

# 알고리즘 실습 보고서

- Dijkstra -

전공 : 컴퓨터공학과

분반 : 05반

학번 : 201502085

이름 : 이규봉

## 1. 실행 환경

과제의 코드를 테스트한 환경은 아래와 같습니다. 따로 사용한 라이브러리는 없습니다.

- Windows 10
- Pycharm 2019.02
- Python 3.7.4

## 2. 과제 설명

- 다익스트라 알고리즘 구현

## 3. 문제 해결 방법

- 문제를 해결하기 위해 자신이 사용한 방법, 아이디어에 대한 설명  
(구현한 코드에 대한 설명 등)

제가 기존에 C++로 짰던 코드를 약간 변경하여 파이썬 코드로 옮겨 작성하였으며, ppt에 나온 형식대로 값을 출력하도록 하였습니다.

ppt에 명시된 대로 입력 값으로 배열과 시작점을 주는 방식이 아닌 코드에 map과 시작점을 직접 주는 방식으로 코딩했습니다.

ppt에 나온 코드와 다른 점으로, ppt에선 갈 수 없는 경로를 나타내는 값으로, inf 값을 주었던 것을 전 가독성을 위하여 -1로 표기하고 사용했습니다.

또한 값의 업데이트 되는 과정을 보여주기 위해, 이전의 dist (d 배열) 값을 prev\_d로 저장하여 사용했습니다.

그 외의 점은 ppt에 작성된 의사코드와 동일합니다. (각 정점들을 돌며 최단 거리를 갱신하고 갱신된 값들을 콘솔에 출력.)

## 4. 결과 화면

```
C:\Users\jopem\Anaconda3\python.exe C:/Algorithm-Hw/Assign10/main.py
다익스트라 알고리즘으로 계산한 결과는 다음과 같습니다.
```

```
-----
S[1]:    d[A] = 0
-----
```

```
Q[0] :    d[A] = 0
Q[1] :    d[B] = -1 => d[B] = 10
Q[2] :    d[C] = -1 => d[C] = 3
-----
```

```
-----
S[2]:    d[C] = 3
-----
```

```
Q[1] :    d[B] = 10 => d[B] = 7
Q[2] :    d[C] = 3
Q[3] :    d[D] = -1 => d[D] = 11
Q[4] :    d[E] = -1 => d[E] = 5
-----
```

```
-----
S[3]:    d[E] = 5
-----
```

```
Q[3] :    d[D] = 11 => d[D] = 14
Q[4] :    d[E] = 5
-----
```

```
-----
S[4]:    d[B] = 7
-----
```

```
Q[0] :    d[A] = 0
Q[1] :    d[B] = 7
Q[2] :    d[C] = 3
Q[3] :    d[D] = 14 => d[D] = 9
-----
```

```
-----
S[5]:    d[D] = 9
-----
```

```
Q[3] :    d[D] = 9
Q[4] :    d[E] = 5
-----
```

```
-----
최종 결과:
[0, 7, 3, 9, 5]
-----
```

```
Process finished with exit code 0
```

ppt에 답으로 나온 것과 같이, 시작점 A에서 각 정점으로의 최단 경로가 정확하게 구해진 것을 확인할 수 있었습니다.

## 5. 느낀점 및 고찰

파이썬으로 다익스트라 알고리즘을 구현해 볼 수 있어 좋았습니다.