

MGT 1006: Programming for Business Computing

Spring 2024 (112-2)

Instructor: Ling-Chieh Kung
College of Management
National Taiwan University

In recent years, information technology (IT) has transformed the way people do commerce and business. Some obvious examples are online channels, digital marketing, automatic replenishment, program trading, display advertisement, social networking sites, business analytics, just to name a few. Understanding the capability of IT obviously brings in huge advantage to a business decision maker: Either you do it by yourself when you are junior, or you know who the right people to delegate to are when you are senior. Being able to communicate with (or even lead) IT people is also critical.

In this course, we will introduce how to write computer programs for business computing. We cannot make you a software engineer, who build software products to sell to consumers. Instead, we plan to enable you to write programs to facilitate your own works (e.g., analyze a huge data set that cannot be done with MS Excel). More importantly, you will know how a computer program works, the ways computer scientists and software engineers think, and how to leverage IT to bring in competitive advantages to your organization and yourself.

The programming language we will introduce is Python, one of the most popular and powerful high-level programming language nowadays. The language Python is just something that facilitates the delivery of the principles of computer programming. What really matter are the conceptual principles, not the syntax or rules. Our objective is not to teach you how to write Python programs; we want to make it easier for you to learn other programming languages (like R, SAS, Javascript, etc.) in the future.

This is an elective course for everyone. If too many students want to take this course, students in the college of management have the highest priority. We do not assume any background in computer programming, and there is no prerequisite for this course. Auditing is welcome. This course is taught in Chinese.

Basic information

Instructors.

- Ling-Chieh Kung (孔令傑): lckung@ntu.edu.tw; 管二 413.

Teaching Assistants.

- 陳柄瑞 (R11725028), 朱盈穎 (R11725011), and many others.

Meetings.

- 18:25–21:05, Monday: Lectures, exams, or office hour:
 - Physical lectures and office hour: 管一 103, 管一 104.
 - Exams: 管一 103, 管一 104, and many other classrooms.
- Labs:
 - Wednesday: 12:25–14:15, 管一 101; 18:25–20:15, 管一 103.
 - Thursday: 12:25–14:15, 管一 104; 18:25–20:15, 管一小電腦教室.

Textbook. A. Downey, *Think Python 2* (<http://greenteapress.com/wp/think-python-2e/>).

On-line Resources.

- Course website: NTU COOL (<https://cool.ntu.edu.tw/>).
- To submit homework: PDOGS (<http://pdogs.ntu.im>).

Grading

Breakdown.

- Following all administrative rules (such as responding to surveys by the due time): 5%.
- Homework: 35%.
- One quiz: 5%.
- Two exams: 25% (15% for the higher and 10% for the lower).
- Final project: 30%.

Conversion rule. The final letter grades will be given according to the following conversion rule:

Letter	Range	Letter	Range	Letter	Range	Letter	Range	Letter	Range
F	[0, 60)	C-	[60, 63)	C	[63, 67)	C+	[67, 70)	B-	[70, 73)
B	[73, 77)	B+	[77, 80)	A-	[80, 85)	A	[85, 90)	A+	[90, 100]

Policies

Lectures in videos. Most lectures will be given in the format of online videos. Before most lectures, the instructor will upload videos containing some materials to be studied by the students themselves. The total length of those videos for one lecture will be around 60 to 90 minutes.

Office hour. For those Mondays without a lecture or event, some TAs will hold office hour in the classroom during the official lecture time. Students are welcome to consult the TAs for any questions and ask for help. Even if they do not want to talk with the TAs, students are also welcome to use the classroom to do discussions or study whatever they like. No booking is required, and we welcome everyone.

Important Note. Because one must have enough painful experiences in debugging and revising programs, the instructor (and TAs) may (and actually should) refuse to debug for any student. Unless there is a special announcement, the office hour on Monday (or occasionally held in labs) is the only exception.

Homework. For most weeks, one homework will be assigned on Tuesday and due in the next week. Please upload your Python source codes (and other files, if required) to the online grading system PDOGS by the due time. No submission in class or in lab. No hard copy. No late submission. While discussions are highly encouraged, each student must turn in her/his own homework. Cheating will result in severe penalty for everyone involved. All homework grades are counted in calculating the final semester grades.

PDOGS. For homework and exams of this course, we rely heavily on the Programming Design Online Grading System (PDOGS, or P-Dogs). After a student uploads her/his Python source file, the system will automatically compile and run the program with respect to some testing data, calculate grades, and display the grades to the student. One may repeatedly modify his program and upload again and again until she/he is satisfied. Only the grades for the *last submission* before the deadline will be recorded.

Labs. In labs, the TAs may review materials covered in lectures and give students on-site practices. These practices do not count for any grade. However, attending labs can be as important as attending lectures if you are a beginner. Each student will be assigned to one of the labs, and she/he should not attend other labs.

A quiz and two exams. For three weeks we will have a quiz and two exams during the lecture time. Students will be asked to write several Python programs during the exam time. Students are expected to use their own laptops, but those who do not have one may contact the instructors (in advance, not on-site) for help. The Internet will remain active throughout the exams, and one is allowed to search whatever she/he wants online. However, no communication with any living person is allowed.

Final project. Students should form teams to do one final projects. For the final project, the instructor will only specify a rough direction. Each team then decides its own topic, build a program for its own objective, and demonstrate its program to the class publicly.

Attendance and class participation. We do not count attendance. If you have something more important to do, feel free to drop a lecture or a lab. Nevertheless, we encourage class participation. During lecture time or office hour, you are more than welcome to ask or answer questions and provide comments. You are also encouraged to use NTU COOL for after-class discussions.

About plagiarism/cheating. For all class activities that must be graded, plagiarism/cheating will result in a severe penalty for everyone involved. For each assignment, all students' submitted codes will be compared through a system. If two students' codes are considered too similar by the system, the instructing team will manually check the codes. If the instructing team also believe that the codes are too similar, each of the two students will be "suspected" and informed. If they do not convince the instructing team that there is no plagiarism/cheating, they will be marked as "one bad record." While one bad record does not affect a student's grade, a student's grade of that plagiarized problem will be zero since the second bad record.

Tentative schedule

Date	Lecture topics	Videos	Textbook	What's due
Module 1: basics				
2/19	Course overview and the basics	1-1, 1-2	Chs. 1 & 2	
2/26	Computers and conditional*	1-3	Appen. B & Ch. 5	HW0 (Tue)
3/4	Iterations*	1-4	Ch. 7	HW1 (Tue)
3/11	Lists and algorithms*	1-5	Ch. 10	HW2 (Tue)
3/18	Applications (1) and <i>quiz</i>	N/A	N/A	HW3 (Sat)
3/25	Midterm exam 1	N/A	N/A	
Module 2: advanced techniques				
4/1	Functions and fruitful functions*	2-1	Chs. 3 & 6	HW 4 (Tue)
4/8	Strings*	2-2	Ch. 8	
4/15	Data structures, file I/O, and exceptions*	2-3	Chs. 11, 12, & 14	HW5 (Tue)
4/22	Applications (2)	N/A	N/A	HW6 (Sat)
4/29	Midterm exam 2	N/A	N/A	
Module 3: applications				
5/6	Classes and plotting	3-1, 3-2	Chs. 15–17	
5/13	Graphical user interface*	3-3	Chs. 18	HW7 (Tue)
5/20	Self-select advanced topics*	3-4 to 3-6	N/A	
5/27	Applications (3) and closing remarks	N/A	N/A	
6/3	No class: recorded project presentations	N/A	N/A	FP (Tue)

- For those dates where the lecture subject is starred, the lectures will be given in on-line videos before the lecture dates. There will be no lecture during the lecture time. Please see the "Policy" section for more details.
- The notes in the "Videos" column refers to lecture videos for that topic, where x - y means the y th week of the x th course "(Programming for Business Computing in Python (x))" on Coursera.
- In the "What's due" column, "HW" stands for "homework", and "FP" stands for "final project".