

Course Syllabus

[Jump to Today](#) [Edit](#)

Computing & Software Systems SKL 342: Programming Issues With Object-Oriented Languages

UW1-041

Friday 1:15pm-5:15pm

COVID-19 Related Resources

Face covering policy: [_ \(https://www.uwb.edu/academic-affairs/covid-policies\)](https://www.uwb.edu/academic-affairs/covid-policies) As we return to in person instruction, I want to acknowledge the difficulties of this past 18 months and inform you of my expectations for our safety as we begin this class together. To ensure the health and safety of the University campus community, face coverings are required to be worn indoors when other people are present regardless of vaccination status, and in all public and common areas, such as lobbies, hallways, stairways, restrooms, elevators, and classrooms. Eating and drinking in the classroom will not be permitted This requirement is in accordance with the University of Washington COVID-19 Face Covering Policy. For the purposes of this policy, a face covering must: Fit snugly against the sides of the face; completely cover the nose and mouth; bandanas and gators are not considered face coverings for this policy. Students who do not wear a face mask will be asked to leave the classroom. If you forget your face mask or refuse to wear one, I will ask you to leave the classroom. Repeated failure to wear a face covering may result in you being referred to the Student Conduct Office for possible disciplinary action. I hope that we can all agree to keep each other safe by wearing our face masks.

This is a in-person class. If necessary, the class will continue as remote learning. Please refer to the home page for accessing Zoom links and recordings.

Please refer to [the home page](#) for Zoom link details.

UW Bothell Coronavirus (COVID-19) resources: <https://www.uwb.edu/coronavirus>
(<https://www.uwb.edu/coronavirus>)

Name: Kelvin Sung

Email: ksung@uw.edu

Office: I will be at [this Zoom URL](https://www.uwb.edu/coronavirus)
(<https://washington.zoom.us/j/97352723875?pwd=cFBsdk92M0JLSXZjSks4V2dHdFhQQT09>)
during office hours (passcode: 1234).

Office Hours: Mon 12-2pm (or email for appointment)

Basic Course Information

Covers language and development/execution environment differences, including data types, control structures, arrays, and I/O; addressing and memory management issues including pointers, references, functions, and their passing conventions; object-oriented design specifics related to structured data, classes, templates, and operator overload.

Instructor

Kelvin Sung: ksung@uw.edu, office

Office Hour: Monday 9 - 11 AM or by appointment.

Class Discord

<https://discord.gg/wMQs9sRfQP> (<https://discord.gg/wMQs9sRfQP>): Qingran, our peer facilitator will be on-line during his office hours to answer questions. Otherwise, questions posted may not be answered in real-time.

Peer Facilitator:

Qingran Shao will be *on our course Discord*

Office Hours:

Wednesday 11am to 1pm [on Jan 12, 19, 26, Feb 2, and Feb 9]

Friday 3pm to 5pm [on Jan 7, 14, 21, 28, and Feb 4]

Course Objectives

The overall goal of CSS 342SKL is to transition students who are competent developing small programs in one object-oriented programming language to developing larger programs in another object-oriented

language. Currently, it is assumed that students are familiar with Java (or C#) and need to become familiar with C++, the language used in CSS 342/343. While the underlying concepts are similar, there are some differences in syntax and many differences in usage, particularly relating to memory management and the software development toolchain. This course is intended as an aid to your survival in CSS 342. The intention is to bring Java/C# students up to speed in C++ as quickly and painlessly as possible. Consequently, this course is dense, taught concurrently with the first half of CSS 342. There may be additional in-class work that focus on specific aspects of C++. This course will be run as part lecture, part hands-on lab.

Suggested Reading and References

- **C++ Reference Book:**
 - Bjarne Stroustrup, *The C++ Programming Language*, Forth Edition, Addison-Wesley, 2013. The canonical reference.
- **C++ Reference Sites:** Alternative is to keep the links of a couple of favorite reference sites around, e.g.,
 - For the language: <https://en.cppreference.com/w/cppreference.com> (<https://en.cppreference.com/w/>), or
 - For libraries: <http://www.cplusplus.com/reference/> (<http://www.cplusplus.com/reference/>).
- **Working on Unix/Linux:**
 - Harley Hahn, *Harley Hahn's Student Guide to UNIX*, 2nd edition, McGraw-Hill, 1996. If you're unfamiliar with Unix or Linux and want to learn more, get a book like this.
- **Software Engineering and Using C++ Tools:**
 - Herbert Schildt, *STL Programming from the Ground Up*, Osborne/McGraw-Hill, 1999. A very good introduction to the Standard Template Library, with lots of examples. However, it is not a reference; for example, it doesn't provide complete lists of methods for each class.
 - Nicolai M. Josuttis, *The C++ Standard Library*, Addison-Wesley, 1999. Much more of a complete reference than the Schildt book. Includes some examples, but is not intended as a tutorial.
 - Scott Meyers, *Effective STL: 50 Specific Ways to Improve Your Use of the Standard Template Library*, 1st Edition, Addison-Wesley, 2001. If you have read "*The C++ Standard Library*", why not this one?
 - Scott Meyers, *Effective C++: 55 Specific Ways to Improve Your Programs and Designs*, 3rd Edition, Addison-Wesley, 2005. The book you must read to become a C++ programmer.
 - Scott Meyers, *More Effective C++: 35 New Ways to Improve Your Programs and Designs*, 1st Edition, Addison-Wesley, 1996. The book you must read to become a good C++ programmer.

Course Schedule

Date	Topics
Jan 7	Lab 1: Toolchains, datatype conversion, pass by value/reference and file I/O issues.
Jan 14	Lab 2: C++ memory management: pointers, arrays, new and delete

	operators.
Jan 21	Lab 3: Class, constructors, inheritance and lifespan.
Jan 28	Lab 4: Copy constructors, assignment operators, deep vs. shallow copy; templates and iterators.
Feb 4	Lab 5: Friends, Operator overload, and C++ Standard Library

Grading

This class is graded as C/NC. There are no exams, but programs must be written, tested (by you), and submitted. The instructor will review the programs in/after class. You must attend and participate seriously in lab activities to receive credits. From a practical point of view, you should focus your effort in each lab on whichever lab topic you are *least* comfortable or familiar with — whatever you'll likely need the most help with. Don't worry so much about submitting "correct answers," but rather understanding the topic at hand and submitting evidences that you worked on that topic.

Access and Accommodations

Your experience in this class is important to me. If you have already established accommodations with Disability Resources for Students (DRS), please communicate your approved accommodations to me at your earliest convenience so we can discuss your needs in this course.

If you have not yet established services through DRS, but have a temporary health condition or permanent disability that requires accommodations (conditions include but not limited to; mental health, attention-related, learning, vision, hearing, physical or health impacts), you are welcome to contact DRS at 425-352-5307 or [uwbdrs@uw.edu \(mailto:uwbdrs@uw.edu\)](mailto:uwbdrs@uw.edu). DRS offers resources and coordinates reasonable accommodations for students with disabilities and/or temporary health conditions. Reasonable accommodations are established through an interactive process between you, your instructor(s), and DRS. It is the policy and practice of the University of Washington to create inclusive and accessible learning environments consistent with federal and state law.

For Our Veterans

If you are a student who has served in our nation's military forces, *thank you* for your service. I hope that you feel comfortable enough to confidentially self-identify yourself to me so I can help you make a successful transition from the military to higher education.

Religious Accommodation Policy

Washington state law requires that UW develop a policy for accommodation of student absences or significant hardship due to reasons of faith or conscience, or for organized religious activities. The UW's policy, including more information about how to request an accommodation, is available

at [Religious Accommodations Policy](#).

(<https://registrar.washington.edu/staffandfaculty/religious-accommodations-policy/>).

(<https://registrar.washington.edu/staffandfaculty/religious-accommodations-policy/>). Accommodations must be requested within the first two weeks of this course using the [Religious Accommodations Request form](#) (<https://registrar.washington.edu/students/religious-accommodations-request/>). (<https://registrar.washington.edu/students/religious-accommodations-request/>).

Student Conduct; Plagiarism and Cheating

You are expected to provide original works based on your own effort for this course. You will receive a zero for any coursework for which you are discovered cheating or plagiarizing. (Given that this is an optional lab class, it's hard to say how cheating could make sense. It is perfectly alright in this class for you to work in small groups and have each group member submit a copy of the group work.) You will be referred to the University for further action. It is your responsibility to know and uphold the Student Conduct Code for the University of Washington, available at




<https://www.uwb.edu/studentaffairs/studentconduct>




(<https://www.uwb.edu/studentaffairs/studentconduct>).

Problems

If you have problems with anything in the course, please come and see me during office hours, make an appointment to see me at some other time, or send me an email. I want to make you a success in this course and, by extension, CSS 342.

Course Summary:

Date	Details	Due
Wed Jan 12, 2022	 Lab 1: Toolchains, type conversion, references, file I/O issues https://canvas.uw.edu/courses/1555841/assignments/6922022	due by 11:59pm
Wed Jan 19, 2022	 Lab 2: C++ memory management: pointers, arrays, new and delete operators https://canvas.uw.edu/courses/1555841/assignments/6922023	due by 11:59pm
Wed Jan 26, 2022	 Lab 3: Class, constructors, inheritance and lifespan https://canvas.uw.edu/courses/1555841/assignments/6922024	due by 11:59pm

Date	Details	Due
Wed Feb 2, 2022	 <u>Lab 4: Copy constructors, assignment operators, deep vs. shallow copy; templates and iterators</u> (https://canvas.uw.edu/courses/1555841/assignments/6922025)	due by 11:59pm
Wed Feb 9, 2022	 <u>Lab 5: C++ Standard Template Library</u> (https://canvas.uw.edu/courses/1555841/assignments/6922026)	due by 11:59pm
Thu Feb 10, 2022	 <u>Course Evaluation</u> (https://canvas.uw.edu/courses/1555841/assignments/6922021)	due by 11:59pm