CENG 218 – Analysis and Design of Algorithms
Homework 4

15 June 2020

Due Date: 26 June 2020

Exercise 1 Minimum Spanning Trees

Write a C++ program in a file `mst.cc` that reads an adjacency list from the standard input and outputs the minimum spanning tree (MST) and its cost on the standard output. The input should start with a single line containing the number of vertices. Afterwards, the input should have one line for each vertex in the graph and starts with a one letter vertex label in the range `a`–`z`. Then it should list labels and weights of the all other vertices connected to this vertex one after the other. Your output should start with a single line containing the cost of the MST followed by lines each containing the vertex labels of each edge in the MST.

All submissions have to follow the following input/output format exactly. The example below has four vertices `a`–`d` and four edges. `a` is connected to `b` with an edge of weight 5, and to `d` with an edge of weight 1.

Example Input:

```
4
a b 5 d 1
b a 5 d 8
c d 15
d a 1 b 8 c 15
```

Example Output:

```
21
a b
a d
d c
```

Exercise 2 Shortest Paths

Write a C++ program in a file named `shortest.cc` that reads an adjacency list from the standard input and outputs the shortest paths from the first vertex to all others and the path costs on the standard output. The input should start with a single line containing the number of vertices. Afterwards, the input should have one line for each vertex in the graph and it should start with a one letter vertex label in the range `a`–`z`. Than it should list labels and weights of the all other vertices connected to this vertex one after the other. Your output must have one line per target vertex that contains the label of the target vertex and the cost of the shortest path.

All submissions have to follow the following input/output format exactly. The example below has four vertices `a`–`d` and four edges. `a` is connected to `b` with an edge of weight 5, and to `d` with an edge of weight 1.

Example Input:

```
4
a b 5 d 1
b a 5 d 8
c d 15
d a 1 b 8 c 15
```

Example Output:

```
b 5
c 16
d 1
```