CSC258: Computer Organization Winter 2026

This course provides an introduction to the underlying digital structures of computers. Topics include digital logic representation and design, computer system organization and microprogramming.

Instructor Information

Name	Office	Phone	Email
Steve Engels	BA4266	(416) 946-5454	sengels@cs.toronto.edu *
	Office Hours: MWF 12pm-1pm (in-person, between classes) ** Online office hours TBD, or by appointment.		
Course Email: csc258-2026-01@cs.utoronto.ca *			

** please write "CSC258" in the subject header of course-related emails.*

*** email the instructor if appointments outside this time are required.*

Course Information

Information pertaining to this course will be available on Quercus. The course website will have course announcements & materials, discussion boards, relevant readings, as well as assignment, lab & project details. Announcements will be made through the email registered on Quercus, but the site is required reading, and it is understood that you will check it multiple times a week.

Mark Breakdown

Component	Weight
Labs	28% (7 total, 4% each)
Project	15%
Midterm exam	19% (closed-book)
Final exam	38% (also closed-book) ➔ *you must get 40% on the final to pass the course*

- **Final Exam:**
  - Final exam grade needs to be at least 40% in order to pass the course.
- **Labs:**
  - The labs consist of hands-on lab exercises that take place in BA3145, BA3155 & BA3165. Lab exercises must be completed and shown to the TA before the end of the lab session.
  - Pre-lab reports are mandatory for each lab, and must be submitted online before attending the lab session. Students who fail to do this will not be allowed to perform the in-lab exercises.
  - Labs take place every week for two months, starting in the second week of class (see dates below).
  - Tutorial sessions will be used to discuss the upcoming lab work.
- **Project:**
  - A large assembly language project for the last month of the course. Marks are given for successful implementation of five milestones, with extra credit for innovation and creativity.
  - Project demos are performed in the lab, with each milestone worth 3% each.

Students are **required** to work in pairs for the labs, but may work individually for the project.

Important Dates

Week	Topics	Milestone(s)
Jan 5 – 9	Overview, transistors, basic logic gates	
Jan 12 – 16	Combinational circuit design, K-maps	Lab 1
Jan 19 – 23	Logical devices (muxes, adders, decoders)	Lab 2
Jan 26 – Jan 30	Latches & flip-flops	Lab 3
Feb 2 – 6	Registers, counters, finite state machines	Lab 4
Feb 9 – 13	Finite state machine design, midterm review	Lab 5
Feb 16 – 20	***-- Reading Week --***	
Feb 23 – 27	ALUs, registers, memory	Midterm exam*
Mar 2 – 6	Architecture & microprogramming	Lab 6
Mar 9 – 13	Assembly language basics	Lab 7
Mar 16 – 21	Assembly language program design	
Mar 23 – 27	Advanced assembly language	Project demo #1
Mar 30 – Apr 3	Assembly functions & recursion	Project demo #2

**The midterm is tentatively scheduled for Mon Feb 23 from 6pm - 8pm.*
Please report any conflicts to the course email by Feb 1, along with your course schedule.

Lateness is generally not accepted, except in cases of medical emergency. Lateness due to personal reasons must be brought to the instructor for consideration, as early as possible.

Course Textbooks

Recommended:	Mano, Kime, *Logic and Computer Design Fundamentals*, 4th ed., Prentice Hall, 2008
Other texts:	Hamacher, Vranesic, Zaky, *Computer Organization*, 5th ed., McGraw Hill, 2002 Null, Lobur, *The Essentials of Computer Organization and Architecture,* 3rd ed., Jones & Bartlett Publishing, 2012

Administrative Details

Plagiarism involves representing another's work as your own, and is unpleasant for everybody involved. In case you need clarification on the university's policies on plagiarism, please consult the *Code of Behaviour on Academic Matters* from this website: www.artsci.utoronto.ca/osai/students

As an extension of this, the use of generative artificial intelligence (AI) tools is prohibited in all course assessments unless explicitly stated otherwise by the instructor. This includes, but is not limited to, ChatGPT, GitHub Copilot, and open-source models that you have trained and/or deployed yourself. You may not interact with, nor copy, paraphrase, or adapt any content from any generative AI for the purpose of completing assignments in this course. Use of generative AI will be considered use of an unauthorized aid, which is treated as an academic offense by the university.

Feedback on the course is solicited during end-of-term evaluations. However, feedback before that point is encouraged, to improve the delivery of the course. Please make sure your concerns are voiced to the course instructor or the teaching assistants whenever possible.