

Computer Science 120
Fundamentals of Computer Science I
10:50 – 11:50 MWF, Trex 374, Fall 2006
2:50 – 5:50 TH Trex 263

Instructor:	Dr. Stephen Hughes
Office	Trexler 365-C
e-mail:	hughes@roanoke.edu
Phone	375-4901
Office Hours:	MW 2:30-4:00, Thurs 10:00-11:00 Also by appointment or open door

COURSE OBJECTIVES

This course is the first in a three course sequence designed to introduce students to the fundamental concepts of computer science including the underlying foundations from discrete mathematics. The course focuses on the design of algorithms to solve problems, the basics of mathematical logic, and the implementation of the algorithms in the programming language Java. Students will learn the basic control structures and data structures provided by the Java language, the concepts of objects, classes and methods, and will gain experience in the use of objects (Java classes) in programming and problem-solving. Students will gain familiarity with the UNIX operating system as they develop programs on systems running Linux, a version of UNIX for personal computers.

COURSE CONTENT

Text

Java Software Solutions: Foundations of Program Design, 5th Edition, by John Lewis and William Loftus, Addison-Wesley, 2006.

Prerequisites

There are no formal prerequisites for this course; however, a strong aptitude for logic usually predicts success in the course. Prior experience with programming or with Linux is **not** necessary.

Lab

This course has a required three-hour lab every Thursday afternoon from 2:50 until 5:50. The purpose of the lab is to give the student a structured experience in software design, implementation, and testing, and to increase the student's ability to use and understand the tools available for software development in the Linux environment. For each lab, there will be a *pre-lab assignment* designed to prepare the student for lab. The assignment must be handed in at the beginning of lab. *Unless otherwise specified, the lab itself must be done during the lab session and turned in before leaving.* Associated with most labs will be a short *post-lab assignment* designed to reinforce lab concepts. Typically the post-lab assignment will be due the second class period following the lab.

Programming Projects

Programming projects are designed to give the student the opportunity to put into practice the problem solving and programming skills they have learned. As such they are one of the most important aspects of the course both for student learning and for assessment. There will be 4 - 6

projects during the semester to be handed in for a grade. You are encouraged to start on your projects immediately when assigned and get help from the instructor as needed.

Important: Projects are to be done individually. You may ask class members, lab assistants, and others for help with system questions (e.g., "How do I save a file in emacs?", "How do I run my Java program?") or general information about a topic covered in class (e.g. "What is the symbol for boolean AND?") provided you can do so without divulging or receiving information specific to the solution of the programming project. *You may not discuss any aspect of the design or coding of a programming project with anyone except me.* This policy will be strictly enforced; see the section on Academic Integrity below.

Quizzes & Exams

Short quizzes will be given weekly to assess your progress in the course. These are designed to ensure that you are keeping up with the pace of the class and to give you a sense of the level of mastery that is expected. Quizzes will generally be given at the beginning of class, and not necessarily on the same day of the week. No make-up quizzes will be offered.

MCSP Conversations

The Math, Computer Science and Physics department offers a series of discussions that appeal to a broad range of interests related to these fields of study. These co-curricular sessions will engage the community to think about ongoing research, novel applications and other issues that face our discipline. A list of these conversations will be maintained on the course blackboard page.

Members of this class are invited to be involved with all of these meetings; however, participation in at least one of these sessions is mandatory. For each of the required sessions, students will submit a one-page paper reflecting on the discussion within one week. This should *not* simply be a regurgitation of the content, but rather a personal contemplation of the experience.

Grading

The course grade will be based on 3 tests, lab work (including pre and post lab assignments), programming projects, and a comprehensive final examination.

30%	Tests	10%	Quizzes				
15%	Lab	5%	MCSP Conversation				
20%	Projects	20%	Final Exam				
	B+ 87-90	C+ 77-80	D+ 67-70	Below 60			
A	93-100	B	83-87	C..... 73-77	D..... 63-67	F	
A-.....	90-93	B-	80-83	C-	70-73	D-	60-63

Test Dates:	Test #1	Monday, September 25
	Test #2	Monday, October 30
	Test #3	Monday, November 20
	Final Exam	Tuesday, December 12 (8:30 am)

Course Topics (Schedule Tentative)

WEEK OF	Topics	Lab	Sections in Text
Aug 30 (W,F)	<ul style="list-style-type: none"> • Overview of Roanoke College Network • Basic terminology • The Linux operating system • UNIX file structures 	Introduction to Linux and Emacs	1.1 - 1.3
Sept 4	<ul style="list-style-type: none"> • Overview of computer architecture, algorithms, programming, programming languages, object-oriented programming, and systems software • Java programs - basic structure and style (documentation, naming conventions, and use of white space) • Introduction to objects, output, variables, and assignment statements in Java 	Editing, compiling, and modifying elementary Java programs	1.1 - 1.6; 2.1 - 2.2
Sept 11	<ul style="list-style-type: none"> • Numeric data types and expressions • Classes and class libraries 	Writing simple programs using expressions	2.3 - 2.6; 3.1 - 3.5
Sept 18	<ul style="list-style-type: none"> • Classes and class libraries, cont. • Introduction to applets and graphics 	Writing programs using classes; Writing applets	2.7 - 2.9
Sept 25	<ul style="list-style-type: none"> • TEST #1 • Representation of numeric data (binary, octal, hex; general concept of a base; twos complement) 	HTML, applets, and graphics	Appendix B and supplementary material 3.8 - 3.11
Oct 2	<ul style="list-style-type: none"> • Writing classes and methods • Graphical Objects 	Writing classes	4.1 - 4.8

Oct 9	<ul style="list-style-type: none"> • Introduction to logic (logic operations, truth tables, laws of logic) • Boolean expressions and if statements in Java 	More programming using classes; GUI programming with listeners; Selection statements	5.1 - 5.3 and supplementary material
Oct 16	<ul style="list-style-type: none"> • Fall Break 		
Oct 23	<ul style="list-style-type: none"> • Logic, continued • Application of logic to computer circuits 	Programming with selection statements	5.4 & supplementary material
Oct 30	<ul style="list-style-type: none"> • TEST #2 • Introduction to loops (while statements) 	Programming with loops	5.5
Nov 6	<ul style="list-style-type: none"> • Other looping structures 	More programming with loops	5.6 - 5.8
Nov 13	<ul style="list-style-type: none"> • Loop wrap-up • Events, dialog boxes and buttons • Graphics: Components, Containers, & Images 	More programming with loops; graphics programming	5.9 - 5.12
Nov 20 (M)	<ul style="list-style-type: none"> • TEST #3 	Thanksgiving - No lab	
Nov 27	<ul style="list-style-type: none"> • Classes and methods, more depth... • Design • Parameters revisited. • Testing • GUIs - Interfaces and Events 	More programming with classes and methods	6.1 - 6.9
Dec 4	<ul style="list-style-type: none"> • Dialog Boxes • Wrap-up 	More practice programming with classes; tracing parameters	

COURSE POLICIES

Academic Integrity

Honesty and integrity are qualities we respect in ourselves and in others. Therefore, you are expected to be fully aware of your responsibility to maintain the highest degree of integrity in all of your work. It is accepted that you have read and understood the standards for academic integrity at Roanoke College.

Collaboration on course assignments is strictly forbidden. By submitting work under your name, you are indicating that **you** have completed the assignment. This means that you should be able to completely explain all the details of your work, i.e. every line of code in computer programs. Failure to be able to account for your decisions (to my satisfaction) will result in referral to the Academic Integrity Council.

All students must abide by the Guidelines for Computer Use as stated on page 15 of the Academic Integrity Handbook. Failure to do so will result in involuntary withdrawal from the course

Attendance Policy

Class attendance is vital to your success in this course; material covered during missed sessions is the responsibility of the student. Conversations held in class illuminate the published class materials and are subject to evaluation on subsequent tests and quizzes. Moreover, quizzes and in-class assignments are not available for make-up.

Late Assignments

I understand that circumstances conspire against us all, and occasionally, deadlines cannot be met. If you need to hand in an assignment late, you must contact me via e-mail 24 hours in advance of the due date to *negotiate* a new submission date. Any late submission without prior approval will be penalized 10% per day. Electronic “glitches” do not waive your responsibility to submit your work in a timely manner.

Office Hours

Office hours are an opportunity for you to clarify details you may have missed in class. If you come to office hours with a problem on the assignment, you should come prepared to answer questions, as well as asking them. Additionally, you need to make sure that you have access to an electronic version of your work.

Electronic Devices

Cell phones and pagers must be *turned off* prior to entering the classroom or lab.

The use of any electronic device during a quiz or exam is strictly prohibited. This includes PalmPilots, Pocket PCs, and Blackberrys. Any use of such device during a quiz or exam will be considered a breach of academic integrity. Basic handheld calculators may be used on certain quizzes and exams only when announced by the instructor.

Special Services

If you are on record with the College’s Special Services as having special academic or physical needs requiring accommodations, please meet with me during my regular office hours or schedule an appointment as soon as possible. We need to discuss any accommodations before they can be implemented. Also, please note that you must make arrangements for extended time on exams and testing in a semi-private setting at least one week before *every* exam.