POTD57.1. Problem of the Day #57

Download and Extract

An initial setup of files is provided to you via a shell script: Download potd-q57

Using a terminal, extract the initial files by running the shell script you just downloaded (you will need to navigate to the directory where you saved the file):

```
sh potd-q57.sh
```

Your files for this problem will be in the `potd-q57` directory.

The Problem

Your goal is to detect cycles in directed graphs.

Your Job

Your job is to determine whether or not a graph contains a cycle. A cycle in a directed graph is a set of nodes $\{x_1, ..., x_n\}$ in which an edge exists from x_i to x_{i+1} for all `1 <= i < n` and an edge from x_n to x_1

You will edit the function `bool hasCycles(const Graph& g)`, which returns true/false based on whether or not a graph `g` contains a cycle.

There are m̀aǹỳ ways to determine whether or not a directed graph contains a cycle. There are brute force ways and there are more elegant ways, but as long as your implementation works, you will get full points!

Example

Consider the following:

```
Graph g(3);
g.addEdge(0, 1);
g.addEdge(0, 2);
g.addEdge(1, 2);
```

This would create a graph with 3 nodes (`0`, `1`, `2`) with edges from `0` to `1`, `0` to `2`, and `1` to `2`.

The output of `hasCycles(g)` should return `false`.

On the other hand:

```
Graph g(3);
g.addEdge(0, 1);
g.addEdge(1, 2);
g.addEdge(2, 0);
```

Would create a graph with 3 nodes in which `hasCycles(g)` returns `true`

Hint

How do you think a search algorithm like BFS/DFS can help detect cycles? Think about a graph that doesn't contain cycles and a graph that does, and then imagine how a generic search algorithm would behave on each. What do you think should happen when a search algorithm is performed on a node in a cycle?

Generic search algorithms are not the only way to detect cycles. If you plan to take `CS374` you will learn more about how to detect cycles in directed graphs. Think of this as a warm up to that! Feel free to implement the `hasCycles` function anyway you want.

Upload Solution

Drop files here or click to upload.

Only the files listed below will be accepted—others will be ignored.

POTD 57

Total points:	0/1
Score:	0%

Question

Value:	1
History:	
Awarded points:	0/1

Report an error in this question

Previous question

Next question

Files

○ cycle_detection.cpp
not uploaded

Save & Grade | Save only