

CSE3081/AIE3051 (1반): 알고리즘 설계와 분석**[숙제 4] Dynamic Programming 연습**

담당 교수: 임 인 성

2024년 11월 18일

마감: 12월 1일 일요일 오후 8시 정각

제출물, 제출 방법, LATE 처리 방법 등: 조교가 과목 게시판에 공지함. // C or C++

목표: 수업 시간에 배운 dynamic programming 방법에 대한 이해를 심화시키기 위하여, 주어진 문제로부터 optimal substructure를 유도한 후 이를 효과적으로 구현하여 본다.

문제1. 다음과 같은 *Minimal Triangulation* 문제를 해결하여 보자.

Given a set of n vertices for convex polygon, find a triangulation such that ^①no two chords cross each other, and ^②the total length of the selected chords is a minimum.

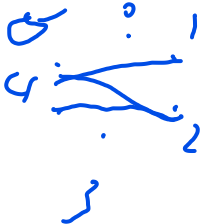
(a) 우선 이름이 MT_command.txt인 ASCII 파일에서 실험에 사용할 입출력 파일의 이름을 읽어 들인 후, 각 다각형에 대하여 문제를 해결하라. ^{→ 4점이 있을 수 있는 테스트 케이스의 파일}

- 이 파일의 첫 줄에는 minimal triangulation을 구할 다각형의 개수가 저장되어 있다. 이후 그 만큼의 각 줄에는 입력 파일의 이름과 출력 파일의 이름이 저장되어 있다.
- 다음은 이 파일의 예를 보여주고 있는데, 입출력 파일 모두 ASCII 파일로 가정한다.

```
3
CP_input00.txt CP_output00.txt
CP_input16.txt CP_output16.txt
CP_input43.txt CP_output43.txt
```

(b) 각 입력 파일에는 다각형 정보가 다음과 같이 저장되어 있는데 (다각형 데이터는 본 숙제와 함께 제공한 ConvexPolygon 프로그램으로 생성할 수 있음),

```
n
x0 y0
x1 y1
⋮
xn-1 yn-1
```



여기에는 다각형 꼭지점의 개수 n 과 i 번째 꼭지점 $v_i = (x_i, y_i)$ ($0 \leq i \leq n - 1$)의 좌표가 순서대로 나열되어 있다. (n 은 200 이하의 정수값으로, 좌표값은 실수값으로 파일에 저장되어 있고, 볼록 다각형 조건을 만족하고 있다고 가정함.)

(c) 프로그램이 수행되면, 각 다각형에 대하여 위의 문제에서 요구하는 방식으로 삼각화 (triangulation)하면서 구한 결과를 출력 파일에 다음과 같이 저장하라.

- 이 결과 파일의 첫 번째 줄에 결과로 산출한 $n - 3$ 개의 chord들의 길이의 총합 total을 소수점 셋째 자리까지의 실수 값으로 출력하라.
- 다음 결과 chord들의 양 꼭지점의 인덱스 (0과 $n - 1$ 사이의 값)를 다음과 같이 한 줄에 한 개씩 정수값으로 출력하라 (총 $n - 3$ 개의 인덱스 쌍이 출력되어야 함).

$$v_i v_j \quad (v_i < v_j)$$

- 채점은 기계적으로 할 예정이므로, 꼭지점의 인덱스는 다음과 같은 조건을 만족하는 순서대로 정렬하여 출력을 해야함: (v_i, v_j) 와 (v_k, v_l) 에 대해 $v_i < v_k$ 이면 전자가 먼저 출력이 되어야 한다. 만약 $v_i = v_k$ 이면 v_j 와 v_l 을 비교하여 값이 작은 쪽을 먼저 출력해야 한다.

(d) 주의: 이 문제를 풀기 위하여 부동 소수점 연산을 수행할 때 double 타입의 변수와 연산을 사용할 것!

2. 다음과 같은 Longest Palindromic Subsequence (LPS) 문제를 생각하자.

“NURSESRUN”처럼 앞에서부터 읽으나 뒤에서부터 읽으나 동일한 문자열 (sequence/string)을 회문 (palindrome)이라고 한다. 어떤 문자열의 부분 문자열 (subsequence)은 원래의 문자열에서 일부 문자만 뽑아 순서를 유지하면서 나열한 문자열으로서, “ABDGI”는 “ABCDEFGHJIJ”의 부분 문자열이다. 한 문자열의 부분 문자열 중 회문이면서 길이가 가장 긴 것을 Longest Palindromic Subsequence (LPS)라 한다. (예를 들어, “BBABCBCAB”의 LPS는 “BABCBAB”임.)

이제 주어진 임의의 문자열의 LPS를 찾아주는 프로그램을 작성하라.

(a) 우선 이름이 LPS_command.txt인 ASCII 파일에서 실험에 사용할 입출력 파일의 이름을 읽어 들인 후, 각 해당 문자열에 대하여 문제를 해결하라.

- 이 파일의 첫 줄에는 풀고자 하는 문제의 개수가 저장되어 있다. 이후 그 만큼의 각 줄에는 입력 파일의 이름과 출력 파일의 이름이 저장되어 있다.
- 다음은 이 파일의 예를 보여주고 있는데, 입출력 파일 모두 ASCII 파일로 가정한다.

```
3
SQ_input00.txt SQ_output00.txt
SQ_input16.txt SQ_output16.txt
SQ_input43.txt SQ_output43.txt
```

(b) 이 문제의 입력 파일과 그에 대한 출력 파일의 형식은 다음과 같다.

- 입력 파일의 첫 줄에는 LPS를 구하고자 하는 문자열의 개수가 주어지 있으며, 다음 그 개수만큼 중복하여 문자열의 길이와 문자열이 한 줄에 한 개씩 주어지 있다. (이 문제의 입력 문자열은 영어 대소문자로 구성되어 있으며, 대문자와 소문자를 구별할 것.)

```
2
9
```

BBABCBCAB

6

XxyZxa

- 출력 파일의 첫 재줄에는 입력 파일과 동일한 문제의 개수가 주어져 있고, 다음 그 개수만큼 자신이 찾은 LPS의 길이와 해당 문자열이 한 줄에 한 개씩 저장되어야 한다. (아래 예로부터 그 내용은 명확하게 알 수 있으며, 기계적으로 채점을 할 예정이므로 출력 형식에 주의를 기할 것.)

2

7

BABCBAB

3

xyx

주의 사항

- 이번 숙제는 이름이 HW_4인 디렉터리에 첫 번째 문제는 이름이 Problem_1, 그리고 두 번째 문제는 이름이 Problem_2인 디렉터리를 만들어 구현한 후 HW_4 디렉토리를 zip으로 묶어 제출하라. 조교는 이 파일을 풀어 Visual Studio를 실행시켜 채점을 할 예정임 (이때 .vs와 x64와 같은 파일들은 제거한 후 압축할 것).
- 그 외의 숙제 제출 방법에 관하여 조교가 사이버 캠퍼스에 공지하는 내용을 반드시 숙지할 것.
- 제출한 원시 코드에서 대해서는 copy-check를 수행할 예정이며, 다른 사람의 코드 또는 보고서를 복사할 경우 관련된 사람 모두에 대하여 (즉 복사한 사람과 복사 당한 사람 모두) 과목 최종 성적의 50%를 감점함.