

CSCI 135/136 ANALYSIS & DESIGN 1 HUNTER COLLEGE CITY UNIVERSITY OF NEW YORK

CSCI 135 SOFTWARE ANALYSIS & DESIGN Monday, Wednesday, Thursday 12:10 – 1:00, Hunter North 118

COURSE DESCRIPTION

This course is: • An introduction to software development, using the C++ programming language. Software development is a skill that involves solving real-world problems by developing computer programs. Breaking a problem down, then creating a series of logical steps (an algorithm) that solves the problem is the conceptual part of the course. The technical component involves translating the solution into a working computer program. A successful student will be able to clearly and logically develop an algorithmic solution to a problem, while being comfortable enough with C++ to transform the the algorithm into a working computer program . • A preparation for further courses in computer science. This course is one of the “ABCs” of computer science. Students who expect to succeed in more advanced courses in computer science need to go beyond understanding the material presented in this course – they need to master it. It’s sort of like the fact that you need to master the alphabet in kindergarten in order to be a successful reader in first grade.

• Time consuming, very, very time consuming. Any programming course takes up a lot of a student’s time. In addition to the time spent in class, most students will need to spend between 10 and 15 hours a week at a computer. That makes for a total of 15-20 hours a week actually developing algorithms, and writing and debugging programs, no kidding!

This course is not: • An introduction to computers in general. We will not cover: operating systems, networks, databases, etc. This course teaches a specialized skill – developing solutions using computer programming – and only that. You should already possess basic computer skills such as compiling simple computer programs, editing files, manipulating files, etc. If you don’t feel comfortable with these sort of basic computer skills you should come talk to me.

• An overview of the C++ language. C++ is a huge language with a lot of highly technical details. We will cover the fundamentals of C++, but the focus is on designing algorithms and solving problems.

• A good idea to take if you are working full time and taking a full course load, or, for any other reason(s) you don’t have a lot of free time to devote to CSCI 13500. The course material requires many hours to master (see above). Be honest with yourself. Make sure this course is for you, now, at this point in your academic life. If you would like to discuss the time requirements further please feel free to come talk with me.

Other than during lecture quizzes I ask that all cell phones be put away in class.

PRE-REQUISITES

The prerequisite is CSCI 12700 or instructor's permission. At the very least, you should have written, compiled, and run a program containing iteration `while`, `for` and selection `if` statements.

TEXT

Please see the [main course page](#) for instructions.

MAIN COURSE PAGE

You should check this page regularly, since all class materials will be gradually posted here.

TUTORING

There are dedicated computer lab for this course: HUNTER NORTH 1001B. It is staffed with tutors, who are there to help you. Please take advantage of them. You can ask for help with labs, projects, and general course material Monday through Friday from 11:00 AM to 6:00 PM.

QUESTIONS

Ask many in lecture! Outside of lecture go to the tutoring labs. The department hired additional tutors especially for students taking this course. Email the instructor of your recitation section with any programming questions that the tutors cannot answer. The beginning of each recitation is also devoted to answering questions. All emails to instructors must start with your full name, day and time of your recitation section, and name of your recitation instructor.

GRADESCOPE

You will submit all labs, homeworks, quizzes, and projects electronically through Gradescope. You will see your grades (including exam grades) on Gradescope as well. An invite email was sent to you before the first lecture. If you have not received it, ask your recitation instructor to send you another one right away: you will need to email him or her your full name, EMPLID, and a working email address.

BLACKBOARD

Make sure that you have configured BlackBoard to use your preferred email address (your Hunter email address, by default), since you are responsible for any email I might send there.

https://okunhardt.github.io/documents/Installing_WSL.pdf

SOFTWARE

This course is taught in Linux and your programs must be able to run on a Linux platform. On campus, you may use the 1001C and 1001B labs to do your work for this course. The standard

CSCI 135/136 ANALYSIS & DESIGN 1 HUNTER COLLEGE CITY UNIVERSITY OF NEW YORK

In these two courses, both of which are required to graduate with a computer science major, you will learn principles of programming, analysis, and design and gain a deep practical knowledge of C++.

[CSCI 135 Syllabus](#)

[CSCI 136 Syllabus](#)

[Gradescope](#)

[Coding Style Guide](#)

Text: *Cay Horstmann, Brief C++, 3rd ed. eText* – please rent the eText for one semester from here – do not buy or rent it from Amazon or Kindle store – you will not get access to the necessary interactive material!

Linux on Windows Tutorial: https://okunhardt.github.io/documents/Installing_WSL.pdf

Tutoring: The tutors for this course are available in the Open Lab Session on BlackBoard Collaborate of CSCI 135. This session runs all week:

Mondays: 11:00 AM – 7:00 PM

Wednesdays: 8:00 AM – 7:00 PM

Tuesdays, Thursdays, and Fridays: 11:00 AM – 6:00 PM

The tutors are there to help you with all of your labs, assignments, and projects! Get as much help as you need.

Lecture: Monday, Wednesday, Thursday 12:10 – 1:00 PM on BlackBoard Collaborate

Lecture Instructor: Genady Maryash HN1047 Office hours: Wednesdays 1:00 – 3:00 PM on BlackBoard Collaborate, gmaryash@hunter.cuny.edu

Email Questions You must get hands-on programming help in person during your lab or from tutors in lab B. We will never debug your code over email. You should ask questions during the lab and the lecture. All other email questions must be sent to your lab instructor listed below. You must always include your section, name of lab instructor, your name and EMPLID.







Section	Lab in HN 1001C	Instructor	E-mail
136.01	Mo 1:10 – 3:00 PM	<i>Minh Nguyen</i>	minh.nguyen@hunter.cuny.edu
136.02	Mo 3:10 – 5:00 PM	<i>Minh Nguyen</i>	minh.nguyen@hunter.cuny.edu
136.03	Mo 5:10 – 7:00 PM	<i>Minh Nguyen</i>	minh.nguyen@hunter.cuny.edu
136.05	We 8:10 - 10:00AM	<i>Subhadarshi Panda</i>	spanda@gradcenter.cuny.edu




136.06	We 10:10 - 12:00C	<i>Subhadarshi Panda</i>	spanda@gradcenter.cuny.edu
136.11	We 10:10 - 12:00B	<i>Hirohiko Kushida</i>	hkushida1@gradcenter.cuny.edu
136.07	We 3:10 – 5:00 PM	<i>Jaspal Singh</i>	Jaspal.Singh62@myhunter.cuny.edu
136.08	We 5:10 – 7:00 PM	<i>Shadman Quazi</i>	Shadman.Quazi62@myhunter.cuny.edu
136.09	Th 1:10 – 3:00 PM	<i>Melissa Lynch</i>	lynch.melissat@gmail.com
136.10	Th 3:10 – 5:00 PM	<i>Melissa Lynch</i>	lynch.melissat@gmail.com

COURSE OUTLINE

DATE	TOPIC	READING: Brief C++	SLIDES	DUE DATE
1/27	Syllabus Gradescope eText features C++ Review	1.3 Machine Code and Programming Languages 1.5 Analyzing Your First Program 1.6 Errors 1.7 HW Algorithm Design	1.5 1.6 1.7	E1.7 1/ LAB 1 Intro to Lin and to C++ 2/
1/29	Introduction Fundamental Data Types	2.1 Variables	2.1	WE 2/ WE 2/ E2.10 1/
1/30		2.2 Arithmetic 2.4 PS <i>First Do It By Hand</i>	2.2	PS 2/ Project 1A 2/
2/3		2.3 Input and Output	2.3-4	LAB 2 Loops a Arrays 2/ Project 1B 2/
2/5	Strings	2.5 Strings	2.5	
2/6	Arrays	6.1 Arrays	6.1	Project 1C 2/
2/10	Loops	4.1 The <code>while</code> Loop 4.2 PS <i>Hand-Tracing</i>	4.1	LAB 3 File I/ Process Data 2/ PS 4/ E4.8 2/
2/13		4.3 The <code>for</code> Loop 4.4 The <code>do</code> Loop 4.5 Processing Input	4.2-3 4.4-5	Project 1D 2/ WE 3/

DATE	TOPIC	READING: Brief C++	SLIDES	DUE DATE
2/19	More Loops	4.6 PS Storyboards 4.7 Common Loop Algorithms	4.6-8	WE 4 WE 4
2/20		4.8 Nested Loops 4.9 PS Solve a Simple Problem First 4.10 Random Numbers and Simulations 3.7 Boolean Variables and Operators	4.9-10 Squares Montecarlo 3.7	LAB 4 File I/O Process Data 3/ PS 4 PS 4
2/24	Decisions	3.1 The <code>if</code> Statement 3.2 Comparing Numbers and Strings 3.3 Multiple Alternatives 3.4 Nested Branches 3.8 Application: Input Validation	3.1 3.2 3.3-4 D.M.L. 3.8	E3.1 2/ WE 3 PS 3 PS 3
2/26	Functions	5.1 Functions as Black Boxes	5.1-3	E3.5 2/ WE 5
2/27		5.2 Implementing Functions 5.3 Parameter Passing 5.4 Return Values 5.5 Functions without Return Values 5.6 PS Reusable Functions	Viz Pyramid 5.4-6	LAB 5 Functions a Prime Numbers 3/ E5.6 3/ WE 5 PS 5
3/2	Scope Static Vars	5.7 PS: <i>Stepwise Refinement</i> 5.8 Variable Scope and Global Variables ↯ Static Variables	5.7 5.8 StatVar Run Code	WE 5 PS 5
3/4	MIDTERM EXAM 1	WEDNESDAY 3/4		
3/5	Streams	8.1 Reading and Writing Text Files	8.1	LAB 6 Strings a Ciphe E8.1 Delay
3/9	Pass by Reference	5.9 Reference Parameters Introduction to Project 2	5.9 Coffee	Project 2 A 3/ E5.14 3/
3/11			Code By Value By Ref.	Project 2 B 3/ E5.15 3/

DATE	TOPIC	READING: Brief C++	SLIDES	DUE DATE
3/19	Arrays	6.1 Arrays 6.2 Common Array Algorithms 0's, Squares, Copy, Sum, Avg, Min, Max, Search, Remove unordered	6.1 6.2	LAB 7 Automata Style 4 <i>Binary Search</i>
3/23		Remove ordered, Insert unordered, Insert ordered, Read inputs and find largest, 6.3 Arrays and Functions Array functions	 6.3	E6.8 4 Project 2 C 4 <i>Selection Sort</i>
3/25		6.4 PS: <i>Adapting Algorithms</i>	 6.4-5	PS 6 WE 6 WE 6
3/26		6.5 PS: <i>Discovering Algorithms by Manipulating Physical Objects</i> 6.6 Two-Dimensional Arrays Print 2D array	6.6 	PS 6 E8.1 4/
4/2	Pointers	7.1 Defining and Using Pointers Pointers Example	7.1 Ptrs 	E7.1 4/ LAB 8 Image Processing 4/ Project 2 D 4/
4/6		7.2 Arrays and Pointers Code	7.2 	
4/7	Dynamic Memory	7.4 Dynamic Memory Allocation Viz	7.4 	
4/13		7.6 PS Draw a Picture	Viz 	LAB 9 Pointers 4/ PS 7.6 (1 and 3 on WE 7

DATE	TOPIC	READING: Brief C++	SLIDES	DUE DATE
4/15		7.5 Arrays of Pointers Galton Board	7.5-6 Viz 	E7.16 4/
4/16	Objects	7.7 Classes of Objects 7.8 Pointers and Objects	7.7-8  Viz	E7.18 4/
4/20	Review	5.9 Reference Parameters	Ptrs 5.9_Slides 	LAB 10 Class Enums 4/
4/22	MIDTERM EXAM 2	WEDNESDAY 4/22		
4/23	Enumerations	The <code>switch</code> Statement Enumerated types: <code>bool</code> , <code>switch</code> , <code>MyBool</code> , <code>LIKELY</code> , <code>Color</code> , <code>Colors</code>	Enums	PS 1
4/27	Classes	9.1 Object-Oriented Programming 9.2 Implementing a Simple Class	9.1-2 Code	WE 1 LAB 11 Me Classes 4/

ChetSheet like the one that will be given to you on the exam

2019 Fall Midterm 1

2019 Fall Midterm 1 Answers
