CSC 2229: Software-Defined Networking, Winter 2020

Department of Computer Science, University of Toronto

Handout # 1 – Information Sheet **Date:** Tuesday, January 10th

Class hours: Mon. 10 AM - 12 PM
Location: GB120 for the first lecture;
BA5205 for the rest of the lectures
Instructor: Prof. Yashar Ganjali
E-mail: yganjali @ cs . toronto . edu

Office hours: Mon. 12 PM – 1 PM, or
by appointment
Office: BA 5238

Class web page:

http://www.cs.toronto.edu/~yganjali/courses/csc2229/

Course Description

This is a MSc/PhD level course introducing Software-Defined Networking (SDN). SDN is an emerging paradigm in computer networks that facilitates change and innovation in infrastructure and network applications by pushing network control functionalities to a logically centralized controller. We will cover the fundamentals of SDN, and focus on various research problems in the context of SDN such as

- controller and switch design;
- network architecture;
- reliability, efficiency and scalability;
- network programming, correctness, and debugging;
- network security;
- new abstractions, and state management;
- network services, network function virtualization; and
- network optimization.

Prerequisites

Basic undergraduate courses in algorithms, networking, and in probability theory are strongly recommended.

Class Mailing List

The class mailing list is csc2229-sdn@cs.toronto.edu. Send me an e-mail if you are enrolled in this course and haven't received a welcome message from that list. Please use this mailing list to ask any questions related to this course.

Textbook

The course is based on recent research material, and we do not have a textbook.

Notes and Handouts

I will use a combination of slides and blackboard. Please take notes when I am using the blackboard. The handouts will be available on class web page, so you don't need to take notes.

Grading

- Active participation in class and discussions: 10%
- Paper presentation: 30%
- Final project: 60%
  - Proposal: 5%
  - Intermediate report: 10%
  - Presentation: 15%
  - Final report: 30%

Late Submission Policy

5% of the mark will be deducted for each day late, up to 20%. Assignments/project reports will not be accepted after 4 days. The deadline for the final report is hard.

Academic Offenses

"Briefly, an academic offence is a bad thing done to get marks you don't deserve. Slightly more formally, an academic offence is an action by a student or course instructor that breaks the rules about academic credit at the University of Toronto."[1] Cheating is considered a very serious offense. Please avoid it! We are all here to teach and learn after all, and concerns about cheating make an unpleasant environment for everyone.

Permitted Collaboration

The following items are encouraged and allowed at all times for all students in this class:

- Discussion of material covered during lecture, problem sessions, or in handouts
- Discussion of the requirements of an assignment
- Discussion of the use of tools or development environments
- Discussion of general approaches to solving problems
- Discussion of general techniques of coding or debugging
- Discussion between a student and a TA or instructor for the course

Collaboration Requiring Citation

Two students engaging in more detailed discussions must be careful to document their collaboration. Students are required to include the names of those who provide specific assistance to properly credit their contribution, in the same manner as one would cite a reference in a research paper. The expectation is that even with a citation, the author must be able to explain the solution.

Examples of Collaboration That Require Citation

- Discussing the "key" to a problem set or programming assignment. Problem set questions are often designed such that the critical concept takes careful thought and gaining that insight from someone else must therefore be documented.
- Discussing the design of a programming project. Design is a crucial aspect of the programming process and discussion can be valuable. Any design input received from others must be cited.

[1] Jim Clark, "Advice about academic offenses", http://www.cs.toronto.edu/~clarke/acoffences/.

- Receiving assistance from another student in debugging code. While the TAs are the preferred source for advice, any detailed assistance from someone else must be credited.
- Sharing advice for testing. For example, if someone provides important information on lessons learned ("my program didn't handle the case where the value was 0") that source must be credited.
- Research from alternative sources. Researching related topics, such as through the Internet, must be documented if the solution submitted is derived from the research information.

Unpermitted Collaboration

All submissions must represent original, independent work. Some examples of activities that do not represent original work include:

- Copying solutions from others. In particular, do not ask anyone to provide a copy of his or her solution or, conversely, give a solution to another student who requests it. Similarly, do not discuss algorithmic strategies to such an extent that you and your collaborator submit exactly the same solution. Use of solutions posted to websites, such as at other universities, is prohibited. Be aware that we photocopy some of the exams prior to handing them back.
- Using work from past classes. The use of another student's solution or the posted class solutions from a previous class constitutes a violation.
- Studying another student's solution. Do not read another solution submission whether in electronic or printed form, even to "check answers."
- Debugging code for someone else. When debugging code it is easy to inadvertently copy code or algorithmic solutions. It is acceptable to describe a problem and ask for advice on a way to track down the bug. [2]

[2] Parts of this note are based on handouts from Nick McKeown, and Tom Fountain, who teach CSC244a and EE182 respectively at Stanford. Some portions are based on similar collaboration policies written by Eric Roberts, Julie Zelenski, and the Computer Science Department at Brown University.