



Program Studi Teknik Elektro ITB

Nama Kuliah (Kode) : Praktikum Pemecahan Masalah dengan C (EL2208)
Tahun / Semester : 2023-2024 / Genap
Modul : 2 – Pointers and Functions
Nama Asisten / NIM : _____
Nama Praktikan / NIM : Pradigta Hisyam Ramadhan / 18322008

Tugas Pendahuluan

1. Definisi *pointers*:

Pointer adalah salah satu jenis variabel yang tersedia pada bahasa pemrograman C dan digunakan untuk menyimpan **alamat memori** dari variabel lain **sebagai nilai variabel pointer** itu sendiri. Untuk menggunakan pointer pada bahasa C, digunakan simbol '*' sebelum variabel.

Contoh penggunaan pointer:

```
#include <stdio.h>

int main(void)
{
    int age = 21;
    int *pAge = &age;    // Alamat dari var. age disimpan
    *pAge = 3;           // isi dari alamat yang sudah disimpan akan diganti

    printf("Address of age      : %p\n", &age);
    printf("Value of age       : %d\n", age);

    printf("Value of pAge      : %p\n", pAge);
    printf("Value at stored address : %d\n", *pAge);

    printf("Size of age        : %d bytes\n", sizeof(age));
    printf("Size of pAge       : %d bytes\n", sizeof(pAge));

    return 0;
}
```

Output yang dihasilkan:

```
Address of age      : 000000f3d43ffb24
Value of age       : 3
Value of pAge      : 000000f3d43ffb24
Value at stored address : 3
Size of age        : 4 bytes
Size of pAge       : 8 bytes
```

2. *Pointers*: Menyimpan alamat dari variabel lain sebagai nilai yang di-assign pada pointer. Apabila kita mendeklarasikan suatu nilai pada sebuah pointer, dapat menyebabkan error karena kita akan mengisi nilai pointer dengan nilai sembarang sebagai alamatnya, dan kita tidak tahu apa isi dari alamat tersebut.

Variabel: Menyimpan nilai aktual yang dideklarasikan oleh pengguna dengan menyediakan alamat memori pada komputer.

3. Keuntungan fungsi dengan parameter pointer

Apabila dalam fungsi yang kita deklarasikan **tidak menggunakan pointer** sebagai argumen fungsi, maka perubahan nilai di dalam fungsi hanya akan berdampak kepada **variabel lokalnya** saja. Sehingga ketika kita memanggil fungsi di program utama kita, pemrosesan fungsi tidak akan berlangsung.

Berbeda apabila menggunakan **pointer sebagai argumen fungsi** dibandingkan menggunakan variabel, pengguna akan **mem-passing alamat memori ke argumen**, bukan nilainya, pada bagian fungsi utama program. Sebagai hasilnya, semua **perubahan** pada fungsi dengan argumen pointer akan **disimpan ke alamat memori** yang dipassing oleh variabel. Dalam bahasa C, hal ini disebut dengan *reference*.

Soal Pemrograman

/*EL2208 Praktikum Pemecahan Masalah dengan C 2023/2024

*TP Modul : 2

*Hari dan Tanggal : Selasa, 5 Maret 2024

*Nama (NIM) : Pradigta Hisyam Ramadhan (18322008)

*Nama File : TP-m02.c

*Deskripsi : Program digunakan untuk menampilkan hasil perkalian array integers yang dimasukkan oleh pengguna

*/

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

```
int multiply(int arr[5], int *result){  
    // Deklarasi nilai awal result  
    *result = arr[0];  
    for(int i = 0; i < 5; i++){  
        *result = *result * arr[i];  
    }  
    return *result;  
}
```

```
int main(){  
    int Array[5];  
    int result;  
    //input elemen array  
    printf("#Input\n");  
    for (int i = 0; i < 5; i++){  
        scanf("%d", &Array[i]);  
    }  
  
    //Output perkalian elemen array  
    printf("\n#Output\n");  
    result = multiply(Array, &result);  
    printf("%d", result);  
  
    return 0;  
}
```

Output yang dihasilkan

```
PS D:\(A) College Materials - ITB 22\'(A) Biomedical Engineering\Semester 4\Pemecahan Masalah dengan C\Praktikum\Modul 2> cd "d:\(A) College Materials - ITB 22\'(A) Biomedical Engineering\Semester 4\Pemecahan Masalah dengan C\Praktikum\Modul 2\" ; if ($?) { gcc TP-m02.c -o TP-m02 } ; if ($?) { .\TP-m02 }
#Input
1
2
3
4
5

#Output
120
PS D:\(A) College Materials - ITB 22\'(A) Biomedical Engineering\Semester 4\Pemecahan Masalah dengan C\Praktikum\Modul 2> 
```

Referensi

C Pointers. (n.d.). Retrieved March 5, 2024, from https://www.w3schools.com/c/c_pointers.php

G. (2022, November 5). *Passing Pointers to Functions in C*. GeeksforGeeks. Retrieved March 5, 2024, from <https://www.geeksforgeeks.org/passing-pointers-to-functions-in-c/>.