**CSE 551 – Assignment 2**

Submitted By:

**Ankit Sharma**

1219472813

ashar263@asu.edu

**Solution for Question (2)**

2

3

4

5

7

* **Give a bound of the form O(f(n)) on the running time of this algorithm**

Outer for-loop (line 1) iterations =

Inner for-loop (line 2) iterates up to =

Number of ‘Add up array entries through ’ operations are up to =

Store the result in operation =

Total operations =

***Therefore, running time of this algorithm on an input of size n is***

* **Show that** **the running time of the algorithm on an input of size is also**

To prove,

To prove above equation, by the definition of - notation, we need to prove that:

where and are positive constants.

We can re-write the above equation as:

We can see that if is a positive fraction less than 1 like 0.01 and , above equation holds true as will be less than

***Hence proved,***

* **Algorithm to solve this problem, with an asymptotically better running time.**

We replace adding all array entries to just adding to which stores sum of all the previous iterations of for a given value.

Since, can be done in , we are only left with two nested for-loop running and (.

***Therefore, running time of this modified algorithm on an input of size n is***

**Solution for Question (3)**

*Given*: Unsorted sequence of numbers

Below is the algorithm to find smallest number in an unordered sequence –

*High level description:*

***min­2***

***min2***

In step 1,  **comparisons** are done in tournament approach as all elements need to be compared. Below is a diagram, explaining tournament approach for finding max element.

Shape, circle

Description automatically generated

In step 2, since the problem is being divided by 2 in every iteration, no of calls =

Therefore, max numbers of elements that can be in

Hence, max numbers of comparisons that can be done in

­

Total comparisons in step 1 and 2 =

*Therefore, above algorithm computes the 2nd smallest number in an unordered (unsorted) sequence of numbers in comparisons in the worst case.*

**Solution for Question (5)**

……………(i)

This recurrence will end when

We are given that , therefore recurrence ends when in eq(i). Substituting value in eq(i) we get,

,

where is given

Since,

Using sum formula for geometric progression, we get:

**Solution for Question (6)**

=

This recurrence will end when

We are given that , therefore recurrence ends when in eq(i). Substituting value in eq(i) we get,

, where is given ………………eq(ii)

Now,

= =

We are given . Substituting value of in above equation we get:

Therefore,

Substituting this value in eq(ii), we get: