CSE3081 (2반): 알고리즘 설계와 분석
<숙제 3 >

담당 교수: 임 인 성

2018년 11월 13일

마감: 12월 1일 토요일 오후 8시 정각
제출물, 제출 방법, LATE 처리 방법 등: 조교가 과목 게시판에 공고할 예정임.

목표: (1) 알고리즘 설계 기법 중의 하나인 dynamic programming 방법에 대한 이해도를 높이도록 한다. (2) 주어진 문제로부터 optimal substructure를 유추하고, 이를 테이블을 사용하여 계산하는 과정에 대하여 연습하여 본다.

1. 다음 글을 읽은 후 아래에 기술하는 gapped alignment 문제를 풀어보자 (출처: Wikipedia).

   > In bioinformatics, a sequence alignment is a way of arranging the sequences of DNA, RNA, or protein to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences. Aligned sequences of nucleotide or amino acid residues are typically represented as rows within a matrix. ***Gaps are inserted between the residues so that identical or similar characters are aligned in successive columns***.

   두 개의 문자열 (string)에 대해 임의로 indel이라고 하는 gap –을 적절히 삽입할 수 있으나, 그러한 삽입을 가급적 방지하기 위하여 매번 -p점 만큼의 감점을 부과한다. 만약 indel이 아닌 두 대응되는 문자가 일치하면 s점 만큼의 점수를, 아닐 경우 -f점 만큼의 감점을 부과한다. 만약 $s = 2$, $f = 1$, $p = 2$라고 가정할 경우, `X = ATCGGATCT`와 `Y = ACGGACT`에 대해 `X = ATCGGAT-CT`와 `Y = A-C-GG-ACT`와 같이 gap이 삽입되었다면 전체 유사성 점수는 1점, 그리고 `X = ATCGGATCT`와 `Y = A-CGG-ACT`와 같이 gap이 삽입되었다면 전체 유사성 점수는 7점이 된다.

   이제 임의로 주어진 문자열 $X = x_1x_2 \cdots x_m$과 $Y = y_1y_2 \cdots y_n$에 대하여, $O(mn)$ 시간 복잡도를 가지는 dynamic programming 알고리즘을 사용하여, 전체 유사성 점수를 최대로 해주는 gap 삽입 방법을 출력해 주는 프로그램을 작성하여 보자. 여러분의 프로그램은 다음과 같은 입출력 요구사항을 만족해야 한다.

   입력 형식
   프로그램이 수행되면 이름이 `input.txt`인 텍스트 파일에서 다음과 같은 형식으로 저장되어 있는 정보를 읽어 들여야 한다.

   ```
   twostrings.bin
   s f p
   ```

   이 파일의 첫 번째 줄에는 두 문자열 데이터를 저장하고 있는 텍스트 파일의 이름이 저장되어 있다. 이 파일에는 이진 형식 (binary format)으로 데이터가 저장되어 있는데, 첫 4 바이트에는 X의 길이 m, 그리고 다음 4 바이트에는 Y의 길이 n이 각각 `int` 타입으로 저장이 되어 있고, 다음 m 바이트에는 X가, 그리고 그 다음 n 바이트에는 Y가 `char` 타입으로 저장이 되어 있다. 다음 세 개의 양의 정수 `s`, `f`, `p`가 (의미는 분명) `int` 타입으로 저장되어 있다.

출력 형식
프로그램 수행 후 이름이 `output.txt`인 텍스트 파일에 다음과 같은 방식으로 계산 결과를 출력하라.

```
1
10
1
8
3
2
4
7
```

여기서 첫 줄에는 자신이 구한 최대 전체 유사성 점수, 두 번째 줄에는 gap을 포함한 전체 문자열의 길이가 저장이 되어야 한다. 다음 X에 삽입된 gap의 개수와 그 개수 만큼 gap이 삽입된 위치를 저장되어야 하며, 그 다음 마찬가지로 Y에 대한 gap 정보가 저장되어야 한다.

2. 임의의 번호를 가지는 카드 묶음을 고려하자. 이 문제에서는 동일한 번호를 가지는 카드가 여러 장 있을 수 있는데, 카드를 섞는 전형적인 방식은 카드 묶음을 둘로 나누어 한 손에 한 묶음씩 잡고 엄지와 검지에 힘을 주어 카드 허리를 휘게 한 다음, 양쪽의 카드를 조금씩 풀어 주면 '섞인 카드 묶음'을 얻게 된다. 이렇게 카드를 섞으면 잘 섞인 것 같지만, 사실 처음 양손에 가지고 있던 카드 묶음에서 카드의 순서는 섞인 카드 묶음에서도 그대로 유지된다. 예를 들어, 왼손에 `2, 4, 6, 8`의 순서로 카드를 가지고 있고, 오른손에 `1, 3, 5, 7`의 카드를 가지고 있다면, 위에서 설명한 방식으로 카드를 섞어서 얻을 수 있는 카드의 순서의 몇 가지 예를 들면 다음과 같다.

```
1, 3, 2, 4, 6, 5, 7, 8
1, 2, 3, 4, 5, 6, 7, 8
1, 3, 5, 7, 2, 4, 6, 8
```

그렇지만 아래와 같은 순서는 절대 얻을 수 없는데, 그 이유는 처음 것은 `3`과 `5`의 순서가 섞기 전과 섞은 후에 다르고, 두 번째 것은 `1`과 `3`의 순서가 다르기 때문이다.

```
1, 2, 4, 5, 6, 8, 3, 7
3, 1, 2, 4, 6, 5, 7, 8
```

이제 왼손에 있는 묶음의 카드 순서, 오른손 묶음의 카드 순서, 그리고 섞은 후 카드 묶음의 카드 순서가 입력될 때, dynamic programming 기법을 사용하여 양손의 카드 묶음을 위에서 설명한 것과 같은 방식으로 섞어서 만들 수 있는지 아닌지를 판별하는 프로그램을 작성해보자 (이 문제에서는 카드 위치가 서로 다르지만, 번호가 동일한 카드가 제한 없이 입력될 수 있음). 여러분의 프로그램은 다음과 같은 입출력 요구사항을 만족해야 한다.

입력 형식
입력 데이터는 이름이 `input.txt`인 텍스트 파일에 저장되어 있다. 첫 째줄에는 왼손에 들고 있는 카드의 개수에 이어서 카드 순서가 입력되고, 두 번째 줄에는 오른손에 들고 있는 카드 묶음, 그리고 셋 째줄과 넷 째줄에는 한 줄에 하나씩 섞인 카드 묶음에 대한 정보가 같은 형식으로 입력된다. 왼손이나 오른손에 들고 있는 카드 개수는 1 이상 1,000장 이하이고, 카드 번호는 1부터 1,000 이하인 정수이다.

출력 형식
계산 결과는 이름이 `output.txt`인 텍스트 파일에 저장되어야 한다. 한 줄에 왼손과 오른손에 들고 있는 카드 묶음을 위에서 정한 방식으로 섞어서 셋째 줄의 카드 순서를 얻을 수 있으면 1 아니면 0을 출력하고, 이어서 넷째 줄의 카드 순서를 얻을 수 있으면 1 아니면 0을 출력한다. 출력은 00, 01, 10, 11 중의 하나이다.

입출력 예 1

입력 (input.txt)
```
4 2 4 6 8
4 1 3 5 7
4 1 3 2 4 6 5 7 8
8 1 2 4 5 6 8 3 7
```

출력 (output.txt)
```
10
```

입출력 예 2

입력 (input.txt)
```
3 5 5 2
4 2 1 2 7
4 3 2 5 2 5 1 2 7
3 5 5 2 1 2 5 2 7
```

출력 (output.txt)
```
01
```

- **[주의]**

1. 각 문제에 대한 구현 여부를 `HW1_S20179999.{hwp,doc,docx,txt}`와 같은 이름의 보고서에 기술하고, 특히 2번 문제의 경우 어떠한 방식으로 dynamic programming 기법을 적용하였는지 설명하라.
2. 조교는 자신의 명령어 파일과 입력 데이터를 사용하여 여러분의 프로그램이 정확한 값을 계산하는지 확인할 예정이니, 자신이 생성한 데이터 (필요 시 적절히 압축하여)와 원시 코드를 조교가 수행하기 편리한 형태로 제출할 것.
3. 숙제 제출 기간 동안 조교가 숙제와 관련하여 중요한 공지 사항을 게시판에 올릴 수 있으니 항상 수업 게시판을 확인하기 바람.
4. 제출 화일에서 바이러스 발견 시 **본인 점수 X (-1)**이고, 다른 사람의 숙제를 복사할 경우 **관련된 사람 모두에 대하여 만점 X (-10)**임.