

MODUL PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 1

# Prosedur dan Penggunaan Tiga File



TIM ASISTEN PEMROGRAMAN  
ANGKATAN 10  
ILMU KOMPUTER FPMIPA UPI

## Ucapan Terima kasih

Penulis mengucapkan terima kasih atas bantuan yang diberikan oleh berbagai pihak yang telah membantu dalam pembuatan modul ini, baik secara langsung maupun tidak langsung.

## Pendahuluan

Prosedur dan penggunaan 3 file sangat bermanfaat sekali dalam efisiensi waktu serta efisiensi dalam penulisan kode. Terlebih lagi apabila terjadi error atau bug maka akan lebih mudah mencarinya.

## Prosedur

Prosedur merupakan seperangkat instruksi yang diproses secara berurutan. Penulisan prosedur **terpisah** dari program utama (main) dan dapat di eksekusi **berulang kali**.

Berikut adalah manfaat dari menggunakan prosedur:

1. Mempersingkat penulisan kode program
2. Memudahkan kita untuk membaca kode program

## Struktur Penulisan Prosedur

```
void [nama prosedur]([tipe data 1][nama variabel 1], [tipe data 2][nama variabel 2], ...){  
    //proses atau kode yang akan di eksekusi  
}
```

Void	Merupakan tanda bahwa perintah yang akan dijalankan merupakan sebuah prosedur.
Nama prosedur	Nama dari suatu prosedur yang dapat di eksekusi dan di panggil. Catatan: <ol style="list-style-type: none"><li>1. Nama tidak boleh diawali dengan angka.</li><li>2. Tidak boleh menggunakan karakter spasi.</li><li>3. Case sensitive.</li></ol>
Tipe data	Tipe data dari variabel yang akan diproses di dalam prosedur. Contoh: integer, float, char, string, struct, dll.

Nama variabel	Nama variabel dan nilai yang akan diproses di dalam prosedur.
---------------	---

## Struktur Pemanggilan Prosedur

```
nama_prosedur (nama_variabel_1, nama_variabel2,...);
```

Nama prosedur	Nama pemanggilan dan nama prosedur yang di deklarasikan harus sesuai.
Nama variabel	Nama variabel atau biasa disebut parameter yang valuenya akan di proses ke dalam prosedur.

Catatan:

1. Nama variabel pada prosedur dan nama variabel saat pemanggilan tidak harus sama.

```
void charProsedur(char huruf){
}
```

```
int main(){
    char input;
    input='C';

    charProsedur(input);
}
```

2. Apabila nama variabel prosedur dan nama variabel saat pemanggilan ada kesamaan, maka perubahan yang terjadi pada variabel prosedur tidak akan mengubah nilai variabel di program utama.
3. Variabel dengan array tidak perlu dituliskan dengan alokasinya, cukup nama variabelnya saja. **Kecuali** prosedur memang dilakukan di dalam perulangan.

<pre> void ContohProsedurI(int deret[]){     //statment }  int main(){     int deret[3]={3,5,40};      ContohProsedurI(deret); } </pre>	<pre> void ContohProsedurII(int deret){     //statment }  int main(){     int i;     int deret[3]={3,5,40};      for(i=0;i&lt;3;i++){         ContohProsedurII(deret[i]);     } } </pre>
---	--

## Penulisan Prosedur dalam 1 File

```

1  #include <stdio.h>
2
3  int min, max=0; //variabel global
4
5  void getmin(int banyak,int angka[]){//prosedur yang berfungsi untuk mendapatkan nilai minimum
6      min = angka[0];
7      int i;
8      for(i=0;i<banyak;i++){
9          if(min>angka[i]){
10             min=angka[i];
11         }
12     }
13 }
14
15 void getmax(int angka){//prosedur yang berfungsi untuk mendapatkan nilai maksimum
16     int tampung= angka;
17     if(tampung>max){
18         max = tampung;
19     }
20 }
21
22 void cetakjumlah(){//prosedur yang berfungsi untuk mencetak jumlah nilai min dan max
23     printf("min: %d max: %d\n", min, max);
24     printf("hasil pertambahan min dan max = %d\n", min+max);
25 }
26

```

```

27  int main(){
28      int n;
29      int i;
30      scanf("%d", &n); //mengisi banyak jumlah data yang akan di input
31      int deret[n];
32      for(i=0;i<n;i++){ //proses penginputan
33          scanf("%d", &deret[i]);
34      }
35
36      getmin(n,deret); //proses pemanggilan prosedur getmin
37      for(i=0;i<n;i++){ //proses perulangan untuk mengakses nilai deret
38          getmax(deret[i]); //proses pemanggilan prosedur getmax
39      }
40      cetakjumlah(); //proses pemanggilan prosedur cetakjumlah
41      return 0;
42  }

```

### Penggunaan 3 File

3 file tersebut terdiri dari:

1. File **Header**, dengan tipe data file .h berfungsi sebagai tempat deklarasi library dasar bahasa c yang digunakan (stdio.h, string.h, malloc.h, dll), variabel global, typedef, serta daftar nama prosedur.
2. File **Mesin**, dengan tipe data file .c berfungsi sebagai tempat untuk menuliskan kode mesin program yang terdiri dari prosedur, dan fungsi.
3. File **main**, dengan tipe data file .c berfungsi sebagai tempat untuk menuliskan kode utama program.

Setiap file dengan tipe data file .c harus melakukan proses pemanggilan header.

**Contoh:**

**Header.h**

```

C header.h > ...
1  #include <stdio.h>
2
3  //deklarasi variabel global
4  int total;
5
6  void penjumlahan(int angka);
7

```

### Mesin.c

```
C mesin.c > ...
1  #include "header.h"
2
3  void penjumlahan(int angka){
4      total = total + angka;
5  }
6
```

### Main.c

```
1  #include "header.h"
2
3
4  int main(){
5      int n;
6      scanf("%d", &n);
7      int deret[n];
8      total = 0;
9      int i;
10     for(i=0;i<n;i++){
11         scanf("%d", &deret[i]);
12     }
13     for(i=0;i<n;i++){
14         penjumlahan(deret[i]);
15     }
16
17     printf("Jumlah Semua Angka : %d\n", total);
18
19     return 0;
20 }
```

### Cara Kompilasi 3 File

```
gcc *.c -o namaoutput.exe
```

Catatan: ketiga file harus dalam satu folder

## Pass By Reference dan Pass by Value

Buka terlebih dahulu <https://bit.ly/2DFgSX8>

**Pass by value**, mengirimkan parameter berdasarkan nilai variabel asalnya yang akan dihubungkan terhadap parameter fungsi pemanggil.

**Pass by reference**, mengirimkan parameter berdasarkan alamat dari nilai tertentu, maka dari itu bila ada nilai yang dirumah dari alamat asalnya maka akan terjadi juga terhadap nilai parameter yang di panggil.

## Latihan

1. Buatlah sebuah program yang mana inputan berupa nama dan genre game (moba, fps, Battle Royale (BR)) favorit sebanyak n buah, lalu pisahkan di antara kelompok game tersebut.

### Contoh:

Input:

5

Fachri moba

Fauzan fps

Barbar racing

Adit moba

Argi fps

Output:

===== MOBA =====

1. Fachri

2. Adit

===== FPS =====

1. Fauzan

2. Argi

===== RACING =====

1. Barbar

===== Battle Royale =====

Tidak ada data.

## DAFTAR PUSTAKA

Tim Asisten Pemrograman Angkatan 9. (2018). Array of String. Modul Praktikum Algoritma dan Pemrograman 1. Bandung, Jawa Barat, Indonesia