CS 551/CS 651 Syllabus

Ramesh Govindan

Logistics: Spring 2022

Class Information

- **Location:** ZHS 352
- **Time:** 4:00-5:50pm MW

Instructor: Ramesh Govindan

- **Office Location:** TBD
- **Office Hours:** Tu 4-6pm
- **Contact mode:** Please contact on Piazza only

Teaching Assistant: TBD

- **Office Location:** Online
- **Office Hours:** Will announce on Piazza
- **Contact mode:** Please contact on Piazza only

Overview

This is a first-year graduate class in computer networks. As such, this class will give you a deeper understanding of wired and wireless networking, from the routing layer and up, and expose you to classic and contemporary literature on the subject. The class also has a series of hands-on labs on an emulated network which will give you insight into what it takes to build real networking subsystems.

The class has the following components:

- **Lab Assignments (50%).** During the course of the semester, you will complete three lab assignments, filling in various pieces of emulated network. Starting from a learning bridge, this sequence of lab exercises asks you to fill in important pieces of routing and transport implementation and configuration. At the end of the course, you will be asked to demonstrate Web downloads across an ISP network that you have configured, across a router you have implemented, and using a transport protocol recently designed by Google. **All lab assignments are to be done individually. *For the lab assignments, you will need to program in C and Python**.
- **In-class quizzes (50%).** There will be four quizzes. Each quiz will test not only the reading materials, but also your understanding of the lab exercises. All quizzes will be closed book (but you may bring one sheet of 8.5x11 paper with anything written on it) and will contain a mixture of multiple-choice and short answer questions.

Prerequisites

Technically, the pre-requisites for this class are CS402/CS350 **and** EE 450/CS353. However, this year, we will admit you into this class if you have taken **one** of these classes. This means that you should determine for yourself whether you have the background necessary for the class. To help you do this, we will:

· Provide you with a self-assessment quiz in the first week of classes
· Assign a programming lab in the first two weeks of class

If you have trouble with either of these, you may not have the necessary background and *you should consider dropping the class*.

Student Responsibilities

By enrolling this class, here is what you're agreeing to.

Preparation

You must keep up with the assigned readings. If you come to class without having read the papers for the corresponding lecture, you're unlikely to learn anything at all from the lecture. In particular:

· You are expected to read all the papers in detail. Not all details will be covered in class. I will assume knowledge of material covered in CS 402 and EE/CS 450, a C/C++ programming proficiency from CS402 or its equivalent, and Python programming proficiency. If you covered the introductory material at some other school it is YOUR responsibility to fill in any missing background. Feel free to ask me for advice on appropriate introductory readings if you feel your background is insufficient.
· I expect you to attend every class meeting. If you do happen to miss a session, you are responsible for finding out what material was covered and if any administrative announcements were made.

Lab assignments and supplementary materials

All work turned in on programming assignments and exams must be that of the individual student. If the instructor finds any evidence of cheating on an exam, he reserves the right to discipline the student appropriately (see below).

This policy also holds for programming assignments. In this class, we will use sophisticated automated program checkers to detect cheating. Be aware that the program checkers have demonstrated very good results and are widely used within the academic community. Any student caught cheating **will be given zero** for the assignment and **will be reported to SJACS**.

For the programming assignments, we will provide you with a github repository with starter code. At the end of the course you will be permitted to make a private copy of the repository for yourself, but you should not make the repository public, or share it with friends, under any circumstances.

We will make available additional supplementary material from the lectures. These include PDFs of papers, PDFs of lecture notes and lecture videos. These will be made available for viewing only (downloads disabled) and only for the duration of the course. You should not make any of this material available online; failure to follow these instructions may result in disciplinary action.

Before class starts, we will ask you to sign an integrity form that records that you have read and understood these rules.

News and Discussion Forum

All interaction with the instructor and with other students for the class will be through its Piazza site. You can access the class Piazza through Blackboard.

You are expected to check Piazza periodically for important announcements.

If you wish to contact me, please use Piazza (rather than my regular email), and I will try to respond to your question quickly. While you should always feel free to contact me, I strongly encourage you to post your question to Piazza; you will usually get much faster responses from your fellow students.

Lab Late Submissions Policy

Three times during the semester, a student may extend the due date of a programming assignment by twenty four hours without needing prior permission. These are known as "grace days." In order to use a grace day, you must fill out this grace day request form **before** the lab's non-extended deadline. (You will need to be logged into your USC account to access this form).

Please note that grace days are in place of "excused late" submissions, not in addition to. If you request additional grace days from the instructor, you must have a documented reason for each grace day used to accompany your request. Once you have used your grace days, any late submission will not be accepted and graded as a 0.

Note: There is no grace period. Even if you submit a few minutes after the deadline, you will need to use a grace day (even if the wireless network in your dorm room is down or you have a github issue, etc.). It is your job to be on time and not cut it too close. Remember Murphy's Law and leave time for things to "go wrong."

Reading List

Reference material

· **[PetersonDavie]** L. Peterson and B. Davie. *Computer Networks: A Systems Approach*. Morgan-Kaufman, 3rd edition, 2003. recommended text
· **[Hanson]** M. J. Hanson. Efficient Reading of Papers in Science and Technology. unpublished manuscript, 2000 (PDF)
· **[Keshav07]** S. Keshav. How to Read a Paper. *ACM Computer Communications Review*, 2007 (PDF)
· **[Levin]** R. Levin and D. Redell. An Evaluation of the Ninth SOSP Submissions. *ACM Operating Systems Review*, 17(3):35–40, 1983 (PDF)

Inter-Domain Routing

· **[BGPIntro]** H. Balakrishnan. Internet interdomain routing (PDF)
· **[BGPPolicies]** M. Caesar and J. Rexford. Bgp routing policies in isp networks. *Netwrk. Mag. of Global Internetwkg.*, 19(6):5–11, November 2005 (PDF)
· **[DelayedConvergence]** Craig Labovitz, Abha Ahuja, Abhijit Bose, and Farnam Jahanian. Delayed internet routing convergence. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '00, pages 175–187, New York, NY, USA, 2000. ACM (DOI)

Transport I

· **[Jacobson]** V. Jacobson. Congestion avoidance and control. In *Symposium Proceedings on Communications Architectures and Protocols*, SIGCOMM '88, pages 314–329, New York, NY, USA, 1988. ACM (DOI)
· **[PadhyeModeling]** Jitendra Padhye, Victor Firoiu, Don Towsley, and Jim Kurose. Modeling tcp throughput: A simple model and its empirical validation. *SIGCOMM Comput. Commun. Rev.*, 28(4):303–314, October 1998 (DOI)
· **[BBR]** Neal Cardwell, Yuchung Cheng, C. Stephen Gunn, Soheil Hassas Yeganeh, and Van Jacobson. Bbr: Congestion-based congestion control. *ACM Queue*, 14, September-October:20 – 53, 2016 (DOI)

Topology

· **[VL2]** Albert Greenberg, Navendu Jain, Srikanth Kandula, Changhoon Kim, Parantap Lahiri, David A. Maltz, Parveen Patel, and Sudipta Sengupta. Vl2: A scalable and flexible data center network. In *SIGCOMM '09: Proceedings of the ACM SIGCOMM 2009 conference on Data communication* (PDF)
· **[Jupiter]** Arjun Singh, Joon Ong, Amit Agarwal, Glen Anderson, Ashby Armistead, Roy Bannon, Seb Boving, Gaurav Desai, Bob Felderman, Paulie Germano, Anand Kanagala, Jeff Provost, Jason Simmons, Eiichi Tanda,

Jim Wanderer, Urs Hölzle, Stephen Stuart, and Amin Vahdat. Jupiter rising: A decade of clos topologies and centralized control in google's datacenter network. *SIGCOMM Comput. Commun. Rev.*, 45(4):183–197, August 2015 (DOI)

· **[ASStructure]** Ricardo V. Oliveira, Dan Pei, Walter Willinger, Beichuan Zhang, and Lixia Zhang. In search of the elusive ground truth: The internet's as-level connectivity structure. In *Proceedings of the 2008 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '08, pages 217–228, New York, NY, USA, 2008. ACM (DOI)

Architecture and Design Principles

· **[ClarkDesign]** David D. Clark. The design philosophy of the darpa internet protocols. *SIGCOMM Comput. Commun. Rev.*, 25(1):102–111, January 1995 (DOI)

· **[SaltzerEndtoEnd]** J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-end arguments in system design. *ACM Trans. Comput. Syst.*, 2(4):277–288, November 1984 (DOI)

· **[ShenkerFundamental]** Scott Shenker. Fundamental design issues for the future internet. *IEEE Journal on Selected Areas in Communication*, 13(7), September 1995 (PDF)

Network Management

· **[Ethane]** Martin Casado, Michael J. Freedman, Justin Pettit, Jianying Luo, Natasha Gude, Nick McKeown, and Scott Shenker. Rethinking Enterprise Network Control. *IEEE Transactions on Networking*, 2009 (PDF)

· **[P4]** Pat Bosshart, Dan Daly, Glen Gibb, Martin Izzard, Nick McKeown, Jennifer Rexford, Cole Schlesinger, Dan Talayco, Amin Vahdat, George Varghese, and David Walker. P4: Programming protocol-independent packet processors. *SIGCOMM Comput. Commun. Rev.*, 44(3):87–95, July 2014 (DOI)

· **[B4]** Sushant Jain, Alok Kumar, Subhasree Mandal, Joon Ong, Leon Poutievski, Arjun Singh, Subbaiah Venkata, Jim Wanderer, Junlan Zhou, Min Zhu, Jon Zolla, Urs Hölzle, Stephen Stuart, and Amin Vahdat. B4: Experience with a globally-deployed software defined wan. *SIGCOMM Comput. Commun. Rev.*, 43(4):3–14, August 2013 (DOI)

· **[NetworkVirtualization]** Teemu Koponen, Keith Amidon, Peter Balland, Martin Casado, Anupam Chanda, Bryan Fulton, Igor Ganichev, Jesse Gross, Paul Ingram, Ethan Jackson, Andrew Lambeth, Romain Lenglet, Shih-Hao Li, Amar Padmanabhan, Justin Pettit, Ben Pfaff, Rajiv Ramanathan, Scott Shenker, Alan Shieh, Jeremy Stribling, Pankaj Thakkar, Dan Wendlandt, Alexander Yip, and Ronghua Zhang. Network virtualization in multi-tenant datacenters. In *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14)*, pages 203–216, Seattle, WA, 2014. USENIX Association (DOI)

Transport II

· **[DemersFQ]** A. Demers, S. Keshav, and S. Shenker. Analysis and simulation of a fair queueing algorithm. In *Symposium Proceedings on Communications Architectures &Amp; Protocols*, SIGCOMM '89, pages 1–12, New York, NY, USA, 1989. ACM (DOI)

· **[DCTCP]** Mohammad Alizadeh, Albert Greenberg, David A. Maltz, Jitendra Padhye, Parveen Patel, Balaji Prabhakar, Sudipta Sengupta, and Murari Sridharan. Data center tcp (dctcp). *SIGCOMM Comput. Commun. Rev.*, 40:63–74, August 2010 (PDF)

· **[MPTCP]** Damon Wischik, Costin Raiciu, Adam Greenhalgh, and Mark Handley. Design, implementation and evaluation of congestion control for multipath tcp. In *Proceedings of the 8th USENIX Conference on Networked Systems Design and Implementation*, NSDI'11, pages 99–112, Berkeley, CA, USA, 2011. USENIX Association (DOI)

Schedule

Approximate Weekly Schedule

Topic	Start Date
Inter-Domain Routing	Jan 10
Transport I	Jan 31
Topology	Feb 16
Architecture and Design Principles	Mar 7
Network Management	Mar 28
Transport II	Apr 11

Schedule of Quizzes

All quizzes are in-class, closed book. Each exam will be 30 mins at the beginning of the lecture on specified date.

Exam	Date
Quiz 1	Feb 9
Quiz 2	Mar 2
Quiz 3	Apr 6
Quiz 4	Apr 27

Lab Submission Deadlines

All labs are due **midnight** of the due date.

Lab	Due Date
1	Jan 28
2a	Feb 11
2b	Feb 25
3a	Apr 1
3b	Apr 29

Grading

Final letter grades will be determined using a "modified" curve. I will assign grades of C and BELOW to individuals who do not perform satisfactorily in the above areas (i.e., you should not assume that because this is a graduate class you will get a B or even B- if you perform unsatisfactorily.).

We will not assign incompletes unless it is for a documented medical reason (in accordance with USC policy).

Statement for Students with Disabilities

Any student requesting academic accommodations based on a disability is required to register with Disability Services and Programs (DSP) each semester. A letter of verification for approved accommodations can be obtained from DSP. Please be sure the letter is delivered to me (or to TA) as early in the semester as possible. DSP is located in STU 301 and is open 8:30 am - 5:00 pm, Monday through Friday. The phone number for DSP is (213) 740-0776.

Statement on Academic Integrity

USC seeks to maintain an optimal learning environment. General principles of academic honesty include the concept of respect for the intellectual property of others, the expectation that individual work will be submitted unless otherwise

allowed by an instructor, and the obligations both to protect one's own academic work from misuse by others as well as to avoid using another's work as one's own. All students are expected to understand and abide by these principles. Scampus, the Student Guidebook, contains the Student Conduct Code in Section 11.00, while the recommended sanctions are located in Appendix A: `http://www.usc.edu/dept/publications/SCAMPUS/gov/`. Students will be referred to the Office of Student Judicial Affairs and Community Standards for further review, should there be any suspicion of academic dishonesty. The Review process can be found at: `http://www.usc.edu/student-affairs/SJACS/`.