

## Lab08

### 100 Points

#### Due date:

11:59pm, Monday 04/08/2019 for Tuesday labs.

11:59pm, Wednesday 04/10/2019 for Thursday labs.

#### Purpose:

The purpose of this lab is to implement a Min-5 Heap class and a Max-5 Heap class in C++.

#### General Requirements:

In this assignment you will work on an array based implementation of Min 5-Heap and Max-5 heap. You are to read in the numbers from a data file of integers and insert each number into a Min 5-heap (and Max-5 heap).

#### In Min 5-Heap (and Max-5 heap):

- The root of T is at  $A[0]$ .
- The parent of  $A[i]$  is at  $A[\text{floor}((i-1)/5)]$ , if it exists.
- The  $j^{\text{th}}$  child of  $A[i]$  is at  $A[5i+j]$ ,  $1 \leq j \leq 5$ , if it exists.

Use the **bottom-up method** for building the heap. (If a different approach is used, the results will be different. For the bottom-up approach, first put all of the elements in the given order into an empty heap, and then try to heapify starting from the last parent, and continue to heapify towards the root. (Heapify is done in reverse level order). The file of numbers you read from will be data.txt. You may hard code the file name in your program if you wish.

In this lab, you should build the heap using the samples which are in the data.txt. After that, your program should have a simple menu like this:

-----  
Please choose one of the following commands:

- 1- Insert
- 2- DeleteMin
- 3- DeleteMax
- 4- FindMin
- 5- FindMax
- 6- LevelOrder
- 7- Exit

The array should be of size 500. **Heaps can have duplicate numbers.** The max element in Min heap (Min element in Max heap) is one of the leaves, and the index of the first leaf can be found using  $A[\text{floor}((\text{last used index in array}-1)/5) + 1]$ .

### Min 5-Heap:

The Min 5-Heap methods should be implemented as follows:

- `buildheap()` - should build the Min-5 heap using the bottom up approach.
- `insert(x)` - should insert `x` into the Min 5-Heap.
- `deletemin()` - should delete the minimum value from the Min 5-Heap.
- `findmin()` - should print the minimum value from the Min 5-Heap.
- `deletemax()` - should delete the maximum value from the Min 5-Heap.
- `findmax()` - should print the maximum value from the Min 5-Heap.
- `levelorder()` - should print out all the elements of the Min 5-Heap using level order traversal.

### Max 5-Heap:

The Max 5-Heap methods should be implemented as follows:

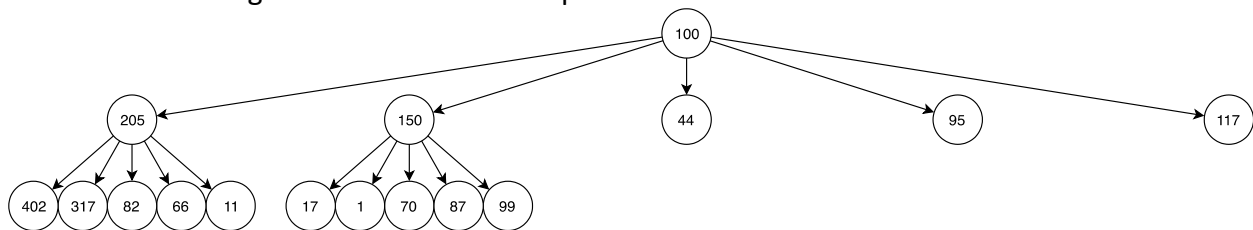
- `buildheap()` - should build the Max-5 heap using the bottom up approach.
- `insert(x)` - should insert `x` into the Max 5-Heap.
- `deletemin()` - should delete the minimum value from the Max 5-Heap.
- `findmin()` - should print the minimum value from the Max 5-Heap.
- `deletemax()` - should delete the maximum value from the Max 5-Heap.
- `findmax()` - should print the maximum value from the Max 5-Heap.
- `levelorder()` - should print out all the elements of the Max 5-Heap using level order traversal

### Expected output for Min-5 Heap:

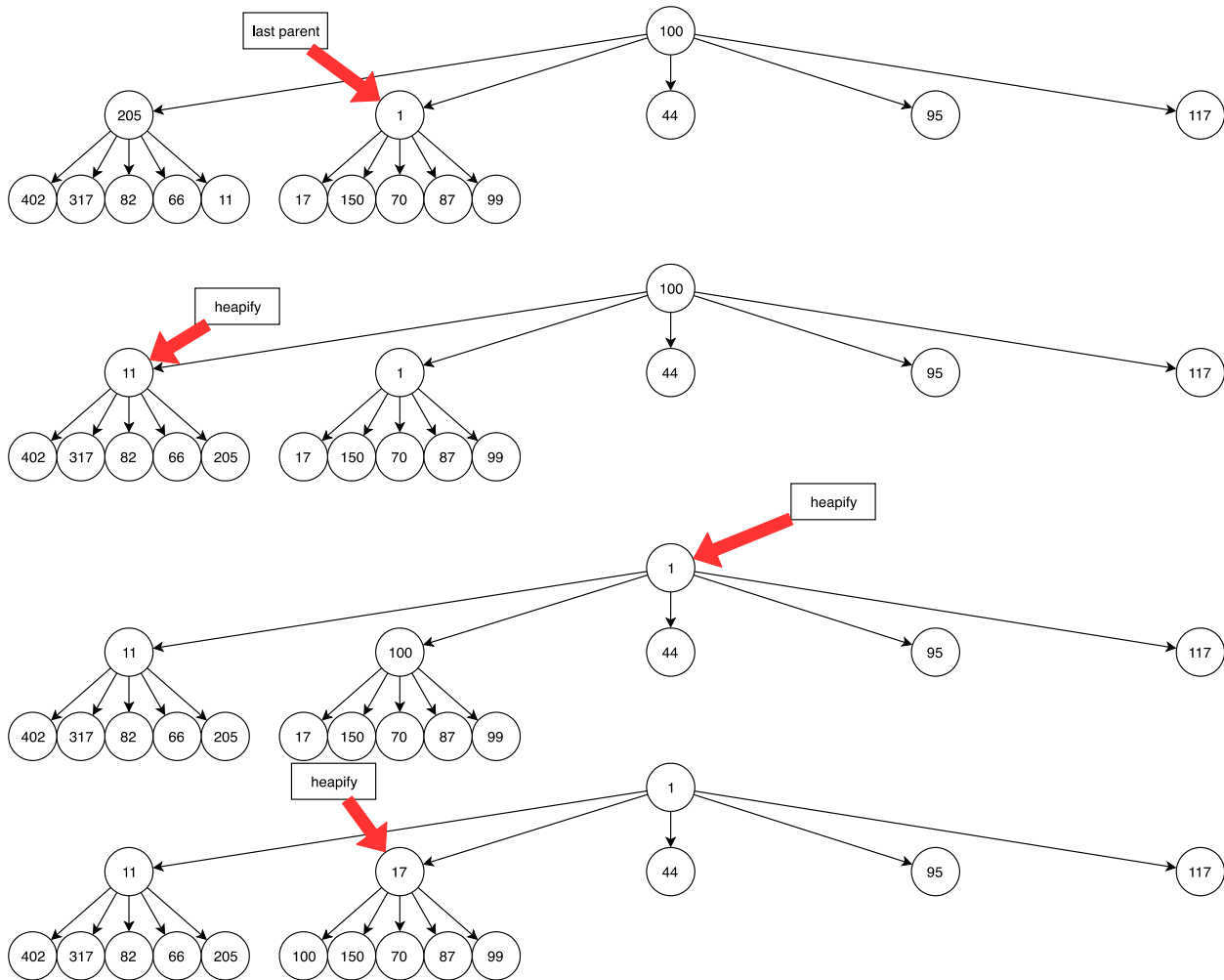
data.txt elements: 100 205 150 44 95 117 402 317 82 66 11 17 1 70 87 99

We will insert these values, in the given order, into the Min-5 Heap. We will use the bottom-up approach to build the heap as given in your lecture notes. **Note the following graphical output only covers the Min-5 Heap:**

Insert all values in given order into the heap:



Start heapifying from the last parent to the root:



Please note that you are not expected to show the Heap graphically in your output. This is just for your reference.

-----  
Please choose one of the following commands:

- 1- Insert
  - 2- DeleteMin
  - 3- DeleteMax
  - 4- FindMin
  - 5- FindMax
  - 6- LevelOrder
  - 7- Exit
- >6  
1

11 17 44 95 117

402 317 82 66 205 - 100 150 70 87 99

-----  
Please choose one of the following commands:

- 1- Insert
- 2- DeleteMin
- 3- DeleteMax
- 4- FindMin
- 5- FindMax
- 6- LevelOrder
- 7- Exit

>4

Min value is: 1

-----  
Please choose one of the following commands:

- 1- Insert
- 2- DeleteMin
- 3- DeleteMax
- 4- FindMin
- 5- FindMax
- 6- LevelOrder
- 7- Exit

>5

Max value is: 402

-----  
Please choose one of the following commands:

- 1- Insert
- 2- DeleteMin
- 3- DeleteMax
- 4- FindMin
- 5- FindMax
- 6- LevelOrder
- 7- Exit

>2

11

66 17 44 95 117

402 317 82 99 205 – 100 150 70 87

-----  
Please choose one of the following commands:

- 1- Insert

2- DeleteMin  
3- DeleteMax  
4- FindMin  
5- FindMax  
6- LevelOrder  
7- Exit  
>3  
11  
66 17 44 95 117  
87 317 82 99 205 – 100 150 70

-----  
Please choose one of the following commands:

1- Insert  
2- DeleteMin  
3- DeleteMax  
4- FindMin  
5- FindMax  
6- LevelOrder  
7- Exit  
> 1  
Enter a value to be inserted: 11  
11  
66 11 44 95 117  
87 317 82 99 205 – 100 150 70 17

-----  
Please choose one of the following commands:

1- Insert  
2- DeleteMin  
3- DeleteMax  
4- FindMin  
5- FindMax  
6- LevelOrder  
7- Exit  
> 7  
Byebye!

**Expected output for Max-5 Heap:**

-----  
Please choose one of the following commands:

1- Insert  
2- DeleteMin

3- DeleteMax  
4- FindMin  
5- FindMax  
6- LevelOrder  
7- Exit  
> 6  
402  
317 150 44 95 117  
205 100 82 66 11 – 17 1 70 87 99

-----  
Please choose one of the following commands:

1- Insert  
2- DeleteMin  
3- DeleteMax  
4- FindMin  
5- FindMax  
6- LevelOrder  
7- Exit  
> 4  
Min value is: 1

-----  
Please choose one of the following commands:

1- Insert  
2- DeleteMin  
3- DeleteMax  
4- FindMin  
5- FindMax  
6- LevelOrder  
7- Exit  
> 5  
Max value is: 402

-----  
Please choose one of the following commands:

1- Insert  
2- DeleteMin  
3- DeleteMax  
4- FindMin  
5- FindMax  
6- LevelOrder  
7- Exit  
> 3

317  
205 150 44 95 117  
99 100 82 66 11 – 17 1 70 87

-----  
Please choose one of the following commands:

- 1- Insert
- 2- DeleteMin
- 3- DeleteMax
- 4- FindMin
- 5- FindMax
- 6- LevelOrder
- 7- Exit

> 2

317  
205 150 44 95 117  
99 100 82 66 11 – 17 87 70

-----  
Please choose one of the following commands:

- 1- Insert
- 2- DeleteMin
- 3- DeleteMax
- 4- FindMin
- 5- FindMax
- 6- LevelOrder
- 7- Exit

> 1

Enter a value to be inserted: 317

317  
205 317 44 95 117  
99 100 82 66 11 – 17 87 70 150

-----  
Please choose one of the following commands:

- 1- Insert
- 2- DeleteMin
- 3- DeleteMax
- 4- FindMin
- 5- FindMax
- 6- LevelOrder
- 7- Exit

> 7

Byebye!

**Submission:**

Follow the conventions below to facilitate grading:

**Source Code**

Place all your source files (\*.cpp, \*.hpp) and input files in a single folder with no subfolders.

- Name your folder using the convention Lastname\_LabX (e.g., Smith\_Lab08).
- Include a functioning Makefile inside the folder. (The makefile should also include the clean command.)
- **Verify that your code runs on the lab Linux machines before submission.**

**Compressed File**

- Compress using .zip, .rar, or .tar.gz.
- Name your file using the convention Lastname\_LabX (e.g., Smith\_Lab08.zip).

**Email**

- Use the following subject for your email: Lastname\_LabX (e.g., Smith\_Lab08).
- Send you code to l290w868@ku.edu if you are in one of Lei's sections or to dhwanipandya1401@ku.edu if you are in one of Dhwani's sections.
- Anytime you have a question about the lab, put the word question in the subject of the email.