

## 2) (5 pts) ANL (Algorithm Analysis)

An  $O(N^2)$  sorting algorithm took 500ms to sort an array of size 5,000. How many values can the same algorithm sort in 8 seconds?

Let  $T(n)$  represent the run time of the sorting algorithm. Then there exists a value  $c$  such that  $T(n) = cn^2$ .

$$T(5000) = c \times (5000)^2 = 500ms$$

$$c = \frac{500ms}{(5000)^2} \quad \text{Grading 3pts}$$

Let  $n$  be the answer to the query, then we have, remembering to convert 8 seconds to milliseconds:

$$T(n) = \frac{500ms}{(5000)^2} \times n^2 = 8000ms$$

$$n^2 = \frac{8000ms}{500ms} \times (5000)^2$$

$$n^2 = 16 \times (5000)^2$$

$$n^2 = 4^2 \times (5000)^2$$

$$n = 4 \times 5000 = \mathbf{20000}$$

**Grading 2pts**