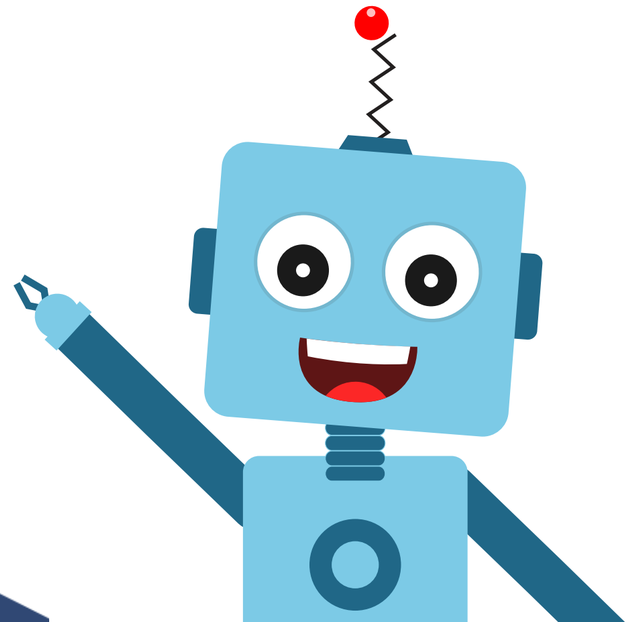
POWERED BY STEMpedia

Conditional Programming

Session 6



Topics covered

POWERED BY STEMpedia

1. Control Flow Structures
2. If-else in python
3. Logical Operators
4. Activities:
    1. Grade Calculator
    2. Is it a Triangle?

Control Flow Structure

Control Flow Structure

POWERED BY STEMpedia

There are three basic types of control structures in programming:

- **Sequential:** In a sequential control structure, the statements in a program are executed sequentially, i.e. step-by-step in the order that they are written.

- **Selection / Conditional:** A selection (conditional) control structure is used to test a condition in a program. This control structure takes a decision to execute one statement/operation over another statement/operation depending on the condition.

- **Iteration:** This control structure executes a set of statements a certain number of times till the mentioned condition is true. Loops are examples of iterative statements.

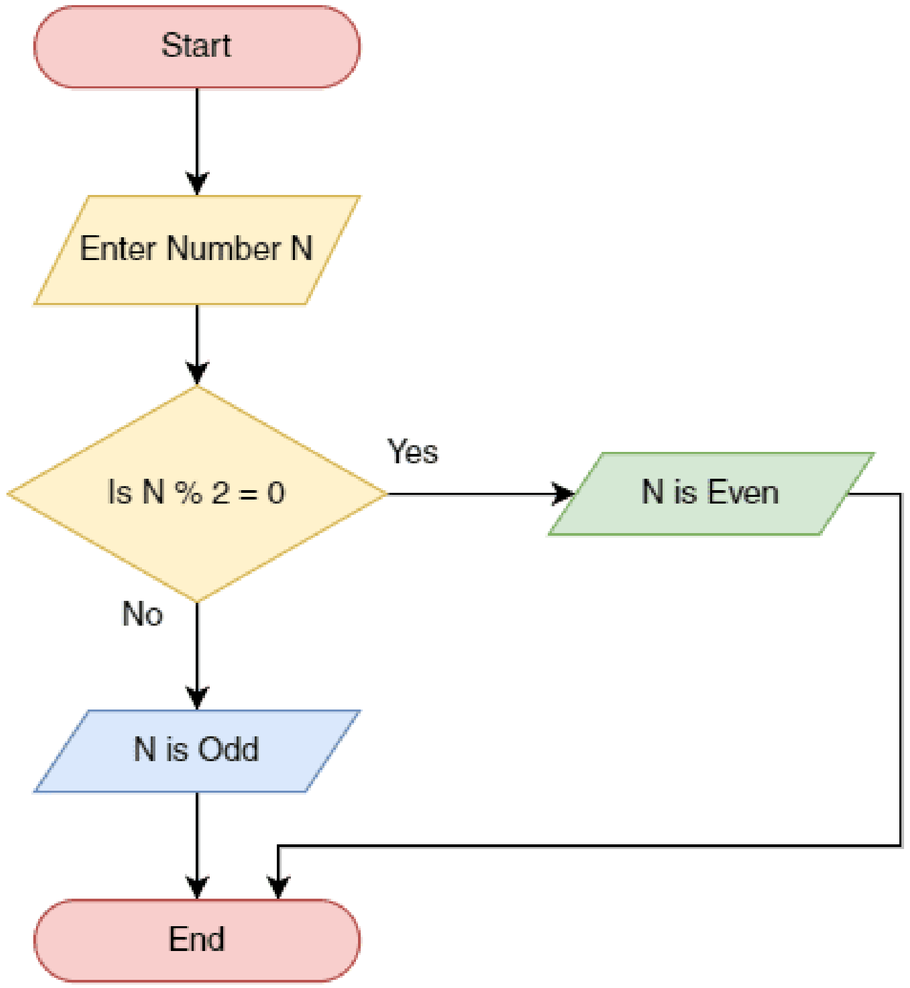
Understanding Conditional Statements

POWERED BY STEMpedia

For example, we decide whether a number is even or odd by dividing the number by 2. If the remainder is 1, we can say the number is odd.

The if-else statement requires three things.

- **Evaluation statement**: a boolean expression that is being checked.
- **Execution statement**: the operations that will be performed if the condition appears to be true.
- **Else execution block**: the operation that will be performed only if the evaluation statement is false.



Syntax of if-statement

POWERED BY STEMpedia

Option 1: if the condition is **True** (i.e. satisfied), the statement(s) written after **if** (i.e. STATEMENT-BLOCK 1) is executed, otherwise statement(s) written after **else** (i.e. STATEMENT-BLOCK 2) is executed. Remember else clause is optional.

```
if condition:
STATEMENTs - BLOCK 1
else:
STATEMENTs - BLOCK 2
```

Syntax of if-statement

POWERED BY STEMpedia

Option 2: Here, we have used '**elif**' clause instead of "else". elif combines if else-if else statements to one if elif … else. You may consider elif to be an abbreviation of **else if**.

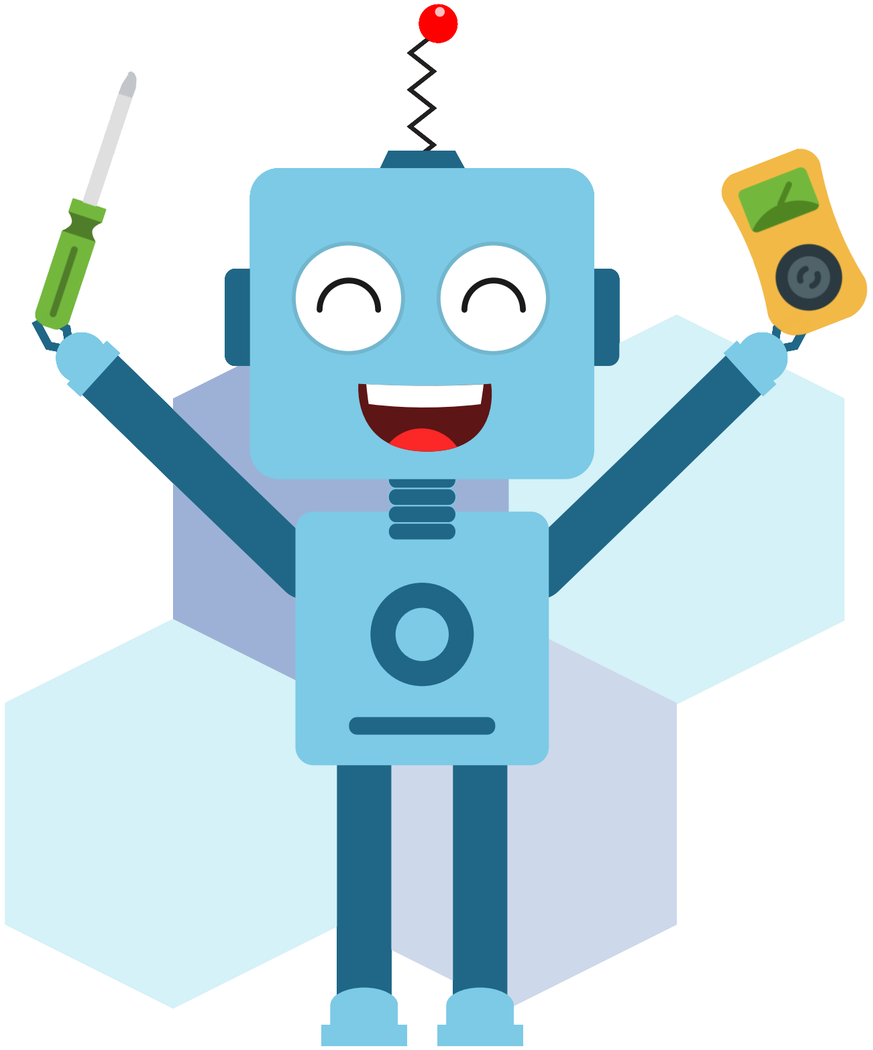
```
if condition:
  STATEMENTs - BLOCK 1
elif:
  STATEMENTs - BLOCK 2
else:
  STATEMENTs - BLOCK 3
```

ACTIVITY

POWERED BY STEMpedia

Grade Calculator

Let’s create code takes input from the user in the form of marks, and based on the value of marks, assigns a grade to the student using conditional statements. Finally, it prints the grade for the given marks.



Let's Code

POWERED BY STEMpedia

- In this activity, user has to enter their marks and stores the input as an integer value in the variable marks.

```
#Taking marks as an input

marks = int(input("Enter the marks: "))
```

- Then we define a condition to check if the marks are greater than or equal to 85. If the condition is true, the variable grades is assigned the value "A".
- Further we define another condition to check if the marks are less than or equal to 84 and greater than or equal to 60. If this condition is true, the variable grades is assigned the value "B".
- Then we define another condition to check if the marks are less than or equal to 59 and greater than or equal to 40. If this condition is true, the variable grades is assigned the value "C".
- Next we define another condition to check if the marks are less than or equal to 39 and greater than or equal to 30. If this condition is true, the variable grades is assigned the value "D".

Let's Code

POWERED BY STEMpedia

- In last it executed if none of the previous conditions are true. If this condition is true, the variable grades is assigned the value "F.
- Add the conditions to determine the grades using the if-else statements inside the function.

```
#Defining condition for grades
if marks >= 85:
  grades = "A"
elif marks<=84 and marks>=60:
  grades = "B"
elif marks<=59 and marks>=40:
  grades = "C"
elif marks<=39 and marks>=30:
  grades = "D"
else:
  grades = "F"
```

Let's Code

POWERED BY STEMpedia

- Then we print the message "The grade for the given marks is:" along with the value of the variable grades

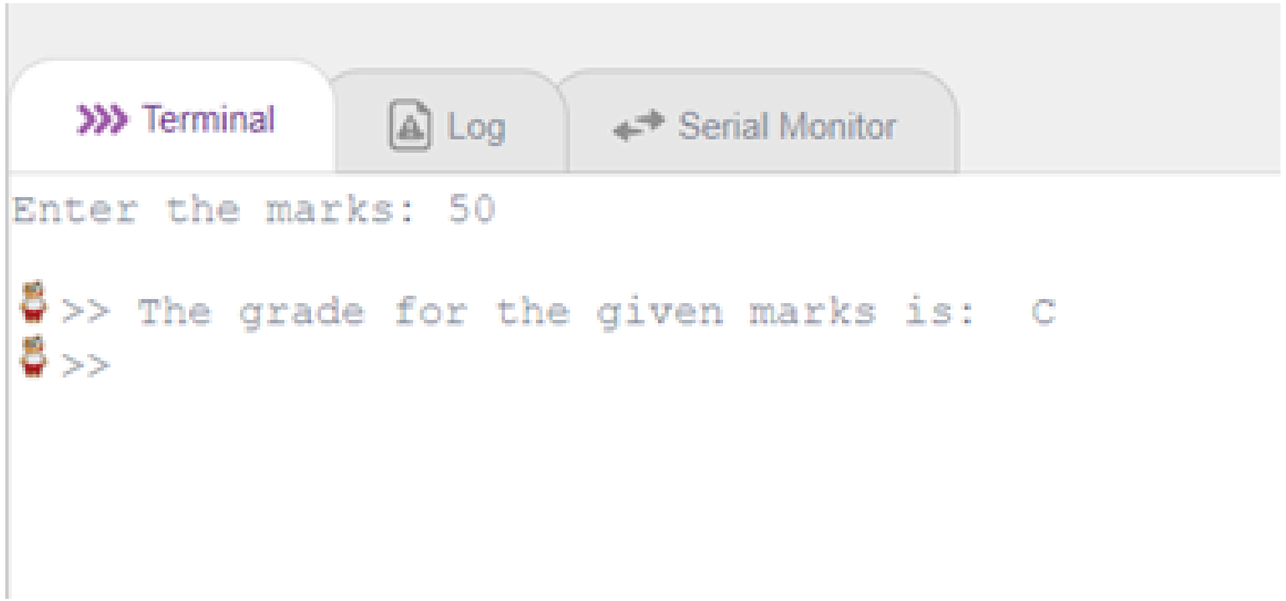
```
print("The grade for the given marks is: ",grades)
```

Final Code

POWERED BY STEMpedia

```
#Taking marks as an input

marks = int(input("Enter the marks: "))

#Defining condition for grades

if marks >= 85:

  grades = "A"

elif marks<=84 and marks>=60:

  grades = "B"

elif marks<=59 and marks>=40:

  grades = "C"

elif marks<=39 and marks>=30:

  grades = "D"

else:

  grades = "F"

print("The grade for the given marks is: ",grades)
```

Final Output



Nested Condition

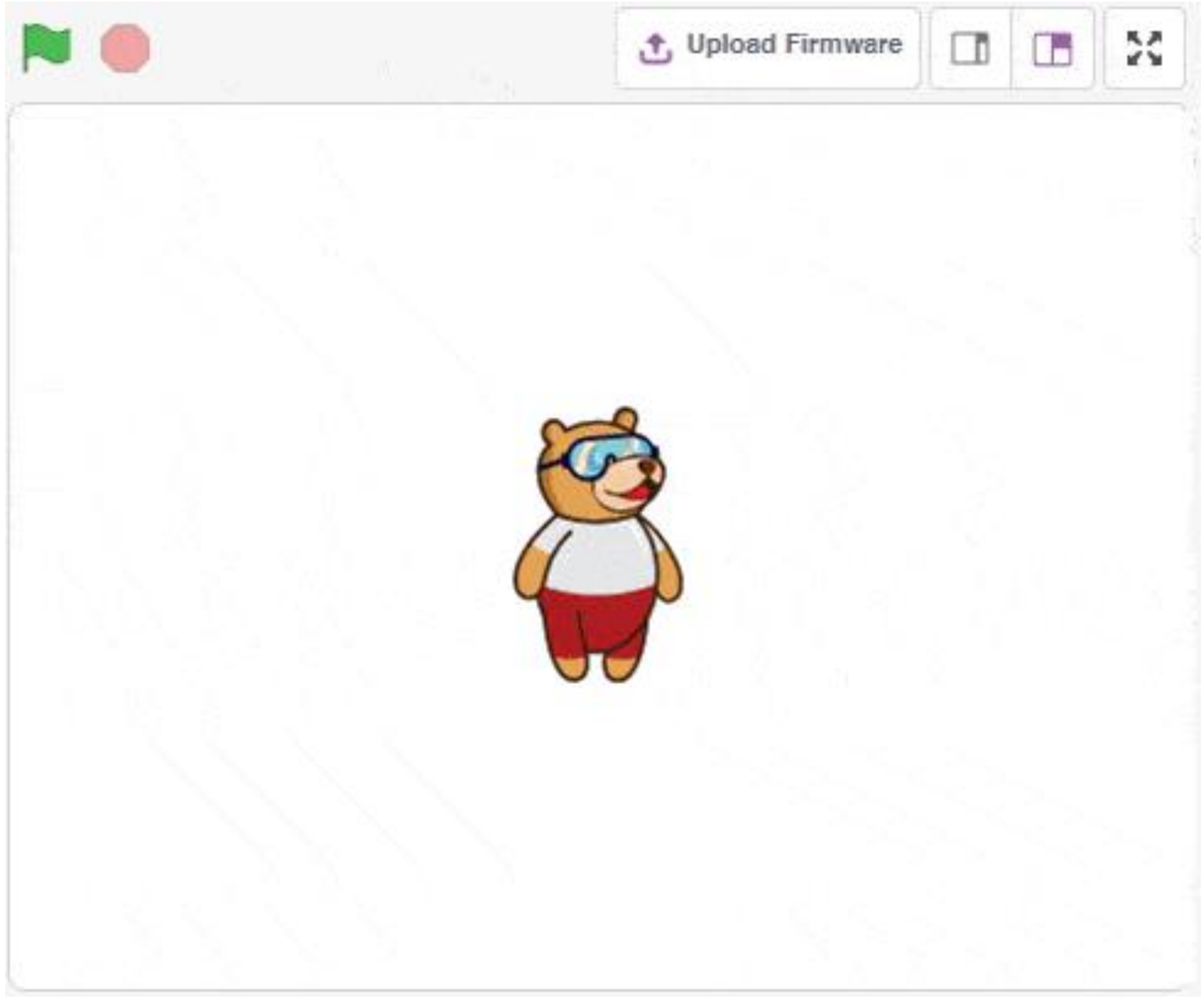
POWERED BY STEMpedia

It is possible to have a condition within another condition. Such conditions are known as Nested Conditions.

Example:

```
sprite = Sprite('Tobi')
def compareNumbers(x, y):
  if x == y:
    sprite.say("Both are Equal", 2)
  else:
    if x<y:
      sprite.say("Y is bigger than X", 2)
    else:
      sprite.say("X is bigger than Y", 2)
  return
sprite.say("Case 1", 1)
compareNumbers(4, 4)
sprite.say("Case 2", 1)
compareNumbers(5, 6)
sprite.say("Case 3", 1)
compareNumbers(6, 5)
```

Output



Logical Operators

POWERED BY STEMpedia

- Logical operators are fundamental statements that can be used to build a decision-making capability in your code.
- When we need to make our decision based on two or more checks, it is necessary to have a combination of logical operators.

The three most important logical operators are:

- AND Operator
- OR Operator
- NOT Operator

AND Operator

POWERED BY STEMpedia

- The AND operator is used to see if two or more conditions are true.

- If all the conditions are true, the AND operator returns TRUE. If any one of the conditions fail, the AND operator returns FALSE.

- In python AND operator is denoted by and keyword. Some other programming languages use “&&” as an AND operator.

OR Operator

POWERED BY STEMpedia

- The OR operator is used to see if either one of two or more conditions is TRUE.
- If any of the conditions is true, the OR operator returns TRUE. If all the conditions fail, the OR operator simply returns FALSE.
- In python syntax OR operator is denoted by or keyword. Some other programming languages use “||” as an OR operator.

NOT Operator

POWERED BY STEMpedia

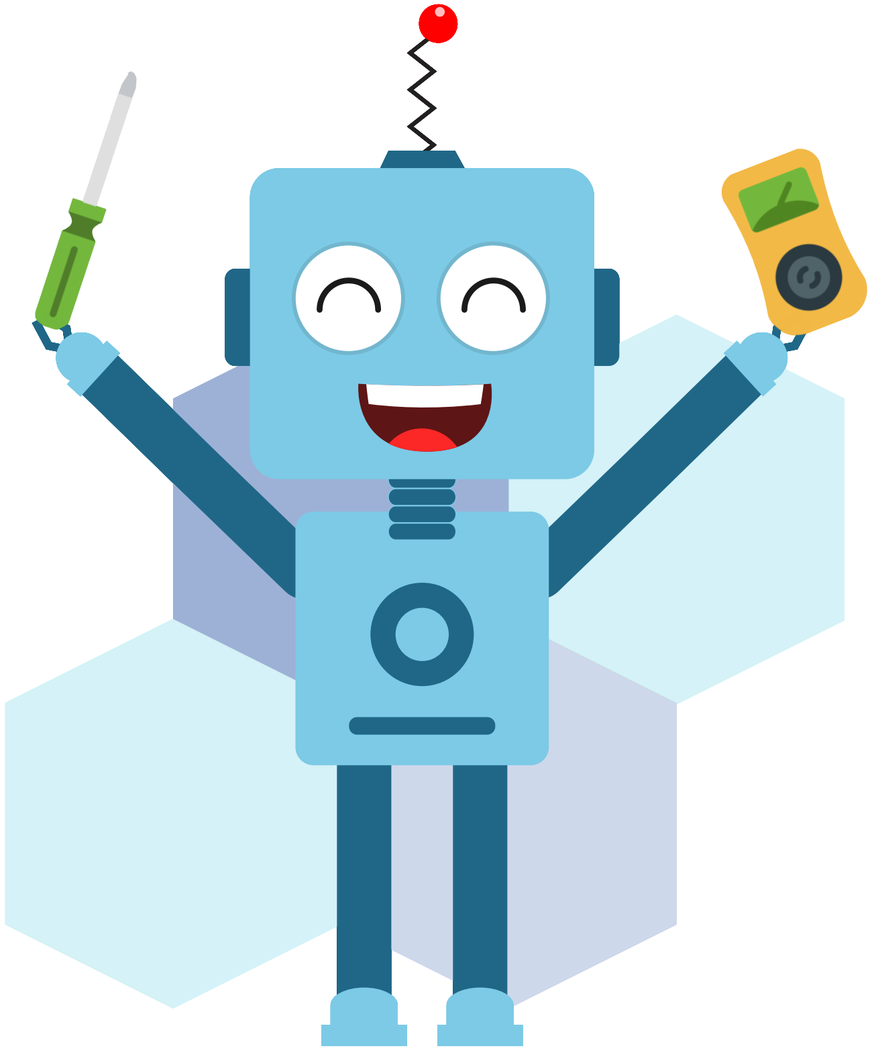
- We use the NOT operator to reverse or negate a condition.
- If the condition is true, NOT will return false and vice versa.
- In python programming, we use no keyword to denote not operator. Some other programming languages use “!” as NOT operator.

ACTIVITY

POWERED BY STEMpedia

Is it a triangle

Let’s write code to find out is the given tringle is perfect tringle or not.



Let's Code

POWERED BY STEMpedia

- Define the sprite.

```
sprite = Sprite('Tobi')
```

- Create a function isTriangle with 3 side length inputs – A, B and C.

```
def isTriangle(A, B, C):
```

- Execute the logic: Check if the sum of two sides is greater than the remaining side. If A + B > C and B + C > A and A + C > B, then it is a triangle

```
if ((A+B) > C) and ((B+C) > A) and ((A+C) > B):
```

Let's Code

POWERED BY STEMpedia

- Now, check whether the triangle is valid or not :

```
sprite = Sprite('Tobi')
def isTriangle(A, B, C):
  if ((A+B) > C) and ((B+C) > A) and ((A+C) > B):
    sprite.say("Its a triangle", 2)
  else:
    sprite.say("Its not a triangle", 2)
  return
sprite.say("Case 1", 1)
isTriangle(15, 20, 25)
```

Final Output

It is a Triangle

POWERED BY
STEMpedia

www.ai.thestempedia.com