# PS1

# RECURSIVE GRAPHICS

We'll be implementing the Sierpinski triangle assignment described at http://www.cs.princeton.edu/courses/archive/fall13/cos126/assignments/sierpinski.html.

Some notes:

- You should create a Sierpinski class that derives from `sf::Drawable`. Then, you can have it just `draw` itself to your main window. If you are familiar with Java, this would be called "implementing the `Drawable` interface":

```
// instantiate
Sierpinski st(7); // parameter is recursion depth
// maybe want to give it a size parameter too
```

- Your executable must read two parameters (integers): recursion-depth and window-size. You should create a square SFML window that's exactly as big as the window-size argument, and your triangle should fill it.

# WHAT TO TURN IN

It's important that you turn in everything needed to build your projects.

Please note that you will have **two projects** for this assignment:

(1) the Sierpinski implementation

(2) your own original work.

Create a directory with all your work and with **both projects in the same directory.**

Your `Makefile` should build both projects. The two resulting executables should be named `sierpinski` and `original`. **Use unique names for the files in the two projects, so you can put all the files in the same directory.**

Each of the two executables should take two arguments:

1. Recursion depth
2. Initial window height

Your `Makefile` should contain two targets: `all` and `clean`. The former should build both executables, and the latter should remove the executables, `.o` files, and all other temporary files created during the build.

The directory should be named **ps1** and contain:

6. A `readme.txt` file that includes:
   - Your name
   - A discussion of what you did—at least 100 words.What you actually implemented, and, what was interesting, hard, fun, or easy about your project.
   - How much time you invested (optional)
   - Anything else you'd like us to know (optional)

Remember, we will have to build and run your code, so make sure to submit all that's needed!

Use `tar` command from the parent directory of your `ps1`:

```
tar czvf ''<archive-file-name>''.tar.gz ps1
```

to compress your directory structure.

# HOW TO TURN IT IN

Submit using the submit utility as follows:

**submit schakrab ps1 ps1**

## But wait, there's more...

Also, we want you to share (in the discussion group) your original image with the class.

# GRADING RUBRIC

**original graphic project** (8)

file name correct 1

reads depth and window-size args 1

is somewhat different from Sierpinski project 2

implements draw function as derived class of `sf::Drawable` 2

screenshot and explanation is posted to course discussion group 2

(include your First and Last name for grading purpose; the discussion group is at https://groups.google.com/forum/#!forum/computing4summer2018 )

**Makefile** (6)

builds objects associated with sierpinski project 1

links "sierpinski" executable 1

builds objects associated with original project 1

links "original" executable 1

"make all" builds everything 1

"make clean" removes temporary files, objects, and executables 1

**tar.gz archive** 2

(all files packaged in .tar.gz file with correct directory structure)

**readme.txt** 4