BLG 336E – Analysis of Algorithms II, Spring 2014

Project 3 – Real Estate Matching

Total Worth : 10% of your grade

Handed Out : 02.05.2014, Friday

Due : 16.05.2014, Friday

Overview

You will develop an algorithm for a real estate agent. Agent has **n** apartments to hire and **m** customers. After visiting the apartments, customers provide a list of apartments that they are willing to hire. They may hire any apartment in their list but they refuse to hire the apartments that are not in their list. There is no priority among the apartments in their list. One apartment can be hired only by one customer.

Agent wishes to find a matching between customers and apartments such that maximum number of apartments are hired. In this project, you will suggest and implement an algorithm which finds this matching.

Hint: First you need to model the problem as a graph, then modify this graph in a way that an algorithm in your lecture slides can be applied to solve it.

Project Definition

Code (60 pts.)

Your input file contains the list of apartments preferred by each customer. Each line represents a customer and each integer represents an apartment id. As explained before, apartments in a line are in no particular order, the customer is willing to hire any of them.

After calculating the matching in which maximum number of apartments are hired, your program should print the customer ID – apartment ID pairs to the standard output (console). Customer IDs are the line numbers of the input file.

Your program should compile and run using the following commands:

```
g++ yourStudentID.cpp –o realEstate
./realEstate input.txt
```

Your program will be evaluated with a different input file, with different number of customers and apartments.

Report (40 pts.)

1. Explain the purposes of all your classes and methods. **[5 pts.]**
2. Explain how you modelled the problem as a graph, draw a sample graph. **[5 pts.]**
3. Explain how you modified this graph, show on your sample graph. **[5 pts.]**
4. Give pseudo code of your algorithm, explain its steps and run it for your sample graph. **[15 pts.]**
5. Calculate time complexity of your algorithm. **[10 pts.]**

Submission

You should be aware that the Ninova system clock may not be synchronized with your computer, watch, or cell phone. Do not e-mail the teaching assistant or the instructors your submission after the Ninova site submission has closed. If you have submitted to Ninova once and want to make any changes to your report, you should do it before the Ninova submission system closes. Your changes will not be accepted by e-mail. Connectivity problems to the Internet or to Ninova in the last few minutes are not valid excuses for being unable to submit. You should not risk leaving your submission to the last few minutes. After uploading to Ninova, check to make sure that your project appears there.

Policy: You may discuss the problem addressed by the project at an abstract level with your classmates, but you should not share or copy code from your classmates or from the Internet. You should submit your own, individual project. Plagiarism and any other forms of cheating will have serious consequences, including failing the course.

Submission Instructions: Please submit your project files through Ninova. Please upload all your code files in a *.zip or *.rar archive. In the archived file, you must include all your program and header files. A second *.pdf file that contains your report must also be uploaded.

All your code must be written in C++, and we must be able to compile and run on it on ITU's Linux Server (you can access it through SSH) using g++. You should supply one yourStudentID.cpp file that calls necessary routines for all questions (Multiple files are acceptable, as long as you state the compilation instructions in your report).

When you write your code, try to follow an object-oriented methodology with well-chosen variable, method, and class names and comments where necessary. Your code must compile without any errors; otherwise, you may get a grade of zero on the assignment.

If anything in this document is not clear, please let the teaching assistant know by email (aralat@itu.edu.tr).