CS 2B Syllabus - Spring 2019

Intermediate Object Oriented Programming in C++

Hello. My name is Anand Venkataraman. I will be your instructor for this course. In this document I have laid out the various policies for our class. Please read it carefully and let me know if you have any questions.

Contents

Course Description

CS 2B continues where CS 2A left off. Students already comfortable with the C++ language will have an opportunity to master essential intermediate programming techniques using C++.

Class inheritance, templates, elementary data structures and the Standard Template Library are among the many topics that will be covered in depth. Successful completion of CS 2B is required to continue with CS 2C, which is the study of algorithmic analysis and data structures, the centerpiece of all C++-based CS degree programs and vocations.

A working facility with simple algebra as well as good written English comprehension skills are both strong advisories.

Are you ready for 2B?

Every quarter a number of students sign up and decide that they aren't ready for the course around Week 2. To help you plan better, I've released the first lab of 2B publicly.

Please click on: http://cs.psme.foothill.edu/c and enter the code duck in the box. This should show you the specs for the very first lab so you can self-calibrate your level of readiness for CS2B.

Read the spec carefully. You should pass all my tests.

Note that this is not where you would formally submit the lab at the end of Week 1. That will be done within Zybooks.

I will be happy to respond to any questions you have about this lab after the first day of the quarter. Save up those questions and post them in the forums once they open up.

Weekly Time Estimate

Many students find the material challenging if they don't allocate enough time for classwork and homework. Even if you are already facile with C++, you may find yourself needing significant time to work through assignments and implement them according to spec. Many past students who underestimated the amount of work this involved ended up with grades that they were not satisfied with.

Make sure you're not one of those students by making the right choice up front. Are you able to allocate enough time for this course or would you rather take it in a later quarter when it better fits your schedule?

Every week, you'll have one or more Zybook chapters or Canvas Modules to study and one or more lab assignments to complete.

If you have some programming experience already, expect to spend about 8-12 hours per week reading and/or attending lectures or watching videos. You will need to spend an additional 10-15 hours working on programming labs. To be on the safe side, budget about 25 hours per week (initially) for this course.

Preparatory Tasks

You must complete the first required task for this course no later than the first day of the quarter. This is just a simple 3-question quiz that **does not require prior knowledge of C++.** If you don't complete this task, you will be dropped and your seat likely given to a student on the waitlist. Consider this the equivalent of showing up to the first lecture. Not doing it will be treated as a no-show to the first lecture.

Text and References

Zybooks will be our primary text for this course. The data suggests that students enjoy a higher success rate with an interactive learning experience and guided activities as compared to traditional media. However, some topics that need to be covered aren't in Zybooks. So I've included Michael Loceff's and my modular material in Canvas for those topics.

The Zybook we'll use is CS 02B: Object-Oriented Programming Methodologies in C++. It is mandatory reading and contains most of the labs, which you will submit directly into it.

To get started with your Zybook:

1. Sign in or create an account at learn.zybooks.com
2. Enter the following zyBook code: FOOTHILLCS02BVenkataramanSpring2019
3. Subscribe

A subscription costs $77. Students may begin subscribing the week before the quarter starts and the cutoff to subscribe is the day before the quarter ends. Subscriptions will last until the end of the quarter. I'm told they can be extended for a nominal fee.

In addition to the Zybook and Canvas Modules, there is a recommended text for the course. It is *Absolute C++* (any edition at least as recent as the 2nd), by Walter Savitch, Addison Wesley. However, you can use any C++ textbook or resource that fits your style and budget.

You can order the recommended text through our bookstore at http://books.foothill.edu/, phone: (650) 949-7305.

Another recommended reference that may help with style is *The Elements of C++ Style,* by Misfeldt, et al, Cambridge University Press.

Important Note

You'll notice that we start the course at Chapter 8 of your Zybook. We already covered chapters 1-7 in CS 2A and we'll pretty much take all that knowledge for granted. That is, we won't have time in this course to go over basic material. We'll assume you already know it and will start building on top of it. If you're not comfortable with this, please consider registering for 2A instead to avoid disappointment later. Remember that learning to program should be as enjoyable as programming. There is no point in hurrying through the learning process.

Typical Routine

One assignment is due each week, most weeks. We'll try and stick to the below activity schedule as closely as possible. But minor deviations are possible to adapt for specific requirements of your section of the course.

Assignments are worth varying amounts of points depending on their difficulty. At the end, the grand total for all labs will be scaled to 70%.

Week	Canvas Module	Topic	Read	Labs	Due	What
0	-	Before you begin	Zybook Ch 1-7	Optional	-	-
1	1	Pointers and Memory	Zybook Ch. 8	Lab 1: Zy 8.16	Sun, 14 Apr	Playlist (Zy)
2	2	Recursion	Zybook Ch. 11	Lab 2: Zy 11.10	Sun, 21 Apr	Hanoi
3	3	Bitwise Operators	Canvas M3	Lab 3: Zy: 16.25	Sun, 28 Apr	CA1D
4	4	Trees, Deep copies	Canvas M4	Lab 4: Zy: 14.10	Sun, 5 May	Cloning Trees
5	5	Exceptions & Operator overloading	Zybook Ch. 12	Lab 5: Zy 12.5	Sun, 12 May	Complex Nums
6	6	Review/Midterm	All above	-		
7	7	Inheritance and Chaining	Zybook Ch. 10.1-10.3	Lab 6: Zy: 10.8	Sun, 26 May	Shapes 1
8	8	Polymorphism & Virtual Fx	Zybook Ch. 10.4-10.7	Lab 7: Zy 10.9	Sun, 2 Jun	Shapes 2
9	9	Streams and Templates	Zybook Ch. 9 & 13	Lab 8: Zy 13.4	Sun, 9 Jun	Efficient Queues 1
10	A	The STL	Zybook Ch.14	Lab 9: in Canvas	Sun, 16 Jun	Efficient Queues 2
11	B	Sorting & Review	Zybook Ch.15	All extra credit work	Sun, 23 Jun	All extra credit due
12	-	Final Exam	All above	-		-

Assessment

Your final grade will be based on programming lab assignments (Total scaled from max to 70%), participation (scaled to 5%) and exams (scaled to 25%).

The assessment has been designed to test both conceptual understanding and knowledge of language features. The labs emphasize the latter and the exams emphasize the former. The idea is that you should be able to get a passing grade by doing well in the labs, but in order to get into A-grade territory, you have to demonstrate a solid grasp of the concepts. An A+ is possible if you truly enjoy programming, program in your spare time for fun, and make the effort to independently look up, discuss (in the forums) and learn topics I will announce from time to time in announcements.

If you're focused solely on your grade and do everything flawlessly by the book, but fail to demonstrate good conceptual understanding, you will likely not get an A in this course. Keep this in mind as you decide whether this quarter is the right time for you to be doing CS 2B.

For an	A+	A	A-	B+	B	B-	C+	C	D	F
You need (%)	97	91	88	86	80	78	75	67	60	< 60

Tests (25%)

There is a midterm exam during the Thursday of the middle week **(Thu May 16)**, and there is a final exam during the Thursday of the last week **(Thu Jun 27)**. Both tests will be administered via Canvas and can be taken remotely. Further details about the tests will be given at the appropriate time.

Attempting the tests implies an acceptance of the Foothill Honor Code which means you agree not to cheat on it.

Labs, exams and quizzes submitted after their deadlines without prior arrangements are for personal edification only. Their scores will NOT contribute to your final grade

Participation (5% of final grade)

F2F students: You should be actively engaged in the classroom in order to receive participation points. I will calculate it based on your attendance and class activities.

Online students: Participation comes from an activity measurement formula used by Canvas and Zybooks. Meaningful contribution in Canvas is at least one non-frivolous post in the discussion forum, or a serious answer to a question posted by a fellow student or me.

Programming Labs (70%)

This quarter, I want to try an experiment: You have a number of labs of varying levels of difficulty worth varying amounts of points. At the end of the quarter, I'll do the following:

1. Add up all lab scores PLUS all extra credit (for each student).
2. Let X = average of the top 3 totals. X will be the cap. Anything over X won't count since you'd be maxed out. (How many students will not be helped by this capping?)
3. Scale X up to 70% and all other lab totals correspondingly.

This means you don't have to ace all labs in order to get the full 70% from labs. But you do have to do as well as you can. You want to be one of the top two lab totals for a guaranteed 70% - Can you prove this (in the forums)? You may get extra credit.

To prevent collusion I'm gonna have to include this caveat: If the maximum total is less than half of the possible total, this scheme is off. I'll fall back on using the possible total as your 70% max. (You don't have to use this caveat in your mathematical proof from the previous para).

Submissions must be turned in via Zybooks using the procedure described in the Zybook.

I encourage you to additionally visit me with your code to get my feedback. There is an objective rubric in determining your lab score. But I can offer subjective feedback on your programming style, which, of course, you are free to disregard with no impact to your score for the lab.

If you don't know how to use a debugger, I'd say that's the first skill you should master. When you ask me or a tutor for help with your code, we'll ask you what you've already done yourself using the debugger. You should be prepared to answer questions that demonstrate a good grasp of what you're supposed to do.

If you have a bug in your code, I will NOT DEBUG IT FOR YOU, even if that means you won't ace the lab. Instead I may give you some hints. Debugging own code is an essential skill that aspiring programmers must learn and enjoy - Yes, enjoy!

Now would be a good time to read and understand Foothill's Academic Integrity Policy. I strongly recommend you do so now. We take plagiarism very seriously and you don't want a permanent record on your transcript.

Extra Credit

From time to time, I'll present challenge coding problems in class and online which you can answer to get extra credit. I'll announce the precise way in which each such challenge should be attempted and submitted at the appropriate times. I'll make a note of your extra-credit work and apply EC points at the end

of the term after the final before the grades are finalized.

Programming style

My personal preference for program formatting is the C++ equivalent of the classic K&R style for C. It's not imperative that you follow the K&R style. I'm ok with any consistent and clean styling/formatting of your programs.

Compilers

In this class, you can write most of your warm-up and practice code within a programming environment made available inside of Zybooks. When you get around to implementing labs that are non-trivial, you can look into installing a desktop IDE. These labs will need to be submitted into Zybooks as multiple files.

I recommend using Microsoft Visual Studio/C++ (Community Edition) for Windows users or Xcode for Mac users. Our F2F students will find that our classroom is equipped with one computer per student and these computers (PCs) have an IDE installed on them. You're welcome to bring your personal laptops to class and use them. I can help you with issues with either of the two IDEs above. In addition, we may also have access to a Unix/Linux command line environment where you can compile and run your programs using g++. Again, this is an environment where I will be able to help you.

If you are facile on another IDE, you are welcome to use that, instead. However, my assistance in the forums regarding compiler specifics will be limited to Visual Studio, Xcode and Linux/g++.

Communication

Please use public discussion forums in Canvas for any question or comment that relates to the class (except questions of a private nature). If you have a confidential question (grades or registration) use the Email feature of Canvas by first clicking on Inbox (in your left-nav), and then selecting me as the recipient of your message. You will be able to attach files to your message.

You'll find it rewarding to engage in dialogue with your classmates in the Discussion Forums, which you can reach by clicking on the Discussions link in your Canvas page for this course. I'll check the discussion forums often and answer any important open questions that no one has yet answered.

I also encourage you to meet with each other after class, set up private study and programming groups and work on independent (non assignment) programming challenges outside of class. I'll give you a few interesting challenges from time to time. Some of these may earn you extra credit.

Feel free to answer your fellow student questions, even if you only have a guess as to what the answer is. It's great to engage in conversation with each other in this manner.

Contacting me

You can email me at anand@fhda.edu. If you are on campus and would like to see me personally during my office hour, my room number is 0x1021 (in hexadecimal). If you're enrolled in 2B, you probably already know how to decode that into decimal.

My office hours are:

- Mon/Wed: 10-11am (online)
- Tue/Thu: 10-11am (in my office)
- Other times by appointment

This information is also posted on my departmental page and you are welcome to stop

by to ask me questions during that time. My preferred way of being contacted at other times is via the private messaging feature in Canvas.

STEM Success Center

If the online forums here are not enough, please visit the STEM Center page, try to make time to meet with a tutor. These people are qualified to help you with assignments or modules without giving you an answer that will short-circuit your learning process.

Room 4213 in the STEM Center will have CS tutors at various times each day. The STEM Center is also a place on main campus where students without their own computers can do their lab work. The schedule for the STEM Center and its tutors is posted on the main STEM Center web page. Please enquire about online computer science tutors. Peruse support material, help and services offered to you FOR FREE by our awesome STEM Success Center - paid for by the division.

Other Resources

Professor Elaine Haight maintains a blog called Opportunities for CS students. It contains announcements of internships, scholarships, free software offers, pertinent public lectures, etc. Bookmark it and visit it frequently.

Course outline and SLOs

You can access the official course outline of record for all CS courses here.

Student Learning Outcomes for this course are:

1) A successful student will be able to write and debug C++ programs which make use of inheritance, i.e., the "is a" relationship, common to all OOP languages. Specifically, the student will define base and derived classes and use common techniques such as method chaining in his or her programs..

2) A successful student will be able to use the C++ environment to define the basic abstract data types (stacks, queues, lists) and iterators of those types to effectively manipulate the data in his or her program.

Disability Resource Center

Foothill College is committed to providing equitable access to learning opportunities for all students. Disability Resource Center (DRC) is the campus office that collaborates with students who have disabilities to provide and/or arrange reasonable accommodations.

If you have, or think you have, a disability in any area such as mental health, attention, learning, chronic health, sensory, or physical, please contact DRC to arrange a confidential discussion regarding equitable access and reasonable accommodations.

If you are registered with DRC and have a disability accommodation letter of accommodations set by a DRC counselor for this quarter, please use Clockwork to send your accommodation letter to your instructor and contact your instructor early in the quarter to review how the accommodations will be applied in the course.

Students who need accommodated test proctoring must meet appointment booking deadlines at the Testing Center:

- Exams must be booked at least three (3) business days/weekdays in advance of the instructor approved exam date/time.
- Finals exams must be scheduled seven (7) business days/weekdays in advance of the instructor approved exam date and time.

Failure to meet appointment booking deadlines will result in the forfeiture of testing accommodations and you will be required to take your exam in class.

Contact the DRC if you cannot find or utilize your MyPortal Clockwork Portal. DRC strives to provide accommodations in a reasonable and timely manner. Some accommodations may take additional time to arrange. We encourage you to work with DRC and your faculty as early in the quarter as possible so that we may ensure that your learning experience is accessible and successful.

To obtain disability-related accommodations, students must contact Disability Resource Center (DRC) as early as possible in the quarter. To contact DRC, you may:

Visit DRC in Building 5400, Student Resource Center:

- On the web: http://www.foothill.edu/drc/
- Email DRC at drc@foothill.edu
- Call DRC at 650-949-7017 to make an appointment

If you already have an accommodation notification from DRC, please contact me privately to discuss your needs.

Important Dates

For a list of important dates for the winter quarter, see the official college page here.

Refer to the orange table in page 3 of this syllabus for dates regarding lab submissions, the midterm and final exam.

Happy Hacking!

& (anand)