

## CSE431/531 (Fall 2016) Project 1

Due date: Oct 30, 2016

You need to implement the heap data structure and Prim's algorithm for the minimum spanning tree problem, using the heap data structure you implemented. You can use C++, Java or Python to implement the algorithm.

**Input** you need to read the input graph from the file "input.txt". In the first line of the file, we have two positive integers  $n$  and  $m$ .  $n$  is the number of vertices in the graph and  $m$  is the number of edges in the graph. The vertices are indexed from 1 to  $n$ . You can assume that  $1 \leq n \leq 10000$  and  $1 \leq m \leq 100000$ . In the next  $m$  lines, each line contains 3 integers:  $u, v$  and  $w$ , with  $1 \leq u < v \leq n$  and  $1 \leq w \leq 10^6$ . This indicates that there is an edge  $(u, v)$  of weight  $w$ . You can also assume that the graph is connected and there are no parallel edges.

**Output** You need to output to the file "output.txt". The first line of the file is an integer indicating the total weight of the minimum spanning tree. From line 2 to line  $n$ , you need to output the  $n-1$  edges in the minimum spanning tree. Each line contains 2 integers between 1 and  $n$ , indicating the two end-points of an edge.

**Sample input (from the course slides)**

```
9 14
1 2 5
1 8 12
2 3 8
2 8 11
3 4 13
3 6 4
3 9 2
4 5 9
4 6 14
5 6 10
6 7 3
7 8 1
7 9 6
8 9 7
```

**Output for sample input**

```
42
1 2
2 3
3 6
3 9
4 5
5 6
6 7
7 8
```

**Grading** Here is how we grade your project. We shall post some sample test cases soon. If you passed the sample test cases (without cheating), you will get 30% of the points. We have some hidden test cases. These test cases are worth another 30% of the points. For the remaining 40%, we will read your code and see if your implementation is correct (e.g., whether you implemented the heap data structure correctly). For each test case, your algorithm needs to terminate in reasonable amount of time (say, 5 seconds).

**Submission** Submit your source file via UBlearns. For convenience, please only submit one source file, with the name MST\_⟨firstname⟩\_⟨lastname⟩\_⟨UBID⟩.{cpp|py|java}, where ⟨firstname⟩ is your first name (with initial letter capitalized) and ⟨lastname⟩ is your last name (with initial letter capitalized) and ⟨UBID⟩ is your UB netID. The extension of the source file is cpp, py, or java, depending on the programming language you use.