|  |  |
| --- | --- |
| **Project 6** | **CMPT 438 Algorithms** |
|  | **Fall 2016** |

**Longest palindrome subsequence**

A palindrome is a nonempty string over some alphabet that reads the same forward and backward. Examples of palindromes are all strings of length 1, *civic*, *racecar*, and *aibohphobia* (fear of palindromes).

Give an efficient algorithm to find the longest palindrome that is a subsequence of a given input string. For example, given the input *character*, your algorithm should return *carac*.

Implement your algorithm using dynamic programming. Submit a MS word document to Moodle (lms.manhattan.edu) that consists of the followings:

* Recurrence of your algorithm
* Source code in either C++ or Java
* Screenshot of the output
* Running time of your algorithm