

1. What's the time complexity in big O notation for the following recurrence relations:

SHOW YOUR WORK

a. $T(n) = 2T(\sqrt{n}) + \log n$

ANSWER _____

$$O(\log n * \log(\log n))$$

2. Finding a missing number: An array of n elements contains all but one of the integers from 1 to $n + 1$.

- a. Give the best algorithm you can for determining which number is missing if the array is sorted, and analyze its asymptotic worst-case running time.

ANSWER _____

Binary search algorithm is the best with $\theta(\log n)$. We are looking for the smallest index i for which $A[i] = i + 1$; this will be our missing number. If $A[n/2] = n/2 + 1$, i is less than or equal to $n/2$, and we can recurse on the first half of A ; otherwise, it is greater than $n/2$, and we can recurse on the second half of A .