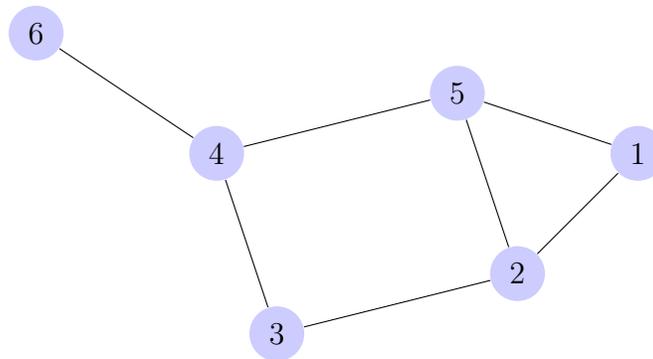

CS 440

ASSIGNMENT 1 (DUE SEPT. 9TH AT 5PM)

The Maximum Clique Problem

This assignment is intended as a warmup - getting acquainted with Python and as a reminder of graphs. Your task is to write a program that computes the size of the maximum clique in a graph. Recall that a clique is a fully connected subgraph. For example, in the graph shown below, the nodes 4 and 5 form a clique of size 2, and the maximum clique is the set of nodes $\{1, 2, 5\}$.



In case you are familiar with NP-completeness in CS320, you will recall that the problem of finding the maximum clique is NP-complete, i.e. there exists no polynomial time algorithm for solving it. Therefore you will need to employ a brute force approach; we suggest simply check all possible subsets of nodes. As a result, don't expect your code to run very fast, and you won't be able to apply it to large graphs.

Your task is to write a python module called `p1.py` that contains a function called `maximum_clique` that returns the size of the maximum clique in a graph. The expected interface is provided below. Your program needs to be able to read an undirected graph from a file. The file format it should recognize is a simplified version of the `dot` format. Here's an example:

```
graph G {
    u -- w; /* a comment */
    w -- r; /* an edge between w and r */
    w -- s;
    s -- u;
    t; /* a node that doesn't have any incoming or outgoing edges */
}
```

Note that we don't guarantee whitespace between the node identifier and the `--`, so don't use that as an aid for parsing. This is an undirected graph called `G` with nodes `r,s,t,u,w`

and four edges. The full spec of the dot language is at <http://www.graphviz.org/doc/info/lang.html>, but you don't need to be able to parse the full language—the example covers the features required to define a graph. For simplicity, assume that the identifiers for the nodes are any string that does not contain `--` as a substring.

As a side-note: you can visualize graphs in dot format using the Graphviz program (<http://www.graphviz.org/>).

Specifications, notes, and hints

Your module needs to have the following classes and functions:

- **Graph.** A class that represents a graph. The constructor should take in a file name that contains the representation of the graph in dot format as shown above. An instance of the class should have an instance variable called `adjacencyMatrix` that represents the adjacency matrix of the graph, i.e. `adjacencyMatrix[i][j]` should be 1 if there is an edge between nodes with indices `i` and `j`. One way to create such a matrix in Python is as a “list of lists”. The following command will create an $n \times n$ matrix of zeros:

```
>>> matrix = [ [0 for i in range(n)] for j in range(n) ]
```

- **maximum_clique:** a function (not part of the `Graph` class) that takes in an instance of `Graph` and returns the size of the maximum clique in the graph.

We will test your code using something like:

```
>>> import p1
>>> g = p1.Graph(fileName) # load a graph in dot format
>>> print p1.maximum_clique(g)
```

Submit your module, `p1.py` using the checkin program.