IC 402
FRI 4 - 6 PM
WEEKLY

CSEC

Scarborough

2016-2017

Learn
Create
Network

CS Enrichment Club - Syllabus

2016-2017 Year

“Everyone should learn to code. It teaches you how to think”

- Steve Jobs

Association of Mathematics and Computer Science Students
University of Toronto Scarborough
1265 Military Trail, BV-308,
Toronto, ON M1C 1A4

Overview

CSEC is a student-ran extracurricular CS Club here at the University of Toronto, Scarborough that focuses on computer science enrichment. It was founded in 2016 by enthusiastic students and professors involved with the UTSC CMS stream. Meetings are held every Friday from 4:00pm to 6:00pm in the Instructional Center, Room 402 (IC402), the Embedded Systems Lab. CSEC is free to join, and open to all students, faculty, alumni, and staff associated with the University of Toronto (UTM/UTSC/UTSG). You don't need to be in the CMS stream to attend - you just need to have a passion for Computer Science, or a will to learn.

Contact Information

If you ever have any questions about CSEC, email us at utsc.csec@gmail.com, contact any CSEC exec, or go to http://www.utsc.utoronto.ca/~csec/.

Brian Chen - President

Office Hours: ...-... at the AMACSS office in BV-308

Email: brian@amacss.org

William Granados - Leader

Office Hours: N/A

Email: william.granadosmatamoros@mail.utoronto.ca

Howard Fung - Lecturer

Office Hours: N/A

Email: howard.fung@mail.utoronto.ca

Prantar Bhowmik - Systems Administrator

Office Hours: N/A

Email: prantar.bhowmik@mail.utoronto.ca

Competitions

I. Get people prepared for the ACM-ICPC Contest

We will be training members for the yearly ACM-ICPC contest. In the first year, we'll be teaching the core concepts for this competition, not looking to compete. We hope to send at least one team. In the coming years, we'll shift focus to preparing as many competent teams as possible to compete in the ACM-ICPC contest.

II. Participate in Hackathons around the Toronto Area

Our goal is to pique the interest of several of our members and coordinate several groups to participate in the Hackathons around Toronto. Some notable Toronto Hackathons include UofTHacks, SportHacks, and more.These teams will normally consist of 4 members, containing the roles of Coder, Designer,and Hustler.

Evaluations and Expectations

Expectations

Everybody is expected to show up on time, enthusiastic, and ready to learn. You must pay attention and remain respectful to those in charge at all times, including any supervisors or senior members. While we are giving lessons, it is very important to take notes. During CSEC, video games, Facebook, Youtube, etc. are not permitted. Please also be respectful of the environment that you are in, especially the computer labs. Failure to adhere to these policies will result in removal from the club. It is heavily recommended that members attend all CSEC meetings.

Homework

CSEC regularly assigns homework to reinforce concepts learned in class. Due dates for homework are usually one week after they are assigned. The quantity of work can range from a simple math problem to a list of challenging problems to solve. Assignments may be collected and marked for evaluation, often counting towards CSEC standings and team selection considerations. Since we are only a club, members should still value their academic work first. We will definitely take special circumstances into consideration - just talk to us! If difficulty is encountered while completing homework, members are encouraged to try their best with what they know, ask us for help, or better yet - research online or use our shared CSEC Resources folder. It is highly recommended that all students complete the homework assigned. Failure to do so will make it very difficult for you to keep up with the fast pace of the curriculum.

Testing

To understand the general abilities of the students in CSEC, we will have several tests throughout the year. These tests may contain questions such as algorithm analysis, pen-and-paper coding, or simple logic questions. These tests are not purely individual evaluations, but rather testing to see where members stand in terms of content comprehension. These tests will be taken into consideration when selecting members for the ACM-ICPC team.

Online Resources and Guidance

General Lesson Plan and Reading Ahead:

algo.is/competitive-programming-course

Online Resource Repository:

https://drive.google.com/folderview?id=0By8UVgxe29XaWUxTMGhoZm9GZEU&usp=sharing

Online Judges and Coding Practice:

open.kattis.com/

wcipeg.com

codeforces.com

topcoder.com

Frequently Asked Questions

Why learn computer programming?
We live in the Information Age, characterized by the mighty capabilities of modern computers. These powerful machines are capable of performing billions of calculations in a flash and yet the average computer user of today knows not much beyond typing an essay or checking their email. By learning to program, you get to play god. You get to completely harness the powers of the powerful machines that most people simply take for granted every day. Still not convinced? Take a look at this video featuring Mark Zuckerberg, Bill Gates, Chris Bosh, and the like, explaining how programming "teaches you how to think": https://www.youtube.com/watch?v=nKIu9yen5nc

How dedicated do I have to be?
Just like any other club, the more you put into CSEC, the more you'll get out. In the end, we are all just students teaching each other (as this club is mostly student-run). While we are very open to personally helping you with understanding the material, we still prefer to go at a relatively fast pace to push members to excellence. The material here is very intensive and you should almost treat this as another course. Falling behind is not a good feeling, so you should do everything you can to stay caught up! Don't be discouraged, because we will always be more than glad to help you. Trying your best is all we expect. You're doing this for you, and nobody else. Programming is a very rewarding and useful skill to have. You can thank us later when you're miles ahead of your CS/Math/Logic classes.

Just what exactly am I in for?
We very briefly familiarize members with the basic ideas of computer science and C++. For the rest of the year, it's all about learning how to tackle algorithmically-heavy, Olympiad-style problems. Your critical thinking skills will vastly improve, and it will really change the way you see the world. Not to mention, nothing is as rewarding as the feeling of solving a challenging programming task.

Scheduling and Lesson Plans

Just a note. This schedule is the advanced schedule, meant for students with a higher understanding of Computer Science. It is recommended that students have taken Intro to Computer Science I & II (CSCA08/48) to understand these concepts.

A less rigorous and more beginner course will be offered to newer individuals, where an instructor will explain the basics of conditional statements, loops, data structures, and other beginner concepts to newer computer science students.

Week 1 - Introduction:

This lesson plan will cover the basic goals, agreements, and expectations of CSEC. The administrative stuff.

Week 2 - Data Structures and Libraries

This lesson plan will go over some important tools in the C++ STL that will aid that will aid them in the future.

Week 3 - Data Structures

This lesson plan will give an overview on more discrete data structures that solve problems like minimum spanning tree, and range queries.

Week 4 - [Continued]

Week 5 - Problem Solving Paradigms

This lesson plan will cover paradigms like complete search, backtracking, divide and conquer, binary search, and exponentiation.

Week 6 - [Continued]

Week 7 - Greedy Algorithms

This lesson Introduces greedy algorithms, and covers coin changing, interval scheduling, and scheduling to minimize lateness.

Week 8 - [Continued]

Week 9 - Dynamic Programming

This lesson will Introduce dynamic programming and will cover some examples as well as applications of Dynamic Programming.

Week 10 - [Continued]

Week 11 - Unweighted Graphs

This lesson will review basic graph theory. Covers depth-first search, breadth-first search, and applications.

Week 12 - [Continued]

Week 13 - Graphs

This lesson covers minimum spanning trees, shortest paths, some graph theory, and some special types of graphs.

Week 14 - [Continued]

And Beyond - To be decided. Let’s have some fun!