



**FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA**

**LAB SHEET PRAKTIK BAHASA PEMROGRAMAN**

SEM. 3

*FUNCTION: PASSING  
PARAMETER BY VALUE &  
REFERENCE*

JOB 8

4x50"

No. EKA/EKA255

Revisi:01

30 Agu 2014

Hal 1 / 5

### A. TUJUAN

Setelah melakukan praktik mahasiswa diharapkan:

1. Memahami perbedaan pengiriman parameter secara nilai dan secara alamat.
2. Memecah program dalam fungsi fungsi yang sederhana.
3. Menjelaskan tentang pemrograman terstruktur.

### B. TEORI DASAR

Pemanggilan dengan nilai merupakan cara yang dipakai untuk seluruh fungsi buatan yang telah dibahas pada praktikum sebelumnya. Pada pemanggilan dengan nilai, nilai dari parameter aktual akan disalin ke parameter formal. Dengan cara ini nilai parameter aktual tidak bisa dirubah sekalipun nilai parameter formal berubah. Untuk lebih jelasnya lihat pada fungsi **tukar()** pada contoh berikut ini.

```
/* File program : tukarl.c
Untuk melihat pengaruh pemanggilan nilai pada fungsi untuk
penukaran dua data */
#include <stdio.h>
void tukar (int, int);
main()
{
    int a = 88, b = 77;
    printf("Nilai sebelum pemanggilan fungsi\n");
    printf("a = %d b = %d\n", a, b);
    tukar(a,b);printf("\nNilai setelah pemanggilan fungsi\n");
    printf("a = %d    b = %d\n", a, b);
}

void tukar(int x, int y)
{
    int z;
    z = x; x
    = y; y =
    z;
    printf("\nNilai di akhir fungsi tukar()\n");
    printf("x = %d    y = %d\n", x, y);
}
```

Tampak bahwa sekeduanya dari pemanggilan fungsi **tukar()**, variabel **a** dan **b** (yang dilewatkan ke fungsi **tukar()** tidak berubah, walaupun pada fungsi **tukar()** telah terjadi

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: PASSING  
PARAMETER BY VALUE &  
REFERENCE*

JOB 8

4x50"

No. EKA/EKA255

Revisi:01

30 Agu 2014

Hal 2 / 5

penukaran antara parameter **x** dan **y**. Mengapa hal ini bisa terjadi? Sebab **x** hanyalah salinan dari **a** dan **y** adalah salinan dari **b** (Lihat gambar 1). Pada saat pemanggilan fungsi, maka :

**x** bernilai 88 (nilai **a**)

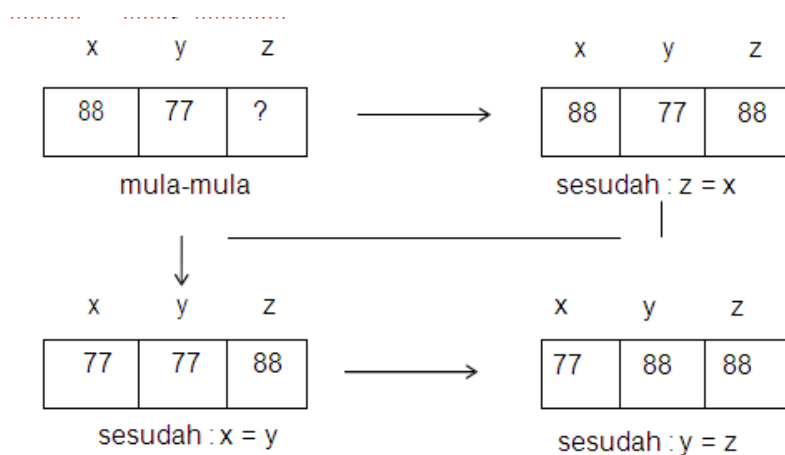
**y** bernilai 77 (nilai **b**)

Sesudah pernyataan-pernyataan berikut dijalankan, maka :

`z = x; x =`

`y; y = z;`

**x** akan bernilai 77 dan **y** bernilai 88.



Gambar 1. Proses penukaran nilai

Gambar tersebut menjelaskan bahwa **a** dan **b** tidak berubah. Yang berubah hanyalah parameter **x** dan **y**.

Pemanggilan dengan referensi (*call by reference*) merupakan upaya untuk melewatkan alamat dari suatu variabel ke dalam fungsi. Cara ini dapat dipakai untuk mengubah isi suatu variabel di luar fungsi dengan pelaksanaan perubahan dilakukan di dalam fungsi. Sebagai contoh perhatikan program **tukar2.c** yang merupakan modifikasi dari **tukar1.c**. Perubahan yang pertama terletak dalam definisi fungsi, yang kini berupa

```
void tukar(int *px, int *py)
{
    int z;
    z = *px;
    *px = *py;
```

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: PASSING  
PARAMETER BY VALUE &  
REFERENCE*

JOB 8

4x50"

No. EKA/EKA255

Revisi:01

30 Agu 2014

Hal 3 / 5

```
*py = z;  
printf("\nNilai di akhir fungsi tukar()\n");  
printf("x = %d    y = %d\n", *px, *py);  
}
```

Adapun perubahan dalam parameter aktualnya menjadi :

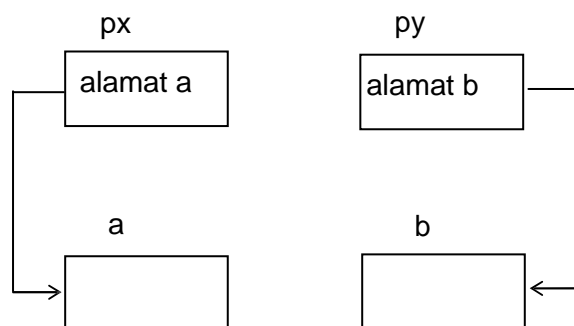
```
tukar(&a, &b);           //alamat a dan alamat b
```

dalam deklarasi parameter

```
int *px, int *py
```

menyatakan bahwa **px** dan **py** adalah suatu variabel pointer. Yang dimaksudkan sebagai variabel pointer adalah suatu variabel yang menunjuk ke variabel lain. Lebih jelasnya, variabel pointer berisi alamat dari variabel lain.

Adapun pada pemanggilan fungsi, **&a** dan **&b** masing-masing berarti "alamat a" dan "alamat b". Dengan pemanggilan seperti ini, hubungan antara variabel pointer **px** dan **py** dengan variabel **a** dan **b** adalah seperti ditunjukkan pada gambar di bawah ini. Dalam hal ini, **px** dikatakan menunjuk variabel **a** dan **py** menunjuk variabel **b**. Ilustrasi penempatan di memory untuk masing-masing variabel ditunjukkan pada gambar 15.2.



Gambar 2. Variabel pointer **px** menunjuk variabel **a** dan variabel pointer **py** menunjuk variabel **b**

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: PASSING  
PARAMETER BY VALUE &  
REFERENCE*

JOB 8

4x50"

No. EKA/EKA255

Revisi:01

30 Agu 2014

Hal 4 / 5

**C. LANGKAH KERJA DAN PERCOBAAN**

**1. Percobaan 1**

Ketik dan kompilasi program berikut;

```
/* File program : tukar2.c
```

```
Untuk melihat pengaruh pemanggilan nilai pada fungsi  
untuk penukaran dua data */
```

```
#include <stdio.h>
void tukar (int *px, int *py); //prototype fungsi
main()
{
    int a= 88,b = 77;
    printf("Nilai sebelum pemanggilan fungsi\n");
    printf("a = %d    b = %d\n", a, b);
    tukar(&a,&b);           //alamat a dan alamat b
    printf("\nNilai setelah pemanggilan fungsi\n");
    printf("a = %d    b = %d\n", a, b);
}

void tukar(int *px, int *py)
{
    int z;

    z = *px;
    *px = *py;
    *py = z;
    printf("\nNilai di akhir fungsi tukar()\n");
    printf("x = %d    y = %d\n", *px, *py);
}
```

Setelah **px** menunjuk **a** dan **py** menunjuk **b**, proses penukaran isi **a** dan **b** dilakukan dengan cara sebagai berikut :

```
z = *px;           /* 1 */
*px = *py;        /* 2 */
*py = z;          /* 3 */
```

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: PASSING  
PARAMETER BY VALUE &  
REFERENCE*

JOB 8

4x50"

No. EKA/EKA255

Revisi:01

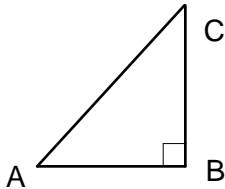
30 Agu 2014

Hal 5 / 5

Pertama variabel **z** diisi dengan nilai yang ditunjuk oleh **px**. Kedua, yang ditunjuk oleh **px** diisi dengan yang ditunjuk oleh **py** (berarti **a** diisi dengan **b**). Ketiga, yang ditunjuk oleh **py** diberi nilai **z**. Dengan melalui tiga pernyataan di atas, nilai **a** dan **b** dapat diubah di dalam fungsi.

**D. PENGEMBANGAN DAN TUGAS**

1. Definisikanlah function untuk menentukan bilangan terbesar dari 2 bilangan yang diinputkan di main(). Function mempunyai parameter berupa 2 buah bilangan yang akan dibandingkan dan memberikan return value berupa bilangan yang terbesar. Sertakan pula prototype function tsb.
2. Perhatikan gambar segitiga ABC di samping ini, dengan sudut ABC adalah siku-siku. Buatlah sebuah program yang menerima input berupa : besar sudut BAC (dalam besaran derajat), dan panjang sisi AB (dalam meter). Program tersebut bias menampilkan output berupa panjang sisi BC (dalam meter).



\*\*\*\*\*

Dibuat oleh :  
Muslikhin, M.Pd.

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

Diperiksa oleh :