CS 202, Summer 2013

Homework Assignment 3

Due: 23.55, July 26, 2013

In this programming assignment, you must write an efficient program to check whether or not a (directed) graph *G = (V; E)* provides following conditions:

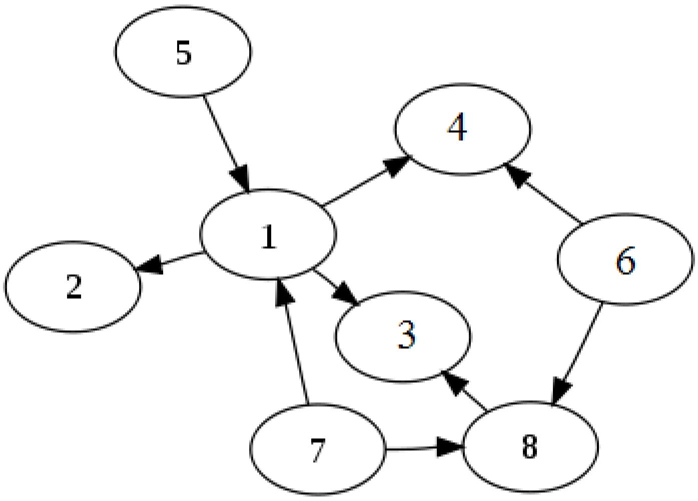
> a) The graph *G = (V; E)* is red-black. A graph is said to be red-black if and only if it is possible to assign one of two colors (red or black) to each of the vertices of the graph such that no two vertices which are connected by an edge are assigned the same color.
>
> b) The graph *G = (V; E)* is cycle-free. A graph is said to be cycle-free with no directed cycles. That is, it is formed by a collection of vertices and directed edges, each edge connecting one vertex to another, such that there is no way to start at some vertex *v* and follow a sequence of edges that eventually loops back to *v* again.

The input file to your program is in the following format. The first line of the file contains the number of vertices (*n*). The vertices are numbered 1 through *n*. Each subsequent line contains a pair of vertices such that each such pair defines a directed edge from first vertex to second vertex.

Sample Input File (input.txt)

```
8
1 2
1 4
1 3
5 1
6 4
6 8
7 1
7 8
8 3
```

Corresponding Graph



You are required to print (stdout) the output of the conditions in following format:

```
The given graph has 8 vertices.
It is (not) a red-black graph.
It is (not) an acyclic graph.
..
End of execution!
```

The name of the input file will be provided as command line arguments to your program. Thus, we call your programs using two command line arguments in the following format:

```
username@dijkstra:~>./graphCheck <inputFileName>
```

Hints: Consider using standard graph traversal techniques and modifying them. Your algorithm should run in O(|V| + |E|) steps.

Notes:

1. This assignment is due by 23.55 on Friday, July 26th, 2013. You should upload your homework using Moodle course page. No hardcopy submission is needed. The standard rules about late homework submissions apply. Please see the course syllabus for further discussion of the late homework policy as well as academic integrity.

2. Your code must not have any memory leaks. You will lose points if you have memory leaks in your program even though the outputs of the operations are correct.

3. You should upload all files as a single archive file (i.e. zip file). The name of archive file should be in following format; *id_surname_name.zip*. The submissions that do not obey these rules will not be graded.

4. You are free to write your programs in any environment (you may use either Linux or Windows). On the other hand, we will test your programs on "`dijkstra.ug.bcc.bilkent.edu.tr`" and we will expect your programs to compile and run on the "`dijkstra`" machine. If we could not get your program properly work on the "`dijkstra`" machine, you would lose a considerable amount of points. Therefore, we recommend you to make sure that your program compiles and properly works on "`dijkstra.ug.bcc.bilkent.edu.tr`" before submitting your assignment.

5. This homework will be graded by your TA Sami Arpa (arpa@cs.bilkent.edu.tr). Thus, you may ask your homework related questions directly to him.