



---

[ComputingIV /](#)  
**COMP2040**

[COMP.2040 Spring 2018](#)

---

[Home](#) [Syllabus](#) [Assignments](#) [Resources](#) [Lecture Blog](#)

## [Dr. Yelena Rykalova](#)

Visiting Faculty, Computer Science

[yelena\\_rykalova@uml.edu](mailto:yelena_rykalova@uml.edu)

- [Note:](#)
  - If you are contacting me
    - regarding class I am teaching please include [the class and section number](#)
    - regarding advising please include [your ID #](#).

### **Office**

Olsen 220A

### [Office hours](#)

MWF 10:45 AM - 11:45 AM

### [Section 201](#)

#### **Class time**

MWF 1:00 PM - 1:50 PM

#### **Class location**

Olsen Hall 109

### [Section 202](#)

#### **Class time**

MWF 2:00 PM - 2:50 PM

#### **Class location**

Olsen Hall 109

### [Office hours](#)

MW 2:00 PM - 3:45 PM

### [TAs](#)

**[Subhajit Chakrabarty](#)**

[Subhajit\\_Chakrabarty@student.uml.edu](mailto:Subhajit_Chakrabarty@student.uml.edu)

Office Olsen Hall 212A

Tue Wed 2:00 PM – 3:00 PM

**Allison Rossetto**

[Allison\\_Rossetto@student.uml.edu](mailto:Allison_Rossetto@student.uml.edu)

Office Olsen Hall 212A

Mon Wed 3:00 PM - 4:00 PM

## Tutoring service

The Centers for Learning and Student Success ("CLASS"): <https://www.uml.edu/CLASS/Tutoring/tutor-schedule/>

## Course Overview

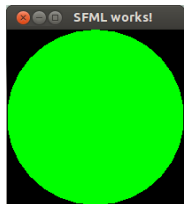
- We will be writing lots of code
- We will be doing it in C++
- We will be using some pretty cool APIs, including [SFML](#), a free, open-source “simple fast media library” for C++, which is available for Mac, Win, and Linux (translation: it's a gaming library)
- Best of all, we'll be working through some awesome problem sets developed over the last 15 years at [Princeton](#), led by Robert Sedgewick.

The Princeton stuff is great because it's all about how computing connects to the larger world.

It's not the usual stuff about using computing to do ever-more complicated things. This is necessary and valuable, but it's kind of self-referential and insular.

We will do the following:

- [PS0](#) Learn how to use to events, windows, and animation with the Simple Fast Media Library ([SFML](#))

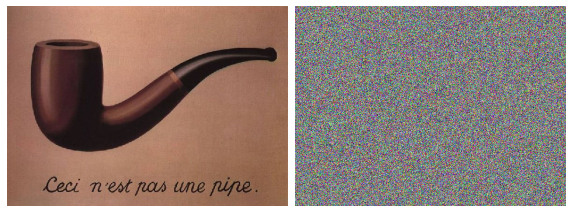
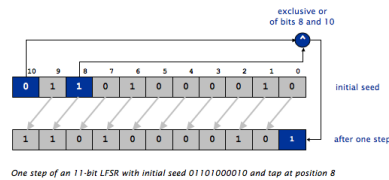


- [PS1](#) Use a recursive class definition to draw Sierpinski's triangle, as well as your own original recursive graphic designs (and learn how to use a code style checker)



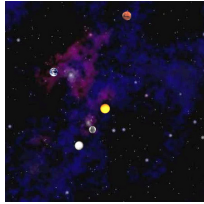
thanks [mathisfun.com](http://mathisfun.com)

- [PS2a](#) [PS2b](#) Implement a linear feedback shift register (LFSR), and use it to cryptographically encode and decode an image (and learn about unit testing)



thanks [Princeton](#)

- [PS3a](#) [PS3b](#) Use Newton's laws and the “leapfrog finite difference approximation” method to create a realistic, animated simulation of planetary motion in our solar system



thanks [Princeton](#)

[video](#)

- [PS4](#) Compare ASCII strings to compute their “edit distance”—a technique widely used in bioinformatics for DNA analysis—and perform space and time analyses of the dynamic programming algorithm employed in the solution

x\y	0	1	2	3	4	5	6	7	8
T	A	A	G	G	T	C	A	-	-
0 A	7	8	10	12	13	15	16	18	20
1 A	6	6	8	10	11	13	14	16	18
2 C	6	5	6	8	9	11	12	14	16
3 A	7	5	4	6	7	9	11	12	14
4 G	9	7	5	4	5	7	9	10	12
5 T	8	8	6	4	4	5	7	8	10
6 T	9	8	7	5	3	3	5	6	8
7 A	11	9	7	6	4	2	3	4	6
8 C	13	11	9	7	5	3	1	3	4
9 C	14	12	10	8	6	4	2	1	2
10 -	16	14	12	10	8	6	4	2	0

thanks [Princeton](#)

- [PS5a](#) [PS5b](#) Use a ring buffer to simulate the vibration of a guitar string, using the Karplus-Strong algorithm, and generate a set of audio files to make a computational synthesizer



thanks [Princeton](#)

- [PS6](#) Analyze a body of text to build a Markov model, and use that model to generate randomized but plausible text output—a practical implementation of Claude Shannon's classic work on information theory, which is used in many present-day applications, including speech recognition and predictive typing

trajectory:	ga	-->	ag	-->	gg	-->	gc	-->	cg	-->	ga	-->	ag	-->	ga	-->	aa	-->	ag
probability for a:	1/5		3/5		1/3		0		1		1/5		3/5		1/5		1/2		
probability for c:	0		0		1/3		0		0		0		0		0		0		
probability for g:	4/5		2/5		1/3		1		0		4/5		2/5		4/5		1/2		

thanks [Princeton](#)

- [PS7a](#) [PS7b](#) In a partnership with [Kronos](#) (Chelmsford, MA), parse large error logs files generated by malfunctioning InTouch devices (their hardware time-clock unit) using regular expression and time-parsing libraries. Note: these log files are not neatly well-formed as is proper XML!



thanks [Kronos](#)

- [PSx](#) (optional assignment) This assignment gives you a chance to resubmit a previous homework (for higher grade.)
- [Portfolio?](#)

---

Page last modified on March 15, 2018, at 02:23 AM