CSE 101 : Introduction to Computers

Lecture 0 – Course Introduction

(Slides courtesy: Antonino Mione)

History of Computers

- https://www.youtube.com/watch?v=sTc4kIVUnoA

Computer Science is Changing Everything

https://www.youtube.com/watch?v=QvyTEx1wyOY

Course Information

- CSE 101 : Introduction to Computers
- Fall 2018
- Instructor: Pravin Pawar
- Meetings: Lecture: Tue/Thur 10:30-11:50 PM

  Lab: Mon: 12:30-1:50PM
- Extra classes maybe held depending on the requirements

Instructor

- Pravin Pawar
- Office: B424
- Email: pravin.pawar@sunykorea.ac.kr
- Course Website:
  http://www3.cs.stonybrook.edu/~amione/CSE101_Course/index.html
- Phone: +82-032-626-1227
- Office Hours: *Mon: 3:00-5:00PM, Wed: 10:30-12:30PM, 3:00-5:00PM*

Announcements

- Please bring a laptop to each class
  - Classes will involve short lecture segments, demos, and some student exercises
- Additional video lectures are noted in the syllabus. These are strongly recommended for extra instruction to help understand Python. Please write questions during these videos to ask in class

Misc Information

- For non-CS majors: This course is an excellent way to get an introduction to what computer science is all about and learn how to program
- For CS majors: This course is the launching point into the CS major for those who have no background in CS at all

Course Overview

- CSE 101
  - introduces the important, central ideas of computer science
  - explores computational thinking and problem solving
  - covers the fundamentals of computer programming
- Computer science is the *study of problem solving with computers*
  - Astronomers don't study telescopes. They *use* telescopes to study the stars!
  - Likewise, computer professionals *use* computers to solve important problems in the modern world
  - Computer scientists also build computers and software that makes the computer hum.
    - In this regard CS is actually a lot more than just studying problem solving with computers.
    - But for CSE 101 that definition is good enough for us now.
- An important thread of this course is **computational thinking**, which is the way computer scientists think about and solve problems

Sorting Example Demo

- Bubble sort
- Insertion sort
- Quick sort

https://www.youtube.com/watch?v=WaNLJf8xzC4

Major Course Topics

- Algorithmic thinking (how to devise solutions to problems)
- Introduction to computer programming using the Python programming language
- Basics of computer hardware
- Data representation (how does the computer save data?)
- Data organization (how do we manage complex data?)
- Software design, implementation (coding) and testing
- Limitations of computers
- Given adequate time, we will also look at:
  - Introduction to information security and cryptography
  - Social, legal and ethical aspects of computing
  - Additional topics
- Also, some of this list may be modified if we find other more interesting topics later

Textbook

- Required: *Explorations in Computing: An Introduction to Computer Science and Python Programming* by John S. Conery. ISBN 978-1466572447
- Required: *How to Code in Python 3* by Lisa Tagliaferri. ISBN 978-0-9997730-1-7
- Optional, free: *Blown to Bits* by Hal Abelson, Ken Ledeen, and Harry Lewis. ISBN 0137135599. Download from www.bitsbook.com
- Necessary software: Python (www.python.org/downloads) and PyCharm
  - These are all free!
- Download links will be available on the course website
- We will use the first lab meeting to setup the software and get familiar with it

Homework

- Over the course of the term you will be required to solve computational problems by writing software in Python
- These homework assignments will reinforce concepts from class and have you explore new concepts, too
- All work will due on fixed dates and times
- All work will be completed on an individual basis (write your own code) *unless otherwise instructed*!
- You will use **Blackboard** to submit your completed assignments
- Please start early on the assignments! Most students find that completing the homework assignments for CSE 101 takes **a lot** longer than they anticipated

Late Homework Policy

- Assignments must be turned in by the due date and time.
  - Any part of an assignment that's late means the entire assignment is late.
  - If your assignment is incomplete or not entirely working by the due date, turn in what you have to get some partial credit.
- If you have an emergency situation, email me before the due date and I may be able to work something out
- Bottom line: Plan ahead, start early!

Cooperation vs. Copying

- Cooperation (talking over problems) is a good way to learn and is encouraged
- ***Do not copy code. Do not let others look at or copy your code.***
- Copying is not allowed on homework or exams no matter the source (written or verbal)
  - When you submit your homework or tests, **you are pledging that the work is your own and you have not copied it.**
  - **You are also pledging that you have not allowed others to copy it**.
- **DO NOT COPY! (*Software tools catch it easily*)**

Lab

- Lab exercises will involve a variety of programming tasks, such as:
  - running existing programs and collecting data about them
  - writing your own, original, short programs to solve problems
  - fixing errors in programs

Examinations

- Examination dates will be posted on the schedule page of the course website. Tentative dates are:
  - Midterm exam 1: Thurs, 9/20
  - Midterm exam 2: Tue, 11/6
  - Final exam: See SUNY Korea Final Exam schedule
  - Oral exams: Will be conducted anytime during lab sessions
- Do not miss exams
- Arrange your work and travel schedules as needed to be present for examinations
- Makeup exams will only be given for verified, officially sanctioned university activities. All makeup examinations may be oral.

Grading

- Problems given in Labs: 20%
- Oral exams: 20%
- Class Participation (Attendance): 5%
- Midterm Exam 1: 15%
- Midterm Exam 2: 15%
- Final Exam: 25%
- Policies:
  - Makeup exams will only be given for verified, officially sanctioned university activities

Re-Grading

- If you feel that your work (exam, homework, quiz, etc.) was not graded correctly, you may request a regrade no later than one week from the day grades are posted or announced
- Requests for regrades made after one week will not be entertained

TA Assistance

- TAs are available almost every day each week
  - Schedule is forthcoming (posted on course web)
  - In "CS Commons" (next to CSD office)
- Come with specific questions and/or code with which you need help
  - TAs strive to spend time with everyone that comes to a session so be courteous and share the TA's attention

Electronics in Class

- Cell phones should be put away during class
- Laptops may be used during periods where you are asked to work on an exercise during class
- Lecture slides are available on the course website for study before class.
- Talk to me after class if there's an issue with this policy

Disability

- If you have a physical, psychological, medical or learning disability, please contact the One-Stop Service Center.
  - Location: Academic Building A201
  - Phone: 626-1117
- The DSS will determine with you what accommodations, if any, are necessary and appropriate
- All information and documentation of disability is confidential

How to Succeed in this Class

- Attend class and be on time!
  - Not all information is in my lecture notes or in the book
  - I sometimes do in-class demos that emphasize non-obvious details
- This is an introductory course, true, but we're going to cover a lot of ground and move quickly starting from scratch
- The assigned work will take a lot of your time, so practice good time management
- Read the reading assignments and review the lecture notes and try out example code
  - Practice is the only way to become proficient at coding
  - Very often your first, second, or third attempt at solving a problem will not be successful. It is **essential** that you give yourself enough time to try different ideas, taking breaks along the way!
  - Those who write extra code for problems not assigned ("for fun") generally do best in this class
  - Learning to code involves learning to read other people's code
- Ask questions right away if confused. Ask in class, ask a TA, come to my office hours or send email. Don't stay confused and don't get behind!
- Welcome and I hope you enjoy the class!

Homework

- Go through the video on Introduction to Computers and Complete History

- https://www.youtube.com/watch?v=z3KnlfATUek