CSC 210 HOMEWORK 6 (DUE FRIDAY OF WEEK 10)

PROFESSOR GODFREY C. MUGANDA

You are to write a program that tries to find an optimal coloring for a given graph[1]. Colors are applied to the nodes of the graph and the only available colors are black and white. The coloring of the graph is called optimal if a maximum of nodes is black. The coloring is restricted by the rule that no two connected nodes may be black. This is illustrated in Figure 1.
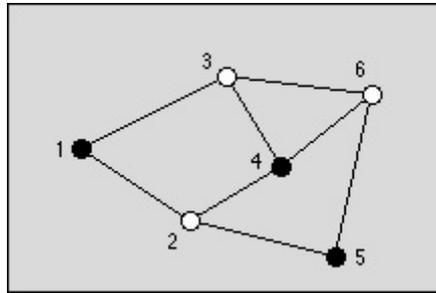

FIGURE 1. A colored graph with the largest number of black nodes

1. INPUT AND OUTPUT

The graph is given as a set of nodes denoted by numbers $1 \ldots n$, where $n \leq 100$, and a set of undirected edges denoted by pairs of node numbers (n_1, n_2) , where $n_1 \neq n_2$. The input file contains m graphs. The number m is given on the first line. The first line of each graph contains n and k, the number of nodes and the number of edges, respectively. The following k lines contain the edges given by a pair of node numbers, which are separated by a space.

The output should consists of $2m$ lines, two lines for each graph found in the input file. The first line of should contain the maximum number of nodes that can be colored black in the graph. The second line should contain one possible optimal coloring. It is given by the list of black nodes, separated by a blank.

Here is a sample input file with just one graph. Your program should handle any number of graphs in the file. You must use a `JFileChooser` object to allow the user to open file of their choosing.

```
1
6 8
1 2
1 3
2 4
2 5
3 4
3 6
```

---

[1]This problem is due to Rujia Liu

```
4 6
5 6
```
The corresponding output is
```
3
1 4 5
```

2. Additional Requirement

Your solution must follow the pattern of a backtracking solution given in the lectures. If you know the number n of vertices of a graph, you can write a backtracking solution that looks for a subset of size n that can be colored black. If that fails, then look for a subset of size $n - 1$ that can be colored black, and so on.