UNIT-I

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT- Basara, T.S, IND.

SYLLABUS

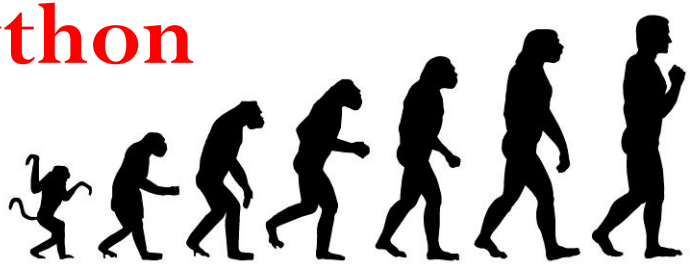
- **UNIT – I [12 Lectures]**

- Introduction to Python Programming
- Algorithmic Analysis
- Downloading and Installing Python
- Python basics: Datatypes, variables and identifiers, comments, variable types, get input from user, equal sign, statements, assignments, expressions
- Operators – Arithmetic operators, precedence, Relational operators, logical operators, Membership operators
- Arrays Vs Lists

python
Programming Language

IIIT BASARA

Evolution of Python



Python was developed by Guido Van Rossum in early 1980s at National Research Institute for Mathematics and Computer Science, Netherlands.

Python language is named after the television show Monty Python's Flying Circus and many examples and tutorial include jokes from the show

Derives its features from many languages likeJava, C++, ABC, C, Modula3, Smalltalk, Algol-68, Unix shell and other scripting languages.

Available under the GNU General Public License (GPL) – Free and open-source software's

Major implementations of Python are Cpython, IronPython, Jpython,MicroPython,PyPy

python
Programming Language

Introduction to Python Programming

•Python is a widely used high-level programming language for general-purpose programming, created by Guido van Rossum.

•python first version is 1.0 and latest version is 3.6

•Python interpreters are available for many operating systems, allowing Python code to run on a wide variety of systems.

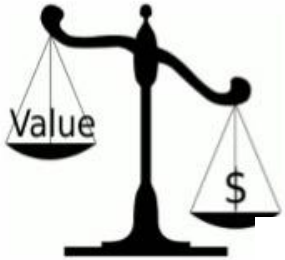
•Python uses whitespace indentation to delimit blocks rather than curly braces or keywords. An increase in indentation comes after certain statements.

•Python file extension is .py

python
Programming Language

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

The primary factors cited by Python users seem to be the following:



->Software Quality

->Developer Productivity

->Program Portability

->Support Libraries

->Component Integration

->Enjoyment

python
Programming Language

Software Quality:

Python code is designed to be readable, reusable and maintainable

Python has deep support for more advanced software reuse mechanisms, such as object-oriented programming(OOP)

Developer Productivity:

Python boosts developer productivity many times beyond compiled or statically typed languages such as C, C++, and Java.

Python code is typically one-third to one-fifth the size of equivalent C++ or Java code.

It is less to type, less to debug, and less to maintain

Python programs also run immediately, without the lengthy compile and link steps of some other tools

Program Portability:

Most Python programs run unchanged on all major computer platforms. Porting Python code between Linux and Windows

Python offers multiple options for coding portable graphical user interfaces, database access programs, web-based systems

IIIT BASARA

Support Libraries:

Python comes with a large collection of prebuilt and portable functionality, known as the *standard library*

This library supports an array of application-level programming tasks, from text pattern matching to network scripting

Python can be extended with both homegrown libraries and a vast collection of third-party application support software

Component Integration:

Python scripts can easily communicate with other parts of an application, using a variety of integration mechanisms such as product customization and extension tool [C,C++,Java,Beans. COM, .NET, Networks, SOAP,XML , CORBA]

Enjoyment:

Python's ease of use and built-in toolset, it can make the act of programming more pl[illegible]re

python

Programming Language

Python is simple

```
print "Hello World!"
```

Python

```
#include <iostream.h>
int main()
{
cout << "Hello World!";
}
```

C++

```
public class helloWorld
{
    public static void main(String [] args)
    {
    System.out.println("Hello World!");
    }
}
```

Java

python
Programming Language

Python Features Include:

–Beginners Language

-Simple & Easy to Learn

-- Free and Open Source

-- Multi Purpose [Web,GUI, Scripting etc..]

--High Level

-- Embedded

–Extensive Standard Library

–Cross Platform Compatibility

–Interactive Mode

-Interpreted

-- Object Oriented

–Portable, Extendable

–Databases and GUI Programming

–Scalable and Dynamic in nature

–Automatic Garbage Collection

python
Programming Language

7/14/2017

Applications of Python

- 1. GUI-based desktop applications
- 2. Web frameworks and applications
- 3. Enterprise and business applications
- 4. Operating Systems
- 5. Language Development
- 6. Prototyping
- 7. Data Science & Analysis

python
Programming Language

IIIT BASARA

Python Users:

- Google makes extensive use of Python in its web search system, and employs Python's creator.
- The YouTube video sharing service is largely written in Python.
- The popular BitTorrent peer-to-peer file sharing system is a Python program.
- Intel, Cisco, Hewlett-Packard, Seagate, Qualcomm, and IBM use Python for hardware testing.
- Industrial Light & Magic, Pixar, and others use Python in the production of movie animation.
- JPMorgan Chase, UBS, Getco, and Citadel apply Python for financial market forecasting.
- NASA, Los Alamos, Fermilab, JPL, and others use Python for scientific programming tasks.
- iRobot uses Python to develop commercial robotic vacuum cleaners.
- ESRI uses Python as an end-user customization tool for its popular GIS mapping products.
- The NSA uses Python for cryptography and intelligence analysis.
- The IronPort email server product uses more than 1 million lines of Python code to do its ...
- The One Laptop Per Child (OLPC) project builds its user interface and activity model in Python.

python
Programming Language

Algorithmic Analysis

•If problem solving is a central part of computer science, then the solutions that you create through the problem solving process are also important. In computer science, we refer to these solutions as **algorithms**.

•An **algorithm** is a step by step list of instructions that if followed exactly will solve the problem under consideration.

•Programming is a skill that allows a computer scientist to take an **algorithm** and represent it in a notation (a program) that can be followed by a computer

•**Algorithm** is a method of representing the step by step logical procedure for solving a problem, where each step can be termed as an instruction

Every algorithm must have to satisfy the following properties:

1. **Finiteness** : Every alg. Must get terminated in known no. of steps
2. **Definiteness** : Each and every step of an alg. Must be easily and clearly understood
3. **Effectiveness** : Every step must be effective in the sense of programming the code & with in span of time
4. **Generality** : The alg. Must be complete in itself so that it can be used to solve all similar kind of problems for i/p data
5. **Input/Output** : Every alg. Must take Zero or More i/p data & produce one or more O/p

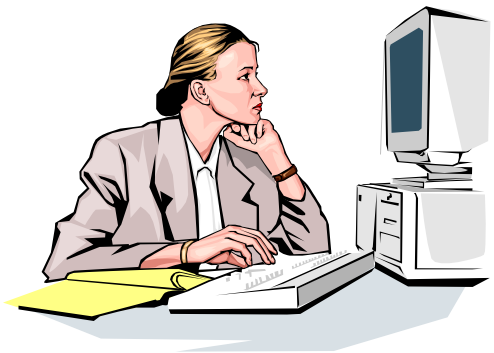
In general , the steps in an alg., can be divided into three Categories/Operations:

1. **Sequence -** A series of steps can be performed one after the other
2. **Selection** – Making a choice from multiple available options
3. **Iteration** – Performing repetitive tasks

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT- Basara, T.S, IND.

Example:

Algorithm for adding two integer numbers :



Step1: Begin

Step2 : Initialize the variables sum with Zero

Step3 : Read two values

Step4 : Add the values & store in variable sum

Step5 : Print sum

Step6 : End

Flowchart:

It is a graphical representation of logical sequence of a given problem ie., it is a pictorial representation of an algorithm. It uses some symbols for specific tasks

Programming Development Lifecycle

The program development life cycle is a set of steps or phases that are used to develop a program in any programming language.

Generally, program development life cycle contains 6 phases, they are as follows….

1. **Problem Definition/ Problem Specification**
2. **Problem Analysis / Outlining the solution**
3. **Algorithm Development / Selecting & representing the Alg.**
4. **Coding & Documentation/ Programming the algorithm**
5. **Debugging (Removing Errors)**
6. **Testing & Validation**
7. **Documentation & Maintenance**

python
Programming Language

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT- Basara, T.S, IND.

Python Tool Introduction

There are two ways to use the Python interpreter:

1. shell mode

In shell mode, you type Python expressions into the **Python shell**, and the interpreter immediately shows the result.

Ex:

```
>>> 2 + 3
5
>>>
```

The >>> is called the **Python prompt**.

The interpreter uses the prompt to indicate that it is ready for instructions

*2. **program mode***

create a source code file named **firstprogram.py**

Ex:

```
print("My first program adds two numbers, 2 and 3:")
print(2 + 3)

My first program adds two numbers, 2 and 3:
5
```

Write a Program to display Hello World! Message on the screen

>>> print(“ Hello, World! ")

Hello World!

Note: The quotation marks in the program mark the beginning and end of the value. They don’t appear in the result.

```
print”Hello World!”
print”Hello Again”
print”I like typing this”
print”This is funny”
print’yay! Its Printing.’
print”I’d much better than you ‘not’.”
print’I “said” do not touch this.’
```

English like commands

```
>>> name =“RGUKT-Basar”
>>> print (name)
```

One-liners

Ex: Swapping 2 variables

Ex: c : Swap x&y

```
int temp =x;
x=y;
y=temp;
```

Ex: Python

```
>>> x,y=y,x
```

Dynamically Typed Language

Ex:

C:

```
int  x=1;
x =(int)x/2;
```

O/p: 0 [can never be equal to 0.5]

Where as in Python

```
>>>x=1
>>> x=x/2
```

Now it is equal to 0.5 [Floating Point Variable]

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

Examples: Python as Calculator

>>>print(“5+2”, 5+2)	**7**
>>>print(“5-2”,5-2)	**3**
>>>print(“5*2”,5*2)	**10**
>>>print(“5/2”,5/2)	**2.5**
>>>print(“5%2”,5%2)	**1**
>>>print(“52”,5**2)**	**25**
>>>print(“5//2”,5//2)	**2**
>>>print(“1+2-3*2= ”, 1+2-3*2)	**-3**
>>>print(“(1+2-3)*2=“,(1+2-3)*2)	**0**

Datatypes

Numeric Values

- Numbers come in two ways

  int – integers

  float – fractional numbers

Ex: 172 , -5 , 23453654 are values of type **int**

42.63 , -0.02 , 37.33232 are values of type **float**

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

int Vs float

- Internally , a value is stored as a finite sequence of 0's and 1's (binary digits, or bits)
- For an int, this sequence is read off as a binary number
- For a float, this sequence breaks up into a mantissa and exponent

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

Operations on numbers

- Normal arithmetic operators : +, - ,* , /
- Note that / always produces a float
- 7/3.5 is 2.0 , 7/2 is 3.5
- Quotient and Remainder : // and %
- 9//5 is 1, 9%5 is 4
- Exponentiation : **
- 3**4 is 81 [3*3*3*3]

- log( ), sqrt( ), sin( ), …
- We must include math " library"
    - from math import*

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

Names, Values and Types

•Names can be assigned values of different types as the program evolves

```
i = 5    #  i is int
i = 7*1   # i is still int
j = i/3  # j is float,  / creates float
```

•**type(e)** returns type of expression e

Ex:

```
>>> i =5
>>> type(i)
<class 'int'>
>>> j= 7.5
>>> type(j)
    < class 'float'>
```

```
>>> i = 2*j
>>> i
15.0
>>> type(i)
<class 'float')
```

Note: Not a good style to assign mixed type values to same name

Boolean Values : bool

- True , False
- Logical operators : not, and , or

  - not True is False, not False is True
  - X and y is True if both of x,y are True
  - X or y is True if at least one of x,y is True

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

Comparisons

x == y ,
a ! = b,
z < 17*5 ,
n>m,

Combine using logical operators

n>0 and m%n ==0

Assign a boolean expression to a name

divisor = (m%n ==0)

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

- variables

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

Identifiers,

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

Comments in Python

•A **comment** in a computer program is text that is intended only for the human reader - it is completely ignored by the interpreter. In Python, the # token starts a comment. The rest of the line is ignored. Here is a new version of *Hello, World!*.

```
#-----------------------------------------------
print("Hello, World!")     # Isn't this easy!
```

variable types

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

get input from user

A **program** is a sequence of instructions that specifies how to perform a computation. The computation might be something as complex as rendering an html page in a web browser or encoding a video and streaming it across the network. It can also be a symbolic computation, such as searching for and replacing text in a document or (strangely enough) compiling a program.

The details look different in different languages, but a few basic instructions appear in just about every language.

input

Get data from the keyboard, a file, or some other device.

output

Display data on the screen or send data to a file or other device.

math and logic

Perform basic mathematical operations like addition and multiplication and logical operations like and, or, and not.

conditional execution

Check for certain conditions and execute the appropriate sequence of statements.

repetition

Perform some action repeatedly, usually with some variation.

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

equal sign

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

statements

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

Assignments

- Assign a **value** to a **name**

  i = 5

  j = 2*i

  j = j + 5
- Left hand side is a name
- Right hand side is an expression
- Operations in expression depend on type of value

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

expressions

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

- Operators – Arithmetic operators, precedence, Relational operators, logical operators, Membership operators

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

Arrays Vs Lists

K.RaviKanth, Ast.Prof., Dept. of CSE, IIIT- RGUKT-Basara, T.S, IND.

IIIT BASARA

Infosys | Campus Connect

For Details Contact Me @ :

9247448766

ravikanth27787@gmail.com

python

Programming Language