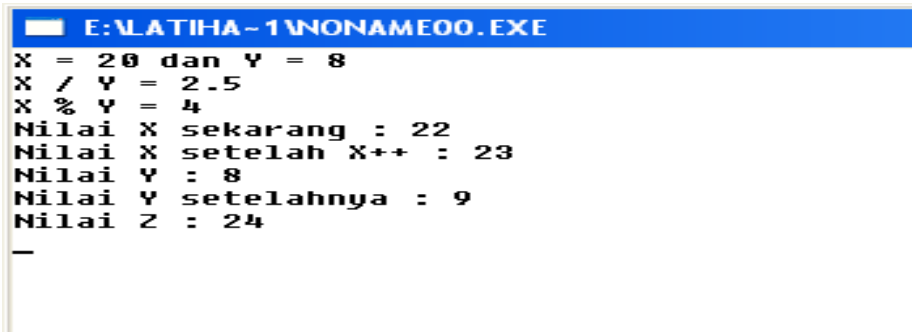
Untuk melakukan perhitungan-perhitungan data, maka diperlukan operator-operator perhitungannya. Operator-operator yang paling umum dipakai dalam pemrograman dengan bahasa C adalah :

Operator	Contoh	Arti
+	c=a+b	Variable c diisi dari isi variable a ditambah isi variable b
-	c=a-b	Variable c diisi dari isi variable a dikurangi isi variable b
*	c=a*b	Variable c diisi dari isi variable a dikali dengan isi variable b
/	c=a/b	Variable c diisi dari isi variable a dibagi oleh isi variable b
++	a++	Isi variable a ditambah 1. Perintah ini sama dengan a=a+1 atau a+=1
--	b--	Isi variable a dikurang. Perintah ini sama dengan a=a-1 atau a-=1
%	c = a % b	Variable c diisi dari sisa pembagian variable a dibagi variable b
+=	c+=a	Variable c ditambah dengan isi variable a. Sama dengan c=c+a
/=	c/=a	Variable c dibagi dengan isi variable a. Sama dengan c=c/a
-=	c-=a	Variable c dikurangi dengan isi variable a. Sama dengan c=c -a
***=**	c*=a	Variable c dikali dengan isi variable a. Sama dengan c=c*a
%=	c%=a	c%=a Variable c diisi dari sisa pembagian c dibagi isi variable a. Sama dengan c=c%a

Contoh – Contoh Program

```
/*
  Program operator
  Nama File: operator.cpp
*/
#include <stdio.h>
#include <conio.h>

main()
{
     int x=20, y=8,z;
     float w;
     clrscr();
     printf("X = %d dan Y = %d\n",x,y);
     w=float(x)/y;
     printf("X / Y = %3.1f\n",w);
     printf("X %% Y = %d\n",x%y);
     x+=2;
     printf("Nilai X sekarang : %i\n",x);
     x++;
     printf("Nilai X setelah X++ : %i\n",x);
     printf("Nilai Y : %d\n",y++);
     printf("Nilai Y setelahnya : %d\n",y);
     z=++x;
     printf("Nilai Z : %d\n",z);
     getch();
     return 0;
}
```



Contoh Program :

```
/* File program : data1.c */
#include <stdio.h>
#include <conio.h>

main()
  {
     int var_bulat = 32767;
     float var_pecahan1 = 339.2345678;
     double var_pecahan2 = 3.4567e+40;
     char var_karakter = 'S';
     printf("Variabel bulat = %d\n", var_bulat);
     printf("Variabel pecahan1 = %f\n", var_pecahan1);
     printf("Variabel pecahan2 = %e\n", var_pecahan2);
     printf("Variabel karakter = %c\n", var_karakter);
     return 0;
     getch();
  }
```

```
/* File program : operator.c */
#include <stdio.h>
main()
  {
     int a, b, c, hasil;
     printf("Masukkan nilai a = "); scanf("%d", &a);
     printf("Masukkan nilai b = "); scanf("%d", &b);
     printf("Masukkan nilai c = "); scanf("%d", &c);
     printf("\n");
     hasil = a - b;
     printf("Hasil pengurangan : a - b = %d\n", hasil);
     printf("Hasil perkalian : b * c = %d\n", b * c);
     hasil = a / c;
     printf("Hasil pembagian : a / c = %d\n", hasil);
     printf("Hasil operasi : a + b * c = %d\n", a + b * c);
     return 0;
  }
```

```
/* File program : modulus.c */
#include <stdio.h>
main()
  {
     int a = 14, b = 2, c = 3, d = 4;
     printf("a = %d, b = %d, c = %d, d = %d\n\n", a, b, c, d);
```

```
      printf("Hasil a %% b = %d\n", a % b);
      printf("Hasil a %% c = %d\n", a % c);
      printf("Hasil a %% d = %d\n", a % d);
      printf("Hasil a / d * d + a %% d = %d\n",a / d * d + a % d);
      return 0;
   }
```

```
/* File program : diskrim.c */
# include <stdio.h>
main()
   {
      float d = 0;
      int a,b,c;
      a = 3;
      b = 4;
      c = 7;
      d = (b*b)-(4*a*c);
      printf("Diskriminan = %f\n",d);
   }
```

TUGAS !

1. Print out tampilan program dari keempat source code diatas !
2. Buat program untuk menghitung luas dan keliling lingkaran, dengan panjang jari-jari yang diinputkan dari keyboard.
3. Buatlah program untuk mengkonversi suhu dari Celcius ke Fahrenheit dengan rumus

   F = C * 1.8+ 32

   Input : Suhu dalam celcius
   Output : Suhu dalam Fahrenheit

4. Apa hasil dari eksekusi program berikut :

```
#include <stdio.h>
main ()
   {
         char kar = 'A';
         kar = kar+32;
         printf("%c\n", kar);
   }
```

   Untuk nomor 4 silahkan anda jelaskan mengapa demikian..

5. Buat program untuk menghitung Luas dan Keliling Bujur Sangkar dengan panjang sisi diinputkan dari keyboard.

Note 	Jawaban di buat berupa source code dan print out tampilan programnya, beserta kesimpulan hasil praktikum.. Kirim jawaban anda (*.doc) via e-mail ke `ekobudisetiawan@ymail.com` Subject : Nama-NIM-Kelas-TC2 Deadline : Jum'at 2 Oktober 2009