

Finding a Minimum Weight Spanning Tree using Kruskals.

Problem:

Given an undirected weighted graph, find a minimum weight spanning tree of the graph using Kruskal's algorithm.

Implementation Details:

The minimum weight spanning tree is to be outputted to standard output. The edges of the tree should be printed one per line. After the tree is outputted, print the total weight of the tree. Note, the order of the vertices listed in the listing of the edges may be different than the order used in the inputting of the edges. Please note, due the size of the text files, your program MUST read from standard in (allow for I/O redirection).

Your program will be compiled using

```
g++ -std=c++14 -Wall -Wextra -O3 -o p4 *.cpp
```

and p4 will be run using

```
time zcat file.txt.gz | ./p4 | grep weight
```

Hence, for this program it is not acceptable to have the filename input on the command line.

Use various data structures. A priority queue is to be used. One point is given for using the STL PQ, while three points are earned for using a priority queue written by the student (not simply downloaded from the internet.) The graph must be stored in either an adjacency list or a multilist. Additionally, checks will be made that a) Kruskals and not Prim's is implemented, and that b) the runtime complexity is $O(n \log n)$ and not $O(n^2)$.

Sample Inputs

0	1	1	2	1	53
0	2	5	3	1	170
1	2	4	4	1	68
2	3	3	5	1	29
2	4	2	6	1	90
3	5	2	7	1	50
4	5	2	8	1	49
			9	1	5
			10	1	5
			11	1	194
			0	1	45
			3	2	81
			4	2	61
			5	2	90
			6	2	149

7	2	61
8	2	4
9	2	82
10	2	26
11	2	05
0	2	32
4	3	34
5	3	117
6	3	104
7	3	52
8	3	01
9	3	181
10	3	05
11	3	8
0	3	41
5	4	25
6	4	26
7	4	2
8	4	89
9	4	65
10	4	49
11	4	34
0	4	5
6	5	17
7	5	13
8	5	106
9	5	16
10	5	32
11	5	113
0	5	26
7	6	20
8	6	181
9	6	65
10	6	89
11	6	80
0	6	45
8	7	85
9	7	45
10	7	37
11	7	52
0	7	5
9	8	82
10	8	26
11	8	173
0	8	52
10	9	16

	11	9	193
	0	9	50
	11	10	153
	0	10	26
	0	11	53

Outputs for the sample input

1 0		
2 1	1 9	
3 5	2 8	
4 2	3 10	
5 4	4 0	
The weight of the MWST is 11.	5 7	The total weight of the tree is 78
	6 5	
	7 4	
	8 3	
	9 5	
	10 1	
	11 2	

Data Structures To Be Used

- AdjList or Multilist
- Disjoint Sets (a.k.a. UnionFind)
- priority_queue (-5 for sorting, -2 for using the STL)