

Technical Report

1829008 김민영

자료구조 과제8

HW8-1 : Implement the 'decrease-key' and 'increase_key'

1. basic ideas and how you applied it into your code, analysis of your code(complexity)

Lecture8 Priority Queue 강의자료를 참고하여 코드를 구현하였다. (min heap 의 삽입연산부분)

먼저, minheap 과 관련한 init, insert_min_heap, build_min_heap 등의 기본 함수들을 불러왔다. 이후 Decrease_key_min_heap 과 Increase_key_min_heap 을 구현하였다. 각 함수 둘다 예외처리를 한 뒤에, 반복문을 이용해 구현하였다. 특히 Increase_key_min_heap 은 parent 와 child 의 관계를 이용했다.

2. result image

```
C:\WINDOWS\system32\cmd.exe
Min heap : 1 4 2 7 5 3 3 7 8 9
Decrease Min heap : 1 3 2 4 5 3 3 7 8 9
Increase Min heap : 1 3 3 4 5 10 3 7 8 9
계속하려면 아무 키나 누르십시오 . . .
```

HW8-2 : Revise Prim algorithm code

1. basic ideas and how you applied it into your code, analysis of your code(complexity)

기존 강의자료에서는 Prim 알고리즘을 unsorted array dist 를 사용해 구현하였다.

본 과제에서는 Prim 알고리즘을 unsorted array dist 를 사용하지 않고 min heap 을 이용해 구현하였다. 이때 build_min_heap, delete_min_heap, Decrease_key_min_heap 을 이용하였다.

첨부된 코드 안에 부모자식관계가 명시되어있다.

2. result image

```
C:\WINDOWS\system32\cmd.exe
Vertex 0 -> 1 edge : 3
Vertex 1 -> 2 edge : 8
Vertex 2 -> 3 edge : 15
Vertex 2 -> 4 edge : 2
Vertex 4 -> 5 edge : 9
Vertex 4 -> 6 edge : 4
Vertex 4 -> 7 edge : 5
계속하려면 아무 키나 누르십시오 . . .
```

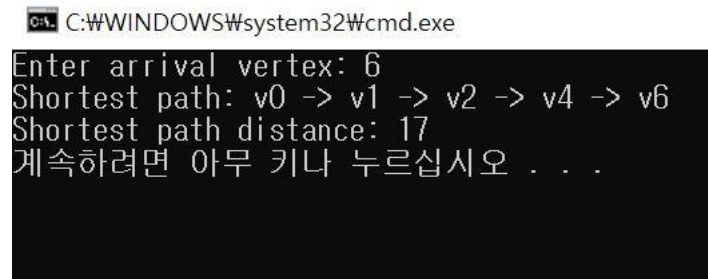
HW8-3 : Revise Dijkstra algorithm code

1. basic ideas and how you applied it into your code, analysis of your code(complexity)

기존 강의자료에서는 다익스트라 알고리즘을 unsorted array dist 를 사용해 구현하였다.

본 과제에서는 다익스트라 알고리즘을 unsorted array dist 를 사용하지 않고 min heap 을 이용해 구현하였다. 이때 build_min_heap, delete_min_heap, Decrease_key_min_heap 을 이용하였다. 첨부된 코드 안에 부모 자식관계가 명시되어있다.

2. result image



```
C:\WINDOWS\system32\cmd.exe
Enter arrival vertex: 6
Shortest path: v0 -> v1 -> v2 -> v4 -> v6
Shortest path distance: 17
계속하려면 아무 키나 누르십시오 . . .
```