Do problem 2.8.5 in the textbook.

In this problem, you have to show that any single-string Turing Machine, deciding the language of palindromes, takes time $\Omega(n^2)$.

The language of palindromes is the set:

$$L_{\mathrm{PAL}} = \{x \in \{0,1\}^* | \text{ s.t. } x \text{ reads the same forward as backward } \}$$

for example, $010111010 \in L_{\mathrm{PAL}}$, but 0011 is not.

So, you have to show that if M is any single-string Turing Machine with the property that:

$$M(x) = \text{“yes” if } x \text{ is a palindrome, and “no” otherwise}$$

then M will take $\Omega(n^2)$ steps on any x with $|x| = n$.

This is a very nice example of a *lower bound*; i.e., an example of a problem for which we know that no solution exists which takes less than a certain amount of resources.