Introduction to Computer Infrastructure

Computer Science 103
Spring, 2022

Professor: Charles Hedrick
Course modality: in person
Prerequisites: none, corequisite: CS111
Meeting time: from scheduling system
Email: hedrick@rutgers.edu

Course Description:
The course equips students with no computer experience other than basic word processing and spreadsheets the computer skills needed for the introductory computer science courses. By the end of the course, students will be able to effectively use the Rutgers computing environment to iteratively design, build, and debug programming assignments.

Early parts of the course will be coordinated with computer science CS111. They will be aimed at giving students specific skills needed to do assignments. Thus, it will normally be taken with CS111. Later sections will involve skills needed to do more complex assignments, as well as assignments in the next few courses. While the material could be useful for other courses, e.g. computer science CS110, it will be focused specifically on skills needed to do the kind of programming used in the first few courses for majors.

Learning objectives:

At the end of the course, students will be able to effectively use the Rutgers computing environment to iteratively design, build, and debug programming assignments. Students will be able to use remote access tools, integrated development environments, the file system, the command line, and debuggers for the purposes of software development.

Text/Resources:
No required text. There will be handouts with instructions, or with pointers to online resources that cover topics.

Requirements:
The course will have weekly assignments that will assess students on current week's learning objectives. The first part of an assignment will be presented in class, at which time students will begin working on the assignment in class. Grading is based on weekly learning objectives.

Grading:
Weekly assignments: 100% of course points.
The course is pass/no credit. There is no curve, students must earn a minimum of 70% of class points to pass the course.

Academic Integrity:
Rutgers University regards acts of dishonesty (e.g. plagiarism, cheating on examinations, obtaining unfair advantage, and falsification of records and official documents) as serious offenses against the values of intellectual honesty. These policies are detailed here:

https://nbprovost.rutgers.edu/academic-integrity-students

In addition, the Computer Science department has established policies for academic integrity that pertain specifically to programming assignments:

https://www.cs.rutgers.edu/academics/undergraduate/academic-integrity-policy/programming-assignments

Syllabus

Week	Topic	**Learning Objectives** - at the end of each week students will be able to:
1	Course introduction	Log into Rutgers systems: Canvas, Email, Box, Zoom. Setup own home system/laptop. Describe the difference between the command line and the GUI tool.
2	Files and directories	Explain the difference between a file and a directory. Navigate the file system using the command line. Explain what the home directory is. List the contents of a directory/folder and explain the output. Explain file extensions. Use wildcards. Explain what a zip file is. Zip a directory. Unzip a zip file. View files using the command line. Take a screen shot and locate the file. Organize a directory for course work. Search for a file within the file system.
3	VSCode IDE	Explain what is Visual Studio Code (VSCode) and what it is used for. Install VSCode and JDK. Write a “Hello World” in Java in VSCode. Compile the “Hello World” program using the VSCode terminal. Execute the “Hello World” program using the VSCode terminal.
4	Using remote facilities	Configure MS Remote Desktop to use the Rutgers Virtual Private Network (VPN). Log into the iLab machines. Run VSCode on the iLab machines.
5	Basic computer architecture	Describe the CPU, the memory, and the hard drive and their interaction. Describe the Java Virtual Machine (JVM). Describe why programs run out of resources.
6	Debugging	How to use print statements effectively. Add breakpoints. Step through the iterations of a loop. Step through the iterations over an array of integers. Step into and out of a method.
7	Debugging	Write a Java program that reads arguments from the command line. Set up the Json file to pass command line arguments to a program.
8	Basics of network	Write a block diagram depicting the network from home through to a computer at Rutgers. Describe IP addresses and hostnames. Explain what is likely to go wrong when connecting to a Rutgers computer and how to report problems.
9	Coding Style and Readability	
10	Coding Style and Readability	
11	Break	Spring break or Thanksgiving break
12	Introduction to JavaDoc	Describe how to find methods appropriate for a class. Write JavaDoc comments for classes and methods.
13	Libraries	Describe what libraries are. Describe how to use libraries in Java programs. Compile and execute through the command line Java programs with library dependencies. Add the library dependency to VSCode and execute.
14	Review	