

## Flowchart If – Then – Else

อนันต์ ผลเพิ่ม  
Anan Phonphoem  
anan@cpe.ku.ac.th

1

## Outline

- Flowcharts
- IF – THEN – ELSE

2

## Boolean Expression

- Two possible values: True, False
- 1. Relation Operator
  - =, <, >, <>, <=, >=
- 2. Boolean Operator
  - AND, OR, NOT

15 = 34	False
15.05 < 17	True
34 <> 34.00002	True

3

## Precedence rules for arithmetic operators

- ( ) parentheses
- Unary + and –
- \*, / , DIV , MOD
- + –
- If equal precedence, left to right

### Examples

-a+j/-w	= (-a) + (j / (-w))
C*23/6+23mod2	= ((C*23)/6) + (23 mod 2)

4

## Precedence rules for Boolean Operators

- ( ) parentheses
- NOT
- and
- Or
- <, <=, >, <=, =, <>
- If equal precedence, left to right

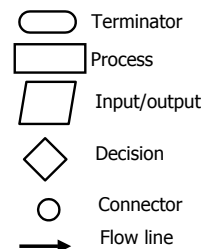
### Examples

(count <= 10) **AND** (Sum <= Limit) **OR NOT** Stop

5

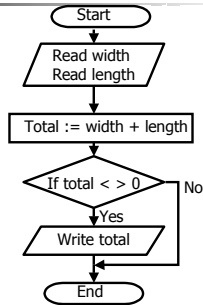
## Flowcharts

- Graphical representation of algorithm



6

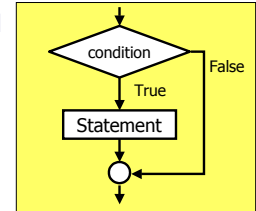
## Flowchart example



7

## IF – THEN

**IF** condition **THEN**  
statement



8

## IF – Then (sum1.pas)

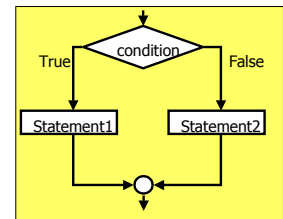
```

Program summation1;
Var
  Sum, number : integer;
Begin
  Sum := 10;
  Write ('Please input a number');
  Readln(number);
  if number < 0 then
    number := 0;
  Sum := Sum + number;
  writeln ('Sum =', Sum)
End.
  
```

9

## IF – THEN – ELSE

**IF** condition **THEN**  
statement 1  
**ELSE**  
statement 2



10

## IF – Then – Else (sum2.pas)

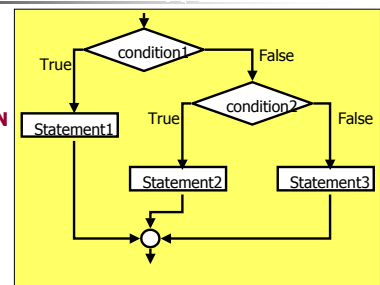
```

Program summation2;
Var
  Sum, number : integer;
Begin
  Sum := 10;
  Write ('Please input a number');
  Readln(number);
  if number < 0 then
    number := 0;
  else
    number := 20;
  Sum := Sum + number;
  writeln ('Sum =', Sum)
End.
  
```

11

## Nested Logic

**IF** cond1 **THEN**  
statement1  
**ELSE**  
**IF** cond2 **THEN**  
Statement2  
**ELSE**  
statement3



12

## Nested Logic (Sum3.pas)

```
Begin
  Sum := 10;
  Write ('Please input a number');
  Readln(number);
  if number > 0 then
    number := number + 1;
    if number > 5 then
      number := 5;
    else
      number := 20;
  Sum := Sum + number;
  writeln ('Sum =', Sum)
End.
```

13

## Nested Logic (sum4.pas)

```
Begin
  Sum := 10;
  Write ('Please input a number');
  Readln(number);
  if number > 0 then
    begin
      number := number + 1;
      if number > 5 then
        number := 5;
    End
  else
    number := 20;
  Sum := Sum + number;
  writeln ('Sum =', Sum)
End.
```

14

## Case

```
Case expression of
  const_value1 : statement1;
  const_value2 : statement2;
  const_value3 : statement3;
Else
  statement4;
End;
```

15

## Case Example

```
Readln (number);
Case number of
  1,2,3 : writeln('small');
  4,5,6 : writeln('medium');
  7,8 : writeln('large');
End;
```

16