

Project 5 Dynamic Programming

Part I: Longest Palindrome Subsequence (LPS)

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. Just print one of them if there are different LPSs with same length.

1)

Input: (will be a file)

1a2bc34def3g2h1ij

Output:

7

1234321

2)

Input:

character

Output:

5

carac

Part II Sequence Alimment

Given two sequences $x[1..m]$ and $y[1..n]$ consist of inserting spaces at arbitrary locations in the two sequences so that the resulting sequences x' and y' have the same length but do not have a space in the same position. Find an optimal alignment of these two sequences to maximize the score. Just print one of them if there are different alignments with same score.

Position j receives a score as follows:

- +1 if $x'[j] == y'[j]$ and neither is a space;
- -1 if $x'[j] != y'[j]$ and neither is a space;
- -2 if either $x'[j]$ or $y'[j]$ is a space.

Input: (will be a file)

GATCGGCAT

CAATGTGAATC

Output: (please output '_' to express SPACE)

G_ATCGGCAT_

CAATGTGAATC

Score: -3

Reference formulation:

$$M[i, j] = \max \begin{cases} M[i-1, j-1] + 1 & \text{match } x_i = y_j \\ M[i-1, j-1] - 1 & \text{skip } x_i \neq y_j \\ \max(M[i-1, j], M[i, j-1]) - 2 & \text{insert space } x_i \text{ or } y_j \end{cases}$$

	X	G	A	T	C	G	G	C	A	T
X	0	-2	-4	-6	-8	-10	-12	-14	-16	-18
C	-2	-1	-3	-5	-5	-7	-9	-11	-13	-15
A	-4	-3	0	-2	-4	-6	-8	-10	-10	-12
A	-6	-5	-2	-1	-3	-5	-7	-9	-9	-11
T	-8	-7	-4	-1	-2	-4	-6	-8	-10	-8
G	-10	-7	-6	-3	-2	-1	-3	-5	-7	-9
T	-12	-9	-8	-5	-4	-3	-2	-4	-6	-6
G	-14	-11	-10	-7	-6	-3	-2	-3	-5	-7
A	-16	-13	-10	-9	-8	-5	-4	-3	-2	-4
A	-18	-15	-12	-11	-10	-7	-6	-5	-2	-3
T	-20	-17	-14	-11	-12	-9	-8	-7	-4	-1
C	-22	-19	-16	-13	-10	-11	-10	-7	-6	-3

Restrictions:

- ◆ This project will be personal, do not copy or modify by the others, or you will get nothing.
- ◆ Use c/c++ only.

Deadline:

- ♦ Due time: 5/19 13:00
- ♦ Demo time: 5/19 16:30-18:30 at EC 324
- ♦ Upload to e3

Please let me know if you have any problem

email: fd3srxs.cs99g@nctu.edu.tw

Lab: EC637

TA Joseph