Master Course Syllabus

For additional course information, including prerequisites, corequisites, and course fees, please refer to the Catalog: https://catalog.uvu.edu/

Semester: Spring	**Year:** 2025
Course Prefix: CS	**Course and Section #:** 2370-001
Course Title: C++ Programming	**Credits:** 3

Course Description

Introduces C++ programming for students with prior programming experience. Covers language fundamentals, core standard library components, error handling, value semantics, pointers and memory management, object-oriented programming, and templates.

Course Attributes

This course has the following attributes:
☐ General Education Requirements
☐ Global/Intercultural Graduation Requirements
☐ Writing Enriched Graduation Requirements
☒ Discipline Core Requirements in Program
☐ Elective Core Requirements in Program
☐ Open Elective
Other: *Click here to enter text.*

Instructor Information

Instructor Name: Frank Jones

Student Learning Outcomes

Upon successful completion of this course, students will be able to:
1 Solve small-to-medium-size problems using the C++ programming language
2 Distinguish between value and reference semantics
3 Use the core algorithms and containers in the standard C++ library when applicable
4 Manage memory via pointers

Course Materials and Texts

Required Materials: Access to the zyBooks learning environment using the C++ Programming online course for this class.
1. Click on the first zyBooks assignment in Canvas. (Do not go to the zyBooks website and create a new account.)
2. Subscribe.

A subscription is $89.

Course Requirements

Course Assignments, Assessments, and Grading Policy

Course Mode:

This is a **face-to-face** course. There will be **no recordings**. You are expected to attend all class sessions.

Description of how course works:

Canvas is where course content, grades, and communications will reside for this course. Links to the **zyBooks** assignments are on the **Modules** page in Canvas (as well as the Assignments page). Clicking on assignments will take you directly to the zyBook (you will need to subscribe the first time). You can develop code offline and then submit it in the zyBook, or you can develop within the zyBook itself. Sample code and other files are available in the **Files** folder on Canvas. Projects 4–6 are submitted to Canvas only. I recommend you use an offline development environment, such as VS Code, Visual Studio, or your favorite programming editor, but you can do all development in zyBooks if you want. If you use use Visual Studio, create only *empty console* projects. We will be using portable C++, that is, no platform-specific extensions.

Each class period will consist of a live, **interactive, lecture/discussion.**

For this **three (3) credit-hour** course students should expect to spend up to **9+ hours a week** completing course activities.

Student Responsibilities:

- *Start class the **first week** of the term.*
- ***Be accountable by setting aside regular time each week to complete course activities and assignments on time as noted per the due dates**.*
- *Learn how to use Canvas including communication tools (e.g. **discussion,** Canvas **inbox,** etc.). If you have technology-related problems contact the Service Desk (Links to an external site.).*
- *Complete assigned exercises in **zyBooks** (or **Canvas**) prior to their respective due dates.*
- *Keep track of important dates in the Student Timetable (Links to an external site.).*
- *Contact your instructor as early as possible if an emergency arises. Do NOT wait until the last minute to ask for an extension.*
- *Abide by **ethical** standards. **Your work must be your own.** In each of your 6 programming projects, include the following statement as a comment near the beginning of your source code (and **mean** it):*

"I declare that the following source code was written solely by me. I understand that copying any source code, in whole or in part, constitutes cheating, and that I will receive a zero on this project if I am found in violation of this policy."

Instructor Responsibilities:

- *Respond to messages within ONE **business day.** If multiple messages are received regarding the same question or concern, they may be responded to with an **announcement** to the entire class.*
- *Provide timely, meaningful and constructive feedback on assignments.*
- *Facilitate an effective learning experience.*
- *Refer students to appropriate services for issues that are non-course content specific. For instance, technical issue, writing labs, accessibility services, etc.*
- *Mentor students through the course.*

- **Grading Scale:**
- The following grading standards will be used in this class:

Grade	**Percent**
A	94-100
A-	90-93
B+	87-89
B	83-86
B-	80-82
C+	77-79
C	73-76
C-	70-72
D+	67-69
D	63-66

D-	60-62
E	0-59

-
- **Assignment Categories**

Activity	**Weight**
zyBooks PAs	5%
zyBooks Labs	10%
Professionalism	5%
Programming Projects	50%
Midterm Exams (2)	15%
Final Exam	15%
Unannounced quizzes (Extra credit)	3%

Late Work Statement:

- All work is due at **11:59pm** on its respective due date. Late work is **automatically penalized 33% per day** by Canvas, so there can be *no exceptions* for late penalties. Plan on getting your work done ahead of time.
- *** **IMPORTANT**: **The due dates are in *Canvas*.** And to reiterate: all grades (except for Projects 4–6) are graded automatically, so there can be *no leniency for late submissions*. Such is life in the computer age.

Non-Late Policy:
Students in this class have the option of utilizing what I refer to as the "non-late" policy. This is an OPTIONAL policy that (when properly applied) supersedes the more traditional late policy just described. The non-late policy is meant to do several things:

1) Emulate/reduce as much as possible the artificial constraints that academic schedules impose on the learning process.

2) Encourage good habits/practices that will benefit students in this class and in life.

3) Prevent students from hitting a "wall" by running out of time, not completing an assignment on time, giving up on it, and subsequently not being able to complete any of the successive coursework that depends upon the concepts explored in the uncompleted assignment.

In the real world, professional people work consistently and diligently to meet practical deadlines. Even under the best of circumstances we sometimes miss these deadlines. When this happens as students this often means "all hope is lost" and we just turn in what we have. In the real world we don't do this. Among other actions that may be appropriate, we generally keep working - HARDER - and push through until the project is completed. This is often not only the right and best thing to do, but also the REQUIRED thing to do.

The non-late policy offers a student the opportunity to continue working on an assignment past the due-date, and to turn it in without late point deductions if three conditions are met. The conditions are as follows:

1) Prior to the due date the student must have met a minimum threshold in number of out of class hours worked towards the completion of the assignment. IN THIS COURSE the non-late policy can ONLY BE APPLIED TO PROGRAMMING PROJECTS. Due to the nature of the material in this class, textbook reading and other homework do NOT count as “work towards completion” of the projects. Only time spent performing technical research, designing, implementing, and testing your project can be counted as hours worked. The minimum threshold of hours worked for an assignment will be specific to, and indicated for, each assignment.

Hours worked must be recorded in a simple "work log" that will be described shortly.

2) Prior to the due date the student must have met a minimum threshold in number of "work days" for the assignment. A "work day" is a day in which the student invested AT LEAST 1 hour of out of class time working towards the completion of the assignment. See the previous requirement for a description of what activities qualify as "time working".

3) Once the due date for an assignment has passed, the student must work at least 2 hours DAILY on the assignment until the day the assignment is completed and turned in (excluding Sundays). Obviously this requirement doesn't apply to the day the student turns in the assignment as they may finish before reaching the 2 hour mark (and that is perfectly fine)

Failure to meet any of these three requirements for an assignment will invalidate the non-late policy for the assignment.

Work Logs and Invoking the Non-Late Policy:
In order for a student to invoke the non-late policy they need only leave a comment with their submission that states
"Invoking the non-late policy." Accompanied by a copy of their work log showing that they have met the three
requirements. Work logs should be a simple listing of the days, amounts of time, and activities

invested. Here is a simple
example

Jan 31st, Worked on design for project X (flow-charts) - 0.5 hour : 0.5 hours total
Feb 2nd, Worked on design for project X (flow-charts) - 1 hour, Started work on class diagrams 0.5 hours : 1.5 hours total
Feb 3rd, Finished class diagrams and started coding project X - 2 hours : 2 hours total
Feb 4th, Coding and debugging project X - 3.5 hours : 3.5 hours total
Feb 5th, Coding and debugging project X - 2 hours : 2 hours total
Feb 6th, Coding and debugging project X - .25 hours : .25 hours - project completed.
Due date: Feb 4th
Work days before project due: 3
Hours worked before project due: 7.5

If the minimum number of days worked for this project was 3 or less - AND the minimum number of hours worked was less than 7.5 - then this student would be able to invoke the non-late policy and not face any late penalties even though they finished on the second day after the due date.

Speak with your instructor or a course IA for more information or with any questions concerning this policy.

NOTE: ALL coursework (with the exception of the final exam) is due by 11:59pm on the last day of class. No late work will be accepted beyond this day and time.

Assignments:

This course is organized into **6 modules**, as follows.

- Module 1: C++ Fundamentals (Chapters 1–4)
- Module 2: Sequences and Functions (Chapters 5–7)
- Module 3: Classes and Exceptions (Chapters 8–9)
- Module 4: Pointers and Streams (Chapters 10–11)
- Module 5: Object-Oriented Programming (Chapter 12)
- Module 6: Generic Containers and Algorithms (Chapters 13–15)

There are **6 programming projects** of varying difficulty. There are numerous Participation Activities (PAs) built into the zyBook, many of which you must complete as you read the book. In addition, there are 15 small programming labs that you do in zyBooks.

Quizzes:

There will be unannounced quizzes throughout the semester. They will happen at the beginning of class, and cannot be made up. So you must attend class to get any credit for the quizzes.

Attending class in person is the most effective way for you to learn, so quizzes encourage you to be in class.

Professionalism:

Classroom learning is a "group activity" - one wherein the active participation of each member of the class is important to the efficient use of time for all parties involved. As such, a certain degree of professionalism is expected from all students. In this context "professionalism" does not mean a stiff and formal environment, such is not necessary or even ideal for what we are trying to accomplish. Rather, professionalism involves behavior that extends common courtesy and consideration to everyone else in the room (including the instructor). This includes being polite and respectful of each other (including other people's questions and comments), as well as being respectful of each others' time.

A great deal of time and energy are wasted by students who are not present, unprepared, or acting in a distracting manner (such as having side-bar conversations or being distracted themselves). In an effort to encourage professional behavior, daily attendance is taken in the form of a simple participation quiz, and points are recorded for being in class (attending class is the lowest bar of professional behavior). Beyond "being there" a student may lose their professionalism point for a day through unbecoming or unacceptable behavior. Generally if an issue arises the instructor with discuss the issue with the student(s) involved and issue and document a warning in private. If the behavior continues then the instructor will be forced to begin deducting points. In serious cases points may be deducted without a warning.

Discussions:

In addition to messaging the instructor and going to the Tutoring Lab, you are encouraged to initiate Discussions in Canvas for topics you feel need out-of-class treatment. This is for your convenience and enrichment but will not be graded. Be sure to ask questions as they arise during class. Class discussion is a Good Thing.

Assessments:

There are 2 midterm exams and a final exam. The exams will mostly be multiple choice, and are administered in Canvas**.** You must be **alone** when taking exams. You have only one attempt for each exam. Exams are **timed.** The final exam is **comprehensive.**

Required or Recommended Reading Assignments

Readings are included in each assignment in zyBooks

General Description of the Subject Matter of Each Lecture or Discussion

This course is organized into **6 modules**, as follows.

- Module 1: C++ Fundamentals (Chapters 1–4)

- Module 2: Sequences and Functions (Chapters 5–7)
- Module 3: Classes and Exceptions (Chapters 8–9)
- Module 4: Pointers and Streams (Chapters 10–11)
- Module 5: Object-Oriented Programming (Chapter 12)
- Module 6: Generic Containers and Algorithms (Chapters 13–15)

Required Course Syllabus Statements

Generative AI

ChatGPT (and similar Tools) in This Course

Use ChatGPT as a learning assistant, not as a crutch. You should not use it to generate any code in this course. If asking ChatGPT (or some other LLM(s)) questions helps you to learn some concepts faster, GREAT because then we can spend class time on more interesting topics; but don't allow the model to do your thinking/working for you. Remember: If you REALLY want to be good, work for it. If

Using Remote Testing Software

☐ This course does not use remote testing software.

☒ This course uses remote testing software. Remote test-takers may choose their remote testing locations. Please note, however, that the testing software used for this may conduct a brief scan of remote test-takers' immediate surroundings, may require use of a webcam while taking an exam, may require the microphone be on while taking an exam, or may require other practices to confirm academic honesty. Test-takers therefore shall have no expectation of privacy in their test-taking location during, or immediately preceding, remote testing. If a student strongly objects to using test-taking software, the student should contact the instructor at the beginning of the semester to determine whether alternative testing arrangements are feasible. Alternatives are not guaranteed.

Required University Syllabus Statements

Accommodations/Students with Disabilities

Students needing accommodations due to a permanent or temporary disability, pregnancy or pregnancy-related conditions may contact UVU Accessibility Services at accessibilityservices@uvu.edu or 801-863-8747.

Accessibility Services is located on the Orem Campus in BA 110.

Deaf/Hard of Hearing students requesting ASL interpreters or transcribers can contact Accessibility Services to set up accommodations. Deaf/Hard of Hearing services can be contacted at DHHservices@uvu.edu

DHH is located on the Orem Campus in BA 112.

Academic Integrity

At Utah Valley University, faculty and students operate in an atmosphere of mutual trust. Maintaining an atmosphere of academic integrity allows for free exchange of ideas and enables all members of the community to achieve their highest potential. Our goal is to foster an intellectual atmosphere that produces scholars of integrity and imaginative thought. In all academic work, the ideas and contributions of others must be appropriately acknowledged and UVU students are expected to produce their own original academic work.

Faculty and students share the responsibility of ensuring the honesty and fairness of the intellectual environment at UVU. Students have a responsibility to promote academic integrity at the university by not participating in or facilitating others' participation in any act of academic dishonesty. As members of the academic community, students must become familiar with their rights and responsibilities. In each course, they are responsible for knowing the requirements and restrictions regarding research and writing, assessments, collaborative work, the use of study aids, the appropriateness of assistance, and other issues. Likewise, instructors are responsible to clearly state expectations and model best practices.

Further information on what constitutes academic dishonesty is detailed in UVU Policy 541: *Student Code of Conduct*.

Equity and Title IX
Utah Valley University does not discriminate on the basis of race, color, religion, national origin, sex, sexual orientation, gender identity, gender expression, age (40 and over), disability, veteran status, pregnancy, childbirth, or pregnancy-related conditions, citizenship, genetic information, or other basis protected by applicable law, including Title IX and 34 C.F.R. Part 106, in employment, treatment, admission, access to educational programs and activities, or other University benefits or services. Inquiries about nondiscrimination at UVU may be directed to the U.S. Department of Education's Office for Civil Rights or UVU's Title IX Coordinator at 801-863-7999 – TitleIX@uvu.edu – 800 W University Pkwy, Orem, 84058, Suite BA 203.

Religious Accommodation
UVU values and acknowledges the array of worldviews, faiths, and religions represented in our student body, and as such provides supportive accommodations for students. Religious belief or conscience broadly includes religious, non-religious, theistic, or non-theistic moral or ethical beliefs as well as participation in religious holidays, observances, or activities. Accommodations may include scheduling or due-date modifications or make-up assignments for missed class work.
To seek a religious accommodation, a student must provide written notice to the instructor and the Director of Accessibility Services at accessibilityservices@uvu.edu. If the accommodation relates to a scheduling conflict, the notice should include the date, time, and brief description of the difficulty posed by the conflict. Such requests should be made as soon as the student is aware of the prospective scheduling conflict.

While religious expression is welcome throughout campus, UVU also has a specially dedicated space for meditation, prayer, reflection, or other forms of religious expression.