

ICE3020 알고리즘설계 과제 2

과제 2 보고서

보고서 작성 서약서

1. 나는 타학생의 보고서를 베끼거나 여러 보고서의 내용을 짜집기하지 않겠습니다.

2. 나는 보고서의 주요 내용을 인터넷사이트 등을 통해 얻지 않겠습니다.

3. 나는 보고서의 내용을 조작하지 않겠습니다.

4. 나는 보고서 작성에 참고한 문헌의 출처를 밝히겠습니다.

5. 나는 나의 보고서를 제출 전에 타학생에게 보여주지 않겠습니다.

나는 보고서 작성시 윤리에 어긋난 행동을 하지 않고 정보통신공학인으로서 나의 명예를 지킬 것을 맹세합니다.

2021년 5월 22일

학부 정보통신공학과

학년 3학년

성명 유지훈

학번 12171810



(1)번

-코드-

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

다음과 같이 코드를 구성했다. 강의 자료에 나와있는 모습처럼 source와 target을 활용한 table 구조를 만들었다. 우선적으로 source와 target의 길이 + 1 만큼의 2차원 array를 동적으로 생성을 해주었고 맨 위의 라인과 맨 왼쪽 라인에 값을 순차적으로 입력을 해주었다. 이후 [1][1] 칸부터 시작해서 왼쪽, 위, 대각선 방향의 값을 계산을 해서 그 중 가장 작은 값을 해당 칸에 부여하는 방식으로 모든 테이블을 채웠다. 이후 테이블이 정확하게 구성 되었는지를 확인하기 위해서 전체 테이블을 출력해주고 맨 마지막 값을 Levensthtein Distance로 출력을 해주었다.

-실행결과-

텍스트이(가) 표시된 사진

자동 생성된 설명 화살이(가) 표시된 사진

자동 생성된 설명

다음과 같이 table과 Levenshtein Distance값이 출력이 되는 모습을 볼 수 있다. (밑에 나와있는 경로와 그에 따른 값은 (2)번 문제를 해결하면서 코드를 수정하면서 생겼기에 무시해도 좋다.) 1번의 경우에는 강의노트에도 나와있던 부분이라 table 값이 정확하게 맞았다는 것을 확인 할 수 있었고, cost를 변경해준 상황 또한 실제 계산과 같았기에 코드 구성이 잘 맞았다는 것을 확인 할 수 있었다.

(2)번

-코드-

텍스트, 녹색, 은색, 스크린샷이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

다음과 같이 코드를 추가 구성했다. 처음 경로 출력을 생각했을 때는 [0][0] 에서부터 시작해서 내려가는 구조를 생각했었다. 그러나 이렇게 할 경우 특정 Data에서 문제가 발생해 원하지 않는 경로로 내려가는 경우가 생겼다. 이러한 경험이 바탕이 되어 역으로 경로를 추적하는 방법을 생각하게 되었다.

역순의 경로를 저장하는 array를 따로 생성을 하고 어떠한 칸 까지 사용했는지를 알려주는 index 변수를 생성하고 이를 마지막에 for문에서 뒤에서부터 출력을 하면 될 것이라는 생각이 있었다. 그래서 초기 구조에서는 79번 라인부터 82번 라인까지의 4가지 경우로 비교를 해주게 만들었다. 여기서의 4가지 경우는 왼쪽, 위, 대각선(2가지경우) 를 말하고, 현 위치의 값에서 각 행동 코스트를 뺀 값과 그 해당 칸과의 값이 같은 경우를 if문을 통해서 확인을 하고 해당 위치를 변수에 담았다. switch 문 혹은 if-else 구조를 채택할 수 있었으나 그런 경우에는 어느 하나의 경로에 큰 영향력을 주기 때문에 애매하다 생각을 했다. 그러나 생각해보니 if를 쓰더라도 맨 마지막 경우에 영향력이 커진다는 것을 알아냈고 이를 통해서 보안책을 마련 해야겠다는 생각이 들었다. 그래서 마련한 장치는 위 혹은 왼쪽의 값이 대각선에 있는 값보다 작은 경우에는 해당 위치를 가는 경로 값을 담는 것이다. 이렇게 처리하면서 대각선으로 움직이는 것에 너무 큰 영향력이 있는 것을 방지했다.

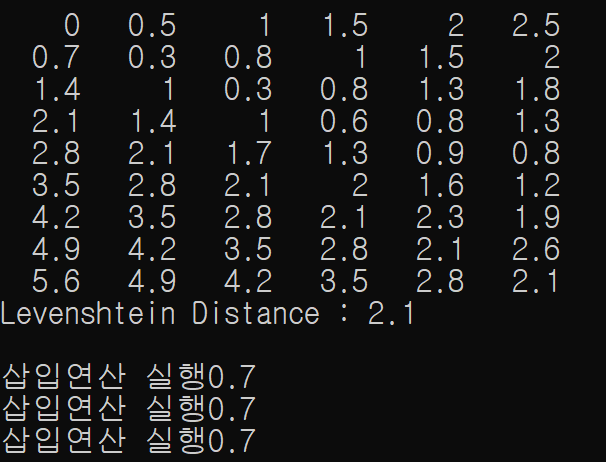
이렇게 처리를 하고 코드를 돌려보니 어느 한 구간에서 무한 루프를 도는 것을 확인했다. 이유를 확인해본 결과 float, double 관련 값 비교에서 오차로 인해 같은 값으로 인지하지 못하는 문제였다. 이를 해결하기위해 6번 7번 라인을 추가하고 79~82번 라인과 88번 91번 라인을 수정했다.(이전에는 해당 라인들이 단순 == 비교로 되어 있었다.) 이후 결과를 확인해보니 원하는 결과 경로를 잘 표현하는 것을 볼 수 있었다.

경로 출력 관련 코드는 139~143번 라인에서 볼 수 있고, 단순하게 카운트 된 가장 마지막 위치부터 0번 위치까지 경로 array 값을 읽고 이에 해당하는 경로와 값을 출력해준다.

-실행결과-

화살이(가) 표시된 사진

자동 생성된 설명



텍스트이(가) 표시된 사진

자동 생성된 설명

3가지 데이터에 대해서 순서대로 다음과 같은 결과를 얻었다. 이 경로가 맞게 출력이 된 것인지를 해석적으로 확인해보았다.

1번 데이터 : Source = Levenshtein, Target = Meilenstein

1. L과 M에서 교환연산이 이루어진다.
2. v와 i에서 교환연산이 이루어진다.
3. 바뀐 i와 e 사이에 l의 삽입연산이 이루어진다.
4. h의 삭제연산이 이루어진다.

2번 데이터 : Source = TACTG, Target = CATGACTG

1. C의 삽입연산이 이루어진다.
2. A의 삽입연산이 이루어진다.
3. T와 A사이에서 G의 삽입연산이 이루어진다.

3번 데이터 : Source = 데이타마이닝, Target = 데이터베이스설계.

1. ‘타’와 ‘터’의 교환연산이 이루어진다.
2. ‘마’와 ‘베’의 교환연산이 이루어진다.
3. ‘닝’과 ‘스’의 교환연산이 이루어진다.
4. ‘설’의 삽입연산이 이루어진다.
5. ‘계’의 삽입연산이 이루어진다.

다음과 같이 출력 데이터와 경로가 순차적으로 잘 표현 되는 것을 볼 수 있었다. 테이블 상에서 다음과 같은 경로를 따라서 이동한 것이 잘 표현이 되었고 각 코스트와 순서가 잘 출력이 되었다.