COMPUTER SCIENCE AND ENGINEERING
Indian Institute of Technology, Palakkad
CS2130: Data Structures and Algorithms Lab
Lab 11 (Hashing)

IIT PALAKKAD

12 Oct, 2018

Time: 3 hrs

General Instructions

- You are expected to follow good programming practices (refer to the book titled *C How to Program* by Dietel and Dietel). Your programs should handle all kinds of test cases - positive, negative, corner cases/boundary conditions, good input, bad input and are never expected to crash on any input.
- You are also expected to provided a high level description of the functions used in your programs. There is no restriction on the format except that it should be concise and precise. Further, the running time and additional space (excluding the space for the arguments) used by the functions should be clearly stated and justified.
- **Please note that poor programming practices and improper documentation can attract a penalty of up to 20% of the marks allocated to a question.**

1. (30 points) Write a program to implement the standard operations of a binary search tree with distinct positive integer values in every vertex as keys.

   Input-Output Format: Each line contains a specific operation to be performed on the tree.

   INS x. Insert **x** into the tree.
   PST. Prints the postorder traversal sequence of the tree with every pair of consecutive elements separated by a space. If the tree is empty, print -1.
   DEL x. Deletes the vertex **x** from the tree. If **x** has 2 children, then replace **x** with its inorder predecessor. If the deletion is unsuccessful, then print -1.
   MAX. Print the maximum element in the tree, if it exists. Otherwise, print -1.
   MIN. Print the minimum element in the tree, if it exists. Otherwise, print -1.

```
Sample Input:
MAX
MIN
PST
INS 20
INS 30
INS 10
INS 1
INS 2
INS 3
INS 5
```

Continued on next page...

```
INS 1
INS 100
INS 110
INS 90
INS 15
PST
MAX
MIN
DEL 10
PST
DEL 20
DEL 100
DEL 110
MAX
MIN
PST
DEL 99
DEL 30
DEL 1
DEL 2
DEL 3
DEL 5
DEL 90
DEL 15
PST
INS 1
MAX
MIN

Expected Output:
-1
-1
-1
5 3 2 1 15 10 90 110 100 30 20
110
1
3 2 1 15 5 90 110 100 30 20
90
1
3 2 1 5 90 30 15
-1
-1
1
1
```

2. (70 points) Write a program to implement a hash table to store distinct keys that are positive integers with collisions resolved using the chaining method.

Question 2 continues on the next page. . .

Input-Output Format: The first line of the input contains the number m denoting the number of slots in the hash table. Each subsequent line contains a specific operation to be performed on the hash table.

INS x. Insert **x** into the table using the hash function **h(x) = x mod m**. Assume that the slots of the hash table are numbered from 1 to $m-1$. If the insertion is not successful or if the element already exists, then print -1.
PRT i. Print the keys that are hashed to slot **i** of the table in the ascending order of their insertion times. If the table has no elements in slot **i**, print -1. If the table does not have ith slot, then print -1.
DEL x. Deletes **x** from the table. If there is no entry with key **x**, then print -1.
SRCH x. Searches for the key **x** in the table. If it is present print Y, otherwise print N.

```
Sample Input:
10
INS 10
INS 5
INS 1
INS 7
INS 100
INS 25
INS 10
INS 7
PRT 0
PRT 7
PRT 1
PRT 7
DEL 7
SRCH 23
SRCH 5
PRT 7
PRT 2
INS 15
INS 14
INS 90
INS 17
PRT 7
PRT 4
PRT 5
```

```
Expected Output:
-1
-1
10 100
7
```

Question 2 continues on the next page. . .

```
1
7
N
Y
-1
-1
17
14
5 25 15
```