National University of San Agustin
System Engineer School

ADA - Lab 05 - Heap Sort

Christian E. Portugal-Zambrano

October 22, 2018

1 Introduction

In this practice we will be introducing some data structures concepts to the development of the Heap Sort Algorithm. First we will be introducing the concept of binary tree, then a heap must be completely explained, finally a example using the heap to perform an ordering will be presented.

2 Pre-requisites

Some linked and array based queues knowledge is required, but we will be explaining the main concepts related to this. The selection of a programming languages is a must and good skill treating arrays is a basis.

3 Programming environment

You can use all the tools described on practices before, but here you will need some graphics tools, I recommend to use Netbeans for Java or Qt for C++, both tool brings up with a graphics library enough for our practices objectives.

4 Algorithms presentation

So, as we are working with LaTeX algorithms presentation will be easy, you just need to add this at the header of your document:

```
\usepackage{algorithm}
\usepackage{algorithmicx}
\usepackage[noend]{algpseudocode}
```

4.1 Queues

A queue is a data structure named as FIFO (First input, first output) and is commonly used in real world e.g banks queues. We can implement it using links (machine pointers) or arrays (limited size). This data structure is used in problems where the order of insertion is a main concern.

4.2 Heaps

A heap is another data structure where the main concerns is that one member is the maximum (maxheap) or the minimum (minheap) of all the elements, its graphical representation is almost like a tree, this representation is showed in Figure 4.1, the root of the three is the value with max or min value from all of them, there are operation which maintain this order like sink, swim, less and compare. We have to remember that any algorithm works with keys attached to values.

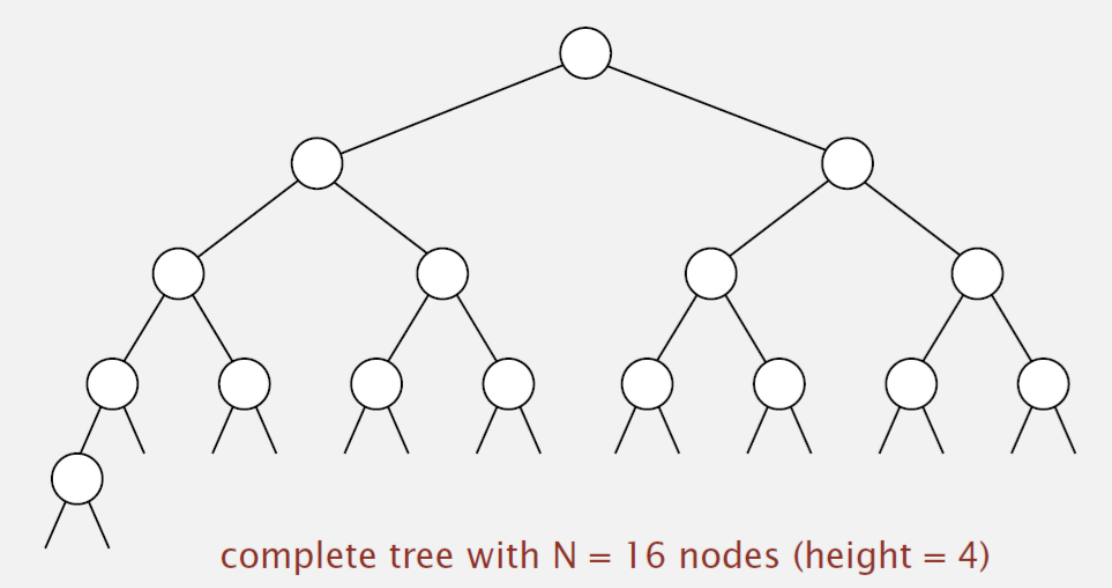

Figure 4.1: Graphical representation of a heap, graphics taken from [1]

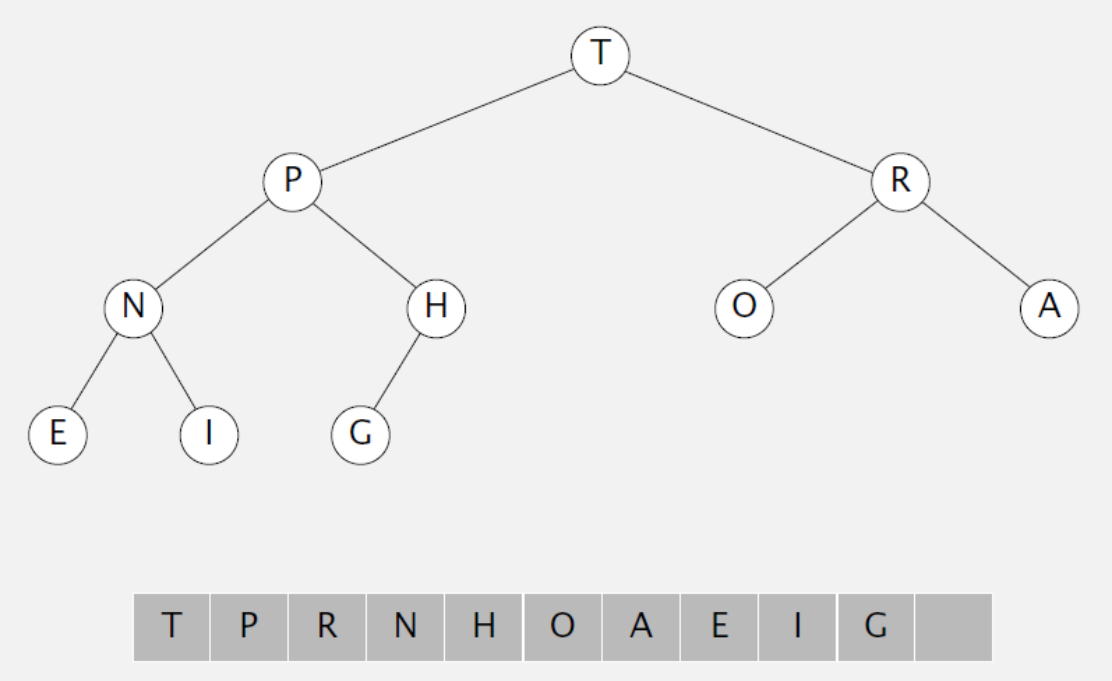

Figure 4.2: Representation of a heap (linked and array based) to perform a heapsort algorithm. Figure taken from [1]

4.3 HeapSort

It is the combination of a queue with a heap, they together make a priority queue. We can use linked or array representation to perform the ordering, in Figure 4.2 we can describe the two types of representation. A heapsort is just an algorithm that order the elements considering the relation parent-child and a max or min value (max or min heap).
To maintain the form of a heap, we need to perform some operations like:

- Promotion.- when a child is larger than its parent.
- Demotion.- when a parent is less than its child.

4.4 Warm-up

For this practice you will have to implement a heap, remember that you can do this through an array or pointer based implementation.

4.4.1 To Do:

For this section you will be implement a Max/Min heap sort using the heap previously, then make an analysis of time using R and compare it with merge and quick-sort for different order of data (nearly sorted, few unique, inverse ordered and random), in which

cases Heap sort overcome the others? which are the differences between those three algorithms?.

5 Deep inside

This section is just for anyone who wants to make a deep inside into the theory and like challenges[1], you will have to prepare a presentation of just 5 minutes to explain and run the algorithm, then you will have to defend yourself another five minutes of questions. I want that all students benefit from your presentation, remember that we are here to learn. If you want to do this please email to cportugalz@unsa.edu.pe

6 Deadline

For this practice one score will be taken at class time at nearly November 05. Remember that plagiarism must be avoided and if it is detected the grade will be zero and repetition informed to superior authorities. The work can be realized with up to 3 people and a demo must be presented. All question and doubts must be done to the same email.

References

[1] R. Sedgewick and P. Flajolet, *An introduction to the analysis of algorithms*. Pearson Education India, 2013.

[1] This must not be included into the report