

CS 2433 – C/C++ Programming

Instructor

Dr. Sachin R. Jain

Teaching Assistant Professor

Office: 223, MSCS

Email: sachin.jain@okstate.edu

Phone: 405-744-2283

Virtual Office Hours: MWF 1:00pm – 2:00pm (Central Time) or by appointment.

Description

***This is an online course.** All lectures, resources, assignments, and correspondence are accessed entirely online through Canvas at canvas.okstate.edu. Sign in using the O-Key username and password provided to you by OSU. Once you sign in, you will see the Canvas Dashboard, which provides you with a list of links for the courses in which you are enrolled. You should see our course listed as CS 2433 - C/C++ Programming **Online Fall 2020**. Click on that link to go to the course homepage.*

Why Learn C/C++?

The C language has formed the basis for many languages including C++, Java, JavaScript, Go, Rust, Limbo, LPC, C#, PHP, Python, Perl, Verilog, C-shell, etc. However, learning C/C++ is still an asset to a programmer, for several reasons:

- C/C++ are middle level languages, combine features of high-level and low-level languages.
- Can be used for low-level programming, such as scripting for drivers and kernels.
- Supports functions of high-level programming languages, such as scripting for software applications.
- C/C++ are structured programming languages which allows a complex program to be broken into simpler programs.
- C/C++ are highly portable languages and is often the language of choice for multi-device, multi-platform app development.
- C/C++ have a rich function library.
- C/C++ are powerful, efficient and fast languages, that finds a wide range of applications.
- GUI applications to 3D graphics for games to real-time mathematical simulations.
- C/C++ have stood the test of time. There are billions of lines of C/C++ out there running in many of the software / applications.
- C/C++ in particular are used frequently for embedded devices.

Course Objectives

Some of the main objectives of the course are as follows:

- Develop a basic understanding of the programming environment.
- Improve programming skills of the students.
- Allow students to design, write and implement programs in C/C++.
- Give students a basic understanding of Object-Oriented Programming.

Course Outcomes

By the end of the course, the students will be able to:

- Write good C/C++ code.
- Use good programming style for writing code in C/C++.
- Design C/C++ programming solutions to problems.
- Acquire basic understanding of algorithms.

Textbook

In this particular course an online textbook and assignment system, called zybooks will be used. Most of the assignments and programs will be covered through zybooks and zylabs, so it is must to subscribe on zybooks. Following steps is to be followed for completing the registration:

1. Sign in or create an account at *learn.zybooks.com*
2. Enter zybook code: **OKSTATECS2433JainFall2020**
3. Subscribe:
 - a. Subscription cost is **\$77**.
 - b. Students may begin subscribing on **Aug 03, 2020**.
 - c. The cutoff to subscribe is **Nov 21, 2020**.
 - d. Subscriptions will last until **Dec 18, 2020**.

Additional Reading (Optional)

- *The C Programming Language, Second Edition, by Kernigan and Ritchie.*
- *C Primer Plus, Sixth Edition, Stephen Prata.*
- *Programming: Principles and Practice Using C++, Second Edition, by Bjarne Stroustrup.*
- *Absolute C++, Sixth Edition, by Walter Savitch.*
- *Professional C++, Fourth Edition, by Marc Gregoire.*
- *Practical C++ Programming, Second Edition, by Steve Oualline.*

Additional Online Resources

- <http://www.cplusplus.com> , <http://www.cplusplus.com/reference/>
- Bjarne Stroustrup's page on C++: <http://www.stroustrup.com/C++.html>.
- Tutorial topics in C++: <http://www.learncpp.com.html>.
- Video tutorials: <https://www.youtube.com/playlist?list=PLAE85DE8440AA6B83>.
- Notes on Object-oriented design: https://en.wikipedia.org/wiki/Object-oriented_design.

Useful Software

- **Bloodshed Dev-C++ IDE (<http://www.bloodshed.net/devcpp.html>).** (Recommended)
- jEdit (<http://jedit.org/index.php?page=download>), a powerful text.
- Code Blocks (<http://www.codeblocks.org/>) - a lightweight IDE.
- The atom editor (<https://atom.io>) - a good free code editor.
- Notepad++ (<http://notepad-plus-plus.org>) - a lightweight text editor for Windows.

Instructor Response Time

As this is an online course, it is expected that, all correspondence should be done through Canvas. But, if you need to contact me on an individual basis, the most preferred way is through email. I will try my best to respond you within 24 hours, though sometimes it could be as long as 48 hours or more, especially in case of a weekend or holiday. To ensure that I perceive your email, please make sure to begin the subject line of your email with the course number in square brackets, followed by the message subject; for example:

[CS 2433] Unable to view grades on Canvas

For those of you located in other than my time zone (Central Time, GMT-6), please be aware that this also may affect the time it takes me to respond to your emails.

Providing grades and/or feedback for assignments may take up to 2 weeks of the due date.

Participation Expectations

Students should expect this course to be more challenging and take a lot of effort. You are going to learn a basic programming language course, the only way to become proficient is to do a lot of practice in developing programs for different problems in it.

The typical rule of thumb is that you should expect to spend 2-3 hours for every hour of online class. It is expected that, you should plan to spend 6-9 hours per week on average reading, doing homework / quizzes and other assignments.

To do well in this course, students are expected to

- Keep up with the zyBooks material, including the participation activity (PA) and challenging activity (CA) as well as zyLabs.
- Read or view the instructional material posted to Canvas frequently.
- Ask for help if any of the material covered is not clear.
- Complete the homework / quizzes / assignments and submit them before their deadlines.
- Regularly check Canvas for announcements.

Grading Policy

Grades in this course will be calculated according to the completion of following assignments:

Assignment	Value in Points	Percentage of Total Grade
Homework / Quizzes	300	30%
Programming Assignments	300	30%
One Mid-term Exam	200	20%
Final Exam	200	20%
Total	1000	100%

Final grades will be assigned according to the following scale:

Grade	Percentage Earned
A	90 – 100%
B	80 – 89.99%
C	70 – 79.99%
D	60 – 69.99%
F	0 – 59.99%

Homework - Quizzes

There will be Homework - Quizzes after completion of section(s) / chapter(s), which may also be completed through the Canvas.

Programming Assignment

- All programming assignments should be done **ONLY** through zybooks (zylabs).
- 10% penalty of available points per day late. However, you cannot receive negative points for an assignment.
- An Incomplete programming assignment may not be considered for grading.
- If you are going to miss an exam or assignment, contact the instructor in advance.
- Exceptions can be made if a serious family or personal emergency arises.

Exams

There will be one midterm exam during the semester and a final exam during finals week; Both exams will be online proctored exams (approx. \$15-20 each) and will be using Examity through Canvas only. You must have a Computer/Laptop with web-cam, microphone, speaker and Internet connection. More information regarding Examity is available on Students Quick Guide on Canvas.

Collaboration

Discussion of concepts, ideas, and techniques is acceptable. After discussion, each student must write up his/her own solution. Copying another person's work, in part or in whole, is not allowed. Giving another student your work, in part or in whole, is considered cheating as well. If you are unsure whether your collaboration is acceptable, speak with the instructor in advance. Take care that your solutions are not exposed to or by other students.

Students who do not comply with the collaboration policies described above will be assigned sanctions in accordance with OSU policy 2-0822 (Academic Integrity). Depending on the circumstances of the violation, the sanctions may result in a score of zero on an assignment, a final grade of F! for the course, or dismissal from the OSU graduate program. In all instances, the violation will be reported to the appropriate institutional officials.

Disabilities Act

If any student feels that he/she has a disability and needs special accommodations of any nature whatsoever, the instructor will work with you and Student Disability Services, 315 Student Union, to provide reasonable accommodations to ensure that you have a fair opportunity to perform in this class. Please advise the instructor of such disability and the desired accommodations at some point before, during, or immediately after the first scheduled class period.

Tentative Course Schedule

SN	Module	Date	Week	Homework & Assignment	Due Date
1.	Introduction to Programming	Aug 17 - Aug 23	1	HW-1 & Assignment-1	Sept 06
2.	Basics of C/C++	Aug 24 - Aug 30	2		
3.	Variables / Assignments	Aug 31 - Sept 06	3		
4.	Branches	Sept 7 - Sept 13	4	HW-2 & Assignment-2	Sept 20
5.	Loops	Sept 14 - Sept 20	5		
6.	Arrays / Vectors	Sept 21 - Sept 27	6	HW-3 & Assignment-3	Oct 04
7.	Functions & Recursion	Sept 28 - Oct 04	7		
MID TERM EXAM					
8.	Objects and Classes	Oct 05 - Oct 11	8	HW-4 & Assignment-4	Oct 25
9.	Pointers	Oct 12 - Oct 18	9		
10.	Streams	Oct 19 - Oct 25	10		
11.	Inheritance	Oct 26 - Nov 01	11	HW-5 & Assignment-5	Nov 08
12.	Exceptions & Templates	Nov 02 - Nov 08	12		
13.	Containers	Nov 09 - Nov 15	13	HW-6 & Assignment-6	Nov 22
14.	Searching & Sorting Algorithms	Nov 16 - Nov 22	14		
15.	Additional Topics	Nov 23 - Nov 29	15	--	--
FINAL EXAM					