Yukon University	**School of Science**
	CPSC 129 **Object Oriented Programming II**
	Term: Winter 2026 **Number of Credits: 3**

Course Outline

INSTRUCTOR: Kate Chatfield-Reed
E-MAIL: kchatfieldreed@yukonu.ca
OFFICE: A2806
PHONE: (867) 456-8563
TIME/DATES: T/R 5:30 pm – 6:50 pm and R 7 – 7:50 pm in A2702
OFFICE HOURS: *anytime*, schedule by email

COURSE DESCRIPTION

CPSC 129 continues the presentation of OOP begun in CPSC 128. Now that students have mastered the core computational constructs the emphasis is on using them to express more subtle algorithms, and to compose more sophisticated data structures and classes.

COURSE REQUIREMENTS

Prerequisite(s): CPSC 128

EQUIVALENCY OR TRANSFERABILITY

Receiving institutions determine course transferability. Find further information at:
https://www.yukonu.ca/admissions/transfer-credit

LEARNING OUTCOMES

Upon successful completion of the course, students will be able to:

- to produce an object-oriented (OO) analysis and design for a problem.
- to apply the principles of class inheritance, composition, and association to construct hierarchies of new classes.
- to use the components and constructs necessary to implement an OO program in efficient, reusable, extensible code.
- to design alternative algorithms to solve a variety of problems.
- to design alternative data structures to represent the information necessary to solve a variety of problems.
- to evaluate programs through the careful application of appropriate testing techniques to assess both their correctness and their memory and processor performance characteristics.

- to document the analysis, design, implementation and testing of a program constructed using OO principles.

COURSE FORMAT

Weekly breakdown of instructional hours

This course will be delivered with the following breakdown per week: 3 hours of lecture and 1 hour of tutorial. Although it will vary from individual to individual, students should expect to spend 6 hours on course material outside of the classroom time (per week) on completing assignments.

Delivery format

This course will be delivered in a face-to-face (in person) format. However, lectures may be attended remotely and recorded to be watched later (upon request). Students will be expected to access the YU online learning platform for additional material (Moodle).

EVALUATION

Tutorials	10 %
Quizzes	10 %
Assignments	30 %
Coding Portfolio	20 %
Final	30 %
Total	100%

Tutorials (10%): Tutorials are on opportunity to work in class with lots of support from classmates and the instructor. Problems will be assigned most weeks and solutions to them will be made available. Students should be certain to do these problems promptly or they risk being unable to understand the material in the next class. Students work on the tutorials in class each week and be graded before leaving. Completion of the tutorials is worth 10% of the final course mark.

Quizzes (10%): Small biweekly quizzes usually done on paper.

Assignments (30%): There will be weekly assignments over the course of the term worth 30% of the final mark. Where the tutorial problems are intended to assist the student in learning new material and are marked for completion, the assignments are meant to reinforce and extend the student's understanding of material that has already been learned. These assignments should reflect the student's own work and should be completed individually without assistance from LLMs (such as ChatGPT). Assignments are subject to a 5% late penalty per day they are late.

Portfolio (20%): The portfolio is an opportunity for students to submit final versions of each thematic assignment cluster (i.e. Conway's game of life, the maze, sorting algorithms, etc.). Students can adjust previous attempts and implement feedback from the instructor and produce a very clean set of finished programs.

Final Examination (30%): A final examination which will cover the entire course, and be worth 30% of the final mark, will be held during the examination period at the end of the semester. The date and time will be announced as soon as it is set.

COURSE WITHDRAWAL INFORMATION

The last date to withdraw without academic penalty is Mar. 5th, 2026. Refer to the YukonU website for other important dates.

TEXTBOOKS & LEARNING MATERIALS

No textbook is required for this course. All the necessary content is provided either through the course Moodle or online resources.

ACADEMIC INTEGRITY

Students are expected to contribute toward a positive and supportive environment and are required to conduct themselves in a responsible manner. Academic misconduct includes all forms of academic dishonesty such as cheating, plagiarism, fabrication, fraud, deceit, using the work of others without their permission, aiding other students in committing academic offences, misrepresenting academic assignments prepared by others as one's own, or any other forms of academic dishonesty including falsification of any information on any Yukon University document.

Please refer to Academic Regulations & Procedures for further details about academic standing and student rights and responsibilities.

ACCESSIBILITY AND ACADEMIC ACCOMMODATION

Yukon University is committed to providing a positive, supportive, and barrier-free academic environment for all its students. Students experiencing barriers to full participation due to a visible or hidden disability (including hearing, vision, mobility, learning disability, mental health, chronic or temporary medical condition), should contact Accessibility Services for resources or to arrange academic accommodations: access@yukonu.ca.

TOPIC OUTLINE

Object-oriented design, search, algorithm analysis, GUIs, MVC architecture, recursion, sorting, data processing, and image processing.