CS 2110: Computer Programming Lab (Odd 2012)
Lab 10

Heuristic Solutions for Graph Problems

Instructor - John Augustine

October 29, 2012

Notes

- This is the second in a 3-part series of algorithms on graphs.
- Please follow the input/output format.
- This assignment is open ended. There is NO known polynomial time optimal solution. Hence, you should devise a solution that is a good approximation of the optimal solution. (Recall that we did this for the knapsack problem.)
- Your solution must run in reasonable time; in particular, your solution must scale well to work on large graphs. If you use high time complexity solutions for large input graphs, then you will lose marks.

Problem Definition

You are given an undirected simple graph $G = (V, E)$. Let V_1 and V_2 partition V. Edges of the form (v_1, v_2), $v_1 \in V_1$ and $v_2 \in V_2$ are called bridges induces by the partition V_1 and V_2. You are to report a two-part partition of V such that the number of bridges is maximized.

0.1 Command Line Arguments

Your command line argument for this assignment is a single input file name.

0.2 The Input File Format

Each input file should contain one graph. The format is as follows. First you must have an integer n which represents the number of vertices on which the graph is defined. Given n, the n-element vertex set is implicitly understood to be $\{0, 1, \ldots, n-1\}$. Then, you will need to have a $n \times n$ binary matrix. The following is an example.

```
4
1 0 0 1
1 0 1 1
0 0 0 0
0 1 0 1
```

Couple of notes regarding the input:

1. The input may not be a symmetric matrix. If either $a_{ij} = 1$ or $a_{ji} = 1$, then you should interpret it as an undirected edge between vertex i and vertex j.

2. The input can be in free format. The elements in the matrix need not be formatted in the proper matrix format. The only guarantee you have is that each element of the matrix is separated from the next (taken in row major sequence) by some white space. For example, the following is equivalent to the above input example.

```
4 1 0 0 1 1 0
1 1

0 0 0

0
0 1

0 1
```

0.3 Output

For this lab, you again get a reprieve from strict output formatting. You must simply report all the required information in an understandable way.

You must report:

1. the elements of part V_1 in increasing order (note that V_2 is implied),

2. the number of bridge edges,

3. a list of bridge edges.

Uploading into MOODLE

Your code should be written as a single .c file. You must first compress the file using `gzip -c filename.c > filename.c.gz` and then the compressed .gz file must be uploaded into moodle. A link will be set up for this purpose in moodle.

Your TA for this lab

CS08B031, CS10B052, CS11B001 — CS11B009	Saurav Kant Jha
CS11B011 — CS11B021	Tejas Kulkarni
CS11B022 — CS11B032	Paresh Nakhe
CS11B033 — CS11B042	Shrikant Polawar
CS11B043 — CS11B053	Sai Sreenivas
CS11B054 — CS11B063	Nishaanth