CSIEB0100 Data Structures, Fall 2015
Final Exam

ID: _____ Dept:_____ Name: _____

1. **(20%)** The most frequently used buttons in a browser are the forward and backward buttons to traverse the URLs visited by the user. Based on the **doubly linked lists** data structure, write a C++ program to maintain the browsing history in the following ways.

(a) Design a class `BrowsingHistory` to maintain the browsing history of a user. Each node should store the URL (string) of a web page. The links of a node represent the previous and next page visited by the user.

(b) Write a `CurrentPage` function to return the URL of the current web page under browsing. Also provide a `NewPage` function to add a new URL into the history.

(c) Write a `Forward` and a `Backward` function to traverse the browsing history and return the URL of the new current page.

(d) Provide a `Display` function to print out the entire browsing history.

(e) Test your functions properly in the main program.

(This page is left empty for you to answer the question.)

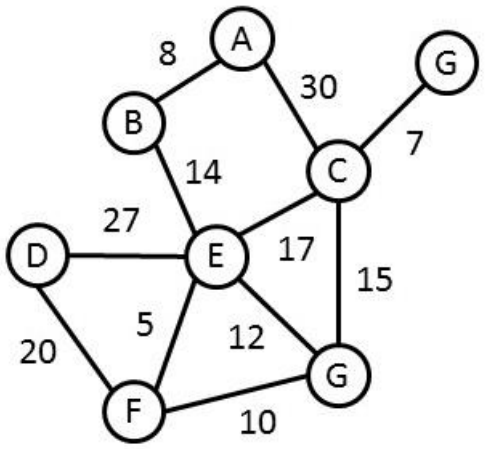
2. **(20%)** Consider the users of Facebook with links representing friendships between users.
   (a) Design a data structure (ADT) to represent the users and friendships.
   (b) Write a C++ class to implement your ADT.
   (c) Write a C++ program to find all direct and indirect friends of a given user using the class you design in (b).

(This page is left empty for you to answer the question.)

3. **(20%)** Answer the following questions about max-heaps.
   (a) How many elements of a max-heap of height *h* can have? Explain your answer.
   (b) Where in a max-heap might the smallest element reside, assuming that all elements are distinct? Explain your answer.
   (c) Draw the max-heaps (initially empty) after each insertion of elements 9, 11, 12, 3, 14, 15, 7, 8, 1 in that order.
   (d) Draw the max-heaps after each of two delete operations.

4. **(20%)** Answer the following questions about binary search trees (BSTs).
   (a) Draw the final BST after inserting elements 29, 38, 13, 6, 21, 17, 25, 30, 10, 8, 32 in that order.
   (b) Draw the tree after deleting 25 from the BST in (a).
   (c) Draw the tree after deleting 38 from (b).
   (d) Draw the tree after deleting 13 from (c).
   (e) List the elements of pre-order, in-order, and post-order traversals of the tree from (d).

5. **(20%)** Answer the questions using the following graph.



(a) Draw the sequence of edges (represented by edge weight) added to the minimal-cost spanning tree generated by the Kruskal's algorithm.

(b) Do the same thing as (a) using Prim's algorithm.

(c) Do the same thing as (a) using Sollin's algorithm.

(d) Find the shortest paths from E to all other vertices using Dijkstra's algorithm.