

자료구조응용 과제

- *LMS 공지사항에 응용과제 제출 시 파일명 정하는 법과 제출 시 주의 사항이 공지되어 있으니 읽어 주세요.
- *실습실 PC에 코드가 남아있지 않도록 해 주세요. - 퇴실 시에 코드를 작성한 본인이 반드시 삭제해 주세요.
- *문제에서 명시하지 않아도 프로그램 실행 시 본인의 학부, 학번, 이름이 처음에 항상 출력되게 해 주세요.

과제#18 (만점: 10 점)

단계 1) Connected graph 로 주어지는 weighted undirected graph G에 대한 adjacency matrix 를 파일(in.txt)로 입력 받은 뒤, 모든 edge 에 대해 weight 의 오름차순으로 quick sort 를 수행하여 정렬된 결과를 출력하라. ('weight 값 (edge 정보)'로 출력하고 edge 는 vertex 번호 작은 것, 큰 것 순으로 표시할 것)

문제 2) 단계 1 의 결과를 이용하여 Kruskal's algorithm 을 구현하라. 아래의 구현 방법을 적용하고, T 에 edge (a, b), where $a < b$,가 추가될 때마다 해당 정보 weight (a, b)를 차례대로 화면 출력한다.

<구현 방법>

*minimum cost spanning tree T 를 구성하는 중간 단계에서, T 에 속한 각각의 connected component 를 서로 다른 집합으로 구분하기 위해 weighted union 과 collapsing find 를 이용한다

1. weight 에 대해 오름차순으로 edge 들을 정렬한다.
2. weight 가 작은 edge 부터 차례대로 선택하여 T 에 추가할 지 결정하는 과정을 반복한다.
 - 1) Edge E 의 양끝점 a,b 에 대해 $\text{Find}(a) == \text{Find}(b)$ 이면, E 는 T 에 추가하지 않는다.
 - 2) 그렇지 않은 경우, $\text{Union}(\text{Find}(a), \text{Find}(b))$ 를 수행하고, edge E 를 T 에 추가한다.

Requirement

1. File(in.txt)로 adjacency matrix 의 원소들을 입력 받으며, in.txt 는 matrix 의 upper triangle 부분만 row-major 로 표시하고 있다.

$n \ m_{1,2} \ \dots \ m_{1,n} \ m_{2,3} \ \dots \ m_{2,n} \ m_{3,4} \ \dots \ m_{3,n} \ \dots \ m_{n-1,n}$
여기에서

$n \ (1 \leq n \leq 20)$ 은 graph 의 vertex 의 개수이며, 각 vertex 는 1 부터 n 까지의 번호로 표시한다.
 $m_{ij} \ (1 \leq i \leq n, 1 \leq j \leq n)$ 는 weighted edge 의 값을 표시한다. 만약 vertex i 와 vertex j 사이에 weighted edge (i,j)가 있으면 m_{ij} 는 weight 를 나타내고, 그렇지 않을 경우 $m_{ij} = 0$ 이다.
 모든 edge 의 weight 는 서로 다르게 주어진다.

Ex1)	Ex2)
<in.txt> 5 9 2 0 4 3 0 1 0 0 5	<in.txt> 8 8 9 0 2 7 0 0 1 3 0 0 0 0 4 0 0 0 0 0 0 0 0 0 5 6 0 0 0
<화면 출력> 학부:... 학번: ... 이름: ... 단계 1 1 (2,5) 2 (1,3) 3 (2,3) 4 (1,5) 5 (4,5) 9 (1,2)	< 화면 출력> 학부:... 학번: ... 이름: ... 단계 1 1 (2,3) 2 (1,5) 3 (2,4) 4 (3,4) 5 (5,7) 6 (5,8)
단계 2 1 (2,5)	

2 (1,3)	7 (1,6)
3 (2,3)	8 (1,2)
5 (4,5)	9 (1,3)
	단계 2
	1 (2,3)
	2 (1,5)
	3 (2,4)
	5 (5,7)
	6 (5,8)
	7 (1,6)
	8 (1,2)

추가과제#18 없음