

2) (5 pts) ANL (Algorithm Analysis)

A sorting algorithm takes $O(n\sqrt{n})$ time to sort n values. The algorithm took .2 milliseconds to sort an array of 1000 values. How many seconds would it take to sort an array of size 900,000?

Let the run time of the algorithm be $T(n)$. It follows that $T(n) = cn\sqrt{n}$, for some constant c . Use the given information to solve for c :

$$T(1000) = c1000\sqrt{1000} = .2 \text{ ms},$$

$$c = \frac{.2}{1000 \times \sqrt{1000}}$$

Now, let us find $T(900000)$:

$$T(900000) = \frac{.2}{1000 \times \sqrt{1000}} \times (900000) \times \sqrt{900000} = 180\sqrt{900} \text{ ms} = 180 \times 30 \text{ ms} = 5400 \text{ ms}$$

Converting 5400 ms to seconds, we get **5.4 seconds** as the final answer.

Grading: 1 pt to set up equation for c , 1 pt to solve for c , 1 pt to set up final equation, 1 pt to solve for the answer in milliseconds, 1 pt to convert to seconds.