



MKP3303 SCIENTIFIC COMPUTATION

Gobithaasan Rudrusamy
Faculty of Ocean Engineering Technology & Informatics,
University Malaysia Terengganu.

email: gr@umt.edu.my

<https://sites.google.com/site/gobithaasan/LearnTeach>

Content:

Topik 1: Pengenalan

- Asas komputeran saintifik
- Sejarah bahasa pengaturcaraan
- Pengenalan kepada perisian saintifik
(Mathematica dan Python)

Topic 1: Introduction

- Basic scientific computing
- History of programming
- Introduction to scientific software (Mathematica and Python)

Minggu	Tajuk: BM	BI	Practical Lab	Report (CLO3)
1 & 2	Topik 1: Pengenalan <ul style="list-style-type: none">• Asas komputeran saintifik• Sejarah bahasa pengaturcaraan• Pengenalan kepada perisian saintifik (Mathematica dan Python)	Topic 1: Introduction <ul style="list-style-type: none">• Basic scientific computing• History of programming• Introduction to scientific software (Mathematica and Python)		
3 & 4	Topik 2: Nombor, Ungkapan dan Fungsi <ul style="list-style-type: none">• Integer, nombor nisbah dan tak nisbah• Nombor titik apungan• Nombor kompleks• Ungkapan, pernyataan umpukan, kesamaan, fungsi• Ralat limpah atas, limpah bawah dan pembundaran	Topic 2: Numbers, Expressions and Functions <ul style="list-style-type: none">• Integers, rational numbers and irrational numbers• Floating point numbers• Complex numbers• Expressions, assignment statements, equalities, functions• Overflow error, underflow error and rounding-off error		
5 & 6	Topik 3: Matriks dan Vektor <ul style="list-style-type: none">• Tatasusunan• Vektor lajur dan baris• Operasi vektor dan matriks	Topic 3: Matrices and Vector <ul style="list-style-type: none">• Lists• Column and row vector• Vector and matrix operation		
	Penilaian Formatif Ujian: 15% (CLO1:15%)			Ujian merangkumi: Topik 1, 2 dan 3.
7 & 8 & 9	Topik 4: Visualisasi, jenis Data Abstrak <ul style="list-style-type: none">• Jenis-jenis Data• Plot data dua dimensi• Plot data tiga dimensi• Visualisasi data berdimensi tinggi• Data animasi	Topic 4: Types of Data and Visualization <ul style="list-style-type: none">• Data types• Simple 2-dimensional plots• 3-dimensional surface plots• Higher dimension data visualisation• Animated data		
	Penilaian Formatif Projek 1: 15% (CLO2)			Projek 1 merangkumi: Topik 1, 2 3, dan 4.
10 & 11	Topik 5 : Penyelesaian Persamaan: Kamiran <ul style="list-style-type: none">• Alat pengamiran• Pengamiran tak tentu• Pengamiran tentu• Kamiran sebagai fungsi	Topic 5: Solution of Equation: Integration <ul style="list-style-type: none">• Integral tools• Indefinite integral, definite integral, integration as function		
12, 13 & 14	Topik 6: Penyelesaian Persamaan:Pembezaan <ul style="list-style-type: none">• Pembezaan: had dan terbitan• Penyelesaian persamaan tunggal• Penyelesaian sistem persamaan	Topic 6: Solution of Equation: Differentiation <ul style="list-style-type: none">• Differentiation: Limit and derivative• Solution for single variable equation• Solution for systems of equation		
	Penilaian Formatif Projek 2: 15% (CLO2)			Projek 2 merangkumi: Topik 1, 2 3, dan 4.
	Penilaian Sumatif Peperiksaan akhir: 40% (CLO1:10% & CLO2:30%)			Peperiksaan merangkumi keseluruhan topik

https://en.wikipedia.org/wiki/Computational_science

Computational science, also known as scientific computing or scientific computation (SC), is an advanced computing capabilities to understand and solve complex problems.

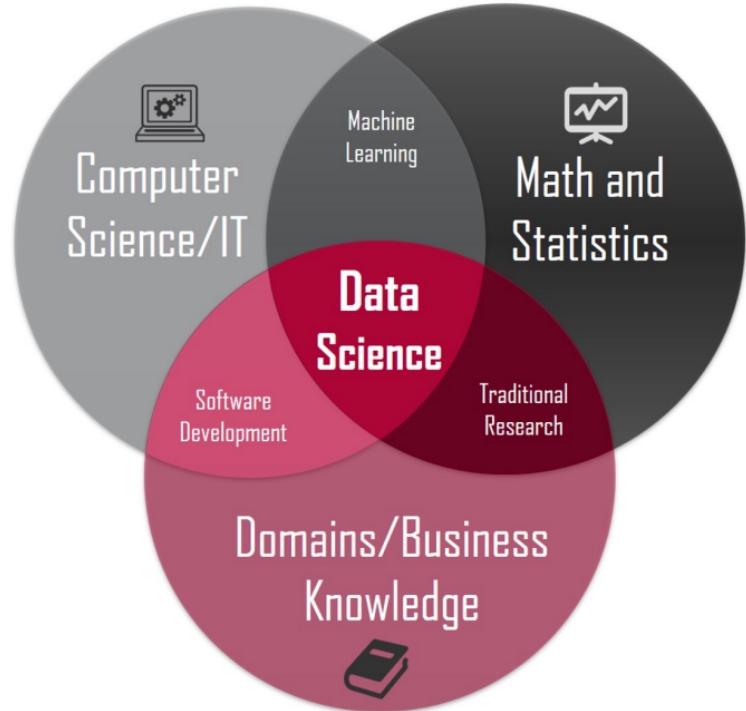
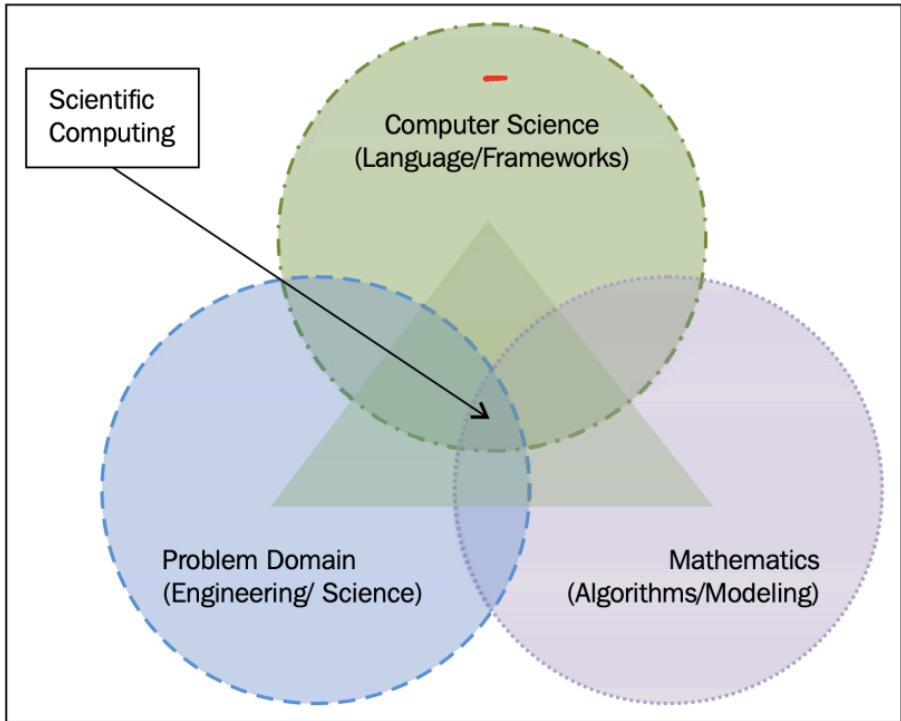
it involves the development of models and simulations to understand natural systems.

Algorithms (analytical and numerical): mathematical models, computational models, and computer simulations developed to solve science (e.g., biological, physical, and social), engineering, and humanities problems.

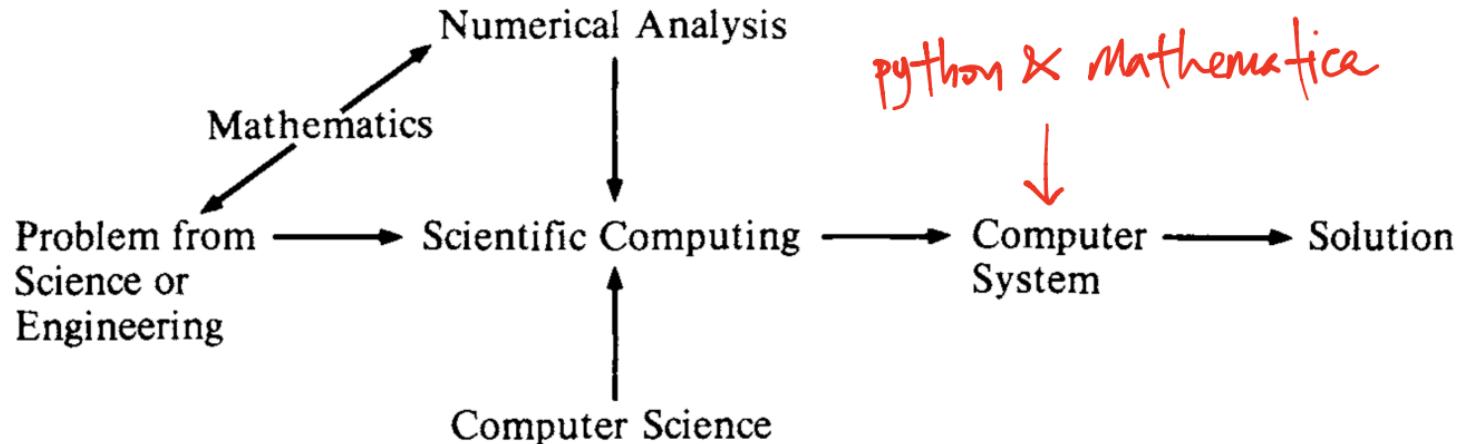
The field is different from theory and laboratory experiment which are the traditional forms of science and engineering.

The scientific computing approach is to gain understanding, mainly through the analysis of mathematical models implemented on computers.

Scientists and engineers develop computer programs, application software, that model systems being studied and run these programs with various sets of input parameters.



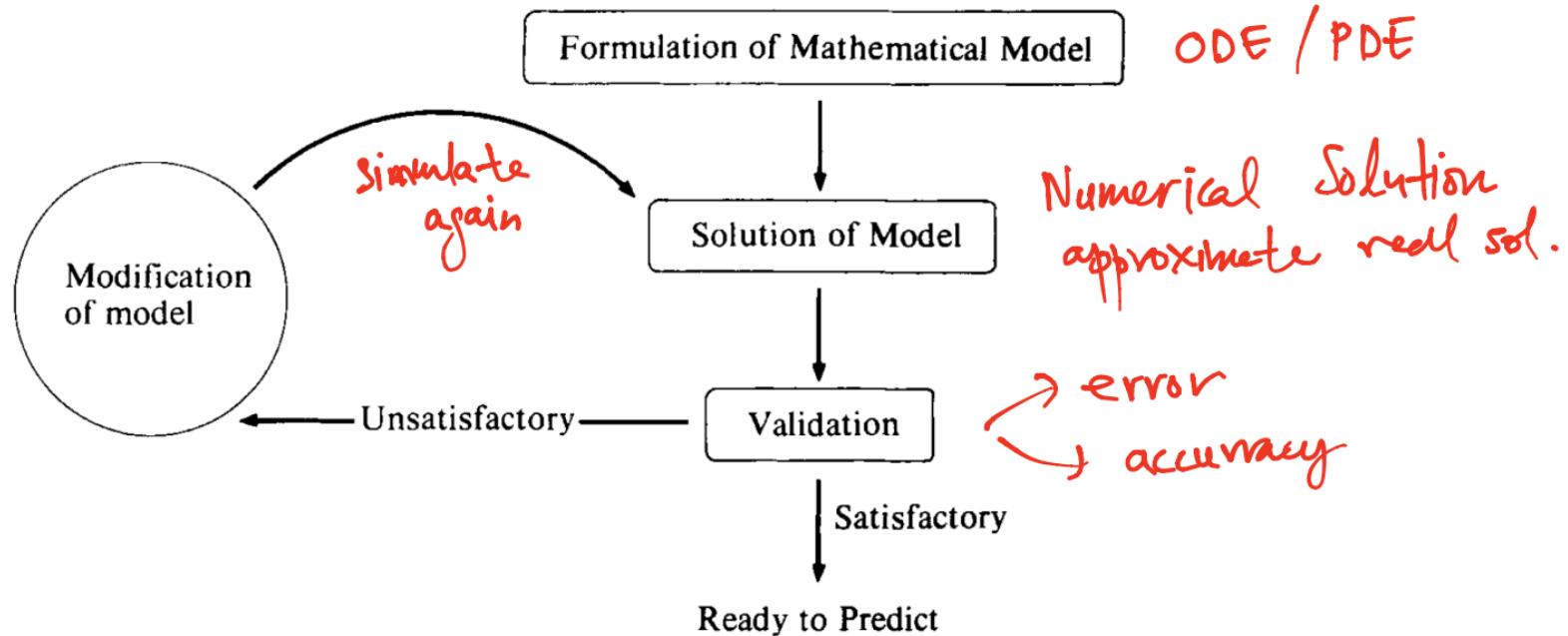
Definition of scientific computing



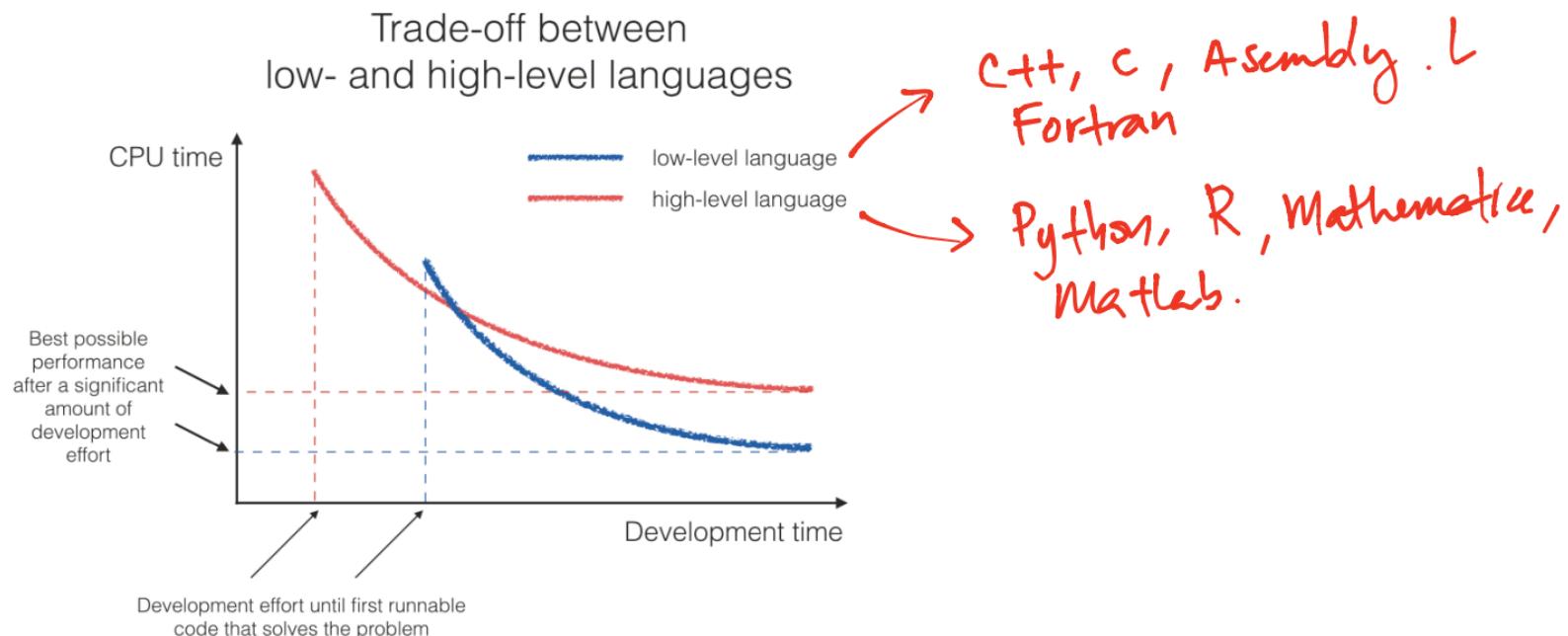
"Scientific computing is the collection of tools, techniques and theories required to solve on a computer the mathematical models of problems in science and engineering."

– Gene H. Golub and James M. Ortega, Scientific Computing and Differential Equations. An Introduction to Numerical Methods (1991, Elsevier Inc).

Mathematical Modeling and Solution Process



Intro to Programming: low- versus high-level languages



Robert Johansson, Numerical Python: Scientific Computing and Data Science Applications with Numpy, SciPy and Matplotlib (2019, Apress)

Python Versus C++

● Python
Search term

● The C++ Programming Langu...
Search term

+ Add comparison

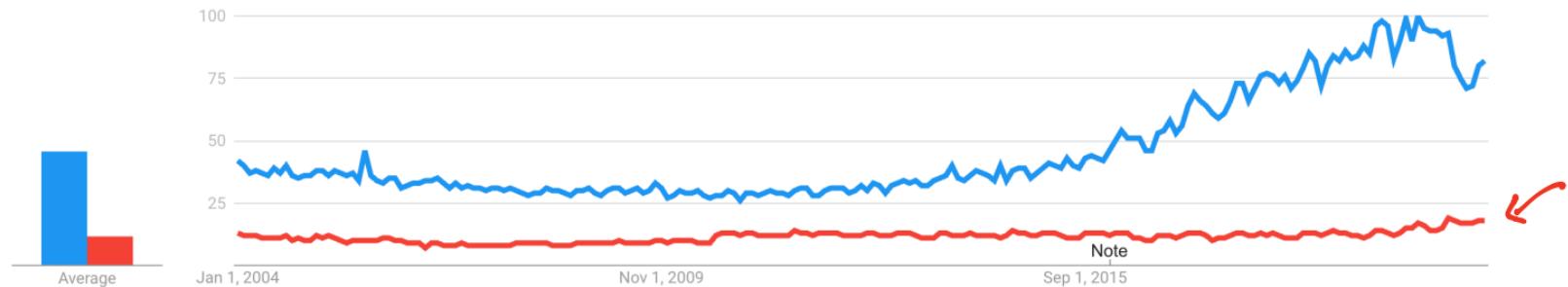
Worldwide ▾

2004 - present ▾

All categories ▾

Web Search ▾

Interest over time ⓘ



Introduction to higher-level languages: Python

Python was conceived in the late 1980s by Guido van Rossum at Centrum Wiskunde & Informatica (CWI) in the Netherlands as a successor to ABC programming language, which was inspired by SETL, capable of exception handling and interfacing with the Amoeba operating system.

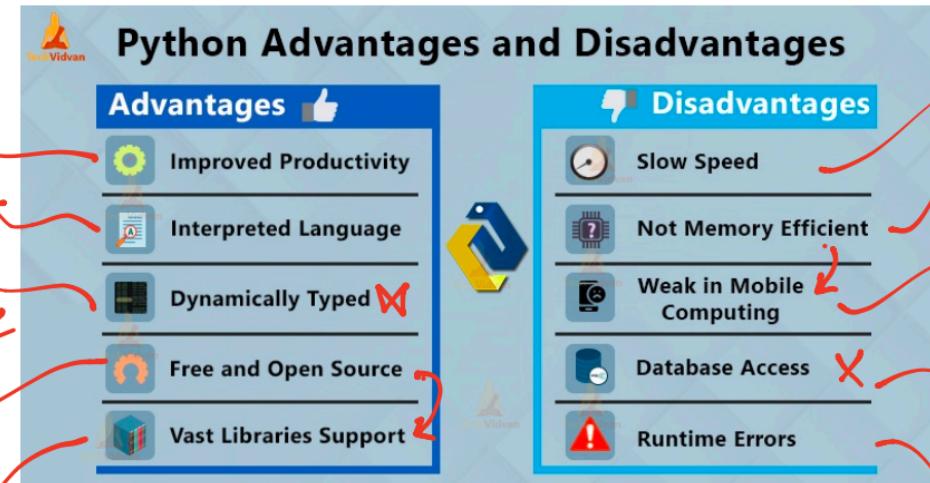
SETL (SET Language) is a very high-level programming language based on the mathematical theory of sets developed by (Jack) Jacob T. Schwartz at the New York University.



[https://en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))

Advantages and Disadvantages of Python

* Easy to read, learn & write.



simplicity

direct execution

automatically assign data type

many libs!

• pip

• Python package index (PyPi)

- interpreted
- dynamically typed

use large amount of memory

not memory efficient

slow processing power

less developed

data type of a variable can change anytime

Cython → C

Introduction to higher-level languages: Mathematica

Wolfram Mathematica is a software system with built-in libraries for several areas of technical computing that allow symbolic computation, manipulating matrices, plotting functions and various types of data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other programming languages.

It was conceived by Stephen Wolfram and is developed by Wolfram Research of Champaign, Illinois.

The Wolfram Language is the programming language used in Mathematica.



https://en.wikipedia.org/wiki/Wolfram_Mathematica

Advantages and Disadvantages of Mathematica

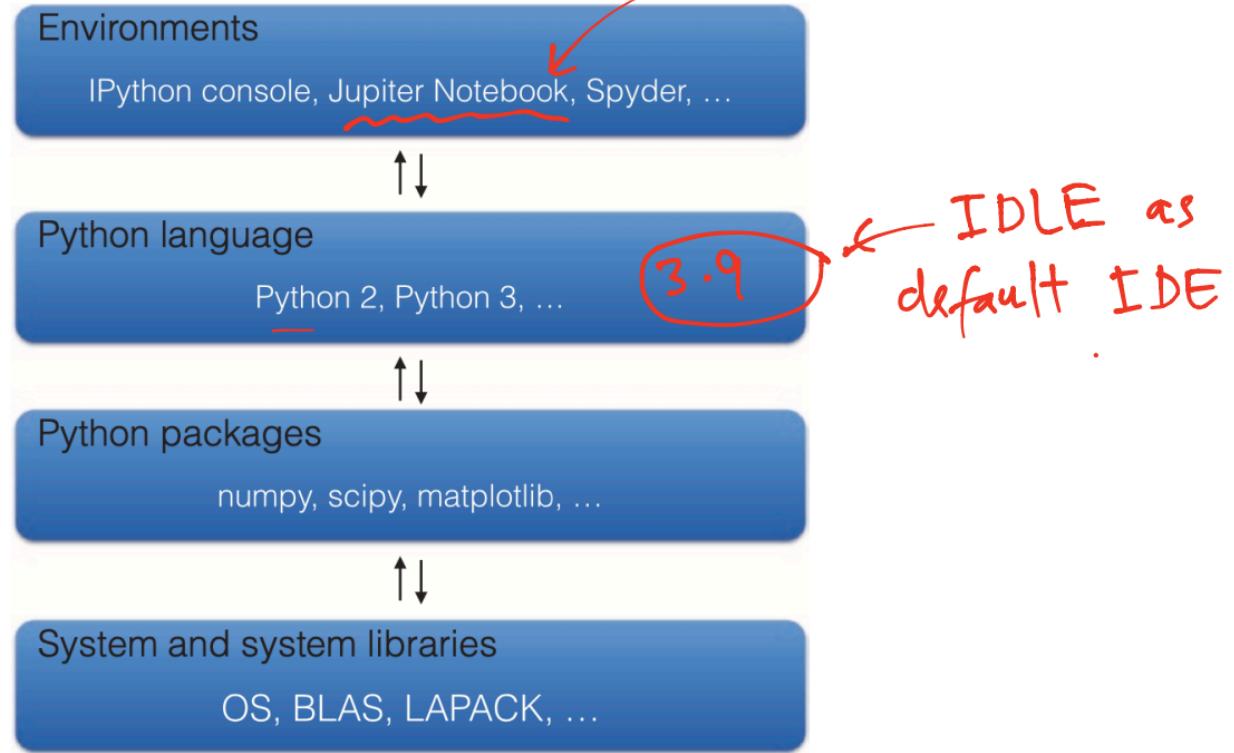
Advantages:

1. Notebooks/Presentation
2. strong symbolic computation
3. Excellent documentation with live examples.
4. Most specialized functions are built in
5. Interactivity/Animation is easy
6. powerful nested data manipulation.
7. very large number of functions it has

Disadvantages:

1. syntax very different to other programming languages.
2. Lots of shortcut notations: @, //, /@, &, #, @@@, @@, /*, etc.
3. Expensive.
4. Closed-source.

Python environment



Python as scientific and technical computing environment

www.scipy.org: the Scipy organization provides a centralized resource for information about the core packages in the scientific python ecosystem, and lists of additional specialized packages, as well as documentation and tutorials. as such, it is a valuable resource when working with in python.

<https://wiki.python.org/moin/NumericAndScientific>. another great resource is the Numeric and Scientific page on the official python wiki:

Python SciPy for Numerical Computing.

Online Python Compiler: ✓

https://www.w3schools.com/python/python_compiler.asp

Complete details on SciPy: <https://scipy.org/>

SciPy reference notes: for hands-on lab: <https://scipy-lectures.org/>

Jupyter Notebook Online: <https://jupyter.org/try>

Google Collab for Python execution with Jupyter notebook:
<https://colab.research.google.com/notebooks/intro.ipynb>

Mathematica Books, Demos, Online documentation

Create a Wolfram Cloud account for online Mathematica usage:

<https://www.wolframcloud.com/>

Do introductory programming lab at:

<https://www.wolfram.com/language/fast-introduction-for-math-students/en/>

Read and try coding using this book:

<https://www.wolfram.com/language/elementary-introduction/2nd-ed/>

Examples Mathematica projects: <https://demonstrations.wolfram.com/>

New: Mathematica Notebooks: <https://www.notebookarchive.org/>

* Start installing & learn



WOLFRAM

WOLFRAM CLOUD

Integrated Access to Computational Intelligence

The Wolfram Cloud combines a state-of-the-art notebook interface with the world's most productive programming language—scalable for programs from tiny to huge, with immediate access to a vast depth of built-in algorithms and knowledge. [Learn more »](#)

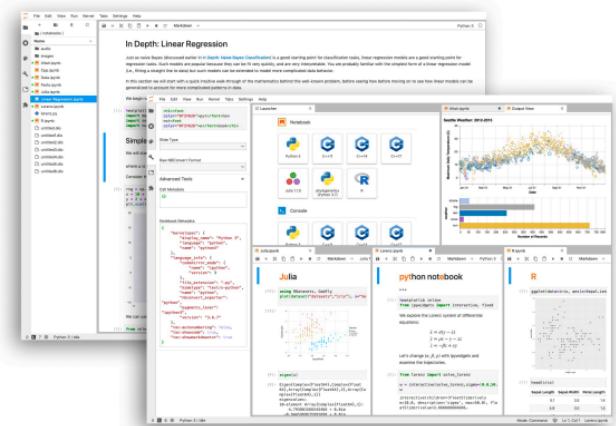
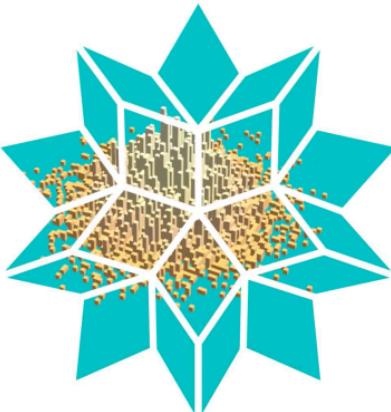
Wolfram Cloud technology powers Wolfram|One, Wolfram Programming Lab, Wolfram Mathematica, Wolfram Enterprise Private Cloud and Wolfram|Alpha, as well as Wolfram instant APIs, instant Web Apps and more...

BASIC PLAN

HAVE AN ACCOUNT?

[Sign up for free](#)

[Sign in](#)



© 2021 R.U. Gobithaasan All Rights Reserved.

R.U.Gobithaasan (2021). Scientific Computing, Lectures for Undergraduate Degree Program B.Sc (Applied Mathematics), Faculty of Ocean Engineering Technology, University Malaysia Terengganu.

Thank you . . .

email: gr@umt.edu.my