

Programming 2 Computer Labs AY 2024-25

Binghuan Li, binghuan.li19@imperial.ac.uk

Table of Contents of the Lab Slides

Week 1: Revision of Programming 1
<ul style="list-style-type: none">▪ Concept of modular programming and use of functions.
Week 2: File I/O
<ul style="list-style-type: none">▪ Namespaces: built-in, global, local scopes▪ File I/O methods: open(), read(), readline(), write(), close()▪ Why need to use close()?▪ Relative path and absolute path in OS▪ Summary of potentially useful string methods and list methods▪ Summary of potentially useful OS commands
Week 3: Modular Programming
<ul style="list-style-type: none">▪ Print formatting with f-string and format() function▪ Use raw string to get rid of Python's escape sequences▪ Function non-keyword argument (*arg) and keyword arguments (**kwarg)▪ Iterating with range() and enumerate()▪ Good coding practices, how to document functions
Week 4: Object-Oriented Programming
<ul style="list-style-type: none">▪ Definitions of class, objects, instances, attributes, methods▪ Basics syntax of OOP▪ Operator overloading and Python's magic methods
Week 5: Object-Oriented Programming & Inheritance
<ul style="list-style-type: none">▪ Four pillars of OOP: abstraction, encapsulation, inheritance, polymorphism▪ Introduction to encapsulation▪ Different forms of inheritance: single, multiple, multi-level; the yoyo-problem
Week 6: Object-Oriented Programming

- other relations between two associated objects: composition

Week 7: Decorators, Properties, Static/Class Methods

- Decorators of a function
- Static methods, Class methods
- OOP getter (property) and setter methods

Week 8: Algorithms

- Implementation of a matching algorithm
- Potential ways to accelerate your algorithm

Week 9: Unit test

- Syntax for creating a unit test class
- Summary of unit test assertion methods and fixture methods

Additional Examples

- Week 5: inheritance in object-oriented programming:
<https://colab.research.google.com/drive/1I9pzmhtwfNx-32kZ95s2AuDFN4nynFe3?usp=sharing>
- Week 6: composition and aggregations in object-oriented programming:
<https://colab.research.google.com/drive/16ZvA4WQWSaHb1XxPbBM9Xi0jW-tVcC?usp=sharing>
- Week 8 - “bits and bobs”, class methods, property, decorators, wrappers:
<https://colab.research.google.com/drive/10SeVwnCiHC3GxhbwUuGPDE6NDJzkSQLC?usp=sharing>