<u>CSCE 2211 Exercises</u> Exercises (5) Heaps & Priority Queues

1. Given the following array of keys:

- Show the steps of building up a minimum heap for that array and the removal from the heap to sort the array.
- What are the numbers of comparisons used in the up-heap and down-heap operations?

Answer

After insertion, the heap array will be:

-inf	4	4	9	6	12	15	13

The insertion steps will use the following number of comparisons:

1+1+1+1+1+1+2 = 8 comparisons

The output of removal will be: 4 4 6 9 12 13 15, i.e. sorted in ascending order.

The removal steps will use the following number of comparisons:

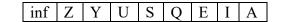
$$4+4+2+2+1+1+0 = 14$$
 comparisons

2. Given the following array of character keys:

- Show the steps of building up a <u>maximum</u> heap for that array and the removal from the heap to sort the array in alphabetical order.
- What is the number of comparisons used in insertion and removal?

Answer

After insertion, the heap array will be:



The insertion steps will use the following number of comparisons:

number of comparisons.

$$1+1+2+3+1+1+1+4 = 14$$
 comparisons

The output of removal will be: Z Y U S Q I E A, i.e. sorted in descending alphabetical order. Work out the number of comparisons in the removal process.

- **3.** What is the complexity (Big-O) of insertion in a minimum heap if:
 - All keys are equal? Answer: $O(n \log n)$
 - Keys are already sorted in ascending order? Answer: O(n)
 - Keys are already sorted in decreasing order? Answer: O(n log n)
- **4.** Outline an algorithm for checking whether an array H[1..n] is a minimum heap.

Answer: A Pseudocode could be

```
Algorithm isMinHeap (H , n) set flag \leftarrow true set i \leftarrow 1 while flag AND i \leq n parent \leftarrow H(i) Lc \leftarrow 2i Rc \leftarrow 2i+1 if (Lc \leq n) if (P > H(Lc)) flag \leftarrow false if (Rc \leq n) if (P > H(Rc)) flag \leftarrow false i \leftarrow i+1 Return flag
```

5. Design an algorithm to find the kth smallest element in an array of size (n) using a minimum heap. What will be the complexity of such algorithm in the worst case?

Answer

```
//To return the kth smallest element in a min heap
int kthsmallest (int X[], int n, int k)
{
    int i, km;
    PQ <int> Heap(n);
    for (i = 1; i <= n; i++) Heap.insert(X[i]);
    for (i = 1; i <= k; i++) km = Heap.remove(); return km;
}
```

Complexity will be $O((k+n) \log n)$

- **6.** Which position could be occupied by the 3rd largest key in a maximum heap of size 32?
- 7. Write a function to merge two minimum heaps into one minimum heap.

<u>Answer</u>

```
void Merge2Heaps(PO<int> & Heap1, PO<int> & Heap2, PO<int> & Heapm, int N,
int M)
{
   int i, x1, x2;
   int K = (M>N)?N:M;
   for (i = 1; i \le K; i++)
        x1 = Heap1.remove(); \quad x2 = Heap2.remove();
        if(x1 < x2) // for efficiency
             { Heapm.insert(x1); Heapm.insert(x2); }
                 Heapm.insert(x2);
                                     Heapm.insert(x1); }
        else {
   if(N>M)
     for(; i \leq K + abs(N-M); i++)
     \{x1 = Heap1.remove(); Heapm.insert(x1); \}
   else
    for( ; i \leq K + abs(N-M); i++)
    \{x2 = Heap2.remove(); Heapm.insert(x2); \}
}
```

- 8. The largest element in a maximum heap must appear in position 1, and the second largest element must be in position 2 or position 3. Give the list of all positions in a heap of size 15 where the k-th largest element can appear for k = 3, 4, 5. Assume all elements are distinct.
- **9.** Show how to implement a first-in, first-out queue with a priority queue. Show how to implement a stack with a priority queue.
- **10.** Give an algorithm using heaps to merge k sorted arrays into one sorted array.