University of Lethbridge
Department of Mathematics and Computer Science

Computer Science 2620 – Fundamentals of programming II
Course Outline – Fall 2025

LECTURES: Mo Wed 10:30 – 11:45 **ROOM:** C610

INSTRUCTORS: Robert Benkoczi (office C556)
`robert.benkoczi@uleth.ca`

TEXTS: *C++ Primer* 5th Ed, by Lippman, Lajoie, Moo.
`https://cpp-primer.pages.dev/`
Open Data Structures, by Morin, available at `http://opendatastructures.org/` (under a CC License)
Mastering modern C++, by Ayman Alheraki
`https://www.slideshare.net/slideshow/mastering-modern-c-c-11-c-14-c-17-c-20-c-23/275446486`

GRADING SCHEME:

Tests (5))	40%
Assignments (5)	20%
Final exam	40%

GRADE DISTRIBUTION: This information is provided as a guideline only and may be revised in this offering.

A+	95	B+	82	C+	70	D+	56
A	90	B	78	C	66	D	50
A-	86	B-	74	C-	62	F	< 50

SCHEDULE:
(as time permits):
1) Classes, abstract data types. Access control and encapsulation, class scope, constructors, static members.
2) Recursive functions.
3) Memory management, pointers. New, delete, STL smart pointers.
4) Templates and generic programming.
5) Searching and sorting.
6) Dynamic arrays, linked lists, skip lists.
7) Binary and general trees, unbalanced search trees.
8) Data structures for storing graphs.

SUBMITTING WORK / GRADE REVISION:

- Work must be submitted at the scheduled time. It may be possible to have some flexibility to submit work after deadlines, but this can vary from assignment to assignment as dictated by the timing for grading the work.
- Make-up tests and late assignments are allowed for medical reasons or emergencies. Missed tests and assignments receive 0 points.
- Requests for remarking tests and assignments are accepted in writing *no later than one week from the date your graded work was returned*. On the request: identify the assignmemnt or midterm, briefly explain why you believe the mark is incorrect, date and sign. Note that if your work is remarked, your grade may go up, down, or remain unchanged.

AI USE POLICY

Generative AI can be a very effective tool for learning. Make sure you use AI appropriately because a significant fraction of your grade is obtained from answers you give personally, in class, on paper. Some concrete measures that define an appropriate use of AI tools are listed below.

- Never submit generated code that you do not fully understand.
- Use AI to generate repetitive, "boiler plate" code. Examples: code to insert 100 random integers into a data structure, or to generate a text file that contains integers obeying some property, etc.
- Use AI to generate code for a procedure you want to understand, in **small increments**. For example, try to have no more than 5-10 lines of newly generated code per query.
- Make sure the code generated is explained. Generated explanations are not always enlightening. Change the code, test, and run, to make sure you know what is going on.
- Keep generated code and code you plan to submit in separate files. Copy from the generated code into your work, adapt, and test. Compile and run your work often.

OTHER COMMENTS

- Plagiarism can lead to severe penalties. Consult the student code of conduct policy, `https://uleth.sharepoint.com/:b:/s/Policy/EYaNcsOfn-NHsfg3NNEIlFUBh7wkC1BPw52MabAXw84JFQ?e=OEeZb2`.
- Please be considerate to your classmates: try not to be late in class, limit your discussion to class discussion during the lecture, and turn off the sound on any electronic device (laptop, cell phone, etc).

LINKS

- Moodle: `http://moodle.uleth.ca/`
- Instructor's page including office hours: `http://www.cs.uleth.ca/~benkoczi/`