

2) (5 pts) ANL (Algorithm Analysis)

An algorithm to process an array of size n takes $O(n\sqrt{n})$ time. For $n = 640,000$, the algorithm runs in 256 milliseconds. How many *seconds* should the algorithm take to run for an input size of $n = 1,000,000$?

Let the algorithm with input array size n have runtime $T(n) = cn\sqrt{n}$, where c is some constant.

Using the given information, we have:

$$T(640000) = c(640000)\sqrt{640000} = 256ms$$

$$c(640000)(800) = 256ms$$

$$c(512 \times 10^6) = 256ms$$

$$c = \frac{256ms}{512 \times 10^6}$$

$$c = \frac{1ms}{2 \times 10^6}$$

Now, solve for the desired information:

$$T(1000000) = c(1000000)\sqrt{1000000}$$

$$= \frac{1ms}{2 \times 10^6} \times 10^6 \times 10^3$$

$$= \frac{10^3ms}{2} = \frac{1second}{2} = \frac{1}{2}second$$

Grading: 2 pts solving for c , 2 pts for plugging 10^6 and canceling to get to $1000/2$ ms, 1 pt to answer $1/2$ second as the question requests.