**Assignment - Week 1 Day 2**

**Question 1 :**

**a) b)**

**Algorithm 1:**

**Algorithm** GetThirdMax(arr):

**Input:** An array ‘arr’ of integers

**Output:** third maximum integer in an array

firstMax ← arr[0] 2

secondMax ← arr[0] 2

thirdMax ← arr[0] 2

n ← arr.length 2

for i ← 0 to n-1 do 2+n

if arr[i] > firstMax then 2n

firstMax ← arr[i] 2n

firstMaxInd ← i n

{increment counter i} 2n

for i ← 0 to n-1 do 2+n

if i = firstMaxInd then n

continue; 1

if arr[i] > secondMax then 2n

secondMax ← arr[i] 2n

secondMaxInd ← i n

{increment counter i} 2n

for i ← 0 to n-1 do 2+n

if i = firstMaxInd or i = secondMax then 2n

continue; 1

if ( arr[i] > thirdMax) then 2n

thirdMax ← arr[i] 2n

{increment counter i} 2n

return thirdMax 1

**Algorithm 2:**

**Algorithm** GetThirdMax(arr):

**Input:** An array ‘arr’ of integers

**Output:** third maximum integer in an array

max ← 0 1

preMax ← 0 1

prePreMax ← 0 1

n ← arr.length 2

for i ← 0 to arr.length-1 do 2+n

if arr[i] >= max then 2n

prePreMax ← preMax n

preMax ← max n

max ← arr[i] 2n

else if arr[i] >= preMax then 2n

prePreMax ← preMax n

preMax ← arr[i] 2n

else if a >= prePreMax then 2n

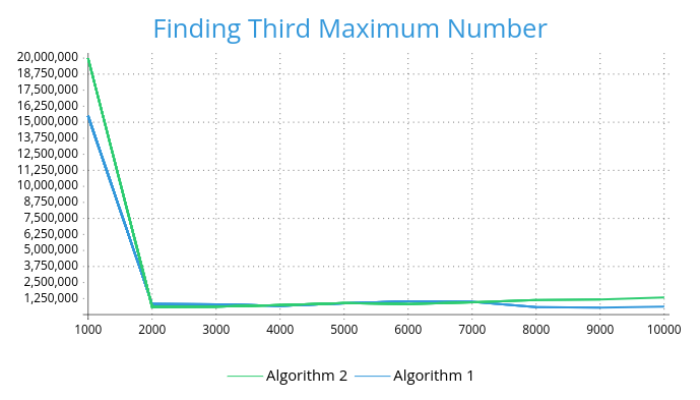
prePreMax ← arr[i] 2n

{increment counter i} 2n

return prePreMax 1

**c) d) Graph :**

Testing array with size of [1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000]



**Question 2 :**

|  |  |
| --- | --- |
| 10,1 | Θ(1) |
| log(log n) | Θ(log(log n)) |
| ln n | Θ(ln n) |
| log n | Θ(log n) |
| log n^n, nlog n | Θ(nlog n) |
| n^(1/2)log n, n^(1/3)log n | O(n^(1/2)log n) |
| n^(1/3), n^(1/2) | O(n^(1/2)) |
| n^(1/k)(k>3) | Θ(n^(1/k)), where(k>3) |
| n^2 | Θ(n^2) |
| n^3 | Θ(n^3) |
| n^k(k>3) | Θ(nk), where(k>3) |
| 2^n, 3^n, n^n, n! | O(n^n) |