General Information Handout

Welcome to ECE244! This handout provides high-level information about the course, including its organization, components, reference material, assessment and topics covered. Please read it and keep it for reference during the term.

Calendar Description

Provides a foundation in programming using an object-oriented programming language. Topics include classes and objects, inheritance, exception handling, basic data structures (lists, tree, etc.), Big-O complexity analysis, and testing and debugging. The laboratory assignments emphasize the use of object-oriented programming constructs in the design and implementation of reasonably large programs.

Instructors

Ding Yuan
ECE Department
Office: 2002E Sanford Fleming Building
Phone: (416) 978-5033
E-mail: yuan@eecg.toronto.edu
URL: https://www.eecg.utoronto.ca/~yuan/

Subhendu Bhadra
ECE Department
E-mail: subhendu.bhadra@utoronto.ca

SK MD Mizanur Rahman
ECE Department
E-mail: sk.rahman@utoronto.ca
URL: https://ca.linkedin.com/in/mizanmizan

Learning Outcomes

At the conclusion of this course, you will be able to:

1. Write C++ code that utilizes encapsulation and code reuse features of the C++ programming language.
2. Understand basic data structures, such as arrays, linked lists and trees, and write C++ code to implement and manipulate them.
3. Express the run-time complexity of simple algorithmic segments using big-O notation.
4. Utilize modern programing tools, such as compilers, debuggers and IDEs.

Industrial Relevance

Object-oriented programming is the most wide-spread programming paradigm today. Every application that runs on a mobile device (phone or tablet) and much of the code that runs on desktops and servers is written using an object-oriented language, such as C++, Java or C#. A firm grasp of this paradigm is key to modern software development.

Course Web Site

Information on ECE 244, including important announcements, a copy of this handout, copies of assignment handouts, and course marks may be found on the web site for the course. Please visit the web site on a regular basis for up-to-date information.

The course website is at: https://www.eecg.utoronto.ca/~yuan/teaching/ece244/

We use Piazza for online Q/A – instructors and TAs will be monitoring the discussion forum and answer your questions. The URL: `https://piazza.com/utoronto.ca/fall2022/ece244`

E-mail

All UofT students are required to have a valid UTORmail email address. Please ensure that your UofT email address is properly entered in the ROSI system.

Forwarding your utoronto.ca email to a Hotmail, Gmail, Yahoo or other type of email account is not advisable. In some cases, messages from utoronto.ca addresses sent to Hotmail, Gmail or Yahoo accounts are filtered as junk mail, which means that emails from your course instructor may end up in your spam or junk mail folder.

Textbook and Other Reference

The following book is recommended, and it will serve as the main reference for the course:

Walter Savitch, *Problem Solving with C++*, 10th Edition, Pearson.

The book is available in as an e-Text (ISBN: 9780134522418) or in hardcopy. A copy of the e-Text can be obtained from the UofT Bookstore using the following link:

https://www.campusebookstore.com/AccessCodes/AccessCodeBrowse.aspx?CODEID=25022

Alternatively, the e-Text can be obtained directly from the publisher by following the steps below:

1. Go to https://uoftbookstore.vitalsource.com/products/problem-solving-with-c-pod-file-walter-savitch-v9780134521176
2. Choose e-Text option (180-day or Lifetime)
3. Click “Add to Cart”
4. Click “Check out”
5. “Sign up” (or sign in if you previously signed up)
6. Proceed with checkout process

You may also use a hardcopy of the book, either new or used. In the latter case, it is ***not necessary*** to buy the latest edition; any edition past the 7th edition (i.e., 8th or the 9th) is adequate for the purposes of the course.

In addition to the textbook, you may consult with a wealth of reference material and excellent tutorials on the C++ programming language on the Web. Here are two examples:

- C plus plus: http://www.cplusplus.com
- Sharam Hekmat (pragsoft.com) C++ Programming: http://www.pragsoft.com/books/CppEssentials.pdf

Lectures

Lectures introduce and cover new concepts related to the topics of the course. It is highly recommended that you attend the lectures. Attending gives you a chance to interact with your instructor, pose questions in real time, as well as participate in answering questions and discussions.

Tutorials

The tutorials are intended to review basic programming concepts and revisit material already covered in the lectures in a more interactive setting, where questions can be asked frequently and freely. In contrast to the lectures, no new material will be introduced, and often only basic material will be reviewed. As such, attending the tutorials is optional.

Laboratory Assignments

The lab assignments consist of several programming exercises using the `C++` programming language. Please refer to the "Laboratory Information Handout" for a list of the assignments, their schedule, due dates as well as general information about the labs.

The lab assignments do take a substantial amount of your time. Hence, ***it is important to start as early as possible on your assignments and to avoid procrastination***. Lab assignments are always due at 5:00pm on their due date (also shown in the Topics Outline section below). You are to do each assignment by yourself, i.e., *individually*. Sharing code or submitting code that is written by someone else will be checked for, will not be tolerated, and will be treated as an academic offence.

Timetable

The following is the timetable for the course's lectures and tutorials. There are three lectures, one tutorial, and one lab session per week. ***Please double check your timetable for the locations since they may change when the term starts***.

Section	Weekday	Hours	Location
Lecture LEC0101 (Prof. Bhadra)	Tuesday	9am-10am	LM162
	Thursday	9am-10am	LM162
	Friday	9am-10am	LM162
Lecture LEC0102 (Prof. Rahman)	Monday	3pm-4pm	MB128
	Wednesday	3pm-4pm	MB128
	Thursday	3pm-4pm	WB116
Lecture LEC0103 (Prof. Yuan)	Monday	3pm-4pm	MC252
	Wednesday	3pm-4pm	MC252
	Thursday	3pm-4pm	MC252
Tutorial TUT0101/0102	Wednesday	11am-12noon	*MC252*
Tutorial TUT0103/0104	Friday	3-4pm	*GB248*
Tutorial TUT0105/0106	Friday	9-10am	*GB119*

Assessment and Evaluation

Assessment is done through a midterm test, programming assignments and a final examination. The date and time of the midterm will be announced shortly after the term starts. The final examination is given during the final exams' period at the end of the term. The exact method of delivery of the midterm test and the final exam will be announced in the lectures.

The composition of the final mark is as follows:

Midterm Test		35%
Programming Assignments		25%
Assignment 1	2%	
Assignment 2	6%	
Assignment 3	7%	
Assignment 4	7%	
Assignment 5	3%	
Final Examination		40%

Topic Outline

The following table is a weekly list of the topics covered in the course, and the corresponding reference in the textbook. Chapters are from the 10th edition of the "Problem Solving with C++" textbook; earlier editions may have slightly different chapter numbers, but essentially the same content. The table also shows labs for each week, along with each lab's due date.

The topics and their scheduled dates are based on the timetable of one of the lecture sections (LEC0103). Please keep in mind that this list of topics and their scheduled dates is ***tentative*** and is intended to serve as only a guide. Varying pace of material delivery may require changes to this list of topics and/or the schedule.

Week	Dates	Lecture	Topics	Reference*	Labs [Due Date**]
0	9/8	L1	Course organization	Handouts	None

1	9/12	L2	Introduction to C++: I/O, strings	1-5	Lab 1
	9/14	L3	C++ functions and parameter passing. Program organization	5, 12.1	
	9/15	L4	Separate compilation (Online demo)	12.1	
2	9/19	L5	Introduction to classes	10	Lab 1 [9/23]
	9/21	L6	Access control in C++ classes	10	
	9/22	L7	C++ constructors/destructor	10	
3	9/26	L8	Organizing/Compiling programs with classes	12.1	Lab 2
	9/28	L9	Online demo for classes	Code	
	9/29	L11	C++ I/O	6	
4	10/3	L11	C++ I/O	6	Lab 2 [10/7]
	10/5	L12	C++ I/O (online demo)	Code	
	10/6	L13	Pointers	9	
5	10/10	L14	**Thanksgiving, no class**		Lab 3
	10/12	L15	Pointers and arrays	7, 9	
	10/13	L16	Pointers and arrays	9, 11.3, 11.4	
6	10/17	L17	Operator overloading	11.2	Lab 3
	10/19	L18	The Complex class example (Online demo)	11.2, Code	
	10/20	L19	Objects with pointers	11.4	
7	10/24	L20	Objects with pointers	11.4	Lab 3 [10/28]
	10/26	L21	The String class example	Lecture notes	
	10/27	L22	Linked lists	13.1	
8	10/31	L23	Linked lists	13.1	Lab 4
	11/2	L24	Recursion	14	
	11/3	L24	Recursion	14	
	11/7	**Engineering Fall Study Break – No lectures, No Labs, No Tutorials**			
	11/9				
	11/10				
9	11/14	L26	Binary trees	Lecture notes	Lab 4
	11/16	L27	Binary trees	Lecture notes	
	11/17	L28	Binary trees	Lecture notes	
10	11/21	L29	Inheritance	15	Lab 4 [11/25]
	11/23	L31	Inheritance	15	
	11/24	L32	Inheritance (Online demo)	15, Code	
11	11/28	L33	Inheritance (Online demo)	15	Lab 5
	11/30	L33	Inheritance (Online demo)	15, Code	
	12/1	L34	Complexity analysis	18.3	
12	12/5	L35	Complexity analysis	18.3	Lab 5 [12/7]
	12/7	L36	Complexity analysis	18.3	

*Chapters in the textbook.
**Labs are due at 5:00pm on their due date.

Term-Work Petitions

During the course you may encounter circumstances beyond your control, such as illness or a personal/family crisis. These circumstances may force you to miss course work, such as assignments or the midterm test. In such cases, you may submit a *term-work petition* through the

Engineering portal (please see https://undergrad.engineering.utoronto.ca/petitions/about-petitions/ for more details on petitions and for access to the portal). Once your petition is approved by the Academic Advisor, *accommodations* for the missed work will be provided. The type of accommodation varies based on the type of missed work.

If you miss the midterm test, the accommodation is to transfer the weight of the midterm to the final exam. If you miss a lab assignment, the weight of the assignment is evenly transferred to the other lab assignments that you have completed. To receive such accommodation, you must have completed lab assignments with a combined weight of at least 15% (out of the 25% allocated to the lab assignments; please see above for the weights of the assignments). If this is not the case, the weight of the missed assignments is transferred to the final exam.

As always, if you have questions or concerns regarding this policy, please contact the course coordinator.

Land Acknowledgement

I (we) wish to acknowledge this land on which the University of Toronto operates. For thousands of years it has been the traditional land of the Huron-Wendat, the Seneca, and most recently, the Mississaugas of the Credit River. Today, this meeting place is still the home to many Indigenous people from across Turtle Island and we are grateful to have the opportunity to work on this land.

Mental Health

It is not unusual that you experience a range of health and/or mental health challenges that could result in significant barriers to achieving your personal and academic goals. This is particularly the case this year, as we transition back to in-person teaching after the shutdown of the COVID-19 pandemic.

The University of Toronto and the Faculty of Applied Science & Engineering offer a wide range of free and confidential services that could assist you during these times. There are resources that you may find helpful are listed on the U of T Engineering Mental Health & Wellness webpage. This is a small selection of these resources:

- Accessibility Services & the On-Location Advisor
- Health & Wellness and the On-Location Health & Wellness Engineering Counsellor
- Inclusion & Transition Advisor
- U of T Engineering Learning Strategist and Academic Success
- SKULE Mental Wellness
- Scholarships & Financial Aid Office & Advisor

If you find yourself feeling distressed and in need of more immediate support resources, consider reaching out to the counsellors at My Student Support Program (MySSP) or visiting the Feeling Distressed webpage.