

# 알고리즘 설계과제 2

## [산책로 디자인 비용계산 프로그램 설계]

### 1. 과제 개요

김인하는 인하공원의 디자인 테마 관리를 담당하고 있다. 크리스마스를 맞이하여 공원에 있는 조형물 사이의 산책로에 크리스마스 전구를 설치하여 디자인하려 한다. 한편, 예산이 한정되어 있어, 모든 산책로에 크리스마스 전구를 설치할 수는 없는 상황이다. 따라서 공원 관람객이 모든 조형물을 관람할 수 있도록, Prim 알고리즘을 이용하여 모든 조형물을 연결하되 최소 비용으로 산책로들에 크리스마스 전구를 설치하고자 한다. 아래의 정보를 토대로 주어진 조건을 만족하는 프로그램을 구현하시오. 단, Prim 알고리즘에서 우선순위 큐(priority queue)는 STL을 사용하지 말고 정렬되지 않은 시퀀스(unsorted sequence) 또는 힙(heap)으로 직접 구현할 것. (STL로 구현 시, 과제 0점)

### 2. 정보 형식

#### (1) 조형물 정보

① 조형물 번호 (기준키, 유일함):  $1 \sim n$  ( $2 \leq n \leq 10,000$ ) 사이의 정수.

#### (2) 두 조형물 간 산책로 정보

- ① 조형물1 번호: 산책로 한쪽 편에 연결된 조형물의 번호  
② 조형물2 번호: 산책로 반대편에 연결된 조형물의 번호  
③ 산책로 디자인 비용:  $10 \sim 10,000$  사이의 정수

### 3. 수행할 기능

#### (1) 데이터 입력받기

프로그램을 실행하면 가중그래프(weighted graph) 정보를 표준입력으로 입력받아, 인접리스트 기반(adjacency list representation) 그래프 자료구조를 생성하여 관리한다. 이때 가중그래프는 무향 연결그래프(connected, undirected graph)로 주어진다.

첫 번째 줄에는 조형물 정보의 수  $n$  ( $2 \leq n \leq 10,000$ )과 조형물 사이의 산책로 정보의 수  $m$  ( $1 \leq m \leq 100,000$ ), 질의의 수  $q$  ( $1 \leq q \leq 5$ )가 공백으로 구분되어 주어진다.

이후  $m$ 개의 줄을 통해, 조형물 사이의 산책로 정보가 아래와 같이 한 줄로 주어지며, 각 필드는 공백으로 구분된다. 두 조형물의 번호는 서로 다르며, 주어진 조형물 정보 내에서 주어진다.

필드 ①	필드 ②	필드 ③
조형물1 번호	조형물2 번호	산책로 디자인 비용

## (2) 산책로 디자인 총비용 및 Prim 알고리즘 진행과정 출력

- 질의형식: “P S”

P: 설계 과정 출력 질의를 나타내는 기호

S: 출발 조형물 번호

- 출력형식: “W  $V_1$   $V_2$  ...  $V_n$ ”

W: 산책로 설치에 들어가는 비용의 총합

$V_1$ : 출발 조형물 번호

$V_i$ :  $i$  ( $2 \leq i \leq n$ )번째로 트리 정점에 추가된 조형물 번호

- 설명: 강의 시간에 배운 바와 같이, 출발 조형물 번호를 시작 정점으로 Prim 알고리즘을 진행하여 트리 정점(tree vertex)이 추가될 때마다 해당 조형물 번호를 출력한다. 단, 시작 정점은 그래프에 존재하는 정점으로 주어진다. 또한 주변 정점(fringe vertex)으로 가는 간선들 중 최소 가중치의 간선이 여러 개 존재할 경우, 건너편 정점 번호가 더 작은 것을 먼저 처리한다.

## 4. 입출력 제한사항

- (1) 프로그램의 한 입력에 대해 정점 정보는 최대  $10^4$ 개, 간선 정보는 최대  $10^5$ 개, 질의는 최대 5개가 입력된다.
- (2) 채점서버의 입력들에 대해 총 4초의 제한시간 이내에 수행되어야 한다.
- (3) 제시한 입출력 형식대로 표준입출력을 사용하여 처리한다.
- (4) 문제에서 설명되지 않은 예외 처리를 해야 할 질의는 입력되지 않는다.

## 5. 프로그램 입출력 예

(1) 표준입출력 예시

파란색: 프로그램 표준입력 내용

빨간색: 프로그램 표준출력 내용

=====

9 14 5

1 2 40

1 8 80

2 3 80

2 8 110

8 9 70

3 9 20

8 7 10

7 9 60

3 6 40

7 6 20

6 4 140

4 5 90

5 6 100

3 4 70

P 1

370 1 2 3 9 6 7 8 4 5

P 2

370 2 1 3 9 6 7 8 4 5

P 3

370 3 9 6 7 8 4 1 2 5

P 4

370 4 3 9 6 7 8 1 2 5

P 5

370 5 4 3 9 6 7 8 1 2

(프로그램 종료)

(2) 표준입출력 예시에서의 각 입/출력 내용 (구현을 검증할 때, 활용할 것)

Input
9 14 5
1 2 40
1 8 80
2 3 80
2 8 110
8 9 70
3 9 20
8 7 10
7 9 60
3 6 40
7 6 20
6 4 140
4 5 90
5 6 100
3 4 70
P 1
P 2
P 3
P 4
P 5

Output
370 1 2 3 9 6 7 8 4 5
370 2 1 3 9 6 7 8 4 5
370 3 9 6 7 8 4 1 2 5
370 4 3 9 6 7 8 1 2 5
370 5 4 3 9 6 7 8 1 2

## 6. 주의 사항 (지키지 않으면, 0점 또는 감점)

### (1) 개발언어 및 채점서버 환경

- ① 허용된 개발언어: C, C++
- ② OS: Ubuntu 18.04 (64-bit)
- ③ gcc 버전: gcc (Ubuntu 7.3.0-16ubuntu3) 7.3.0 (c++14 지원)

### (2) 제출 파일

- ① 보고서 (2차 과제에서는 과제의 부담을 줄여주고자, 보고서 없음. 단, 소스코드 주석에 알고리즘을 자세히 설명할 것)
    - (a) 파일형식: 아래아한글 문서(.hwp), MS Word 문서(.doc, .docx), PDF 문서(.pdf)
    - (b) 양식: 첨부된 파일 참조
    - (c) 파일명: "학번\_이름"으로 통일 (예: 12059876\_홍길동)
  - ② 소스코드 (채점서버에 제출)
    - (a) 하나의 소스코드 파일로 구현하여 서버에 제출 및 채점 (예: 12059876\_GildongHong.cpp)  
(채점서버에서 파일명에 대한 한글 인식이 되지 않기 때문에, 파일명은 영문이름으로 제출)
    - (b) 소스코드에 반드시 주석이 기재돼 있어야 함
    - (c) 소스코드 제출횟수는 상관없으며, 마지막에 제출한 소스코드를 기준으로 평가함
- ※ 파일명 형식 어길 시, 감점

### (3) 기타

- ① 제출 마감: 11월 28일 일요일 23:59까지 (마감 후에 제출되는 과제는 받지 않음)
- ② 마감 직전에는 I-class 및 채점 서버에 학생들이 많이 몰리기 때문에 혼잡할 수 있으니, 마감 시간보다 1시간 정도 이전에 여유 있게 제출하는 것을 권장
- ③ 제출 후에는 제대로 제출되었는지 반드시 확인
- ④ 부정행위가 적발될 경우, 배깅 학생과 원본을 제공한 학생 모두 0점 처리 및 성적의 총점에서 -10점 감점. 인터넷이나 책(교재, 참고서 등)의 소스코드, 예전에 제출했던 과제 등을 베껴서 제출해도 부정행위로 처리