

COP3363

INTRODUCTION TO PROGRAMMING IN C++ FOR MAJORS

Fall 2024

Instructor:	Shibo Li	Time:	TBD
Email:	shiboli@cs.fsu.edu	Place:	TBD.

Course Pages:

- <https://sites.google.com/view/cop3363-fall2024>

Teaching Assistants:

- Shibo Li (shiboli@cs.fsu.edu)
- Shibo Li (shiboli@cs.fsu.edu)

Office Hours:

- Instructor: Wed 12:30-1:30pm MEB 3346
- Shibo Li: Wed 12:30-1:30pm MEB 3346
- Shibo Li: TBA

Mode of Delivery

This course is delivered in traditional synchronous in-person format.

Description

This course covers fundamental concepts and skills of programming in C++ in the Unix Environment. This course is primarily for Computer Science majors who are taking upper division CS courses. Students are also instructed on efficient program design using a combination of procedural and Object Oriented paradigms. Students will also learn how to evaluate and/or interpret digital data and/or their implications, demonstrate the ability to use digital technology effectively and demonstrate the knowledge of how to use digital technology correctly, safely and/or ethically.

Course Content And Technology Requirements

Course content will be accessible through Canvas. Students will need to be able to view videos, write and upload assignments and take exam assessments online. Students will need to have access to high-speed internet and up-to-date software. To view the most current technology requirements, visit the FSU Canvas support site. Note that mobile devices do not necessarily work well with many aspects of Canvas, so a current desktop or laptop is required for this course, with current Windows or Apple OS. FSU ITS recommends only Firefox or Chrome as a browser.

Official course announcements, lecture materials, assignments and help resources will all be online on the Canvas site. Note that all registered students should have the Canvas course website listed on their my.fsu.edu portal page upon logging into Canvas. Be sure to test this and resolve any difficulties no later than the first week of classes.

Attendance and Class Schedule

This class is offered as two 75 minute or three 50 minute lectures per week, and one recitation section per week. Attendance at all lectures and recitations is necessary to do well in this course.

First Day Attendance: This attendance will be taken online on the course Canvas site, by each student completing the first day attendance quiz on Canvas, during the first day of the lecture meeting. Any student registered as of the first day of the first week of this class, and not present according to taking this quiz, will be dropped from the class as per the FSU First Day Attendance policy. This quiz does not count towards your final course grade, it only records first day attendance for each student completing it on time.

Note: all drops are each student's responsibility. If you want to drop, and to be sure you are dropped, you must process and verify the drop yourself using the FSU course drop/add system. The teaching staff cannot guarantee drops will occur due to non-attendance as there are too many normally occurring uncertainties and time delays in the relevant computer and software systems.

Credit Hours

This undergraduate course is taken for 3 credit hours.

Course Focus and Objectives

This course is intended for majors in computer science or related areas and focuses on the fundamental concepts of computer programming using the C++ language. This course may be used as the programming pre-requisite for COP 3330. The course uses the UNIX operating system and the g++ compiler for all programming tasks. A UNIX "primer" will be provided on the course Canvas site for reading, learning and reference.

By the end of the semester, a student who has completed this course with a passing grade should be able to:

- Demonstrate a basic understanding of fundamental computer science concepts, including software and hardware.
- Solve computing problems using a top-down approach in a well-structured design in the procedural paradigm, and utilize some basic object-based programming techniques.
- Design, implement, test and debug a C++ program to solve a given problem.
- Demonstrate knowledge and use of control structures including sequence, selection, iteration and functions.
- Make use of data types and structures in C++ including integer and floating point types, arrays (one-dimensional, two-dimensional, strings) and structs; arrays of structs and structs containing arrays. Have an introductory-level understanding of the C++ class and be able to utilize the standard IO and string classes and their member functions.
- Utilize fundamental algorithms studied to perform tasks such as finding the max and min in a data set, counting, summing, tracking a previous value, searching and sorting, processing until EOF, etc.
- Consider, compare and evaluate code segments or algorithms for relative efficiency in a basic fashion.
- Evaluate and interpret digital data and/or their implications
- Demonstrate the ability to use digital technology effectively
- Demonstrate the knowledge to use digital technology correctly, safely and/or ethically
- Time permitting, be able to understand the basics of recursion.
- Time permitting, be able to understand the basics of setting up and using linked lists.

Regarding UNIX, students should be able to:

- Create and use an FSU CS user account and use it to work on CS UNIX servers
- Use essential UNIX commands to work with directories and files
- Create and edit files using a standard UNIX text editor (e.g. pico/nano or vi/vim)
- Transfer files to and from their laptop to the FSU CS UNIX servers (e.g. Tectia or ftp)
- Complete programming tasks with the g++ compiler to compile, run, debug and test programs, using gdb for debugging

Course Mathematics Pre-requisite

All students taking COP 3363 are required to have previously taken and passed (with a C- or higher final grade) MAC 1140 or MAC 2311 or MAC 2233. If you have not taken and passed one of these courses or its equivalent, you will likely be automatically dropped by the CS department staff.

Required Textbook

Title: Starting Out with C++ from Control Structures to Objects

Format: Redshelf etext with FSU Follet Access

Edition: 10

Author: Tony Gaddis

Status: Required

Publisher: Pearson Copyright: 2021

ISBN: 978-0-13-7450626 (print edition)

Be sure to check the full ISBN number of any book you buy. That is the only way to ensure that you obtain the correct required textbook. To ensure that you only purchase a valid, copyrighted edition, purchase only from FSU bookstores or from the Pearson web site. Note that the Follet Access ebook edition may have a somewhat different ISBN from the print version.

Canvas Website

Course materials are made available on the class Canvas web site and course website (<https://sites.google.com/view/cop3363-fall2024>), and are all considered required reading. In the "Syllabus" section you will find this handout as well as other important information. **Read this syllabus entirely and carefully** as it contains vital details which will affect your grade in this course. All course program assignment files will be turned in to the Canvas course site.

Other web site materials will include

- course lecture slides
- additional required and reference handouts
- helpful resources and links
- current assignment write-ups and related information
- links to turn in your programming projects
- vital course announcements
- potentially more to be announced later

Announcements and Email

Vital class announcements will be provided via the course Canvas site and course website (<https://sites.google.com/view/cop3363-fall2024>). The best way to contact the teaching staff outside of class and office hours is *via FSU account email*. Canvas Announcements and email sent to course students are considered required reading. The class email is to be used by the teaching staff for official course matters only. For example, if we need to send information about exam details, we will create announcements on Canvas and

course website which will also send email to everyone in the class. It is required that everyone in the class does receive and read these in a timely way. Please note that email will be sent to your FSU email address only. When you send email to the teaching staff, only email sent from an FSU email account is normally going to be read. Email sent from other services such as gmail, hotmail etc. will often be filtered out by automated spam filters and deleted without ever having been seen. In all email to teaching staff, include your full name, the course number for this course, and your recitation section number. The teaching staff typically teach more than one course, so this information needs to be provided in every email.

Please note that course teaching staff typically only answer email during FSU standard business hours, Monday through Friday 8:00 am to 5:00 pm. We do our best to answer all emails in a timely way. Email volume in this course is always quite large. Be sure to use your FSU email to contact teaching staff, and not Canvas messaging.

Also be sure to keep the tone and content of all email professional, and follow the FSU Code of Student Conduct in all course communications.

Overview

In this course you will learn to write programs in the C++ programming language using good style, structure and design. No prior programming knowledge is assumed, however this is not a course in how to use computer software such as word processors and web browsers. This course is concerned with how to design and write computer software. Six programming projects will be assigned during the term, some of which will require a large time commitment on your part. Many students find the workload in this course to be heavy, especially during the last half of the course. Assignments will begin with simple problems, and the difficulty level and length will increase as the term progresses. The first project will not require you to write a program yourself, but will consist of a program exercise which will familiarize you with our computer systems and the C++ compiler we use in this course.

We will use the UNIX g++ compiler exclusively, available on computers at the CS Majors Lab and accessible to students with a required FSU CS Department user account from any personal computer. All projects turned in must be completed using standard C++ with this compiler, and must compile and run successfully using this compiler, on a UNIX machine. Note that the supported platform in this course is UNIX, using Windows machines to work with and demo, and the only supported compiler is g++. You should be ok in the course using any Windows, Apple/Linux, or other Linux laptop, but keep in mind you are responsible for course requirements being met and you may need to make adjustments and look up some commands on your own at times if you are using a laptop which is not Windows based. For example, your instructor will demo with a Windows machine using Tectia to connect to the FSU CS machines; if you are on an Apple laptop you would normally use the terminal program, ssh and ftp. The required textbook uses standard C++11.

An FSU CS Department user account is required in this course. Programs will be graded on FSU CS UNIX machines using g++. Note to Windows laptop users: you may wish to install Cygwin as another option, for local use.

Topics

We will start discussing programming fundamentals and the C++ language almost immediately. Some time will be spent addressing basic computer concepts, UNIX, and the use of the C++ compiler, but for the most part these topics will be covered in the early assigned readings and in your recitation sections. In addition the use of UNIX and g++ will be demonstrated in live lecture examples and in recitations. The following topics will be covered, in this approximate order:

- Introduction to computers (hardware and software) and programming
- Computing facilities we use: FSU computing systems, UNIX, the g++ compiler

- Introduction to C++, problem solving and algorithms; declarations
- Simple data types: integer and floating point, char, bool
- Input/output, including file input
- Assignment, operators, expressions, standard mathematical functions
- Conditional statements and logical expressions; if, switch statements
- Looping structures; while, for, do while statements
- Functions, parameter passing, scope, storage classes, macros
- Number representation and common calculation errors
- Arrays: 1-dimensional and multi-dimensional
- Sorting (selection sort) and searching (linear search, binary search)
- Pointers, references and an introduction to dynamic memory allocation
- Strings (both C-style strings and the C++ string class)
- Structs
- Abstract Data Types; using data structures built from combining structs and arrays
- Time permitting: introduction to classes, recursion and more on dynamic data structures
- Throughout the course we will discuss good programming style, design, testing and debugging, documentation, and efficiency

Program Assignments

A sincere word of advice: we have found from much teaching experience that the most common reason for not doing well on the projects is not starting them early enough. You will be given plenty of time to complete each project. However, if you wait until the last minute or the last weekend before the due date to start, you may not be able to finish. Plan to do some work on each project every day, and follow a step-by-step approach with a focus on good time management. Also plan to have it finished a few days ahead of the due date as many unexpected problems can arise during programming, especially in the debugging phase. In addition, computing facilities can become overloaded and/or crowded and make it difficult to get a computer to use. Canvas can become overloaded when projects are due and it can then be a very slow process to submit your work. When you have your own laptop, you cannot predict when it may fail to work. Plan for these things to happen, think about your backup plans in case of equipment problems, and allow ample time to meet course due deadlines. Meeting course deadlines and submitting assignments successfully and correctly to Canvas is each student's responsibility and will determine a large component of your course grade.

There are many sources of help for you while programming for this course. First, recitation sections and TA office hours are provided. You may send questions to the teaching staff via email, while keeping in mind that many types of questions cannot be answered without seeing your work, and some questions may be too complicated for email. Also keep in mind that if everyone puts off working on their programs until the last minute, we will likely not be able to help everyone adequately in office hours; the teaching staff typically can't handle all students in the course asking complex questions all at once in the day or two before a program is due, with respect to staff normal work constraints.

Please note that whether in person or on Zoom, there can at times be some longer wait times for office hours. This is best avoided by starting all of your programming assignments early – as soon as they are available on Canvas – working incrementally every day – and not waiting until the last minute to seek help. With some office hours, if a student has been in office hours for 10 minutes and still needs more time, it may be necessary to go back into the waiting area and allow for some other students to be helped before continuing.

Note: after lecture, for classes that meet in person, is not a good time to ask questions about your assignments. Attend office hours and ask during those scheduled times. The teaching staff all have multiple work duties and do their very best to help, but keep in mind staff work schedules and other academic duties need to be respected.

Some computing sites, when open, may have staff consultants who are available to help you at the site. They are good sources of help with email problems, using the computers etc. However they are not there to help you with your program design or with writing and debugging your program.

Whenever you seek help with your program in office hours, you must be able to quickly access your program file via a portable drive or on your laptop. Having a current print-out of the program, your input, and the printed output results of a run is often useful in face-to-face time if that is available. Do bring your program on a USB flash drive to staff office hours as teaching staff may often need to copy your program onto their office desktop to view it best.

Deadlines & Following Course Instructions

A large part of your grade in this course will be determined by whether or not you can meet deadlines and follow course instructions correctly. Both are very important in the world of professional programming and employment, and they are critical skills for this course.

Be sure to follow instructions for submitting to Canvas exactly. Program assignments will only be accepted when submitted correctly and on time to the course Canvas site. You are required to verify your Canvas submissions and check yourself using the verification process that you turned in the right file and that Canvas received it successfully. Missing files, unreadable files, and incorrect versions of files typically all receive a score of zero.

Backup Copies of Your Programming Assignments

Always make multiple backup copies (on a flash drive, in the university's file space, in your private cloud space, and printed) of your work!!! This is a course requirement, professional convention in CS and IT, and good common sense. The teaching staff may under certain circumstances have to ask you to produce your backup file copies. Note that many business IT departments maintaining large databases make it a habit to maintain 8 or more backup copies, on multiple types of media and in different physical locations. Get in that habit and start now, with at least 3 copies that you maintain for yourself.

Homework Deadlines

- **Due Times:** Each programming and homework assignment will be given a due date. Your work must be turned in correctly by **11:59 pm on the due date** to be considered on time.
- **Late Grace Period:** Work turned in after the due time but by 11:59 pm by 48 hours after the due date will be accepted but will automatically lose 20% of the total possible points. Work not turned in by this time will not be accepted and will automatically receive a grade of zero.

For example, if an assignment is due on a Monday, you must turn it in by 11:59 pm on that Monday for it to be considered on time. If it is turned in by 11:59 pm on the subsequent Wednesday, it will be accepted, considered late, and will incur a 20% point penalty. After this Wednesday deadline, the assignment **will not be accepted, and you will receive a grade of zero for it.**

- **Programming Project 6** must be turned in by the *absolute* due deadline; it cannot be turned in late because it is due at the end of the term and the teaching staff needs adequate time to grade it before final course letter grades are due.

- **Extension Policy:** The above grace period is designed to cover normal expected and unexpected delays. Deadline extension requests will only be considered if you contact the course instructor or your specific recitation instructor TA **during office hours**. Such requests will only be considered if they are made **before the assignment is due (the on-time due deadline)**, and will only be granted for medical or personal **emergencies**. In all cases, be prepared to substantiate any extension request with **written proof consisting of original and verifiable documents**, for example, a letter from your medical doctor, a hospital emergency room receipt, or written funeral home documentation. In most cases you will be required to present proof of your situation to the FSU Department of Student Support and Transitions (DSST) office for formal validation.

Extensions are typically not given via email, but if you are too sick or contagious to come in to campus, or if the course is being held online, send the course instructor or your recitation TA an email about your situation before the program due time.

Note that extension requests made on the project due date will only be considered if they are provided to the instructor at least 8 hours prior to the due deadline.

Extensions are **not** granted for reasons such as:

- You could not get to a working computer
- Canvas access became slow on the due date or even came to a halt around the due time
- You (or your dog, ferret, roommate, etc.) erased your files by mistake, or you lost your work somehow
- You had other course work or job commitments which interfered with your work in this course, or similar
- Your personal computer/laptop failed to work; if it does fail, go to an FSU computing site and/or the CS Majors Lab when they are open and do your work there

There will be no extensions whatsoever on the last assignment, Programming Project 6, except in extremely unusual circumstances such as emergency hospitalization.

You can avoid all problems by starting the assignments early and working on them incrementally, as discussed in class. If you are having trouble understanding the material or working on your assignments, we encourage you to attend office hours and recitations for help right away. Keep in mind that the course TAs are your primary source of help on the programming assignments.

Programming Project Grading and Submission

Programming projects will be graded in part for running correctly (doing all the required tasks and giving the correct results while running to completion successfully), and in part based on your program style, documentation, design and code efficiency.

If a program you turn in does not compile without errors under the required platform, for whatever reason, a penalty of 50 points may be taken off. If your program does not compile successfully, it indicates that you have not tested it properly before turning it in. Be focused when you turn in your program as even a single missing punctuation mark can cause your program to fail. During the grading process, if a trivial syntax error is found, the TA/graders may, at their discretion, take off -5 for each trivial error, up to three errors maximum. If more errors are found, or more serious errors are found, the penalty will be the -50 points.

All programming assignments **must** be submitted by the deadline in electronic form, and verified by you, according to class requirements, to the course Canvas site. Canvas will allow multiple submissions per assignment. **Only the last file submitted will be graded.**

Note that no programming assignments will be accepted via email under any circumstances.

Always use a reliable, up to date laptop or desktop, and a reliable and up to date web browser, to submit assignments and take exams (if exams are online) in this course. FSU ITS recommends Firefox or Chrome, not Safari. This technology is required in this course. Never submit any work using a mobile phone, notepad, or similar. Doing so is at your own risk and may result in a zero score as such devices are not reliable with Canvas. Note that if class exams are online for any reason, Safari cannot be used for these exams as it has known bugs.

Academic Integrity

We follow the Florida State University Academic Honor Policy. Information about this policy follows in the "Important FSU Policies" section of this syllabus.

The basic course rule is that you may not give or receive substantial assistance for any work you are submitting as your own. In all cases in which we have reason to believe that cheating has occurred, we will submit relevant materials to appropriate university authorities for evaluation. If a violation of university academic standards has occurred, a zero score will be assigned on the project or exam in question and other sanctions will be determined as well.

Students are expected to uphold the Academic Honor Policy published in the Florida State University Bulletin and the Student Handbook. The Academic Honor System of Florida State University is based on the premise that each student has the responsibility (1) to uphold the highest standards of academic integrity in the student's own work, (2) to refuse to tolerate violations of academic integrity in the university community and (3) to foster a high sense of integrity and social responsibility on the part of the university community.

Keep this in mind: If you are having trouble finishing an assignment, it is far better to do your own work and receive a low score than to go through an academic integrity investigation and suffer any penalties which may be involved, which can be very severe, and may have a serious impact for many years into your future.

What is cheating on a programming project? (a few examples)

- having someone else write your program, in whole or in part
- copying a program someone else wrote, in whole or in part
- collaborating with someone else to the extent that the programs are identifiably very similar, in whole or in part

In all of the above, it is not relevant whether the "someone else" is a friend, a "tutor," a complete stranger, a textbook other than the required book specified for this course (if you use it, cite it!), a web site or any other media.

In this course, all programming projects are to be done ON YOUR OWN unless otherwise stated in writing by the instructor on the assignment write-up itself.

What is not cheating? (a few examples)

- talking to someone in general about topics and concepts involved
- getting help with the specifics of C++ language syntax and semantics
- utilizing information given to you by the teaching staff of the course, for example copying a paragraph describing the program from the assignment write-up we provide
- copying parts of code from the required textbook(s) used this semester in this course; you would cite as a reference the textbook and page(s) used in your program comments

Generally speaking, talking about assignment work is ok; *sharing, using, looking at or reading ANY form of printed, written, electronic or hand-drawn material is typically a violation* of academic integrity policies. Obtaining material from the internet and submitting it as your own work is considered a violation. Note: for your own security, never post your own code on the internet, anywhere. In addition don't leave your program files or printouts on lab computers or in any public location, including trash cans!

The course teaching staff will regularly examine all class assignments turned in using software plagiarism detection software systems and methods.

Grading and Exams

Final grades will be based on the total points earned on six programming projects and two exams.

You are required to take all course exams at the scheduled times. If a serious medical or personal emergency is not involved in your missing an exam, you will receive a zero for that exam. If a serious medical or personal emergency results in a missed exam, you must attend the course instructor's office hours and substantiate any related request for special consideration with **written proof consisting of original and verifiable documents**, for example, a letter from your medical doctor, a hospital emergency room receipt, or written funeral home documentation. In most cases you will be required to present documentation to the FSU Department of Student Support and Transitions office (DSST) for validation.

If you anticipate an exam in another course or a religious holiday which conflicts with an exam time, you must notify the instructor **at least two weeks before the exam date in person during office hours, or the situation will not be considered**. The exam dates are given at the beginning of the term so that you can avoid scheduling job interviews or other commitments on exam days, hence job interviews, travel etc. are not considered valid reasons for missing an exam. The last exam is normally given according to the FSU Registrar's final exam schedule and there will be no exceptions. Under *no* circumstances may any student take any exam before the class group does.

Attendance and Participation

Attendance at all lectures and recitations, whether in person or online (when available), is necessary to do well in this course. If you do not attend, you will miss important information which is vital for course assignments and exams, at your own risk. Attendance and participation for a face-to-face class means being there in person and participating. Our class lectures and recitations are face-to-face.

It is against university policy and academic principles to register for two classes which meet at the same time, whether in whole or in part. The teaching staff of this course will make no special arrangements for anyone who somehow registers for overlapping synchronous course meetings.

All students must attend the recitation section that they are officially registered for. When face-to-face, the recitation rooms are small and are set up to have exactly one computer per student who is officially registered.

Grading Questions

All grading questions involving all course work must first be discussed with your recitation instructor TA. The first step is always to ask your TA why the points were deducted, so that you understand why the points were deducted. The goal of all university courses is for students to learn why points were taken off so that they can do better work in the future, and in this course, learn to be better programmers.

If the questions involve a re-grading request, because you think a verifiable error has occurred in grading, for example, the points were not added up correctly, then **you must attach a cover page** to the

printed assignment or exam, where the latter describes your concerns and is specific about just what you are contesting and why you think it deserves re-grading. Your recitation instructor will then consider your request, make changes if appropriate, note these on the cover page, and then record a grade change (if it differs). If you remain unsatisfied, then bring your printed assignment and the cover sheet (as already annotated by your recitation instructor) to the course lecture instructor in person during scheduled office hours.

Note that “re-grades” are not done upon student request. The course teaching staff must verify that there is a valid reason for a re-grade, and that will be the decision of the teaching staff.

If you ask that any part of an assignment be reconsidered and re-graded, the entire assignment will be re-graded, therefore the assignment grade may go up or it may go down.

All requests to review the grade for a particular item **must be made within 7 days** after the date that your assignment grade or an exam score is provided to you, otherwise they will not be considered. Special deadlines will be used for project 6 (and possibly project 5) and these will be announced when needed. FSU emails may serve for written communications in most cases and digital media will serve instead of print for some situations.

Tentative Point Distribution and Course Grades

It is not likely that this will change, but circumstances might occur which would make changes necessary, at the discretion of the instructor. Points are allocated and weighted as follows:

Six Program Assignments 60% total points (P1 2%, P2 10%, P3 thru P6 12% each)
Exam #1 20% total points
Exam #2 20% total points

There are no letter grades for individual projects or exams. The final course letter grade is based on the weighted total of all points earned relative to the entire class group. Final grades are based on a straight scale if appropriate (that is, 90 through 100 is an A- or A; 80 through 89 a B-, B or B+, and so on). Depending on overall class performance including the class averages and class achievement relative to prior semesters, these cutoff points may be adjusted. Class grading will never be “worse than” the straight scale. That is, a 90 will always be at least an A-. Rank in the class as a whole will be relevant if a “curve” is used, which it often is. If a curve is inappropriate due to class statistics, typically the mean or median class group score would be used as the bottom cutoff for a B-, particularly if a curve is used which is more generous than a straight scale.

Please note that doing poorly in a course is not a reason for an incomplete (grade of “I”), and that the incomplete grade is not intended for use in cases of extended illness. Read the FSU Bulletin for a discussion of the policy on incompletes. If you are having problems in the course, talk to your course and/or recitation instructor as soon as you possibly can. It is also best to talk with your FSU advisors and other university resources in such situations.

Note that individual students cannot ever be given “extra work” or “extra credit” in a course as it violates university policies as well as universal academic standards of fairness to all students.

Some Tips for Doing Well in this Class

Your goals probably include to learn a lot, to complete the course, to be well prepared for future courses and jobs, and to receive a good letter grade. Our goal is for everyone in the class to learn as much about computer science and programming as possible and be well-prepared for future studies and career work. These goals truly are compatible! You can maximize your grade, and learn a lot at the same time, if you

- Attend all lectures and recitation sections whether online or face to face; many tips and hints about

projects are given during lectures and recitations, and if you miss them, it is your responsibility. If you do not attend and participate, you will not be able to earn as high a grade as someone who does.

- Be aware that many students who have taken programming classes before often lose points on exams and projects due to non-attendance and missed material. Do not assume you "know it all already" as that is almost always a major mistake resulting in poor course grades.
- Read the assigned readings (textbooks, handouts, email, web pages, etc.).
- Submit your work on time and verify it; even if a program does not work, turn in whatever you have done for possible partial credit.
- Start the projects early, work in increments, and get help as soon as you need it; don't wait until it's too late.
- Follow the program style, documentation, and efficiency guidelines carefully.
- Work through as many of the exercises in the textbook as you can to aid your learning and for programming and exam practice.
- Don't rely on public web sites for information. They were not created to help you in this specific course – our course required materials were created to help you do your best!

MANY students who have taken this class previously have said that there is no reason not to do well, because everything you need to do well is provided to you. Your instructors work very hard to make sure that this is the case. If everyone utilized what is provided and everyone got an A in this class, your instructors would be very happy, and you too!

We also hope that you will **HAVE SOME FUN** while learning a lot from this course.

Vital FSU Policies

- **University Attendance Policy:** Excused absences include documented illness, deaths in the family and other documented crises, call to active military duty or jury duty, religious holy days, and official University activities.
- **Academic Honor Policy:** The Florida State University Academic Honor Policy outlines the University's expectations for the integrity of students' academic work, the procedures for resolving alleged violations of those expectations, and the rights and responsibilities of students and faculty members throughout the process. Students are responsible for reading the Academic Honor Policy and for living up to their pledge to "...be honest and truthful and...[to] strive for personal and institutional integrity at Florida State University." (Florida State University Academic Honor Policy, found at <http://fda.fsu.edu/Academics/Academic-Honor-Policy>)
- **Academic Success:** Your academic success is a top priority for Florida State University. University resources to help you succeed include tutoring centers, computer labs, counseling and health services, and services for designated groups, such as veterans and students with disabilities. The following information is not exhaustive, so please check with your advisor or the Department of Student Support and Transitions to learn more.
- **Americans With Disabilities Act:** Florida State University (FSU) values diversity and inclusion; we are committed to a climate of mutual respect and full participation. Our goal is to create learning environments that are usable, equitable, inclusive, and welcoming. FSU is committed to providing reasonable accommodations for all persons with disabilities in a manner that is consistent with academic standards of the course while empowering the student to meet integral requirements of the course.

Students with disabilities needing academic accommodation should:

1. register with and provide documentation to the Office of Accessibility Services; and
2. request a letter from the Office of Accessibility Services to be sent to the instructor indicating the need for accommodation and what type; and,

3. meet (in person, via phone, email, skype, zoom, etc...) with each instructor to whom a letter of accommodation was sent to review approved accommodations.

Please note that instructors are not allowed to provide classroom accommodations to a student until appropriate verification from the Office of Accessibility Services has been provided.

This syllabus and other class materials are available in alternative format upon request.

For the latest version of this statement and more information about services available to FSU students with disabilities, contact the:

Office of Accessibility Services
874 Traditions Way
108 Student Services Building
Florida State University
Tallahassee, FL 32306-4167
(850) 644-9566 (voice)
(850) 644-8504 (TDD)
oas@fsu.edu
<https://dsst.fsu.edu/oas>

Instructor's note on course exams: any requests for specific special exam arrangements due to a registered disability must be brought to the course instructor at least two weeks prior to the exam date, or they will not be considered. In addition students must follow all rules and procedures set forth by the FSU OAS.

- **Free Tutoring from FSU:** On-campus tutoring and writing assistance is available for many courses at Florida State University. For more information, visit the Academic Center for Excellence (ACE) Tutoring Services' comprehensive list of on-campus tutoring options - see <http://ace.fsu.edu/tutoring> or contact tutor@fsu.edu. High-quality tutoring is available by appointment, online and on a walk-in basis. These services are offered by tutors trained to encourage the highest level of individual academic success while upholding personal academic integrity. Instructor's Note on FSU ACE: the Academic Center for Excellence is a terrific source of help to students, providing multiple resources to help you succeed in school. See <https://ace.fsu.edu/>.
- **FSU Student Conduct Code:** Students are expected to follow the FSU Student Conduct Code in all interactions and situations at the university. See <https://dos.fsu.edu/srr/conduct-codes/student-conduct-codes>.
- **Confidential Campus Resources:** Various centers and programs are available to assist students with navigating stressors that might impact academic success. These include the following:

- Victim Advocate Program

University Center A
Room 4100
(850) 644-7161 (Available 24/7/365)
Office Hours: M-F 8 am-5 pm
<https://dsst.fsu.edu/vap>

- Counseling and Psychological Services

Askew Student Life Center, 2nd Floor
942 Learning Way
(850) 644-8255
Office Hours: M-F 8 am-5 pm
<https://counseling.fsu.edu/>

– Counseling and Psychological Services

Health and Wellness Center

(850)644-6230

Office Hours: M-F 8 am-5 pm

<https://uhs.fsu.edu/>

- **Syllabus Change Policy:** Except for changes that substantially affect implementation of the evaluation (grading) statement, this syllabus is a guide for the course and is subject to change with appropriate notice.