Basic Programming II:

Iteration

Quote of the day

"The only way to learn a new programming language is by writing programs in it."

Table of Content

What will We Learn Today?

1. Introduction to Iteration
2. Flowchart of Iteration
3. While Statement
4. For Statement
5. Nested Iteration
6. Loop Control

Introduction

For each package, they will deliver it to each customers until there is no more packages left in the box.

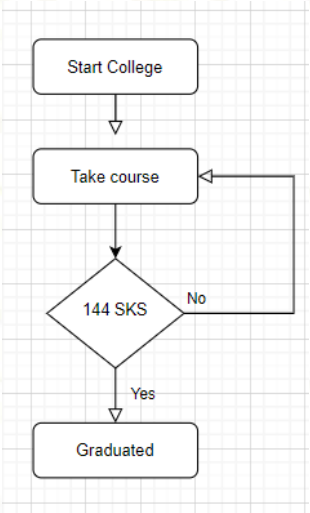
https://mobilbox.id/

Introduction

College students who go to their college and aiming to finish 144 SKS

Introduction

College students who go to their college and aiming to finish 144 SKS



Iteration / Loop

In programming, iteration is a sequence of instruction that is continually repeated. The programming language will tracks what step of the iteration is currently being executed. Basically, the iteration will repeated until the termination condition is met.

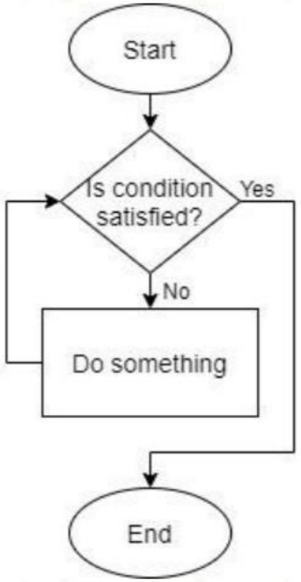
In Python, we could do these types of iteration:

1. Iterate until a certain condition is reached.
2. Iterate until certain number of times.
3. Iterate through elements in a list or array.

Flowchart

Every iteration must have these two parts:

1. Condition checking
2. Iteration process



← Condition checking

Iteration process →

While Loop

In Python, While Loops is used to execute a block of statements repeatedly until a given condition is satisfied and when the condition becomes false, the line after the loop in the program is executed.

While loop falls under the category of indefinite iteration. Indefinite iteration means that the number of times the loop is executed isn’t specified explicitly in advance.

```
while <expression>:
      <statement>
```

For Loop

For loops, in general, are used for sequential traversal. For example: traversing a list or string or array etc. It falls under the category of definite iteration. Definite iterations means the number of repetitions is specified explicitly in advance

```
for i in iterable:
        <statement>
```

Nested Loop

Nested loop is a loop inside a loop.

The inner loop will be fully executed for each iteration of the outer loop.

```
while <expression>:
       <statement>
       while <expression>:
             <statement>
```

```
for i in iterable:
       <statement>
       for j in iterable_2:
             <statement>
```

Loop Control

Sometimes, instead of following our loop condition, we just want to do another thing besides of our defined loop condition. These statements might help.

- break, will terminate the current loop and continue to the next line after the loop
- continue, will reject the remaining statements in the current loop and continue to the next iteration
- pass, used when a statement is required syntactically but do not want any code to be executed

Things We've Learned

Congratulations! You have finished learning about conditions.

1 Introduction to Iteration

2 Flowchart of Iteration

3 While Statement

4 For Statement

5 Nested Iteration

6 Loop Control

DigitalSkola
Uncover The World Of Digital Skills With Us

Thank YOU