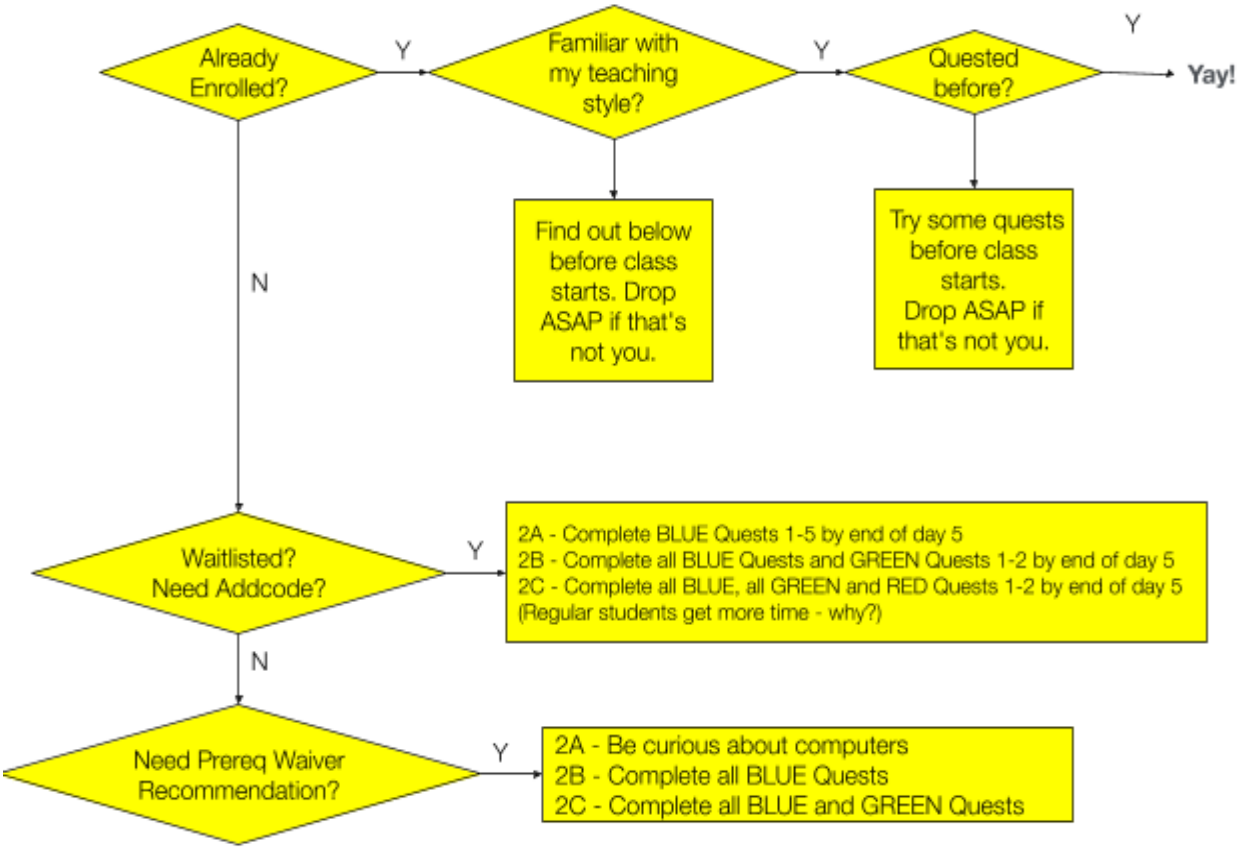
# CS 2C Syllabus - Spring 2021

*Algorithms and Data Structures in C++*

Hey there. My name is Anand. Please read this syllabus carefully. You should especially read it if you have never taken a class from me before.

## Jump To

# First Things First




**My teaching style:** (1) Highly hands-off (2) Will NOT debug your code for you (3) Nobody else is allowed to debug your code either (4) You will largely help each other in class (5) I will simply be an active observer in the forums to make sure you don't say anything silly or bad and find out how much value you have contributed to OTHERS at the end of the quarter. (6) Very occasionally, I may comment in the forums to answer unanswered questions or correct wrong suggestions (7) Your participation scores are worth 15%, and they are SUBJECTIVELY determined by me and you can't find out how much you will get. (8) 1-1 Office hours are ONLY for discussing personal/confidential issues. (9) Asking a private question of general value to all students will get an answer along with compensatory negative points in my private spreadsheet.

## What you're signing up for

Here is the way I approach teaching: I don't think I have ever been fond of stuffing knowledge down the throats of people who don't want it. But I believe that chances are high that if you've already sampled CS2A and CS2B and have come back for more, this is most likely your cup of tea and you want more of it.

If you're doing the quests for your own edification at your own pace - awesome. That's the way to be. No fun being stressed-ful when you're questful.

If you're just now venturing into real C++, and decided to dip your toes into CS2C, *be prepared to do a lot of self-learning*. If you think you are at risk of finding this class overwhelming, please consider restarting at the tiger asap and by solving each quest on your own. If you reach the RED quests that way rather than through the freebie password you got somewhere, you'll likely find it more rewarding.

On the positive side, you CAN always try to get the help you need by posting in our moderated 2A or 2B subs in addition to our section's 2C sub. They're not restricted to active Foothill students.

Note the following **essential** prerequisite skills.

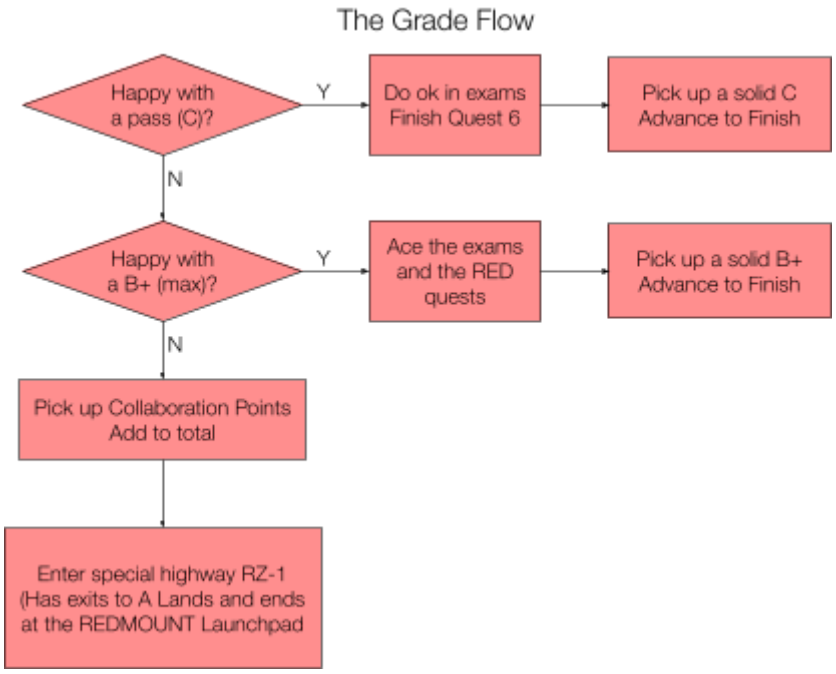
- Looking up, evaluating, and using information on the net
- Following simple directions correctly (e.g. creating a subreddit user according to some requirements)

If you don't have these skills yet, take some time to learn them first (usually, quite easy) and then enroll.

This course provides a systematic treatment of advanced data structures, algorithm analysis and abstract data types in the C++ programming language. Coding topics include building ADTs on top of the STL templates, vectors, lists, trees, maps, hashing functions and graphs. Concept topics include searching, big-O time complexity, analysis of major sorting techniques, top down splaying, AVL tree balancing, shortest path algorithms, minimum spanning trees and maximum flow graphs.

A working facility with simple algebra as well as good written English comprehension skills are both strong advisories to get the most out of this course.

# First Things Second - General Matters




If you just want a taste of life in the fast lane, you should complete quests 1-6, and do ok in the quizzes and exams. No need to participate in the forums. You'll get your passing grade.

For a B+, you should do well in your quizzes and exams, and also complete all 9 quests, without necessarily acing anything.

To get to A Territory, you absolutely need to earn collaboration points, which you can assume to be a proxy for evidence of productive collaborative programming, which is an important skill to have in the real world. How, then, can you earn these lucrative collaboration points?

# How to get collaboration points

I know of two ways: (1) You can participate *wholesomely* in f2f class lectures (when offered) and (2) You can participate *wholesomely* in the class discussion forums on reddit.

What constitutes wholesome participation? Wholesome participation is when you are both a giver and a receiver. You not only ask questions, but also provide answers, courteous help, and useful directions without compromising your classmates' pleasure of finding out themselves.

There is a subjective component to the collaboration points. By signing up to this course you agree that you accept my subjective evaluation of your collaborative input to award suitably normalized collaboration points towards your final grade. You will not know exactly how much it is. So please don't ask me.

Wholesome f2f class participation is when all of the following conditions are met:

1. You are present during class
2. You ask interesting questions
3. Your camera is switched on and you are "live" on it (ok to switch off for short breaks) - if virtual
4. You volunteer to drive or co-drive during at least one class (see past lectures on our youtube channel to find out what this means; may not always get a chance)

As before, if you're not aiming for A Territory, it's ok to attend class anonymously with your camera off (or not attend at all). You can still pass this class.[1]

Wholesome forum participation, which can be in addition to class participation is when your contribution to forum discussions are constructive, productive, and encouraging for those who are trying hard. To be on the safe side, consider adapting one of the following templates for your subreddit posts (customize as necessary).

Here is a useful strategy for those shooting beyond B+: Don't look up a solution on the sub. Try to solve it yourself. If and when you have to hit the sub, do it to ask a fresh new question in your own words, describing your particular problem. Only refer to past subreddit posts in answers to someone else's question (or other kinds of posts, like psas).

**Note**: Whether you're asking or answering, it's perfectly acceptable to say "*I tried u/XYZ's solution at [link to post] and it worked for me. My understanding of it is ABC. Happy to receive more light on this topic from someone who understands it better.*"

[1] You can contact me beforehand if you are unable to be present on camera during classes.

## Template 1 (Asking a question)

Use something like the following template to ask questions. Don't refer to previous posts on the subreddit unless you're answering someone else's question.

> Hiya folks,
>
> I was trying to access XYZ in ABC in this quest, and kept getting my donkey bitten.
>
> I suspect that it's because of UVW. I tried to check by doing MNO, but it OMG.
>
> Help please?
>
> - eternally_grateful_all_of_this_year_2021

## Template 2 (Giving an answer)

Use something like the following template to answer someone else's question. You can refer to past posts and comments here. You can post your answer *even if someone else has already posted one* as long as it is not the same perspective on the same answer.

> Hey eternally,
>
> I feel your pain. Have you tried IJK? Maybe share a screenshot of your error (without the password) and I can help better?
>
> From what you describe, it seems like you're having the same issue as this poster here:
>
> [ Link(s) to past reddit posts ]
>
> What's going on is that MNO goes out of bounds in the array when you use the -- operator on a size_t. I hope this helps.
>
> Let us know,
>
> - always_glad_to_help_today_0405.

## Template 3 (General Banter)

Use something like the following template to share your wonder. You may refer to past and external posts freely here.

> Hello,
>
> Recently I was browsing past reddit posts, waiting for the next Black Mirror episode, and came across this incredibly helpful observation by a past student (u/UVW).
>
> When we pop an item off the stack in Quest 8, we don't return it to the caller. Instead we destroy it. Ever wonder why? Well I did.
>
> The reason has to do with: ABCDE...
>
> I hope you are as surprised as I was when I learned it.
>
> Check it out: [ Links to past and other external posts ]
>
> - ever_curious_at_this_moment_00

## Template 4 (Initiating discussion about something in a f2f class)

Use something like the following template to post something about the content for that week from previous lectures and external sources.

> Hello questers,
>
> This week we were supposed to play around with BCE and CED. These topics are discussed in the Week N youtube channel.
>
> 1. https://youtube.com/[nonlinearmedia channel]/... watch?v=ABCDE&t=34m23s (from Winter 2021)
> 2. https://youtube.com/[nonlinearmedia channel]/... watch?v=ABCDE&t=34m23s (from Spring 2020)
> 3. https://youtube.com/[somewhere else]/... watch?v=ABCDE&t=42m3s
>
> So I can understand why we say "EFG" whenever we check for "PQR", but it seems to contradict the other video (#3) from XYZ.edu. In fact, later in video #2 (around 43m) it seems to contradict what it said earlier at 34m23s.
>
> Anyone care to shed more light on this?
>
> - bored_with_linear_content_2021

Refer to past questers whenever possible (and not asking a question). You can credit them by tagging their username in addition to sharing their link.

And, ofc, you don't have to use the exact same words as these templates. Say it however you want. They just show you the critically required elements of your posts and/or comments.

# Common issues in getting participation points

**"I found the answers to my issues in the subreddit. So I didn't have to post for the answer."**

> Of course you found the answer there. I'd be surprised if you didn't. Unfortunately, that means that you didn't struggle to find it yourself. Using help from past questers without overtly acknowledging them may not be a nice thing to do, but it certainly won't get you any collaboration points.
>
> By doing this you're implicitly suggesting to me that you're driving in the B+ lane. You can switch lanes, but it is difficult to switch into a faster lane unless you've been driving as fast for a little while before the lane change.

**"I'm uncomfortable with showing my face on published videos of our class lectures (and so my camera has to be off)"**

> It's perfectly fine as long as you tell me beforehand and we arrange for a suitable alternate way to establish your active presence. Otherwise, *enrolling in this class is implicit permission to publish videos in which you may appear as a student.*
>
> Why not turn this to your advantage and make it a bullet in your portfolio. E.g.
>
> - ...
> - Youtube link in which I coded live - [ link to vid ]
> - ...

# MHM Contest[2]

This quarter I'm not running an MHM contest. If there is enough enthusiasm in nominating people, let me know in Week 11.

## Winners of the Winter 2021 MHM contests:

- CS2A - Tom D (https://reddit.com/u/Tom_D3000)
- CS2B - Huzaifa Beg (https://reddit.com/u/huzaifa_b39)
- CS2C - John Vicino (https://reddit.com/u/JohnCVicino)

[2] Very important notes: (1) Foothill college is not liable for anything I promise you. (2) The money is from my personal post-tax income. No receipt is necessary. However, you may be required to report it as income.

# Assessment

If you're doing this course for kicks, or other fun reasons, you can skip this section.

If this course is offered for a grade, and you are taking it for a grade, then, your final grade will be based on programming quests (scaled to 60%), participation (scaled to 15%) and exams (scaled to 25%). I will then use the absolute grading scale below:

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

The assessment has been designed to test both conceptual understanding and knowledge of practical issues. The quests emphasize the latter and the exams/quizzes emphasize the former. The idea is that you should be able to get a passing grade by doing well in the quests and moderately well in everything else, but in order to get into A-grade territory, you have to demonstrate a solid grasp of the concepts and good class citizenship[3]. An A+ is possible if you truly enjoy programming, program in your spare time for fun, and take the trouble to independently look up, discuss (in the forums) and learn topics I will announce from time to time in announcements.

With that said, 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.

In this course there will be:

- 9 Mystery Quests you will solve at the average rate of about one per week (your own pace). These quests need to be solved using C++ (worth 60%)
- 1 midterm and 1 final exam (worth 25%)
- Online participation (worth 15%)

# Operational details

Canvas is our hub to coordinate some activities and take online exams. Most of the rest of our work will happen at other online locations, including youtube, zoom, quests, reddit, etc. You will start your adventure in Foothill's class by posting an introductory note (required) about yourself. You can simply reply to my own introductory post if you prefer.

Other than that, we will use publicly available resources with discretion, courtesy and efficiency to share information and help each other.

Don't say anything that you'll end up regretting. OTOH do try to let your natural genuine curiosity shine through. Maintain your profile on our subs as you would if you were a professional and it will free up a lot of your time.

[3] "How does good class citizenship contribute to learning?" you ask. Good question. I'm using it as a suitably weighted proxy for confidence in a person's conceptual knowledge. In the past, I noticed a good correlation between a person's understanding of a concept and their willingness to explain it to someone else.

I will try to remove posts that I deem (in my subjective opinion) to be a liability to your future self. But you can't rely on it. Best to be helpful, courteous, informative and only post useful tips, tricks and observations. They usually have more lasting value.

- Participating and collaborating earns points (max 15).
- Not participating does not earn any points.
- Participating negatively by souring up someone else's experience earns negative participation points (no min).

## Face to Face Classes

My CS2B and CS2C sections are typically online. However, I am happy to schedule and offer f2f lectures on focused 2C topics provided there's at least five of you willing to attend (like a regular class). You should pick a single topic and we'll live-code in class with one of you as the driver. If you haven't volunteered to drive a class like this before, check our recorded CS2A lectures on our youtube channel (nonlinearmedia). There are many examples to show you how it would go (except that we'll be coding more advanced problems).

Once we settle on a topic and participants, we can select a mutually convenient time.

## Mystery Quests

You will solve these at the public questing site (https://quests.nonlinearmedia.org). You have the entire first week to review concepts and earn your password. You can request the password from me any time during the first week (or try to earn it yourself starting at Fangs).

Each quest will give you a certain number of trophies. You can check your total trophy count at any time by visiting your personal scoreboard at the /q site (It will be wiped on the 1st of Jan, Apr, Jul, and Oct). *It will show you all your trophies, but only the ones you earned for 2C (RED ones) count for this course.*

The quests are set up such that the password to each quest is given out upon scoring a certain number of trophies in the preceding quest. However, I found that a few students were getting stuck in the lower numbered quests pounding away at them to eke out every remaining trophy before moving on, even though they had already earned the password. This is a bad strategy. Keep moving when you get a password. You can always come back to polish your previous quests when you have free time before the freeze date.

At the end of the quarter, your total trophy count will be capped at 210 and scaled from 210 to 60%. You can win AT LEAST that many trophies if you make it through to the last one. If you spend a lot of effort getting up to high numbers by the time you get to Quest 7 already, then you'll be close to getting burned out right in time for two of the funnest quests of all. So plan your time and effort wisely. It's not like your old quests are going to disappear when you move on.

## Exams

You will have one midterm exam on the Thu of Week 6 (May 13) and one final exam on the Thu of Week 12 (Jun 24). The midterm is worth 20 points and the final 40. Together, their combined score will be scaled from 60 to 20%.

These exams are objective style and will be administered via Canvas. You will typically have a window of time (18+ hours) during which you can begin these exams. But once you begin, the current version of Canvas does

not allow you to *pause* your exam and come back to it. The 1h (or 2h for final) timer cannot be stopped once you start it, until you hit finish.

All exams are open-book and can be taken anywhere you get a decent Internet connection. I don't recommend taking it on mobile devices.

I'm not going to be able to prevent cheaters from cheating. But keep in mind that cheaters only cheat themselves. Copying is a waste of your time. Few good software companies employ programmers based upon their qualifications if their demonstrated competence doesn't measure up to their stated expertise.

Besides, you'll find that copying robs you of a great opportunity to really learn the language and having a load of fun.

## College Recommendations

Many students who complete my CS2B or CS2C successfully ask me to write college recommendations for them. I don't write or make comparative recommendations for students, nor provide my opinion or evaluation of your current or future abilities. I do not share the grade you earned in my classes.

However, I can help you help yourself by giving you the chance to point the admissions officers at your work (e.g. your reddit posts).

## Preparatory Tasks

You must complete the first required task for this course by midnight of the first day of the quarter. This is just a simple 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.

Also, if you think you may be dropping this course, I urge you to drop ASAP so I can give your seat to someone else on the waiting list.

# Weekly Time Estimate

Programming, like all art, is not a 9-5 job. Sometimes you're on a roll and killing it. Other times, not so much.

I know how it is.

So there are no regular papers or labs due every day or week in this course. Rather, like real projects, there are deadlines you should strive to meet. You can plan your own time in your own way. Below is one suggestion:

| Week | Read References | Complete | Notes |
|---|---|---|---|
| 1 | ADTs review | Mystery Quest 1 | |
| | Algorithm analysis | Mystery Quest 2 | |
| 3 | Time complexity and Big-O | Mystery Quest 3 | |
| | General trees (and BSTs) | Mystery Quest 4 | |
| 5 | AVL Balancing and Splaying | | Quests 1-4 Freeze |
| | **Review/Midterm (Canvas)** | | |
| 7 | Hash tables Quadratic probing | Mystery Quest 5 | |
| | Sorting | Mystery Quest 6 | |
| 9 | Priority Queues, Heaps, Heapsort | Mystery Quest 7 | |
| | Dijkstra's and Kruskal's algorithms | Mystery Quest 8 | |
| 11 | A Maxflow algorithm | Mystery Quest 9 | Quests 5-9 Freeze |
| | **Final Exam (Canvas)** | | |

Every week, give yourself one or more topics to study and one or more programming quests 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. Budget an additional 10-15 hours for working on programming quests. To be on the safe side, budget about 25 hours per week (initially) for this course.

# Learning Resources

Rather than prescribe any particular resource,

- Refer to the list of topics for each week in the red table (ask in the subreddit to drill down into specific details you need clarification on).
- You should refer to sources (including the recommended text) to find out more about these topics.

My first resource suggestion is the book: *Data Structures and Algorithm Analysis in C++*, any Edition ≥ 2nd, by Mark Allen Weiss, Pearson. You can order it through our bookstore. Or get an online one.

The second is a past fork of CS2C modules that ex-prof Michael Loceff created when he taught this course. Thanks to Michael, I'm able to make these available to you.

In week 1, I will share the location of these modules. I'm hosting it outside of Canvas. Although a couple of revisions behind, much of it is still relevant to this course. It is essentially a *distillation* of selected topics from the text. But be aware of salient differences between the content of his modules (or the text) and what some of our quests require. This shouldn't be a problem if you understand the concepts. But it will be a problem if you don't.

As always, hit our sub, when in doubt.

Actually, that's not quite right.

When in doubt, try it out.

If you still just don't get it. Then hit our subreddit.

# Other Resources

The department maintains a blog called Opportunities for CS students. It contains announcements of internships, scholarships, free software offers, public lectures, etc.

# Lane's Lane

The Foothill STEM Center already provides fantastic assistance by making experienced CS tutors available for 1-1 real-time (synchronous) assistance almost 24/7 (via zoom) and generous hours in the STEM Center when the campus is open. Within the STEM Center, Lane Johnson hosts two special workshops each week focused especially on helping questers. Look for their actual hours on our sub, or simply check into the STEM Center sometime and ask for Lane.

# Canvas

This quarter, we will be using Canvas ONLY for the following:

- Reading announcements
- Introducing yourself with your reddit handle (your only required post in Canvas)
- Taking quizzes and exams
- Reviewing quest/test scores when they are ready (will be announced)
- Accessing virtual learning resources such as the STEM center, online tutorial rooms, etc.

Make sure your Canvas configuration settings are such that you get notified when there is a new announcement.

If you're aiming for a grade, make sure your reddit username matches the requirements.

I am not using the Discussion Forums feature of Canvas.

# Discussion Forums

This quarter, we'll continue to use our RED subreddit for all quest related discussions. Please note the following important information:

1. DO NOT SHARE personally identifying information of any kind. However,
2. No matter what your avatar's name, you must sign your posts with your first name (I strongly discourage unsigned posts. A reply should be able to start with something like "Hi John"
3. Your avatar name should start with your first name and an underscore, followed by your initial (or full last name) + some optional digits (Example: John_S0101)
4. You should never post your student ID (CWID) online. A lot of personal information about you can be unlocked by someone who has it.
5. If you have something negative to say about someone's post in the forums, you should direct your concern to me, not to the person in the forum.
6. KEEP IN MIND that these discussion posts will persist into the next quarter and later for future students. So everything helpful you say will help far more students than just your current classmates.
7. Use Canvas for anything not quest-related (enrollment, exams, modules, etc.)
8. No posting source code and fishing for answers. Debug your own code.

Keep this in mind: ANY user anywhere in the world can quest and post/discuss in our subreddits. So you may see posts and replies by users with anonymous names like *coding_lion*, *bat_girl* and such. All posts are subject to the same rules like *Johnny be good*, but only the ones with avatar names matching the spec in this syllabus will get collaboration points towards a grade.

# Getting started on your Mystery Quests[4]

The password to the first quest can be discovered by solving ALL of the BLUE and GREEN quests. If you are proficient in CS2A and CS2B concepts, this should be straightforward. You get 3 days of relatively low course load from the start of the quarter. Use it to discover this password. After that, it's your responsibility to not only find the password, but also catch up to the pace that the rest of your class may have reached (You can give up and request the password any time). If you're not all that confident with C++ yet, you better start already - this syllabus should be in your hands at least a week before class starts.

Passwords for subsequent quests will be automatically revealed upon *satisfactory progress* (as the machine sees it) in each preceding quest.

In order for rewards from a quest to count towards your total, you must have completed all previous quests. If you leave a hole in your trail of completed quests, then your total reward earnings is the sum of all rewards you earned before the first incomplete quest.

[4] "Where can I find these quests?" Hmm… that be yore first quest. Or metaquest.

## Bugs in your code?

Getting your code debugged by someone else is not allowed. That includes me, your tutors, teachers, friends, enemies or relatives. Debugging your own code is an essential skill that aspiring programmers must learn and enjoy - Yes, enjoy!

Of course, I can't police this. But your enrollment in this class signifies acceptance of this condition (in addition to being bound by Foothill's Academic Integrity Policy). You cannot send your code to me, a tutor, a friend or relative and ask them what the issue is. What you can do is:

1. Check our subreddit to see if others have had similar issues
2. Explain (in our subreddit) what you're trying to do
3. Describe in English the detailed steps you would need to undertake (pseudocode)
4. Describe the behavior of your program and ask why it diverges from (2) if it does

Sometimes, a tutor, a fellow student or I may get curious about your code and want to see it. Under these exceptional circumstances, you can share your code on request.

Sometimes it is also ok to post your code on our subreddit. Mostly, exercise good judgment regarding what can be shared. You want a fun and fulfilling learning experience. The best way to get it is to keep it fun and fulfilling for everyone. You wouldn't give away a movie's ending to a friend who's going to watch it. Why give them the solution to a problem when they can feel good finding it themselves?

## Extensions

Extensions don't make sense because the quests are self-paced. You just have to complete each by their "freeze" dates to get credit. After their freeze dates, you can still complete them, but not for credit. There's a LOT of time to complete these quests even if you have to take some breaks. So, please don't ask for extensions.

## 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

Use an IDE/compiler of your choice. But you'll find better support from me and the STEM center if you stick to one of the environments we know about (ask).

# Communication

Please use our sub for any question or comment that relates to the quests (except questions of a private nature). If you have a confidential question (grades or registration) you can email me. If you have a question that only makes sense with material you can find in Canvas (e.g. modules, syllabus, exams, etc.) then it makes sense to post that question in Canvas rather than our sub.

Try to meet with each other after class (even if virtually), 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.

I'm generally online and able to chat in real time M-Th 10am to 11am.[5] Most other times, you can reach me via messaging in Canvas, Reddit or by email. While on campus, my room number is `0x113d` (in hex). If you did my CS2A successfully, you know how to decode that into decimal.

One-on-one meetings are only for discussing confidential stuff. You cannot privately ask me for an explanation that is bound to be generally useful. And you cannot show your quest code to anyone (including me).

# 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 involving advanced data structures such as Lists, Trees, Graphs. They will be familiar with the use and implementation of algorithms for balancing binary trees, creating splay trees, minimum spanning graphs, finding the shortest path through a graph, and maximum flow through a network. They would also be familiar with the most common sorting algorithms and know the advantages and tradeoffs of each.
2. A successful student will be able to reason about the running time and derive properties of computer programs using precise mathematical terminology. Specifically they will be conversant with the Big-o notation and be able to craft efficient algorithms using the appropriate data structures to solve non-trivial computational problems.

[5] Times are in decimal, not binary. I kid you not, but someone did stop by my office at 2am once.

# STEM Center

As the quarter progresses, I'll post Foothill-specific information on Canvas regarding any special access to the STEM Center or tutoring services arranged by the division.

# 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 it to me and discuss accommodations..

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:

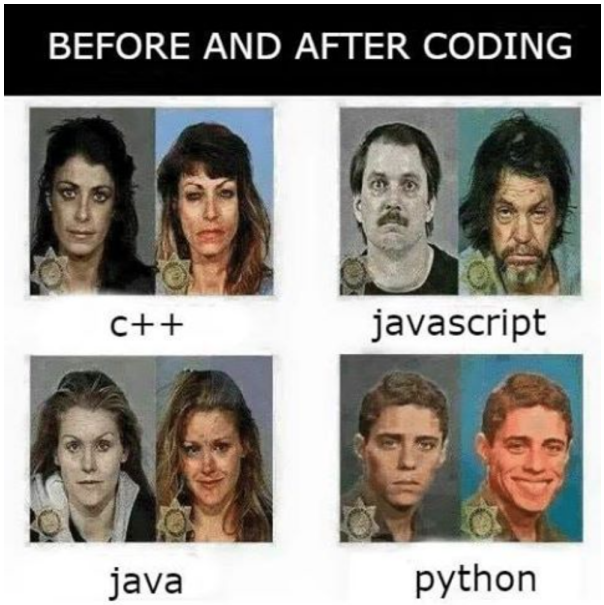
Visit Darcy in Building 5400, Student Resource Center: (physical visits are suspended during college closure)

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

# Important Dates

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

And here's a funny pic to cheer you up before you get wasted with sharks (in a good way).




Happy Hacking!

&