



**FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA**

LAB SHEET PRAKTIK BAHASA PEMROGRAMAN

SEM. 3

*FUNCTION: PENGGOLONGAN
VARIABEL DAN PROGRAM
TERSTUKTUR*

JOB 9

4x50"

No. EKA/EKA255

Revisi:01

30 Agu 2014

Hal 1 / 6

A. TUJUAN

Setelah melakukan praktik mahasiswa diharapkan:

1. Mengetahui perbedaan antara variabel lokal, eksternal, statis dan register
2. Memahami komunikasi antar sejumlah fungsi
3. Mengenal konsep pemrograman terstruktur

B. TEORI DASAR

Penggolongan Variabel berdasarkan Kelas Penyimpanan

Suatu variabel, di samping dapat digolongkan berdasarkan jenis/tipe data juga dapat diklasifikasikan berdasarkan kelas penyimpanan (*storage class*). Penggolongan berdasarkan kelas penyimpanan berupa :

- a. variabel lokal
- b. variabel eksternal
- c. variabel statis
- d. variabel register

Variabel Lokal

Variabel lokal adalah variabel yang dideklarasikan dalam fungsi, dengan sifat : secara otomatis diciptakan ketika fungsi dipanggil dan akan lenyap ketika eksekusi terhadap fungsi berakhir. Hanya dikenal oleh fungsi tempat variabel tersebut dideklarasikan tidak ada inisialisasi secara otomatis (saat variabel diciptakan, nilainya tak menentu).

```
void fung_x(void)
{
    int x;
    .
    .
    .
}
```

↑
x adalah variabel lokal bagi
fungsi fung_x()

Pada **fung_x()**, deklarasi

```
int x;
```

dapat ditulis menjadi

```
auto int x;
```

Penerapan variabel lokal yaitu bila variabel hanya dipakai oleh suatu fungsi (tidak

Dibuat oleh :
Muslikhin, M.Pd.

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin
tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh :



**FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA**

LAB SHEET PRAKTIK BAHASA PEMROGRAMAN

SEM. 3

*FUNCTION: PENGGOLONGAN
VARIABEL DAN PROGRAM
TERSTUKTUR*

JOB 9

4x50"

No. EKA/EKA255

Revisi:01

30 Agu 2014

Hal 2 / 6

dimaksudkan untuk dipakai oleh fungsi yang lain). Pada contoh berikut, antara variabel **i** dalam fungsi **main()** dan **fung_1()** tidak ada kaitannya, sebab masing-masing merupakan variabel lokal.

Variabel Eksternal

Variabel eksternal merupakan variabel yang dideklarasikan di luar fungsi, dengan sifat: dapat diakses oleh semua fungsi kalau tak diberi nilai, secara otomatis diinisialisasi dengan nilai sama dengan nol. Variabel eksternal haruslah dideklarasikan sebelum definisi fungsi yang akan mempergunakannya. Untuk memperjelas bahwa suatu variabel dalam fungsi merupakan variabel eksternal, di dalam fungsi yang menggunakannya dapat mendeklarasikan variabel itu kembali dengan menambahkan kata kunci *extern* di depan tipe data variabel. Kalau dalam suatu program terdapat suatu variabel eksternal, suatu fungsi bisa saja menggunakan nama variabel yang sama dengan variabel eksternal, namun diperlakukan sebagai variabel lokal.

Variabel Statis

Variabel statis dapat berupa variabel internal (didefinisikan di dalam fungsi) maupun variabel eksternal. Sifat variabel ini: Kalau variabel statis bersifat internal, maka variabel hanya dikenal oleh fungsi tempat variabel dideklarasikan. Kalau variabel statis bersifat eksternal, maka variabel dapat dipergunakan oleh semua fungsi yang terletak pada file yang sama, tempat variabel statis dideklarasikan. Berbeda dengan variabel lokal, variabel statis tidak akan hilang sekeluanya dari fungsi (nilai pada variabel akan tetap diingat). Inisialisasi akan dilakukan hanya sekali, yaitu saat fungsi dipanggil yang pertama kali. Kalau tak ada inisialisasi oleh pemrogram secara otomatis akan diberi nilai awal nol. Variabel statis diperoleh dengan menambahkan kata kunci *static* di depan tipe data variabel.

Variabel Register

Variabel register adalah variabel yang nilainya disimpan dalam register dan bukan dalam memori RAM. Variabel yang seperti ini hanya bisa diterapkan pada variabel yang lokal atau parameter formal, yang bertipe char atau int. Variabel register biasa diterapkan pada variabel yang digunakan sebagai pengendali loop. Tujuannya untuk mempercepat proses dalam loop. Sebab variabel yang dioperasikan pada register memiliki kecepatan yang jauh lebih tinggi daripada variabel yang diletakkan pada RAM.

Dibuat oleh :
Muslikhin, M.Pd.

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin
tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh :



**FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA**

LAB SHEET PRAKTIK BAHASA PEMROGRAMAN

SEM. 3

*FUNCTION: PENGGOLONGAN
VARIABEL DAN PROGRAM
TERSTUKTUR*

JOB 9

4x50"

No. EKA/EKA255

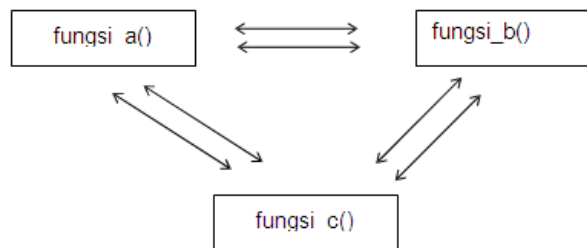
Revisi:01

30 Agu 2014

Hal 3 / 6

Menciptakan Sejumlah Fungsi

Pada C, semua fungsi bersifat sederajat. Suatu fungsi tidak dapat didefinisikan di dalam fungsi yang lain. Akan tetapi suatu fungsi diperbolehkan memanggil fungsi yang lain, dan tidak tergantung kepada peletakan definisi fungsi pada program. Komunikasi antara fungsi dalam C ditunjukkan di bawah. Gambar tersebut menjelaskan kalau suatu fungsi katakanlah **fungsi_a()** memanggil **fungsi_b()**, maka bisa saja **fungsi_b()** memanggil **fungsi_a()**. Contoh program yang melibatkan fungsi yang memanggil fungsi yang lain ada pada program **kom_fung.c**, yaitu **fungsi_1()** dipanggil dalam **main()**, sedangkan **fungsi_2()** dipanggil oleh **fungsi_1()**.



Pengenalan Konsep Pemrograman Terstruktur

Fungsi sangat bermanfaat untuk membuat program yang terstruktur. Suatu program yang terstruktur dikembangkan dengan menggunakan "*top-down design*" (rancang atas bawah). Pada C suatu program disusun dari sejumlah fungsi dengan tugas tertentu. Selanjutnya masing masing fungsi dipecah-pecah lagi menjadi fungsi yang lebih kecil. Pembuatan program dengan cara ini akan memudahkan dalam pencarian kesalahan ataupun dalam hal pengembangan dan tentu saja mudah dipahami/ dipelajari. Dalam bentuk diagram, model suatu program C yang terstruktur adalah seperti yang tertera pada bagan berikut ini. Namun sekali lagi perlu diketahui, bahwa pada C semua fungsi sebenarnya berkedudukan sederajat.

Fungsi **main()** terdiri dari **fungsi_a()** sampai dengan **fungsi_n()**, menegaskan bahwa dalam program fungsi **main()** akan memanggil **fungsi_a()** sampai dengan **fungsi_n()**. Adapun fungsi-fungsi yang dipanggil oleh fungsi **main()** juga bisa memanggil fungsi-fungsi yang lain.

Dibuat oleh :
Muslikhin, M.Pd.

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin
tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh :



**FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA**

LAB SHEET PRAKTIK BAHASA PEMROGRAMAN

SEM. 3

*FUNCTION: PENGGOLONGAN
VARIABEL DAN PROGRAM
TERSTUKTUR*

JOB 9

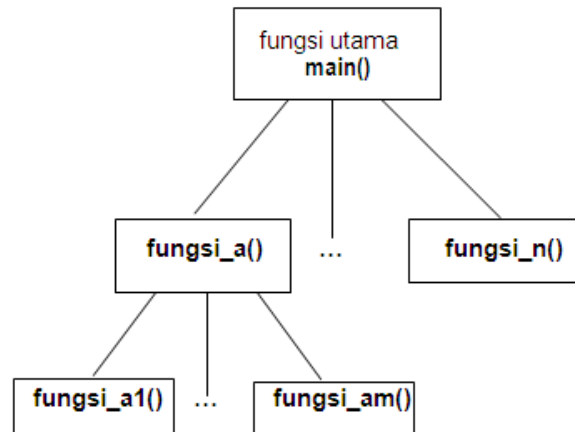
4x50"

No. EKA/EKA255

Revisi:01

30 Agu 2014

Hal 4 / 6



C. LANGKAH KERJA DAN PERCOBAAN

1. Percobaan 1

Ketik dan kompilasi program berikut;

```
int OddEvenTest(int);
```

```
main()
{
    int a, hasil;
    a = 5;
    hasil = OddEvenTest(a);
    printf("a=%d; hasil=%d\n", a, hasil);
}
```

```
OddEvenTest(int b)
{
    int a;
    a = b % 2;
    return a;
}
```

Dibuat oleh :
Muslikhin, M.Pd.

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin
tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh :



**FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA**

LAB SHEET PRAKTIK BAHASA PEMROGRAMAN

SEM. 3

*FUNCTION: PENGGOLONGAN
VARIABEL DAN PROGRAM
TERSTUKTUR*

JOB 9

4x50"

No. EKA/EKA255

Revisi:01

30 Agu 2014

Hal 5 / 6

2. Percobaan 2

Ketik dan kompilasi program berikut;

```
void demo(void);
main()
{
    int i=0;
    while(i < 3) {
        demo(); i++;
    }
}

void demo(void)
{
    auto int var_auto = 0;
    static int var_static = 0;
    printf("auto=%d,static=%d\n",var_auto,var_static);
    ++var_auto;
    ++var_static;
}
```

3. Percobaan 3

Ketik dan kompilasi program berikut;

```
void fung_a(void);
void fung_b(void);
int x = 20;

main()
{
    x += 2; fung_a();
    fung_a();
    printf("\nNilai x dalam main() = %d\n\n",x);
}

void fung_a(void)
{
    static x = 5;
    x++;
    printf("Nilai x dalam fung_a() = %d\n", x);
}
```

Dibuat oleh :
Muslikhin, M.Pd.

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin
tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh :



**FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA**

LAB SHEET PRAKTIK BAHASA PEMROGRAMAN

SEM. 3

*FUNCTION: PENGGOLONGAN
VARIABEL DAN PROGRAM
TERSTUKTUR*

JOB 9

4x50"

No. EKA/EKA255

Revisi:01

30 Agu 2014

Hal 6 / 6

```
    fung_b();  
}  
  
void fung_b(void)  
{  
    x--;  
    printf("Nilai x dalam fung_b() = %d\n", x);  
}
```

D. PENGEMBANGAN DAN TUGAS

Definisikanlah fungsi-fungsi sebagai berikut :

- a. Fungsi `f_to_i()` untuk mengubah ukuran dari satuan kaki (*feet*) ke inci
- b. Fungsi `i_to_cm()` untuk mengubah ukuran dari satuan inci ke centimeter
- c. Fungsi `c_to_m()` untuk mengubah ukuran dari satuan centimeter ke meter

Dalam `main()` mintalah masukan ukuran dalam satuan kaki (*feet*) kemudian lakukan konversi sampai mendapatkan keluaran berupa ukuran dalam meter. Tentukan jumlah dan tipe parameter dan return value yang dibutuhkan

Keterangan : 1 kaki = 12 inchi, 1 inchi = 2.54 cm, 100 cm = 1 meter

Dibuat oleh :
Muslikhin, M.Pd.

Dilarang memperbanyak sebagian atau seluruh isi dokumen tanpa ijin
tertulis dari Fakultas Teknik Universitas Negeri Yogyakarta

Diperiksa oleh :