CS340 Winter 1996

Ming Li

Assignment #3, Due Date: March 6

1. (1 mark) Do Problem 6.2 in the textbook (Lewis&Denenberg).

2. (1 mark) Give a formal and complete proof of the following fact I presented in the class: it takes at least $\frac{3n}{2} - 2$ comparisons to choose the maximum element and the minimum element from an unsorted array of n integers.

3. (8 marks) Programming. The goal of this problem is to program and use a set of functions (or procedures) to perform various operations on *B-trees.* This assignment deals with the B-trees of of order $m = 5$ (treat this as a symbolic constant, so it is easy to change). The following is a list of functions and their descriptions, where P is always a pointer pointing to the root of the B-tree (of order 5):

   a. **Create**(P) {Initializes P pointing to an empty B-tree.}

   b. **Insert**(P,x) {Insert element x into the B-tree. Return 1 (success) or 0 (failure, x is already in the tree).}

   c. **Delete**(P,x) {Delete element x from the B-tree}

   c. **Search**(P,x) {Search for x in the B-tree. Return 1 (found) or 0 (not found).}

   d. **Print**(P) {Print the B-tree according to the example in Figure 1. Print the keys in the tree in increasing order, one per line, and use indentation (as in Figure 1) to show the level of each key in the tree.}

   To test your functions, write a main program that reads a sequence of commands from its standard input. The commands should be:

   - c

     Create an empty B-tree. Should be used only once, as the first command in the sequence.

   - i x

     Insert key x into the B-tree.

   - d x

     Delete key x from the B-tree.

   - s x

     Search for x.

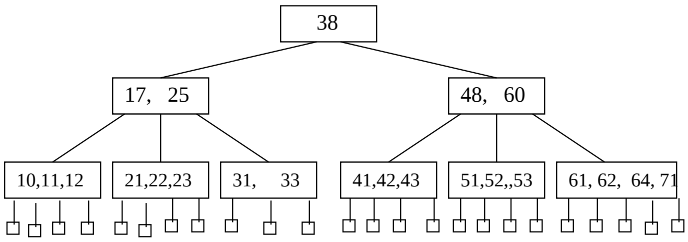
   - p

     Print the tree.

   Each command should print a line describing what it is doing, and the result of the operation. For example:

   Inserting 3: success
   Searching for 4: found

   Run your program using as input the five files found under directory `~cs340/AA3`. This assignment might not be easy, it is probably the most difficult one for this semester (for this course). Try to implement easy functions



(a) A B-tree of order 5.

10
11
12
17
21
22
23
25
31
33
38
41
42
43
48
51
52
53
60
61
62
64
71

(b) Output of the B-tree in (a).

Figure 1: A B-tree of order 5 and its output

first. We will allow you to omit the "delete" function at the cost of 2 marks. That is, if you do not implement the delete function, we will deduct only 2 marks. (If you do not do "delete", then you should not use in4 and in5 in AA3.)

Electronically submit your program, and hand in hard copies. For the programming part, hand in documentation including a cover page, the user documentation (what the program does), the design documentation (how it does it), the source code, and the test results.