# *Challenge Problems I*

September 24, 2018
CS: DS&A
Proof School

This is a list of fun challenge problems. Some of them are theoretical and some of them involve programming. See how many of them you can do over the course of the year. Needless to say, these are all *optional*!

## Some O(1) space problems

1. Design an algorithm that takes two linked lists and determines whether they intersect; i.e. determines whether they share a Node object in common. (Not whether they have data in common, but whether they have an actual Node object in common.)

Your algorithm should:

- Not alter the input lists.
- Use $O(1)$ space.
- Take $O(n)$ time.

2. We've seen in class that the stack data type has two operations, `pop()` and `push()`. When the stack is implemented by an array or a linked list, these both take O(1) time.

Design a data structure that behaves like a stack, with `pop()` and `push()`, but that also has a third command `min()`, which tells you the *minimum value contained in the stack*. (You may assume that all your values are real numbers.) For example, suppose you ran:

```
> push(5)
> push(7)
> push(10)
> push(1)
> push(33)
```

At this point, `min()` should return 1. But after running `pop()` twice, `min()` should return 5.

Oh, I forgot to tell you the thing that makes this problem challenging! The `min()` function must run *in O(1) time*.

3. Find a constant-space algorithm that determines whether a linked list has a loop. (That is, a node that points to an earlier node in the list.) Then find an algorithm that finds where the loop begins. How efficient can you make it?

4. $N$ robots are arranged in a row. They would like to begin exterminating the humans, but need to start their revolution in sync. Every second, each one can pass a command to its neighbors (the robots immediately to the left and the right).

The robot on the left end believes that it's time for things to begin, and he needs to communicate this with the other robots. What algorithm can they come programmed with that enables them to all begin extermination at the same second?

Note: you guessed it–they only have O(1) space in their heads. They don't know how many robots there are in the line, nor do they have space to even store such a number!

## Coding

5. Try the famous Great Tree-List Recursion problem! I'll give a link in Google Classroom.