ICDT 1201Y_Computer Programming

Control Structures (Week 4)

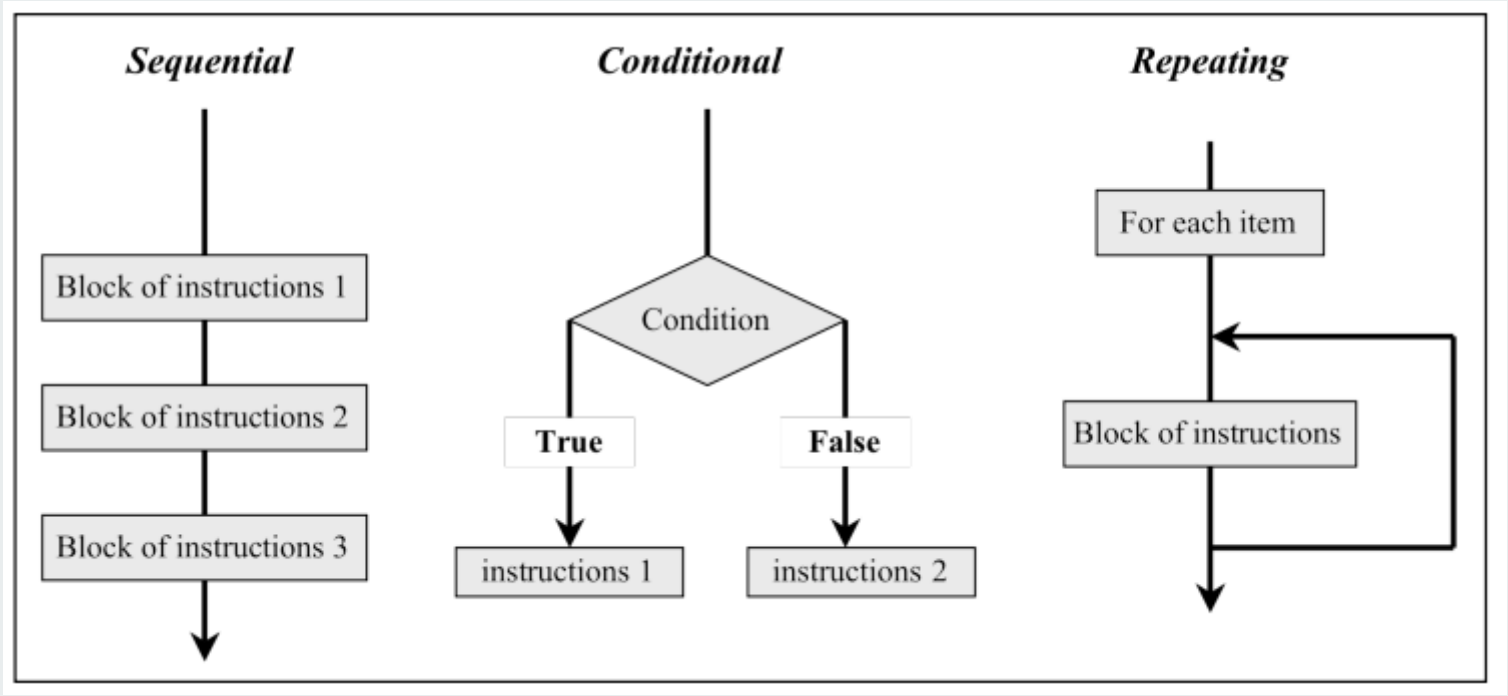
Learning Objectives

- Understand the objectives of decision in programs
- Illustrate a given problem using flowchart and pseudocodes
- Write programs using if… elif …else statement in Python

Control Structures

- The three main types of flow in a computer program:
  - sequential, in which instructions are executed successively,
  - conditional, in which the blocks “instructions 1” and “instructions 2” are executed if the Condition is True or False, respectively, and
  - repeating, in which instructions are repeated over a whole list.

Control Structures



1.Decision Structure

Decision making statements

- Python programming language provides the following types of decision making statements.
  - if statements
  - if....else statements
  - if..elif..else statements
  - nested if statements

Simple Decisions

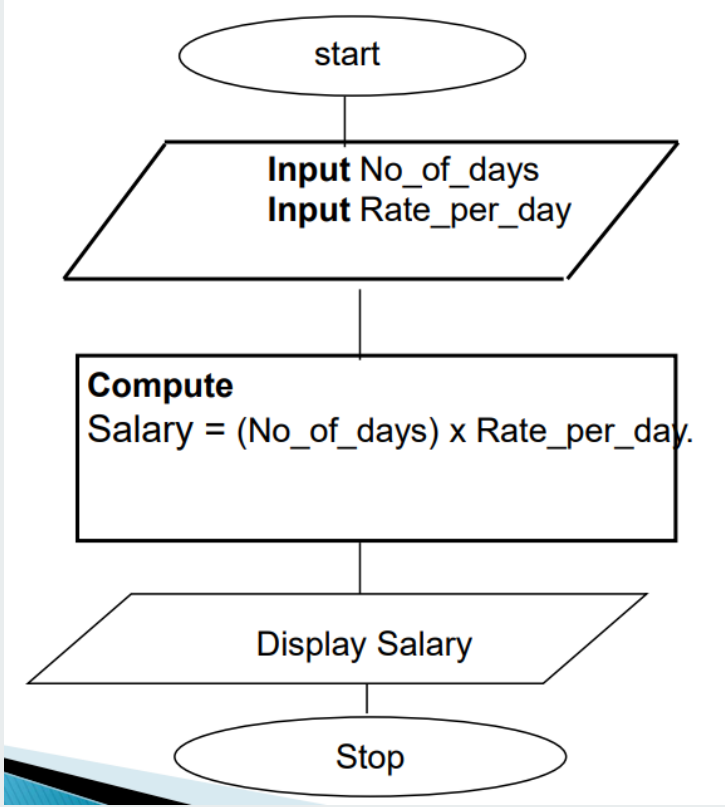
- Sequencing is not sufficient to solve every programming problem.
- To alter the sequence of flow in a program, a decision structure/control structure can be used to suit the needs of a particular situation.
- Conditional constructs are used to incorporate decision making into programs.
- The result of this decision making determines the sequence in which a program will execute instructions.

Decision in Programs

Consider the following Problem:

- Draw the flowchart and write the program to calculate the salary of a person at the end of a month. The program should take as inputs the no_of_days and the rate_per_day.
- Does this problem involve any decision?

Flowchart for the program



The program in Python

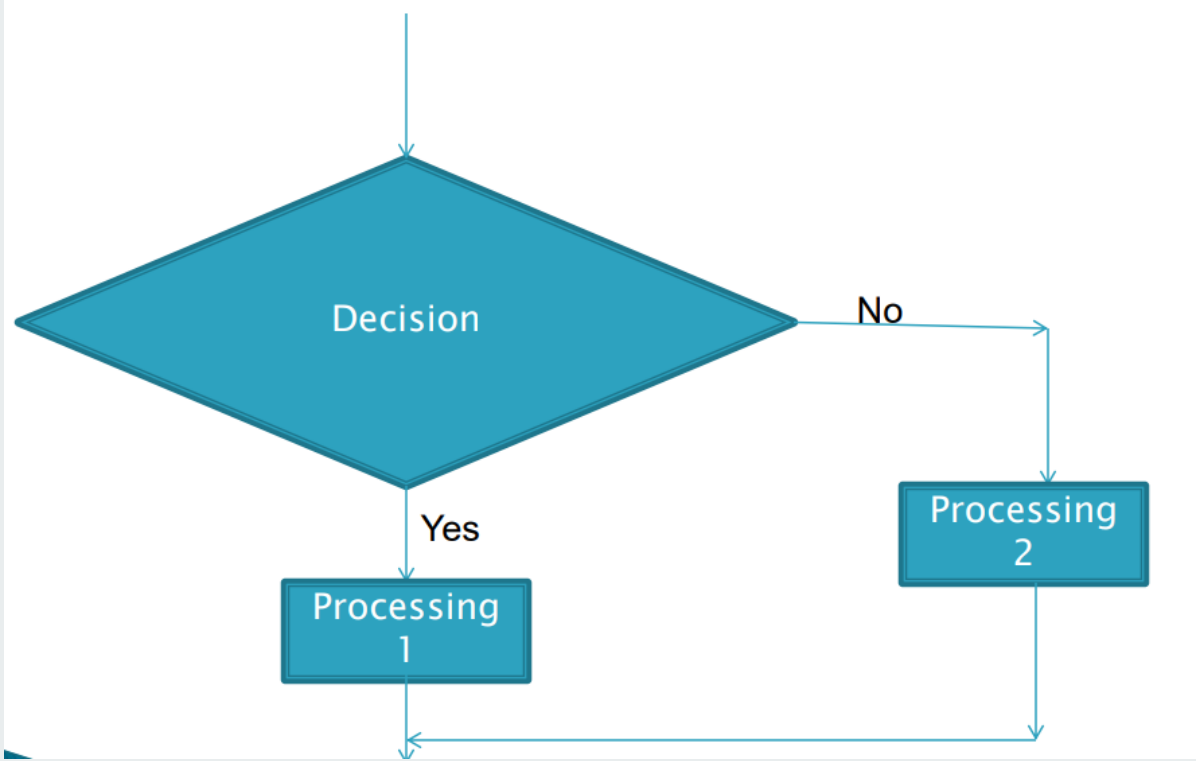
```
# Input num of days and rate per day
no_of_days = int(input("Enter the number of days worked: "))
rate_per_day = float(input("Enter the rate per day: "))

# Calculate the salary
total_salary = no_of_days* rate_per_day

# Display the result
 print(f"The salary at the end of the month is:
{total_salary:.2f}")
```

Problems involving decisions

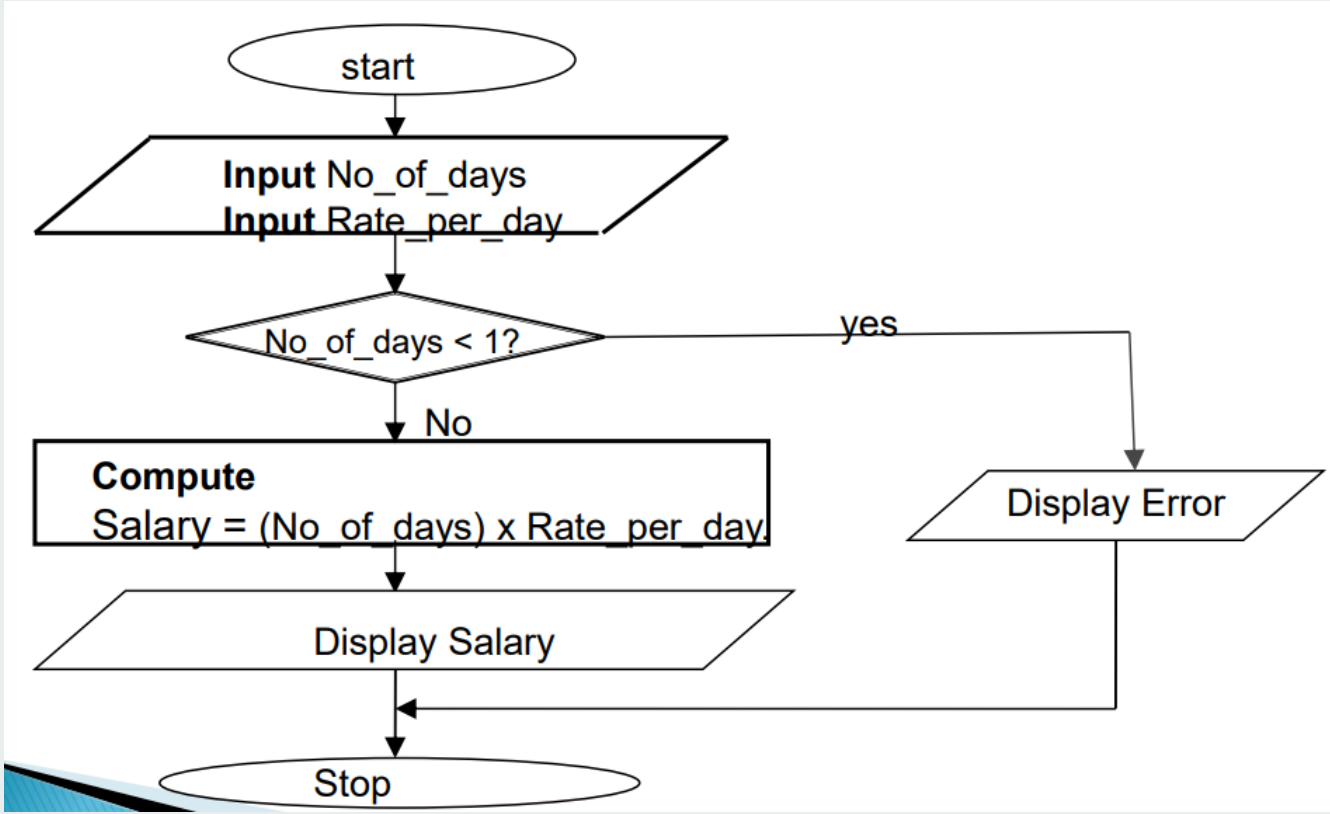
Consider the problem below:

- Draw the flowchart and write the program to calculate the salary of a person. You know the no_of_days and the rate_per_day. If the no_of_days is < 1 , then your program should display an error.
- Does this problem involve any decision now?

The Decision symbol

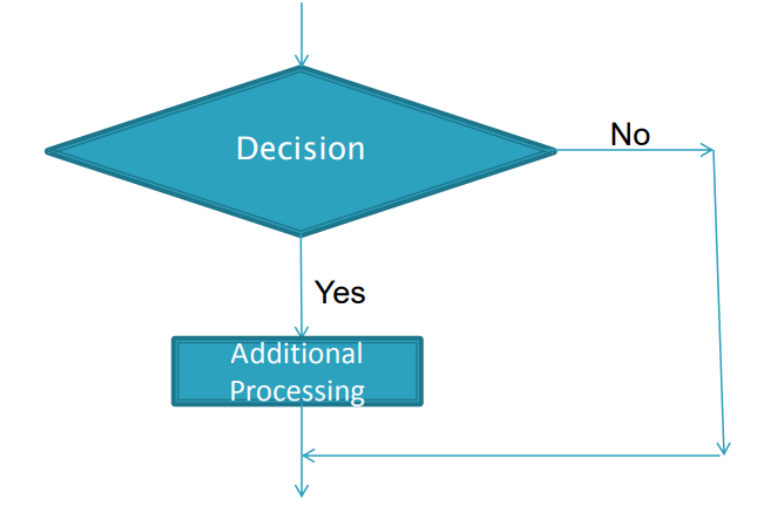


Flow chart including the decision part



Simple decision

- Simple Decisions are cases where there is a path to be followed (processing to be done)if a condition is true. If the condition evaluates to false, the path is simply not followed.

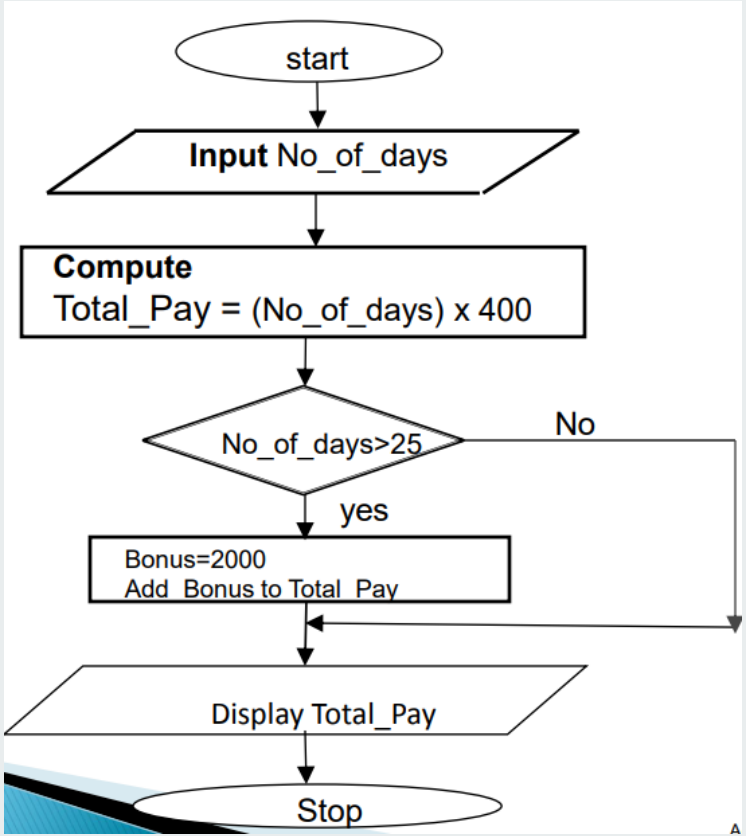


Simple Decisions

Example 5.1 :

- Consider a person being paid on a daily rate of Rs. 400. At the end of the month, he obtains his salary. Additionally, he obtains a bonus of Rs 2000 if he has worked for more than 25 days. We need a program to calculate his total pay at the end of a month.
- Inputs: No. of days.
- Outputs: Total Pay.
- Processing:
  - General : Total Pay = No.of days * 400
  - Conditional: Total Pay = Total Pay + Bonus
- The bonus part is conditional and is based on a simple decision.
- Let's draw the flowchart

Flowchart for example 5.1



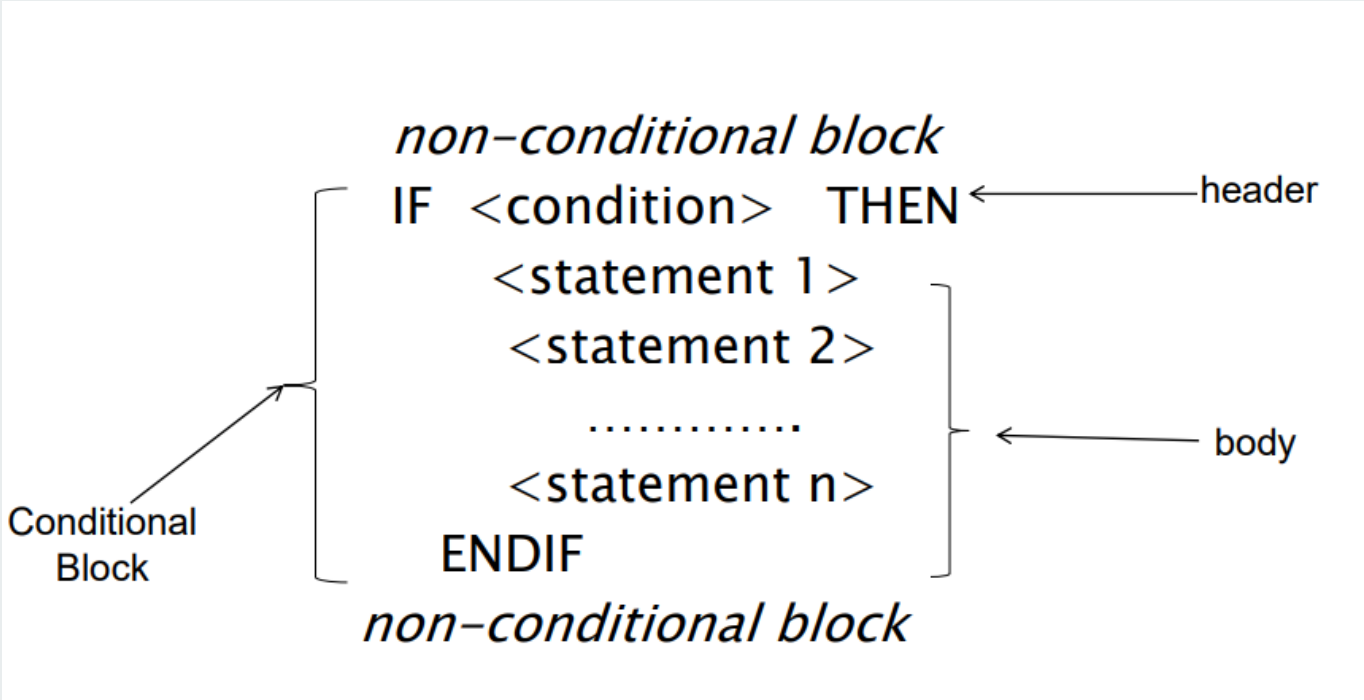
The if statement

- In Programming, to be able to solve problems which involve decision making, we shall make use of the If statement.
- In the simplest form, the if statement (in pseudocodes) is as follows:

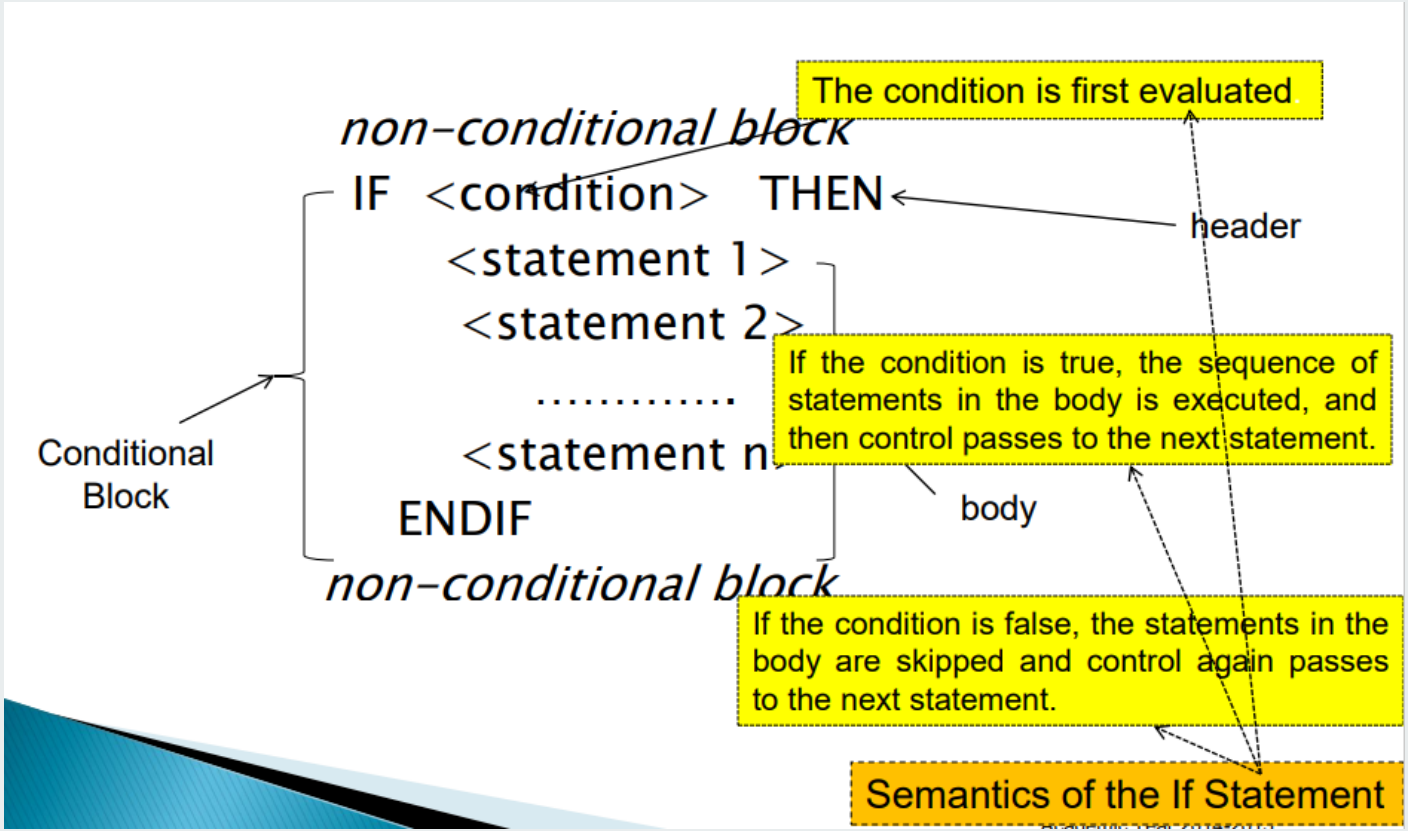
```
If <condition> Then
    <statement(s)>
End If
```

- The condition is evaluated first. If the ***condition*** is ***true then*** the statement(s) block is executed. ***Otherwise***, the next statement following the statement(s) block is executed.

General Form of the basic if statement



Semantics of the if statement

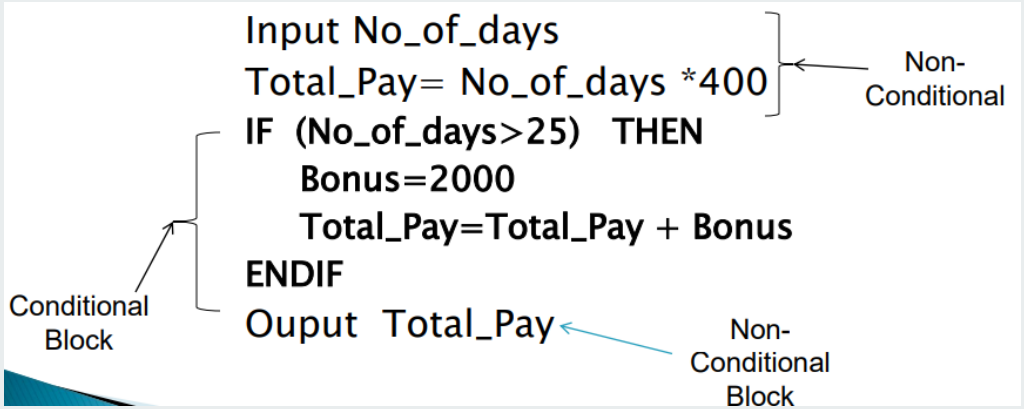


Semantics of the *if* statement

- The condition is first evaluated.
- If it is true, the sequence of statements in the body is executed, and then control passes to the next statement.
- If the condition is false, the statements in the body are skipped and control again passes to the next statement.

Pseudocode for example 5.1

- Problem: Consider a person being paid on a daily rate of Rs. 400. At the end of the month, he obtains his salary. Additionally, he obtains a bonus of Rs 2000 if he has worked for more than 25 days. We need a program to calculate his total pay at the end of a month



Syntax of if in Python

```
if condition:
    #block of codes
```

Program in Python for example 5.1

```
# Given daily rate and bonus condition
daily_rate = 400

# Taking input from the user
no_of_days = int(input("Enter the number of days worked in the month: "))

# Calculate the total pay
salary = no_of_days * daily_rate

# Calculate the bonus based on the condition
if no_of_days > 25:
        bonus = 2000
else:
        bonus = 0

total_pay = salary + bonus

# Display the result
print(f"The total pay at the end of the month is: Rs. {total_pay}")
```

Example 5.2

- Write a Program that takes, as input, a temperature in fahrenheit, converts it to celsius and displays:
  - The value of the temperature in Celsius
  - The message “It’s very hot” if the celsius value exceeds 35 .
  - The message “Program Over” at the end of the program
- Write the pseudocodes design of the program, draw the flowchart and write the Python program.

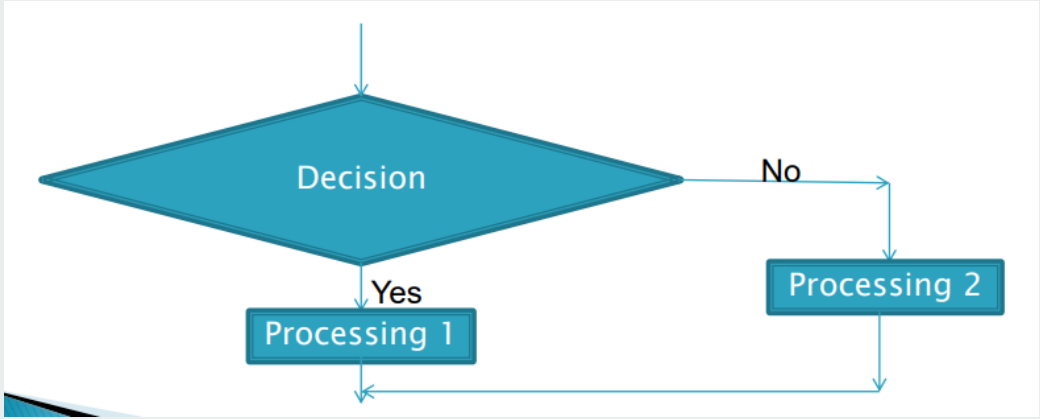
Note: Conversion: celsius = (fahrenheit –32) x 5/9

Exercise 5.3

- Modify the program in Exercise 5.2 so that it still displays the previous outputs, but additionally, if the temperature is below 12C, it displays “It’s quite cold”.
- Again write the pseudocode design, draw the flowchart and write the Python program

Using two way decisions

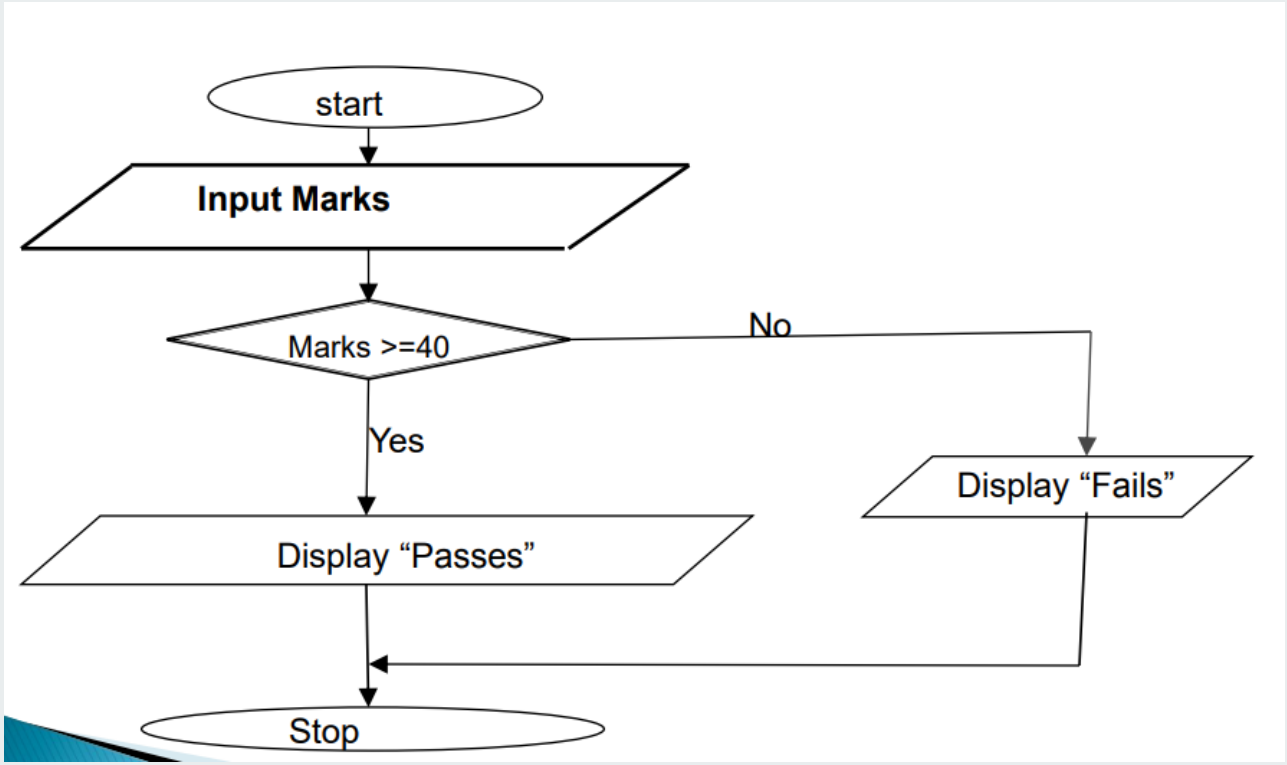
- The simple decision is used when we have no processing in case the condition evaluates to false.
- However, the more general case is when there is different processing depending on whether the condition is true or not.



Using two way decisions

- Consider the following example:
- Example 5.2
  - A program allows the input of a student's marks in a module. If the marks are 40 or above, the program displays "Passes". If the marks are below 40, the program displays "Fails".
  - Draw the flow chart.

Flowchart- example 5.2



If … else statement:

- An if statement can be combined with an else statement.
- An else statement contains the block of code that executes if the conditional expression in the if statement resolves to 0 or a false value.
- The else statement is an optional statement and there could be **at most one** else statement following an if .

The form of the if...else statement

```
if <condition>Then
        <Statement(s)>
 else
        <Statement(s)>
end if
```

- The condition is evaluated first. If the condition is true then the statement(s) in the *if* block are executed. Otherwise, the statement(s) in the *else* block is executed

Pseudocode if….else (example 5.2)

```
Input marks
If marks >= 40 Then
        output “Passes”
else
        output “Fails”
End if
```

Syntax of if..else in Python

```
if condition:
    #block of codes
else :
    #block of codes
```

Program in Python example 5.2

```
# Taking input from the user
marks = float(input("Enter the student's marks: "))

# Checking the condition and displaying result
 if marks >= 40:
     print("Passes")
 else:
     print("Fails")
```

Exercise:5.4

- A man stitches garments for a particular factory. If he stitches 100 pieces or more in a day, he is paid at the rate of Rs 35 per garment. Otherwise, he is paid at the rate of Rs 20 per garment.
- Write the pseudocode design, draw the flowchart, and write the Python program to calculate the total amount of money earned by the man for a given day.
- The input to the program should be the number of garments stitched

Multi way decision

- In some cases, our decision can take more that two possible paths. In such cases we have to make use of the elif statement

The form of if…elif and else statement

```
If <condition1> Then
        <statement(s)>
elif <condition2> Then
        <statement(s)>
else
        <Statement(s)>
End if
```

Example for illustration

```
if x == 3:
    print “X equals 3.”
elif x == 2:
    print “X equals 2.”
else:
    print “X equals something else.”
print “This is outside the ‘if’.”
```

Note:

- Use of indentation for blocks
- Colon (:) after boolean expression

Another if form

- An alternative if form returns a value
- This can simplify your code
- Example:
  - return x+1 if x < 0 else x -1
  - return ‘hold’ if delta==0 else sell if delta < 0 else ‘buy’

The elif statement in Python

- The if/elif statement is a chain of if statements. They perform their tests, one after the other, until one of them is found to be true.
- The *elif* statement allows you to check multiple expressions for truth value and execute a block of code as soon as one of the conditions evaluates to true.
- Like the else, the *elif* statement is optional. However, unlike else, for which there can be at most one statement, there can be **an arbitrary number of** elif statements following an if.

If elif else :Example 5.3

Consider the following problem.

- A student is asked to input his marks. If his score is >= 70, he passes with Grade A. Else If his score is below 70, but >= 40 he passes with Grade B. Else if his score < 40 he fails with Grade F. Write the pseudocode for the above problem to output the Grade of the student depending on his marks.

Pseudocode for example 5.3

```
Input marks
If marks >= 70 Then
        output “Grade A”
Else If marks >=40 Then
        Output “Grade B”
Else
        Output “Grade F”
End if
```

Program in Python

```
# Input the student's marks
marks = float(input("Enter the student's marks: "))
# Determine the grade based on the marks
if marks >= 70:
      grade = 'A'
elif marks >= 40:
      grade = 'B'
else:
      grade = 'F'
# Output the grade
print(f"The student's grade is: {grade}")
```

Key points

- Conditional statements, such as if, elif, and else, enable us to execute specific blocks of code based on certain conditions. They help us create branching paths in our program.