

Longest Palindromic Subsequence

A *palindrome* is a sequence of characters which reads the same backward as forward, e.g., “level”, “noon”, “racecar”. Given a sequence A of n characters, your task is to compute the *length* of the **longest palindromic subsequence** of A , i.e., the *length* of the *longest subsequence* of A that is a *palindrome*. For instance, if A is “ETZHEEHU”, “HEEH” is the longest palindromic subsequence of A , and its length is 4.

Grading (16 points):

- An $O(n^2)$ -time implementation gets 16 points, while an $O(2^n)$ -time implementation still gets 6 points.

Note: The input array A is indexed from 0, and all characters are capitals.

Attention:

- You are NOT allowed to use imports explicitly or implicitly other than the imports already in the code template.
- You see the **percentage** of your submission in the CodeExpert system instead of points.