Start Assignment

Due Jan 22 by 11:59pm **Points** 30 **Submitting** a text entry box
Available Jan 9 at 12am - Jan 29 at 11:59pm 21 days

Fundamentals

Assignment: Percolation Part 1

Learning Objectives

- Select the appropriate data structure(s) to solve a problem given a set of programming specifications.
- Implement and use a Monte Carlo simulation to estimate the percolation threshold

Overview

This, as well as upcoming assignments, should be completed with a partner.
The expected collaboration in CSIS 2420 is different from the teamwork you might have experienced in other courses. In many software projects, team members divide up the code and then merge it into a working project. In CSIS 2420, this is not the case. instead, both assignment partners are expected to work on all of the code. Please read the **CSIS-2420 Collaboration Guideline**.

In this assignment, you will write a program to estimate the value of the percolation threshold via Monte Carlo simulation.

The assignment is broken up into two parts. In this first part of the assignment, you are asked to implement one out of two classes.

Instruction

Find a partner unless you got one assigned already. Try to have a different one for each of the team assignments. This allows you to work with multiple classmates, strengthen your teamwork skills, and build a network of friends and colleagues that can support you as you transition to a 4-year college or the workplace.

Assignment Instructions:

The assignment instructions **A01 Percolation.pdf** are based on an assignment of Princeton's algorithm course. It comes with additional resources: a checklist that includes FAQ and a video that helps you get started.

This first part of the assignment requires the implementation of the class **Percolation**. It should be created in a **package called a01**.

In order to earn full points, the class needs to pass most of the JUnit tests on CodePost (see rubric). You can submit as often as you like but only up until the specified deadline.
Before you submit via Canvas, take a screen-shot of the CodePost website that shows your bruinmail, the class that is tested, and the number of passing/failing tests. Here are two examples:

- ***Sample screenshot where all tests pass***
- ***Sample screenshot where some tests fail***

Additional Resources:

Important: Whenever there is a difference or inconsistency between the assignment instructions (pdf) and one of the additional resources, the instructions need to be followed.

Why are there occasional inconsistencies between the instructions and the resources?
It helps you prepare for the workforce. It is often possible to find resources that include helpful information, but those resources are rarely targeted to a specific task a company asked you to perform. It is an important skill to identify information that is relevant and to discern information that does not apply to a given task.

Checklist:

Here is a **checklist** that includes frequently asked questions, test clients, and possible progress steps.

Videos:

Here is a **video from Professor Sedgewick** **(https://cuvids.io/app/video/101/watch)** where he introduces percolation as an example

Here is another video that you might find helpful. The beginning shows the course site from Princeton. Even though your instructions are based on theirs there are some important differences. Accessing information directly from the Princeton course site might provide you with information that does not match the requirements of this course.

video **(https://www.youtube.com/watch?v=o60oHXesOuA)**

(https://www.youtube.com/watch?v=o60oHXesOuA)

To keep the scope more manageable I removed the requirement to analyze the runtime and memory usage.

Submission

One team member embeds the screenshot from CodePost in the text entry submission.
How to embed images in Canvas **(https://community.canvaslms.com/t5/Student-Guide/How-do-I-embed-images-from-Canvas-into-the-Rich-Content-Editor/ta-p/356)**

Both team members submit the name of the partner and the discussed pebble distribution. If the pebble distribution is not 50/50, include a description that explains the difference.

A01 Part 1 Rubric

Criteria	Ratings								Pts
JUnit tests for class Percolation	**30 pts Full Marks** No more than 2 tests fail	**26 pts 3 Tests fail**	**22 pts 4 tests fail**	**18 pts 5 tests fail**	**14 pts 6 tests fail**	**10 pts 7 tests fail**	**6 pts 8 or more tests fail**	**0 pts Insufficient** No submission or the tests could not be executed.	30 pts
									Total Points: 30