



ICE3020 알고리즘설계 프로젝트

설계 프로젝트

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 6월 20일

학부 정보통신공학과

학년 3학년

성명 유지훈

학번 12171810

<1번> NP-Complete문제 중 TSP문제를 Dynamic Programming기법과 유전자 알고리즘

(Genetic Algorithm)을 이용하여 구현한 후 성능(수행시간)을 비교분석하라. (100점)

-코드-

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

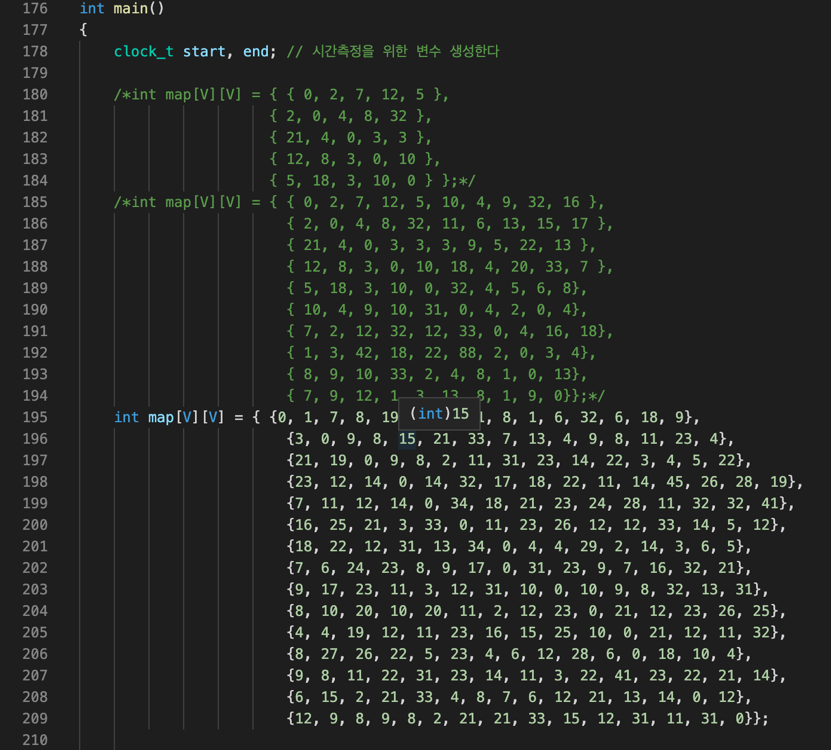
자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명



텍스트이(가) 표시된 사진

자동 생성된 설명

다음과 같이 코드를 구성했습니다.

DP의 상황과 GA의 상황을 하나에 담아서 작성했습니다.

DP의 경우에는 시작 위치를 지정해주고 이후 visited를 이용해 해당 노드를 방문했는지 아닌지를 판별하고 그에 따라서 반복이 일어나는 구조로 만들었습니다. 우선 시작 노드부터 시작해서 카운트(도시의 개수)가 다 안 찼을 경우에는 다음 노드들을 살펴보고 방문을 안했을 경우에는 비교에 들어갑니다. 비교를 넣지 않아도 괜찮으나 시간적으로 이득을 보기 위해서 다음 노드를 더해서 얻어지는 값이 이미 구한 최소 값 보다 큰 경우에는 해당 경우를 스킵하고 바로 다음 노드의 결과를 다시 tspDP를 재귀적으로 보내서 확인하는 구조로 만들었습니다. 이렇게 시작 지점부터 계속 노드를 방문하면서 카운트가 다 찼을 경우에는 그때의 최소값을 리턴해줍니다.

GA의 경우에는 아래와 같은 구성을 했습니다.

encoding : 하나의 structure를 만들고 경로를 string으로, 해당 경로의 총 거리를 int값으로 처리했습니다. 시작 지점과 도착 지점은 0으로 설정을 해두고 유전자를 사이에 있는 노드들로 랜덤하게 생성을 할 수 있도록 했습니다.

유전자 선택 및 변형 : 유전자 풀에 있는 유전자들을 모두 거리를 확인을 해보고 지금 유전자 풀에 존재하는 모든 유전자들을 변형시켜봅니다. 이때 현형은 기존에 있던 경로에서 시작과 끝 사이에 있는 값들을 서로 랜덤하게 변경하는 방법을 사용했습니다. 다른 유전자와 경로를 바꾸는 방법도 있었지만, 이런 경우에는 해당 경로가 올바른 경로인지 확인을 해봐야 하고 이때 맞지 않으면 다시 변형된 경로를 생성해야 될 것이라 판단이 들었기 때문에 다음과 같은 방법을 사용했습니다. 이렇게 생성된 변이 유전자와 기존 유전자를 비교해서 더 좋은 경로 값을 가지는 경우에는 기존 유전자와 교환을 해줬습니다. 그러나 이렇게 진행을 시키면 진짜 최소 값이 아닌 다른 위치의 최소값을 찾을 수 있기 때문에 적절한 비율을 설정해서 새로 만들어진 유전자가 더 안좋다고 하더라도 새로운 유전자 풀에 넣어주는 방식을 사용했습니다.

세대를 얼마나 만들지, 그리고 얼마만큼의 유전자 개수를 만들지를 설정을 해야 했었는데, 임의적으로 최대 30번의 세대, 유전자는 10개를 생성하고 진행을 하도록 만들었습니다. 이 숫자는 더 증가시킬 수 있었으나 더 많은 경우를 생성하더라도 엄청 큰 차이가 없었기 때문에 다음과 같이 진행을 했습니다.

그래프의 경우에는 3가지 경우를 진행했습니다. 노드가 5개, 10개, 15개 인 경우를 임의적인 데이터 셋을 만들었습니다. 이렇게 적은 수를 진행한 이유는 DP로 최소값을 찾을 경우 20개가 넘어가는 인풋을 주었을 경우에 O(2^N\*N^2)의 시간복잡도를 가지고 있어서 기하급수적으로 증가했기 때문입니다. 또한 결과를 확인해봐도 3가지 경우에 시간 차이가 ns 기준으로 극명하게 발생했기 때문에 더 큰 데이터 셋을 사용하지 않았습니다.

-결과-

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

<인풋 노드가 5개일 경우 두 알고리즘의 비교>

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

<인풋 노드가 10개일 경우 두 알고리즘의 비교>

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

<인풋 노드가 15개일 경우 두 알고리즘의 비교>

출력 결과는 다음과 같았습니다. GA의 경우에는 30번의 과정 중 가장 작은 값을 가지는 상황을 선정해서 해당 세대까지 진행을 하도록 했습니다.

GA의 경우 세대 수와 유전자들을 입력 개수가 늘어나는 것에 비례하게 증가시켜주거나 그렇게 진행하지 않았기 때문에 나름의 최의 값을 얻기까지의 시간이 큰 폭으로 늘어나지는 않았습니다. 대신 입력의 수에 따라서 변이와 생성이 N에 비례하기 때문에 그에 따른 시간 증가가 있는 모습을 볼 수 있었습니다.

DP의 경우에는 O(2^N\*N^2)이기 때문에 단순하게 값을 3배 증가시킨 것인데도 큰 폭으로 증가한 모습을 볼 수 있습니다. 5개의 입력에서 4ns, 10개의 입력에서 대략 600ns, 15개의 입력에서 대략 900000ns 가 걸리는 모습을 볼 수 있었습니다.

이러한 결과를 비교해보면 적은 수의 노드를 tsp 하는 경우에는 정확한 답을 확실하게 얻을 수 있다는 장점으로 DP를 사용하면 되겠지만, 많은 수의 노드의 경우에는 tsp시 기하급수적인 시간이 걸리기 때문에 대략적인 최적 값을 찾기 위해서 GA를 사용하는 방법이 더 좋다고 판단이 됩니다.

<2번> 최단경로 문제를 Dijkstra알고리즘과 유전자 알고리즘(Genetic Algorithm)을 이용하 여 구현한 후 성능(수행시간)을 비교분석하라. (200점)