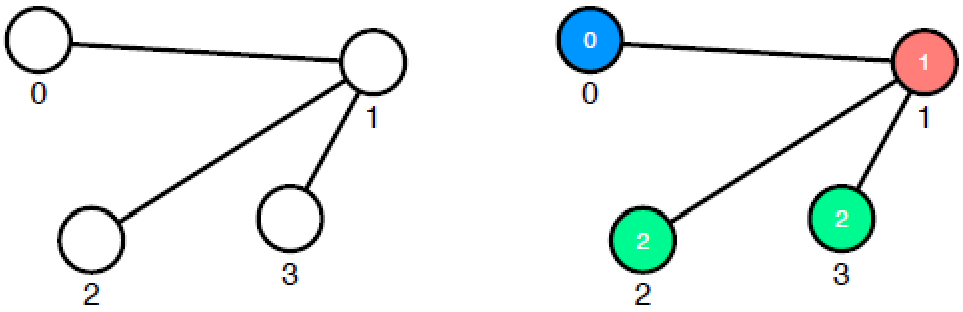
CS490
Assignment 3

Graph Coloring

In this assignment you will design an algorithm to find the smallest coloring of a (undirected) graph. (This is sometimes called the Chromatic Number of a graph). You are provided with a graph and your task is to label the graph's nodes with as few colors as possible such that all pairs of nodes joined by an edge do not have the same color. Figure 1 illustrates a graph and a three coloring of that graph.



(a) A graph.

(b) A three-coloring of the graph.

Figure 1: A Graph Coloring Example

The nodes of the graph are labeled with black numbers while the coloring of the graph is labeled with white numbers. You may notice that a three coloring is not a minimal coloring of this graph. In fact, a two coloring is possible.

The problem is mathematically formulated in the following way. Given a graph G = (N,E) with nodes N = 0 .. n-1 and edges E, let $c_i \in$ N be a variable denoting the color of node i. Then the graph coloring problem is formalized as the following optimization problem:
Minimize the maximum c_i from i=0 to n-1 such that if the edge (u,v) ∈ E, then $c_u \neq c_v$.

The input consists of |E| + 1 lines. The first line contains two numbers: the number of vertices and the number of edges. It is followed by |E| lines, each line i represents an edge (u_i, v_i).

The output has two lines. The first line contains the number of colors used to color the graph.
The next line is a list of color values for each vertex. Your colors should start at 0 and increase by 1 from there. **Make sure the order of vertices in your output matches the numberings of vertices used in the edge definitions of the input!**

Input Example (corresponding to the figure above)

```
4 3
0 1
1 2
1 3
```

Output Example

```
3
0 1 2 2
```