

**1. Course Code**

2208

**2. Course Title**

F2e:Fundamentals of Computer Programming

**3. Teacher**

SUN, Yi

**4. Term**

Fall 2

**5. Course Requirements (Courses / Knowledge prerequisite for this course)**

Fundamentals of Computer Systems (both courses can be taken concurrently)

**6. Course Overview and Objectives**

Programming is the foundation of every other subject in ICT. By becoming proficient in programming, students will be able to actively participate in projects involving system creation. Programming is also necessary for testing ideas, constructing and maintaining networks and servers, and in many other areas.

The course first reviews the fundamentals of procedural programming through experimental exploration, using the dynamic, interactive, object-oriented Python language.

**7. Course Outline**

- 1 Course orientation, the concept of Programming
- 2 The Python Basic knowledge
- 3 Program Flow Control
- 4 List, Ranges & Tuples in Python
- 5 Dictionaries and Sets in Python
- 6 Input and Output operation in Python
- 7 Modules and Functions in Python
- 8 Python programming exercises 1
- 9 Object Oriented Programming with Python
- 10 Python programming exercises 2
- 11 Using Databases in Python
- 12 Python programming exercises 3
- 13 Data analysis with Python basic
- 14 Data analysis with Python advance
- 15 Final Presentation.
- 16

**8. Textbooks (Required Books for this course)**

None.

**9. Reference Books (optional books for further study)**

Python Crash Course, 2nd Edition: A Hands-On, Project-Based Introduction to Programming

Author: Eric Matthes

Publisher: No Starch Press

## 10. Course Goals (Attainment Targets)

- (1) Become able to read, understand, and modify programs written in Python.
- (2) Become able to develop a small application.
- (3) Can write and use Python scripts for everyday tasks.
- (4)
- (5)
- (6)
- (7)
- (8)

## 11. Correspondence relationship between Educational goals and Course goals

Educational goals of the school		Course Goals	
High level ICT skills	Basic academic skills	(1) (2) (3)	
	Specialized knowledge and literacy	(2) (3)	
Human skill (Tankyu skill)	Ability to continually improve own strengths	(3)	
	Ability to discover and resolve the problem in society	Problem setting	
		Hypothesis planning	
		Hypothesis testing	
		Practice	
	Fundamental Competencies for Working Persons	Ability to step forward	(3)
Ability to think through		(3)	
Ability to work in a team			
Professional ethics			

## 12. Evaluation

Goals	Evaluation method & point allocation					
	examination	Quiz	Reports	Presentation	Deliverables	Other
(1)			○			
(2)			○			
(3)				○	○	
(4)						
(5)						
(6)						
(7)						
(8)						
Allocation			40	30	30	

## 13. Evaluation Criteria

Examination	
Quiz	
Reports	The code is written without running errors.
Presentation	Include a specific description of what you want to achieve. Have specific instructions for code implementation.
Deliverables	The results required by the exercise can be achieved.

Other	
-------	--

#### 14. Active Learning

Hourly percentage of active learning within the whole class time		0%
1	Active learning such as problem solving assignment using the knowledge and skills acquired in class.	Sometimes
2	Active learning such as group works and discussions.	Not at all
3	Outcome presentations and feedbacks.	All the time
4	Students actively make decisions on how the class should be conducted.	Not at all

#### 15. Notes

#### 16. Course plan

(Notice) This plan is tentative and might be changed at the time of delivery

---

Lesson 1: Orientation, introduction, motivation

(lecture and demonstration, 90 min)

---

What is programming? We consider solving everyday tasks, first by natural language, then by writing exact and detailed instructions. Students learn the basics of the Python language through a few simple exercises and use it to test their understanding of the elements of programming principles.

---

Lesson 2: The Basics of Python

(lecture and exercise, 90 min)

---

Introduce the fundamentals of Python Language, and write your first Python program.

---

Lesson 3: Program Flow Control

(lecture and exercise, 90 min)

---

Introduce the basic flow control in programming, like If-Else and For Loop. and exercises it.

---

---

#### Lesson 4: List, Ranges & Tuples in Python

(lecture and exercise, 90 min)

---

Introduce the additional built-in sequence types, List, Ranges & Tuples in Python.

---

#### Lesson 5: Dictionaries and Sets in Python

(lecture and exercise, 90 min)

---

Introduce the additional built-in sequence types, Dictionaries and Sets in Python.

---

#### Lesson 6: Input and Output

(lecture and exercise, 90 min)

---

Introduce the method for file input and output in Python. you're gonna learn how to read and write text files and also how to look to read and write binary files using pickle and shelve which are tools that are part of Python.

---

#### Lesson 7: Modules and Functions

(lecture and exercise, 90 min)

---

Introduce the modules and functions in Python. we're gonna find out about modules, imports, obviously functions as well, and really more the things like parameters, arguments. We'll learn that how to creating your own modules and also your own functions.

---

#### Lesson 8: Python programming exercises 1.

(exercise, 90 min)

---

Summarize the previous knowledge and practice comprehensively.

---

#### Lesson 9: Object Oriented Programming

(lecture and exercise, 90 min)

---

Introduce the the concept of Object Oriented Programming. include Class, Instance, Self and Inheritance

---

---

Lesson 10: Python exercises 2

(exercise, 90 min)

---

Summarize the previous knowledge and practice comprehensively.

---

Lesson 11: Using Databases in Python

(lecture and exercise, 90 min)

---

Learn how to use the Python library to connect to database, and operate the data in Python.

---

Lesson 12: Python exercises 3

(exercise, 90 min)

---

Summarize the previous knowledge and practice comprehensively.

---

Lesson 13: Data analysis in Python 1

(lecture and exercise, 90 min)

---

Introduce the fundamental packages for scientific computing with Python: Numpy, Pandas and matplotlib.

---

Lesson 14: Data analysis in Python 2

(exercise, 90 min)

---

Use pre-prepared data sets to practice the basics of data analysis.

---

Lesson 15: Final Presentation.

(group wrk and discussion, 90 min)

---

Presentation and discussion of the final individual work.

---