BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI

Second Semester 2013-2014

CS /IS F211 Data Structures and Algorithms

Lab Sheet – 8 [Duration: 150 minutes]

General Instructions for Programming

1. All inputs to the program must be either (a) command line arguments (b) or read from a file (other than stdin). DO NOT READ anything from stdin and DO NOT USE PROMPTS like “Please enter a number …”.
2. You are required to write the output to a file (other than stdout) and errors if any to a different output channel (stderr or another file).
3. Use standard C coding conventions for multi-file programs. Separate the following: interfaces of functions (use a “.h” file), data type definitions (use another “.h” file), ADT / algorithm implementation (use a “.c” file), and driver/test code (use another ”.c” code). In general, each module has to be written in **separate** c files.
4. All files related to a lab **must** be put inside a single directory by the name of the lab (lab1, lab2, etc.).
5. Valid makefile must be present in the directory.
6. Ensure that all the code written by you are compiling correctly. Preferably use gcc with the options **`-W -Wall -O2`**, while compiling your code.
7. Instructions for uploading the files shall be provided separately.

Problem 1

A B-tree is a tree data structure that keeps data sorted and allows searches, sequential access, insertions, and deletions in logarithmic time. The B-tree is a generalization of a binary search tree in that a node can have more than two children. Unlike self-balancing binary search trees, the B-tree is optimized for systems that read and write large blocks of data. It is commonly used in databases and file systems.

Use Large data for lab exercise from course webpage and perform the following tasks

a). Implement BST and B-Tree.

b). Perform time complexity comparison in both cases.