Adam Mukharil Bachtiar
English Class
Informatics Engineering 2011

Algorithms and Programming

Procedure and Function

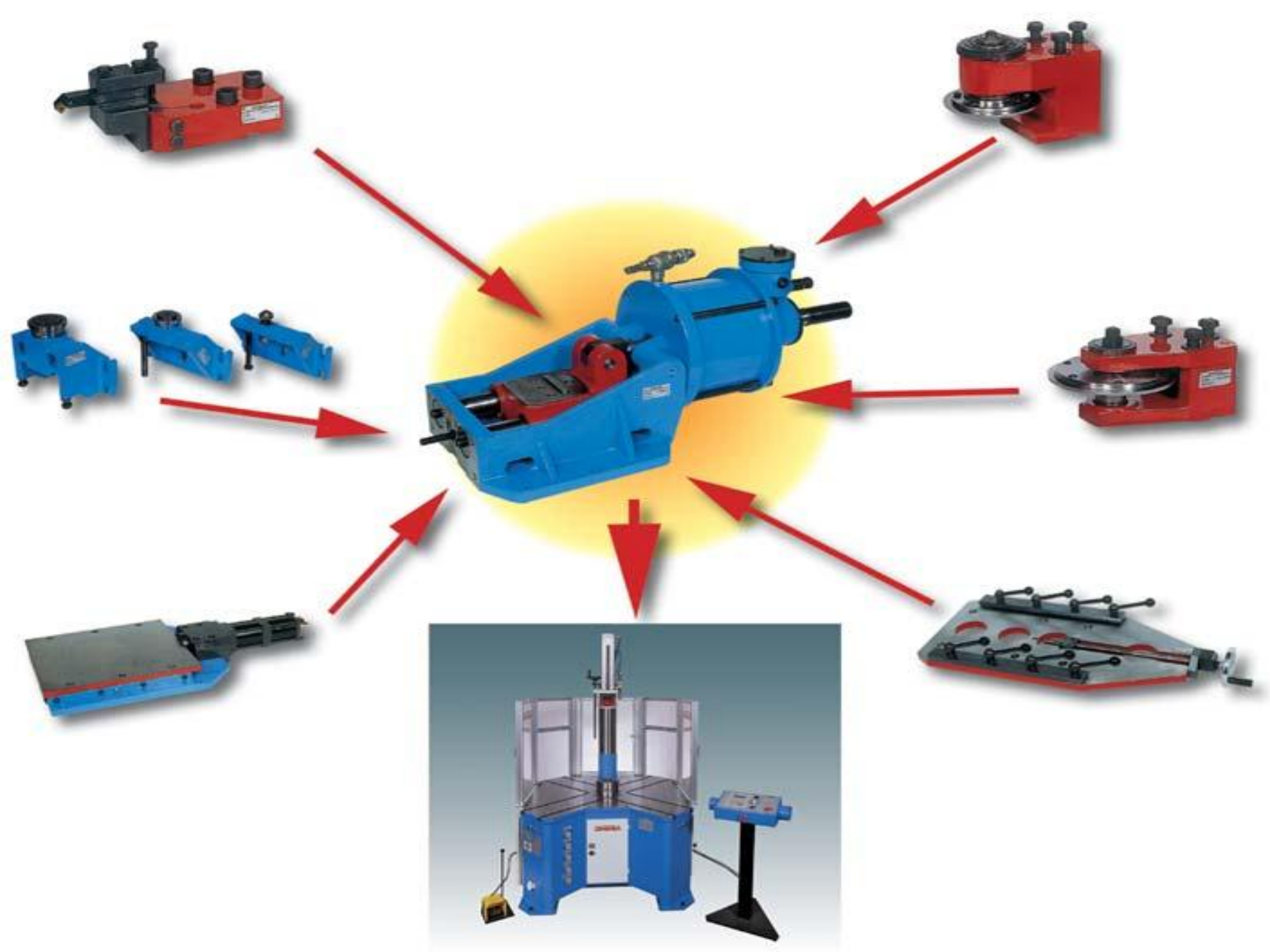
Steps of the Day

Let's Start

Modular Programming

Definition and Types of Modular Programming

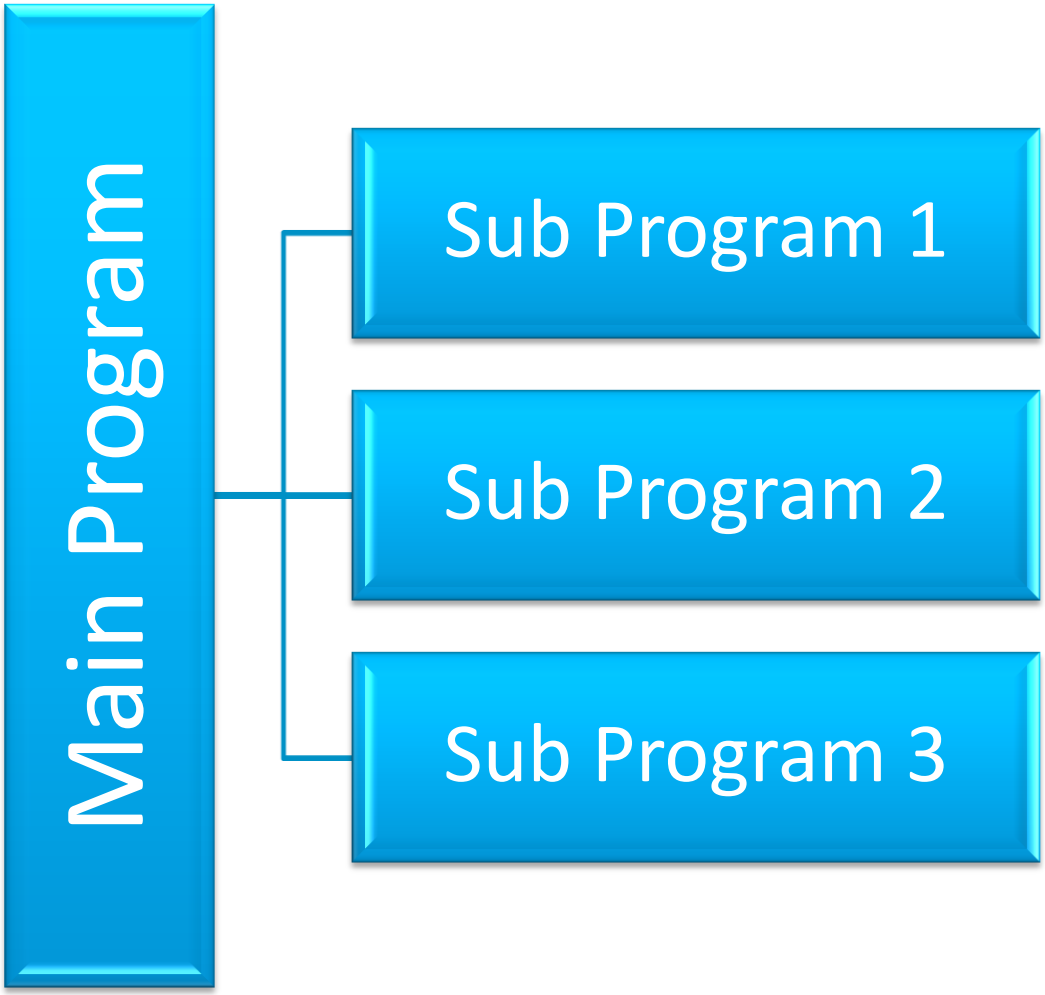
Make a program to solve all problems in your calculus book! **WHAT WILL YOU DO?**



Thinking in Modular Programming

Breakdown **a big problem** into **several small problems**. Small Problems can be reconstructed to solve the big problem.

Thinking in Modular Programming



What is Modular Programming

Programming technique that break **main program** into **several sub program**.

Benefits of Modular Programming

- Eliminate **repetition of same syntax** in program or algorithm.
- Can find **syntax error** easily.
- Easy to make **big program**.

Types of Looping Structure

- Procedure
- Function

Procedure

Definition and Structures of Procedure

What is Procedure

Instruction block that was made specially to do **specific job**.

Example of Procedure (Sleep Procedure)

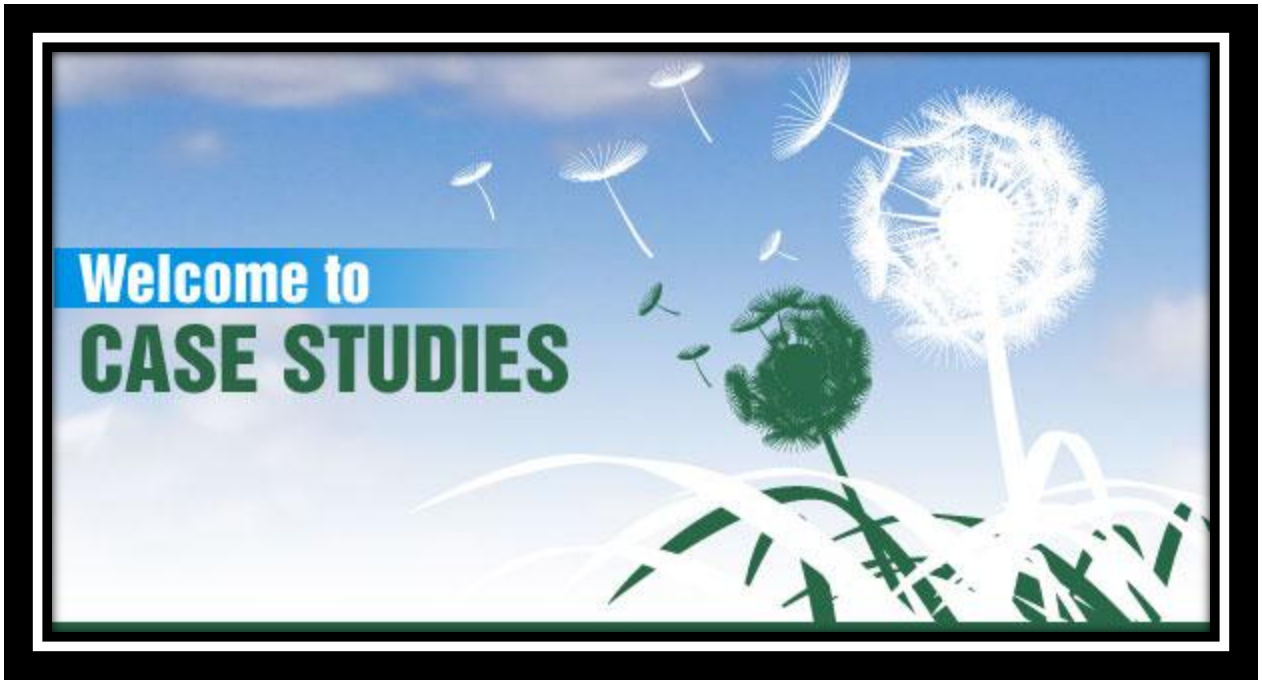
- Brush your teeth
- Go to bed
- Pray
- Cover your body with bedcover
- Count the sheep (if you are insomnia)
- Start to dream
- Wake up (if you are not death)
- Pray again

Format of Procedure (Algorithm Notation)

```
Procedure NamaProsedur (Parameter jika ada)
{I.S.: Keadaan awal sebelum prosedur dijalankan}
{F.S.: Keadaan akhir sesudah prosedur dijalankan}

Kamus:
    {Variabel, konstanta, tipe buatan lokal}

Algoritma:
    {Badan Prosedur, Berisi instruksi}
EndProcedure
```

Format of Procedure (Pascal Notation)

```
procedure NamaProsedur (Parameter jika ada);

{Variabel, konstanta, tipe buatan}

begin
    {Badan Prosedur, Berisi instruksi}
end;
```


Welcome to
CASE STUDIES

Example of Procedure (Algorithm)

```
1   Procedure HitungLuasPersegi
2   {I.S: Diinputkan sisi oleh pengguna}
3   {F.S: Menampilkan hasil perhitungan luas persegi di layar}
4
5   Kamus:
6       sisi:integer
7       luas:integer
8
9   Algoritma:
10    input(sisi)
11    luas ← sisi * sisi
12    output('Luas Persegi = ',luas)
13  EndProcedure
```

Example of Procedure (Pascal)

```
1   procedure HitungLuasPersegi;
2
3   var
4      sisi:integer;
5      luas:integer;
6
7   begin
8     write('Masukan sisi : ');readln(sisi);
9     luas ← sisi * sisi;
10    writeln('Luas Persegi = ',luas);
11    write('Tekan sembarang tombol untuk keluar...');
12    readkey();
13  end;
```

Call the Procedure

Procedure was **useless** until you call it in **main algorithm**, **main program**, **or** **the other modul**.

Format of Calling Procedure (Algorithm)

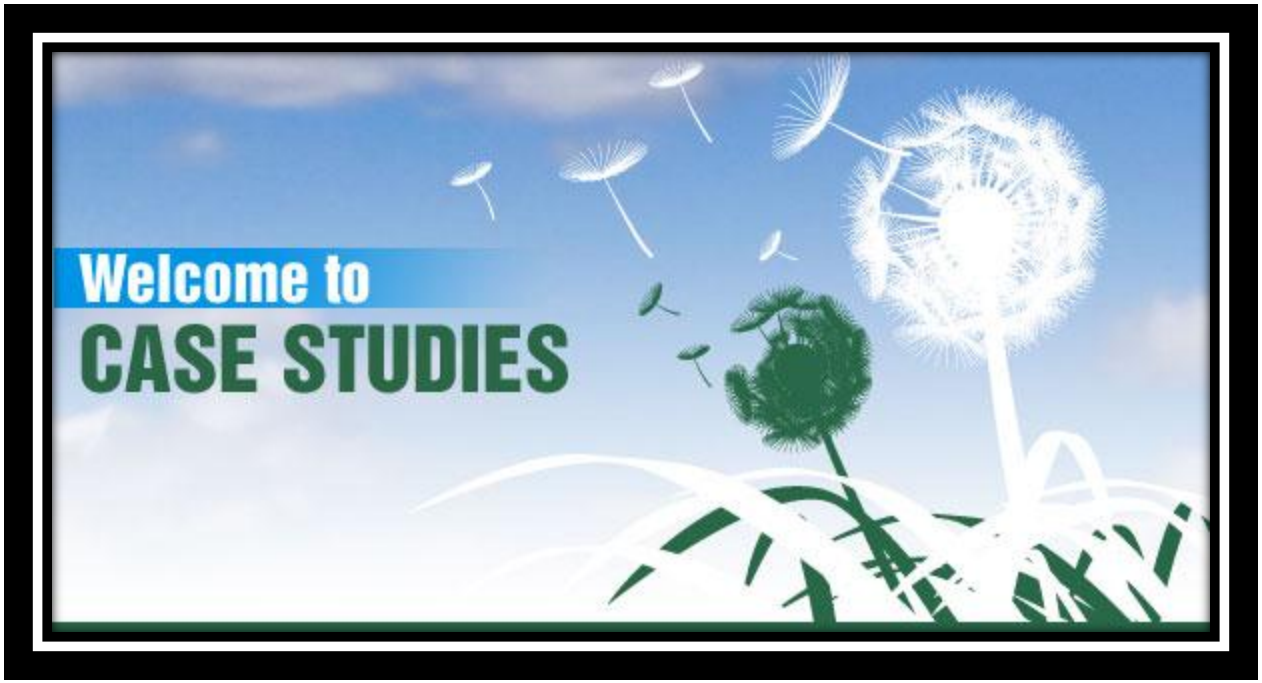
```
NamaProsedur

Atau

NamaProsedur(parameter jika ada)
```

Format of Calling Procedure (Algorithm)

```
NamaProsedur;
```

Atau

```
NamaProsedur(parameter jika ada);
```


Welcome to
CASE STUDIES

Example of Calling Procedure (Algorithm)

```
1   Algoritma PanggilHitungLuasPersegi
2   {I.S: Diinputkan sebuah bilangan oleh pengguna}
3   {F.S: Memanggil prosedur sebanyak bilangan}
4
5   Kamus:
6      i,bil:integer {kamus global}
7      procedure HitungLuasPersegi {Cukup Headernya saja}
8   Algoritma:
9     input(bil)
10    for i ← 1 to bil do
11       HitungLuasPersegi {memanggil prosedur}
12    endfor
```

Example of Calling Procedure (Pascal)

```
1   program PanggilHitungLuasPersegi;
2   uses crt;
3
4   var
5     bil:integer;
6     {Prosedur HitungLuasPersegi kamu diletakkan di sini}
7
8   begin
9     write('Masukan bilangan = ');readln(bil);
10    for i ← 1 to bil do
11       HitungLuasPersegi; {memanggil prosedur}
12    {Baris penutup jangan sampai lupa!!!}
13  end.
```

Types of Variable

- Global Variable
- Local Variable

Global Variable

Varible that **was known by entire program** or algorithm. This variable was declared in **main program** or **main algorithm**.

Local Variable

Varible that **was known only by its owner**. This variable was declared inside procedure or **function**.

Local Variable (Algorithm Notation)

FORMAL PARAMETER

```
Procedure NamaProsedur (Parameter jika ada)
{I.S.: Keadaan awal sebelum prosedur dijalankan}
{F.S.: Keadaan akhir sesudah prosedur dijalankan}

Kamus:
    {Identifier lokal diletakkan di sini}

Algoritma:
    {Badan Prosedur, Berisi instruksi}
EndProcedure
```

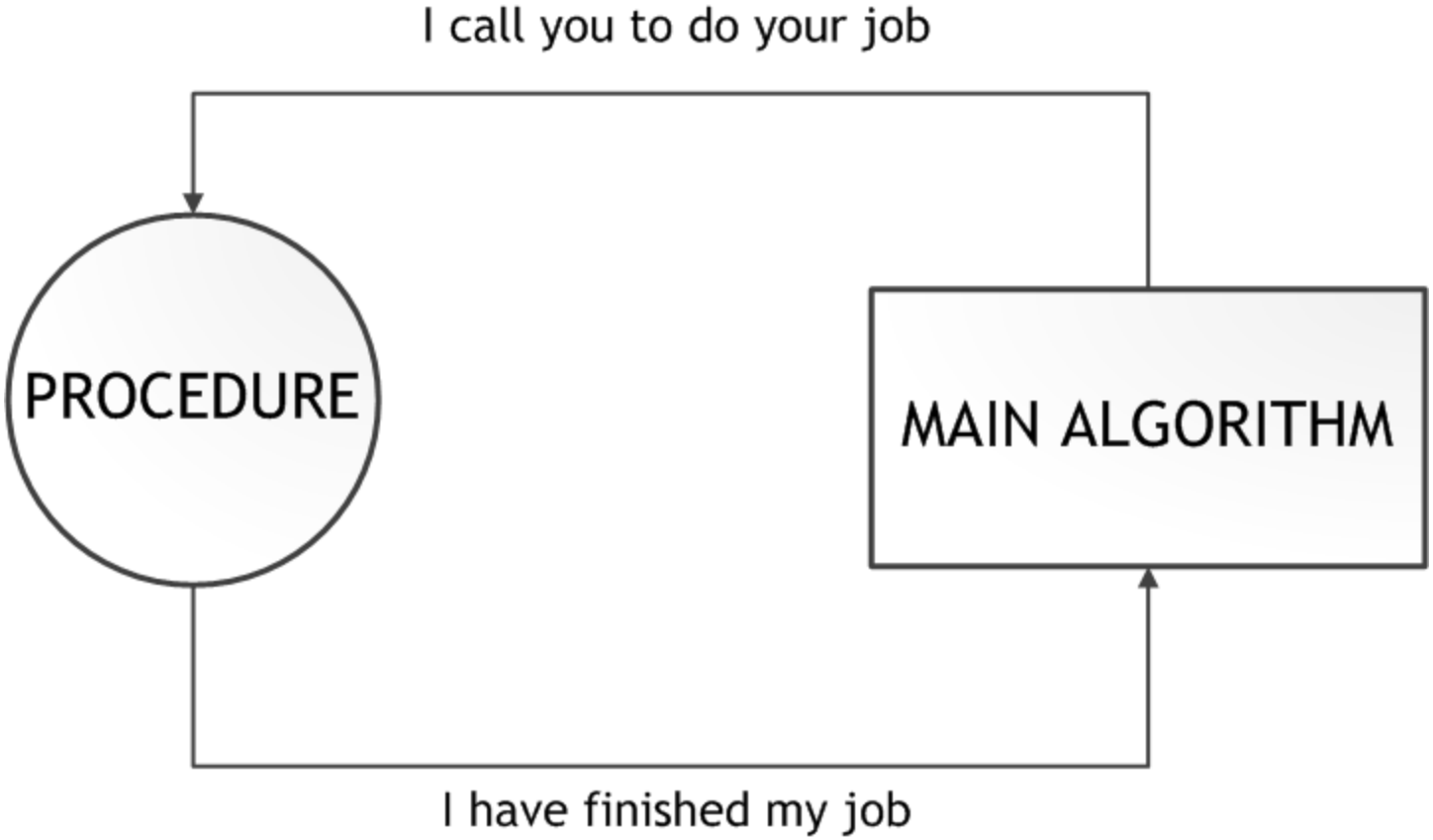
Global Variable (Algorithm Notation)

```
Algoritma judul_algoritma
{I.S.: →diisi keadaan yang terjadi di awal algoritma}
{F.S.: →diisi keadaan yang terjadi di akhir algoritma}

Kamus/Deklarasi:
  {Identifier global diletakkan di sini}

Algoritma/Deskripsi:
  {diisi dengan input, proses, dan output}
```

Local and Global Variable (Pascal Notation)

```
program nama_program;

var
      {identifier global di sini}

      procedure nama_prosedur (parameter jika ada);

      var
          {identifier lokal di sini}

      begin

      end;

begin

end.
```

Ordinary Communication Between Procedure and Main Algorithm



What is Parameter

Variable that allow us to have **more than just ordinary communication** to procedure or function.

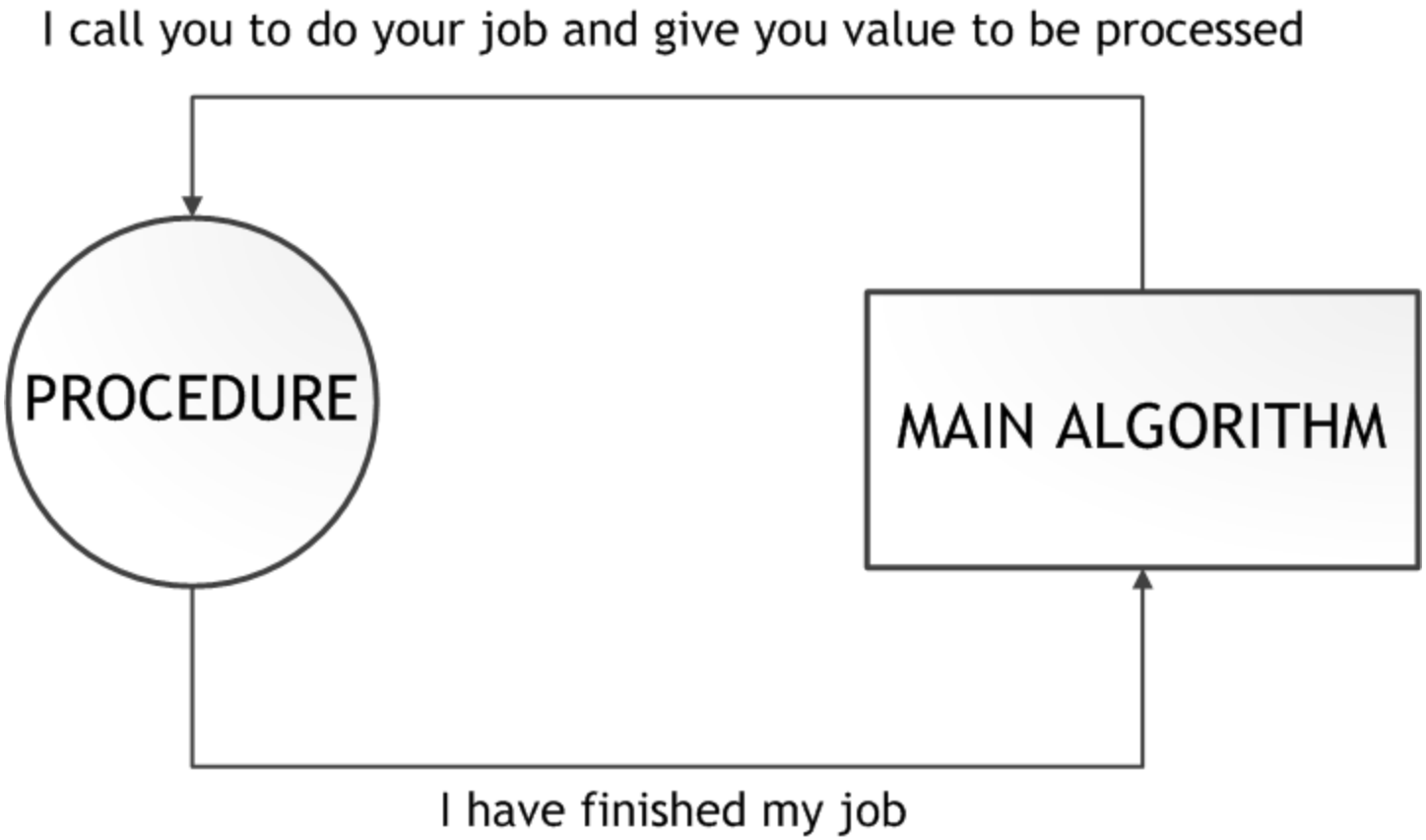
Types of Parameter

- **Input Parameter**
- **Output Parameter**
- **Input/Output Parameter**

 : means parameter by value

 : means parameter by reference

Communication Using Input Parameter



Input Parameter (Algorithm Notation)

```
Procedure NamaProsedur (Input NamaVariabel:TipeData)
{I.S.: Keadaan awal sebelum prosedur dijalankan}
{F.S.: Keadaan akhir sesudah prosedur dijalankan}

Kamus:
    {Identifier lokal diletakkan di sini}

Algoritma:
    {Badan Prosedur, Berisi instruksi}
EndProcedure
```

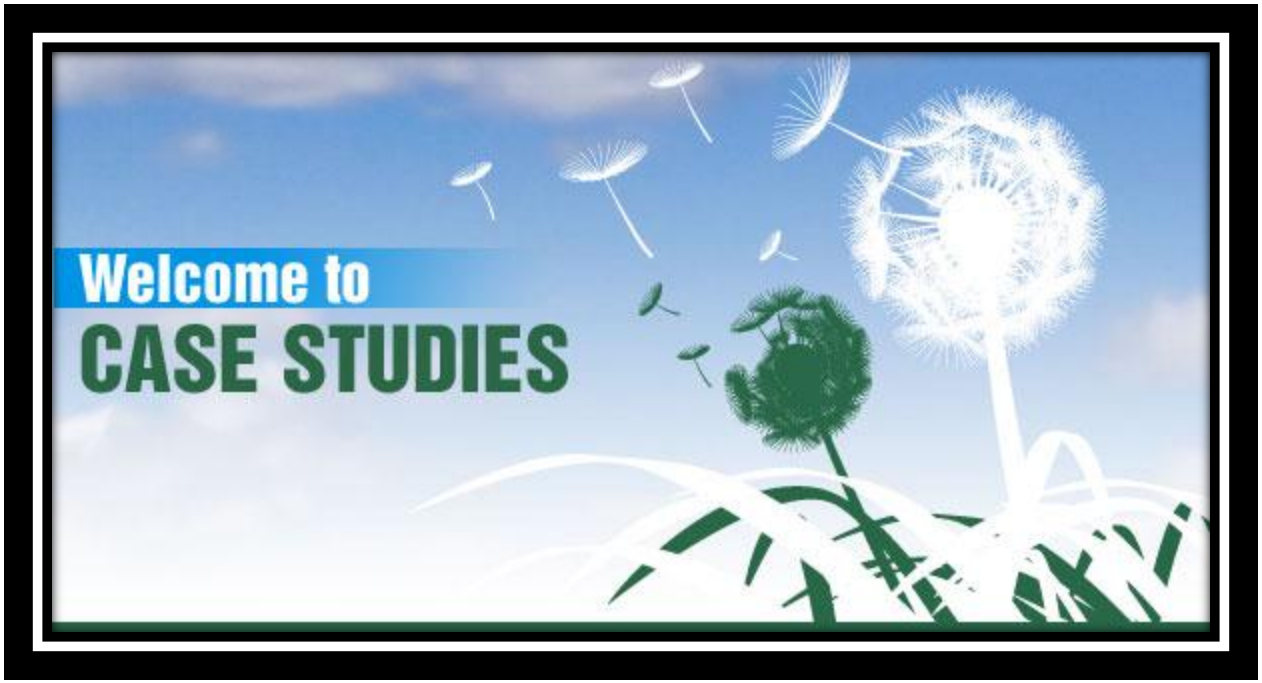
Calling Input Parameter (Algorithm Notation)

```
Algoritma NamaProsedur
{I.S.: Keadaan awal sebelum algoritma dijalankan}
{F.S.: Keadaan akhir sesudah algoritma dijalankan}

Kamus:
    {Identifier global diletakkan di sini}
    Procedure NamaProsedur (Input NamaVariabel:TipeData)

Algoritma:
    NamaProsedur(NamaVariabel) {pemanggilan prosedur}
EndProcedure
```

ACTUAL PARAMETER

Input Parameter (Pascal Notation)

```
program nama_program;

var
      {identifier global di sini}

      procedure nama_prosedur (variabel:tipedata);

      var
          {identifier lokal di sini}

      begin

      end;

Begin
      nama_prosedur(variabel);{pemanggilan prosedur}
end.
```


Welcome to
CASE STUDIES

Example of Input Parameter (Algorithm)

```
1   Procedure Persegi(Input sisi:integer)
2   {I.S: Menerima input berupa sisi}
3   {F.S: Menampilkan luas dan keliling persegi}
4
5   Kamus:
6       luas,keliling:integer
7
8   Algoritma:
9       luas ← sisi * sisi
10      keliling ← 4 * sisi
11      output(luas,keliling)
12  EndProcedure
```

Example of Calling Input Parameter (Algorithm)

```
1   Algoritma PanggilHitungLuasPersegi
2   {I.S: Diinputkan sisi oleh pengguna}
3   {F.S: Memanggil prosedur persegi}
4
5   Kamus:
6       sisi:integer
7       procedure Persegi(Input sisi:integer)
8
9   Algoritma:
10    input(sisi)
11    Persegi(sisi)
```

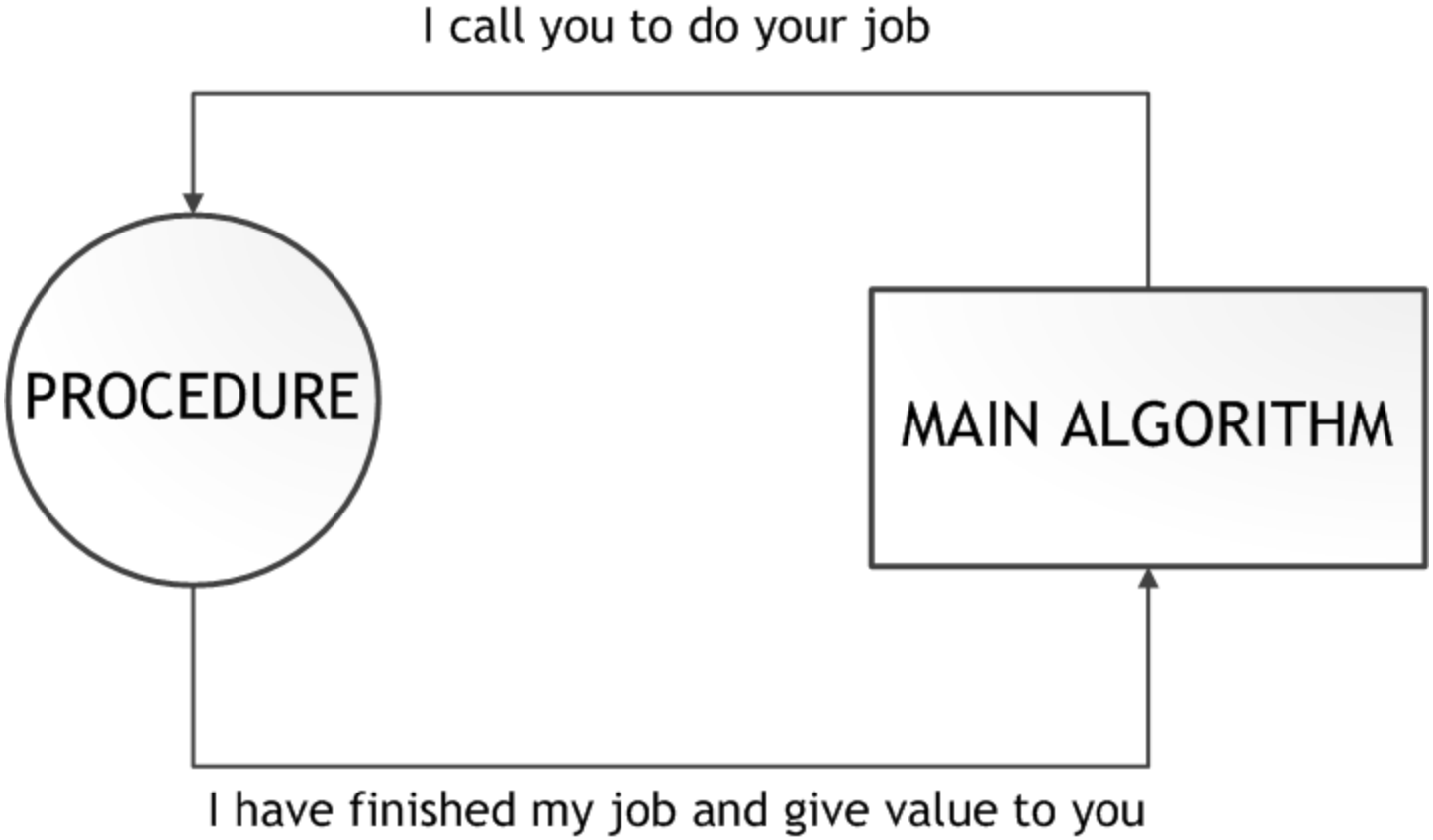
Example of Input Parameter (Pascal)

```
1   program HitungPersegi;
2   uses crt;
3
4   var
5     sisi:integer;
6     procedure persegi(sisi:integer);
7
8     var
9        luas,keliling:integer;
10
11    begin
12       luas := sisi * sisi;
13       keliling := 4 * sisi;
14       writeln('Luas Persegi : ',luas); {bersambung}
```

Example of Input Parameter (Pascal)

```
15      writeln('Keliling persegi : ',keliling);
16    end;
17
18  begin
19    write('Masukan sisi persegi= ');readln(sisi);
20    persegi(sisi);
21    writeln();
22    write('Tekan sembarang tombol untuk menutup...');
23    readkey();
24  end.
```

Communication Using Output Parameter



Output Parameter (Algorithm Notation)

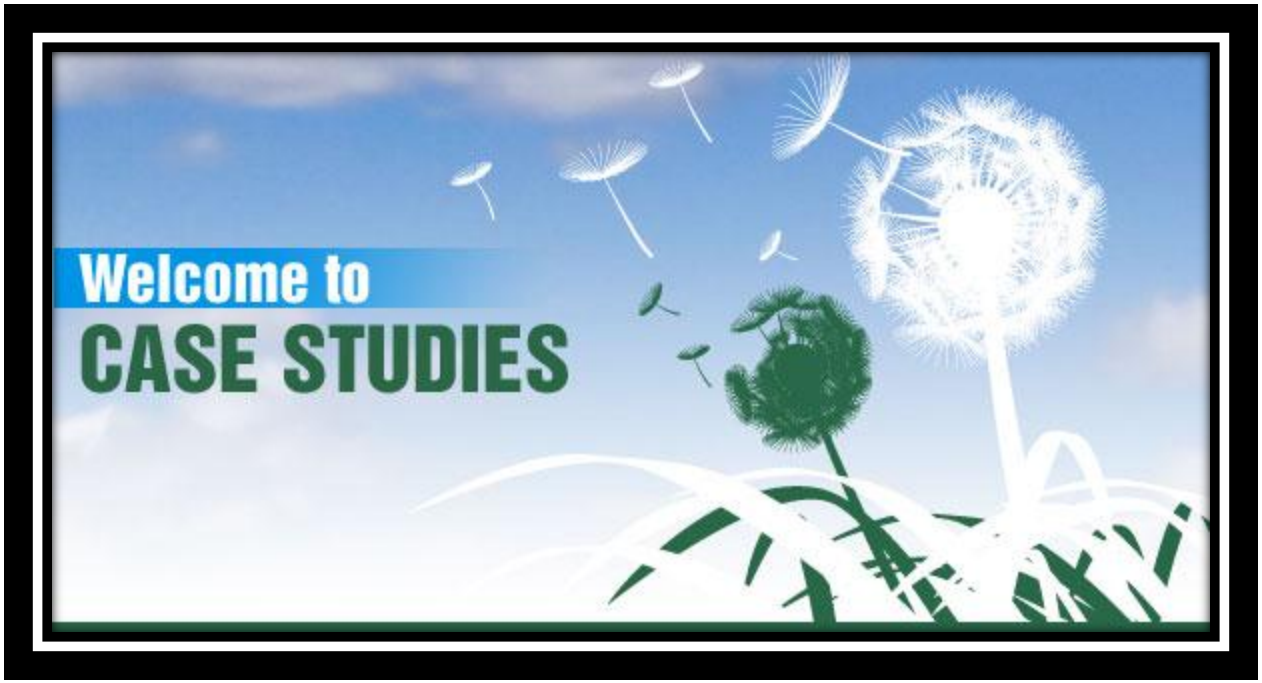
```
Procedure NamaProsedur (Output NamaVariabel:TipeData)
{I.S.: Keadaan awal sebelum prosedur dijalankan}
{F.S.: Keadaan akhir sesudah prosedur dijalankan}

Kamus:
    {Identifier lokal diletakkan di sini}

Algoritma:
    {Badan Prosedur, Berisi instruksi}
EndProcedure
```

Calling Output Parameter (Algorithm Notation)

```
Algoritma NamaProsedur
{I.S.: Keadaan awal sebelum algoritma dijalankan}
{F.S.: Keadaan akhir sesudah algoritma dijalankan}

Kamus:
    {Identifier global diletakkan di sini}
    Procedure NamaProsedur (Output NamaVariabel:TipeData)

Algoritma:
    NamaProsedur(NamaVariabel) {pemanggilan prosedur}
EndProcedure
```

Output Parameter (Pascal Notation)

```
program nama_program;

var
      {identifier global di sini}

      procedure nama_prosedur (var variabel:tipedata);

      var
          {identifier lokal di sini}

      begin

      end;

Begin
      nama_prosedur(variabel);{pemanggilan prosedur}
end.
```


Welcome to
CASE STUDIES

Example of Output Parameter (Algorithm)

```
1   Procedure Persegi(Output luas,keliling:integer)
2   {I.S: Meminta input sisi dari pengguna}
3   {F.S: Mengirimkan nilai luas dan keliling persegi}
4
5   Kamus:
6       sisi:integer
7
8   Algoritma:
9       input(sisi)
10      luas ← sisi * sisi
11      keliling ← 4 * sisi
12      EndProcedure
```

Example of Calling Output Parameter (Algorithm)

```
1   Algoritma PanggilHitungLuasPersegi
2   {I.S: Memanggil prosedur persegi}
3   {F.S: Menampilkan nilai dari prosedur persegi}
4
5   Kamus:
6       luas,keliling:integer
7       Procedure Persegi(Output luas,keliling:integer)
8
9   Algoritma:
10    Persegi(luas,keliling)
11    output(luas,keliling)
```

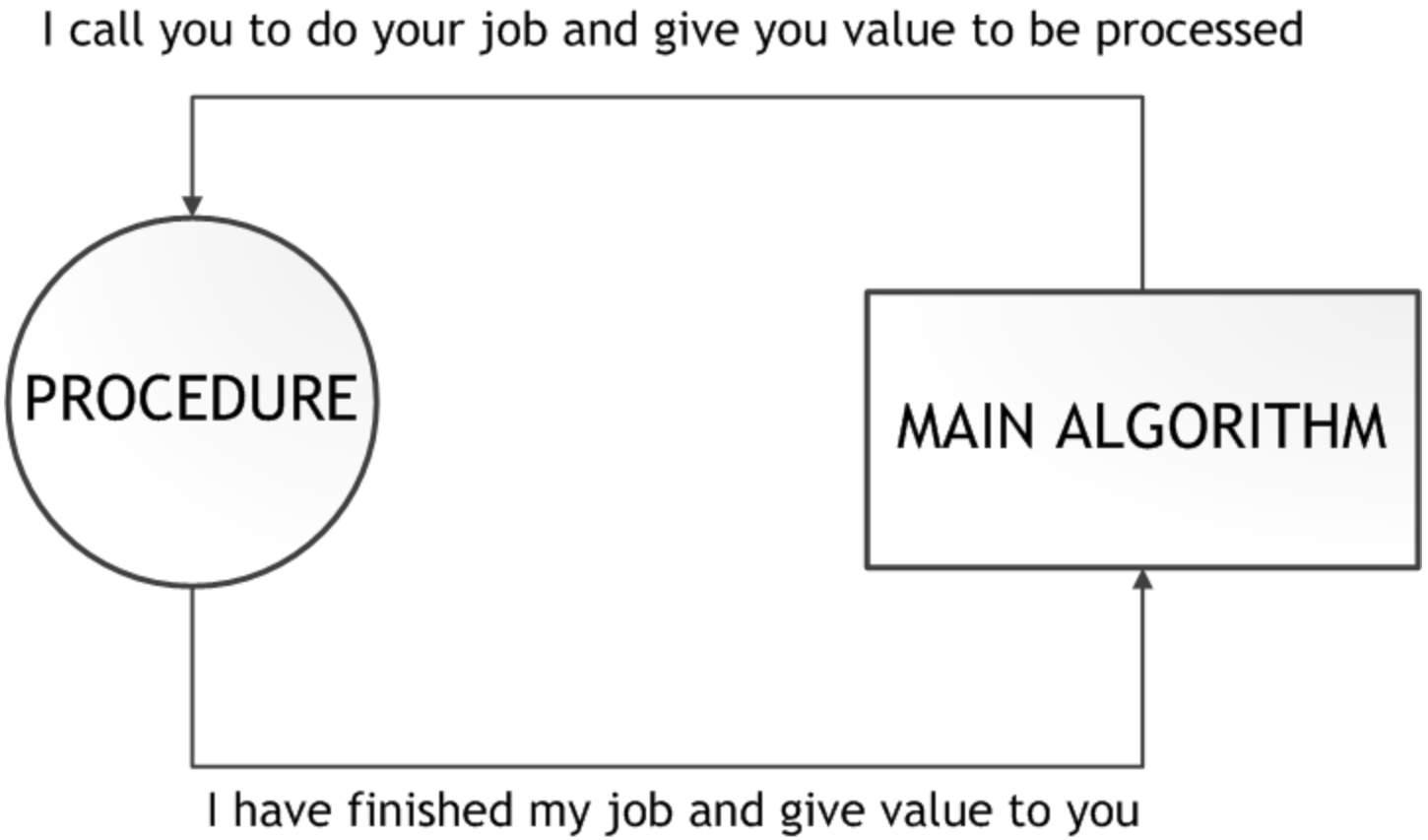
Example of Output Parameter (Pascal)

```
1   program HitungPersegi;
2   uses crt;
3
4   var
5     luas,keliling:integer;
6     procedure persegi(var luas,keliling:integer);
7
8     var
9        sisi:integer;
10
11    begin
12       write('Masukan sisi persegi= ');readln(sisi);
13       luas := sisi * sisi;
14       keliling := 4 * sisi; {bersambung}
```

Example of Output Parameter (Pascal)

```
15    end;
16
17  begin
18    persegi(luas,keliling);
19    writeln('Keliling persegi : ',keliling);
20    writeln('Luas Persegi : ',luas);
21    writeln();
22    write('Tekan sembarang tombol untuk menutup...');
23    readkey();
24  end.
```

Communication Using Input/Output Parameter



Input/Output Parameter (Algorithm Notation)

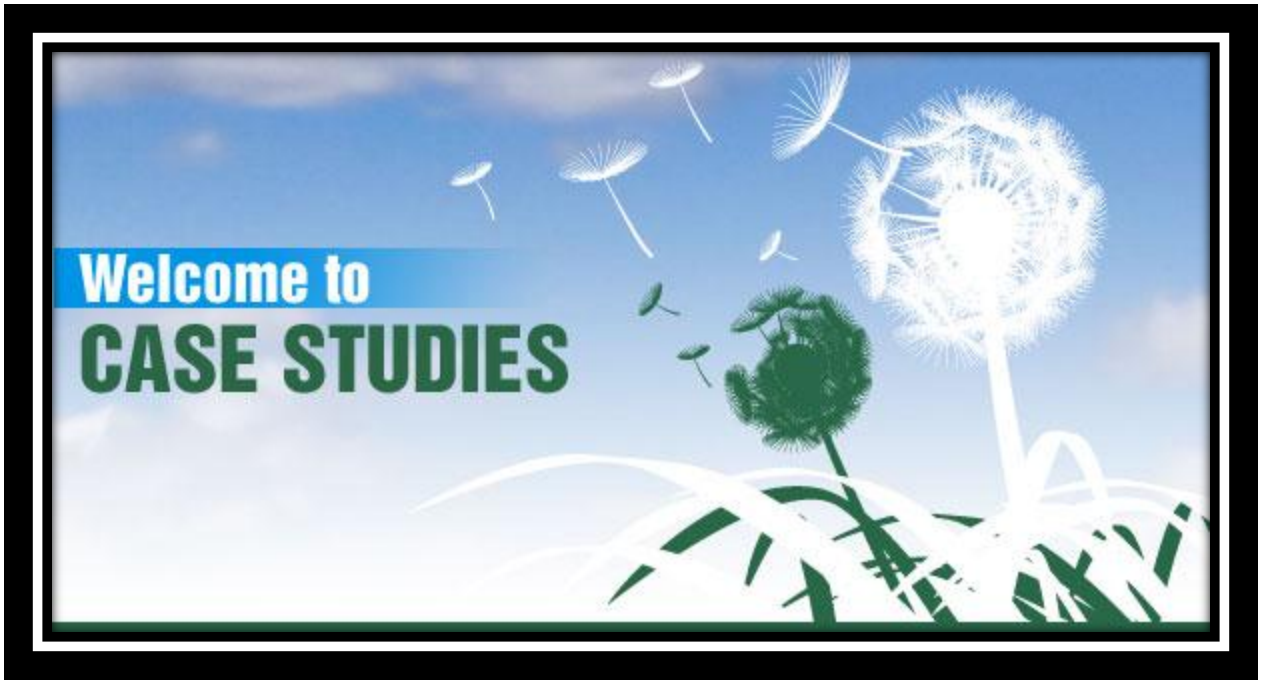
```
Procedure NamaProsedur (I/O NamaVariabel:TipeData)
{I.S.: Keadaan awal sebelum prosedur dijalankan}
{F.S.: Keadaan akhir sesudah prosedur dijalankan}

Kamus:
    {Identifier lokal diletakkan di sini}

Algoritma:
    {Badan Prosedur, Berisi instruksi}
EndProcedure
```

Calling Input/Output Parameter (Algorithm Notation)

```
Algoritma NamaProsedur
{I.S.: Keadaan awal sebelum algoritma dijalankan}
{F.S.: Keadaan akhir sesudah algoritma dijalankan}

Kamus:
    {Identifier global diletakkan di sini}
    Procedure NamaProsedur (I/O NamaVariabel:TipeData)

Algoritma:
    NamaProsedur(NamaVariabel) {pemanggilan prosedur}
EndProcedure
```

Input/Ouput Parameter (Pascal Notation)

```
program nama_program;

var
      {identifier global di sini}

      procedure nama_prosedur (var variabel:tipedata);

      var
          {identifier lokal di sini}

      begin

      end;

Begin
      nama_prosedur(variabel);{pemanggilan prosedur}
end.
```


Welcome to
CASE STUDIES

Example of Input/Output Parameter (Algorithm)

```
1   Procedure Persegi(I/O sisi:integer,Output luas,keliling:integer)
2   {I.S: Menerima input sisi}
3   {F.S: Mengirimkan nilai sisi, luas, dan keliling persegi}
4
5   Kamus:
6
7   Algoritma:
8       luas ← sisi * sisi
9       keliling ← 4 * sisi
10      sisi ← sisi + 1; {lihat apa yang terjadi}
11      EndProcedure
```

Example of Calling Output Parameter (Algorithm)

```
1   Algoritma PanggilHitungLuasPersegi
2   {I.S: Memanggil prosedur persegi}
3   {F.S: Menampilkan nilai dari prosedur persegi}
4
5   Kamus:
6       sisi,luas,keliling:integer
7       Procedure Persegi(I/O sisi:integer,Output luas,keliling:integer)
8
9   Algoritma:
10    input(sisi)
11    Persegi(sisi,luas,keliling)
12    output(sisi,luas,keliling)  {Berapa nilai sisinya?}
```

Example of Input/Output Parameter (Pascal)

```
1   program HitungPersegi;
2   uses crt;
3
4   var
5     luas,keliling:integer;
6     procedure persegi(sisi:integer;var luas,keliling:integer);
7
8     begin
9         luas := sisi * sisi;
10        keliling := 4 * sisi;
11        sisi := sisi + 1; {Lihat apa yang terjadi}
12    end;
```

Example of Input/Output Parameter (Pascal)

```
13  begin
14    write('Masukan sisi persegi= ');readln(sisi);
15    persegi(sisi,luas,keliling);
16    writeln('Keliling persegi : ',keliling);
17    writeln('Luas Persegi : ',luas);
18    writeln('Sisi persegi : ',sisi);
19    writeln();
20    write('Tekan sembarang tombol untuk menutup...');
21    readkey();
22  end.
```

Function

Definition and Structures of Function

What is Function

Instruction block that was made specially to do **specific job** and **return a value**. Such as:

F(x)= 2x+4 → it will return 6 for x=1.

Difference Between Procedure and Function

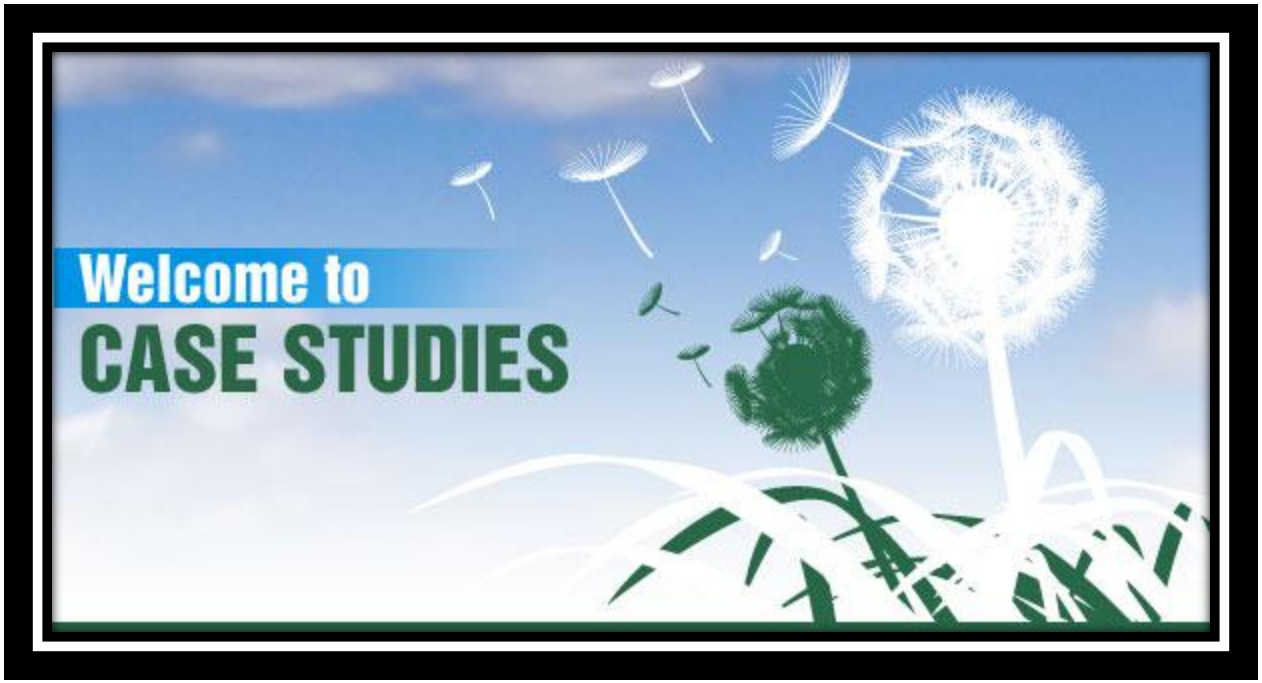
Procedure **was not made to return a value** (only do specific job) but function **was made to return a value** (More specific than procedure).

Format of Function (Algorithm Notation)

```
FUnction NamaFungsi (Parameter jika ada) → tipefungsi
{I.S.: Keadaan awal sebelum fungsi dijalankan}
{F.S.: Keadaan akhir sesudah fungsi dijalankan}

Kamus:
    {Variabel, konstanta, tipe buatan lokal}

Algoritma:
    {Badan fungsi, Berisi instruksi}
    return VALUE {tipenya sama dengan tipe fungsi}
EndFunction
```

Format of Procedure (Pascal Notation)

```
function NamaFungsi (Parameter jika ada):tipefungsi;


{Variabel, konstanta, tipe buatan}


begin
    {Badan Fungsi, Berisi instruksi}
    NamaFungsi := VALUE; (tipenya sama dengan tipe fungsi}
end;
```


Welcome to
CASE STUDIES

Example of Function (Algorithm)

```
1   Function LuasPersegi(Input sisi:integer) → integer
2   {I.S: Menerima input berupa sisi}
3   {F.S: Menampilkan luas dan keliling persegi}
4
5   Kamus:
6
7   Algoritma:
8       return sisi * sisi
9   EndFunction
```

Example of Function (Algorithm)

```
1   Algoritma PanggilLuasPersegi
2   {I.S: Diinputkan sisi oleh pengguna}
3   {F.S: Menampilkan nilai fungsi luas persegi}
4
5   Kamus:
6       sisi,luas:integer
7       Function LuasPersegi(Input sisi:integer) → integer
8
9   Algoritma:
10    input(sisi)
11    luas ← LuasPersegi(sisi)
12    output(luas)
```

Example of Function (Pascal)

```
1   program HitungPersegi;
2   uses crt;
3
4   var
5     sisi,luas:integer;
6     function LuasPersegi(sisi:integer):integer;
7
8     begin
9        LuasPersegi := sisi * sisi;
10    end;
11
12  begin
13    write('Masukan sisi persegi= ');readln(sisi);
14    luas := LuasPersegi(sisi); {Pemanggilan Function}
```

Example of Function (Pascal)

```
15    write('Luas persegi : ',luas);
16    writeln();
17    write('Tekan sembarang tombol untuk menutup...');
18    readkey();
19  end.
```

THANK YOU

GRACIAS

Contact Person:
Adam Mukharil Bachtiar
Informatics Engineering UNIKOM
Jalan Dipati Ukur Nomor. 112-114 Bandung 40132
Email: adfbipotter@gmail.com
Blog: http://adfbipotter.wordpress.com