

Prosedur dan Fungsi

Bahasa C

Baharuddin Aziz

Semester I 2019/2020

Tujuan Perkuliahan

- Mengetahui konsep prosedur dan fungsi
- Memahami penerapan prosedur dan fungsi pada bahasa C

Pendahuluan

Pemrograman Modular

- **Pemrograman modular** (*modular programming*): teknik pemecahan program menjadi sejumlah upa-program (*subprogram*)
- Beberapa bahasa pemrograman menamakannya:
 - sub-rutin (*subroutine*)
 - modul
 - prosedur
 - fungsi
- **Prosedur**: modul program yang mengerjakan tugas/aktivitas yang spesifik dan menghasilkan suatu efek *netto*.
- **Fungsi**: upa-program yang memberikan/mengembalikan (*return*) sebuah nilai dari tipe tertentu (tipe dasar atau tipe bentukan).

Prosedur dan Fungsi (Bahasa C)

Contoh Program: prosedur.c

```
#include <stdio.h>

// membuat fungsi say_hello()
void say_hello(){
    printf("Hello Selamat Datang!\n");
}

void main(){
    // memanggil fungsi say_hello()
    say_hello();
}
```

```
D:\>gcc -o prosedur.exe prosedur.c
```

```
D:\>prosedur
Hello Selamat Datang!
```

Contoh Program: prosedur1.c [1/3]

```
#include <stdio.h>

// membuat fungsi say_hello()
void say_hello(){
    printf("Hello Selamat Datang!\n");
}

void main(){
    // memanggil fungsi say_hello()
    say_hello();
    say_hello();
    say_hello();
}
```

```
D:\>gcc -o prosedur1.exe prosedur1.c

D:\>prosedur1
Hello Selamat Datang!
Hello Selamat Datang!
Hello Selamat Datang!
```

Contoh Program: prosedur1.c [2/3]

```
#include <stdio.h>

// membuat fungsi say_hello()
void say_hello(){
    printf("Hello Selamat Datang!\n");
}

void main(){
    // memanggil fungsi say_hello()
    say_hello();
    say_hello();
    say_hello();
}
```

```
#include <stdio.h>

// membuat fungsi say_hello()
void say_hello(){
    printf("Hello Selamat Datang!\n");
}

void main(){
    // memanggil fungsi say_hello()
    say_hello();
}
```

Contoh program: prosedur.c

Contoh Program: `prosedur1.c` [3/3]

```
D:\>gcc -o prosedur1.exe prosedur1.c
```

```
D:\>prosedur1  
Hello Selamat Datang!  
Hello Selamat Datang!  
Hello Selamat Datang!
```

```
D:\>gcc -o prosedur.exe prosedur.c
```

```
D:\>prosedur  
Hello Selamat Datang!
```

Subprogram dengan Parameter

- **Parameter:**
variabel yang menyimpan nilai untuk diproses di dalam subprogram
- Parameter akan menyimpan nilai yang akan dimasukkan ke dalam subprogram.

Contoh Program: prosedur2.c

```
#include <stdio.h>

void say_hello(char name[]){
    printf("Hello %s!\n", name);
}

void main(){
    say_hello("Agus");
    say_hello("Budi");
    say_hello("Caca");
}
```

```
D:\>gcc -o prosedur2.exe prosedur2.c

D:\>prosedur2
Hello Agus!
Hello Budi!
Hello Caca!
```

Contoh Program: prosedur3.c

```
#include <stdio.h>

void add(int a, int b){
    printf("%d + %d = %d\n", a, b, a+b);
}

void main(){
    add(1, 4);
    add(8, 2);
    add(3, 2);
}
```

```
D:\>gcc -o prosedur3.exe prosedur3.c
```

```
D:\>prosedur3
```

```
1 + 4 = 5
```

```
8 + 2 = 10
```

```
3 + 2 = 5
```


Contoh Program: fungsi.c

```
#include <stdio.h>

float bagi(int a, int b){
    float hasil = (float)a / (float)b;
    return hasil;
}

void main(){
    printf("Hasil 5/2: %.2f\n", bagi(5, 2));
}
```

```
D:\>gcc -o fungsi.exe fungsi.c
```

```
D:\>fungsi
Hasil 5/2: 2.50
```

Contoh Program: fungsi1.c [1/2]

```
#include <stdio.h>

int tambah(int a, int b);
int kurang(int a, int b);

void main(){
    printf("5 + 2 = %d\n", tambah(5,2));
    printf("5 - 2 = %d\n", kurang(5,2));
}

int tambah(int a, int b){
    int hasil = a + b;
    return hasil;
}

int kurang(int a, int b){
    int hasil = a - b;
    return hasil;
}
```

```
D:\>gcc -o fungsi1.exe fungsi1.c
```

```
D:\>fungsi1
```

```
5 + 2 = 7
```

```
5 - 2 = 3
```

Contoh Program: fungsi1.c [2/2]

```
#include <stdio.h>

int tambah(int a, int b);
int kurang(int a, int b);

void main(){
    printf("5 + 2 = %d\n", tambah(5,2));
    printf("5 - 2 = %d\n", kurang(5,2));
}

int tambah(int a, int b){
    int hasil = a + b;
    return hasil;
}

int kurang(int a, int b){
    int hasil = a - b;
    return hasil;
}
```

```
#include <stdio.h>

float bagi(int a, int b){
    float hasil = (float)a / (float)b;
    return hasil;
}

void main(){
    printf("Hasil 5/2: %.2f\n", bagi(5, 2));
}
```

Contoh program: fungsi.c

Pertanyaan?