

Tutorial 02: January 23

This tutorial focuses on runtime analysis and priority queues. There are 4 questions in total - 2 easy [E], 1 medium [M], and 1 hard [H].

1. [M] Loop Analysis - Iteration

Provide a tight Θ bound on the following pseudocode as a function of n :

Algorithm 1: ITERATIVE PSEUDOCODE

```

1  $k \leftarrow 1$ 
2 for  $i$  FROM 1 TO  $n$  do
3    $j \leftarrow 0$ 
4   while  $j \leq n$  do
5      $j \leftarrow j + k$ 
6   end
7    $k \leftarrow 2k$ 
8 end

```

2. [E] Algorithm Design - Heaps

How would you implement a stack using a heap? Analyse the complexity of the push and pop operations.

3. [H] Loop Analysis - Recursion:

Provide a tight O bound for the runtime of the following algorithm. You may assume that at each recursive step, n is an integer.

Algorithm 2: STOOGESORT(A, i, j)

Input: Array A of size n , index i (initially 0), index j (initially $n - 1$)

Output: No output but the subarray $A[i \dots j]$ will be sorted

```

1 if  $A[j] < A[i]$  then
2   SWAP( $A[i], A[j]$ )
3 end
4 if  $j - i + 1 > 2$  then
5    $t \leftarrow \lfloor \frac{j-i+1}{3} \rfloor$ ;
6   STOOGESORT( $A, i, j - t$ );
7   STOOGESORT( $A, i + t, j$ );
8   STOOGESORT( $A, i, j - t$ );
9 end

```

4. [E] Max-Heap Operations:

1. Insert 27 and 9 into the following heap, and then perform a delete-max operation on the resulting heap.

