

COP 3503C Programming Assignment # 5

Time Limit: 4 seconds (per input case)

Read all the pages before starting to write your code

What should you submit?

Write all the code in a single Main.java file. **Submit your Main.java file to Codegrade. Also, submit 2 test files (test_1.txt and test_2.txt) along with your test strategy test_strategy.txt . In the test strategy, write your test strategy and discuss about the two test files you have generated and what is the correct output for those test files.**

Please include the following commented lines in the beginning of your code to declare your authorship of the code:

```
/* COP 3503C Assignment 5
```

```
This program is written by: Your Full Name */
```

Compliance with Rules: UCF Golden rules apply towards this assignment and submission. Assignment rules mentioned in syllabus, are also applied in this submission. The TA and Instructor can call any students for explaining any part of the code in order to better assess your authorship and for further clarification if needed.

Caution!!!

Sharing this assignment description (fully or partly) as well as your code (fully or partly) to anyone/anywhere is a violation of the policy. I may report to office of student conduct and an investigation can easily trace the student who shared/posted it. Also, getting a part of code from anywhere will be considered as cheating.

Deadline:

See the deadline on webcourses. After that the assignment submission will be locked. **An assignment submitted by email will not be graded and such emails will not be replied according to the course policy.**

What to do if you need clarification on the problem?

I will create a discussion thread in webcourses and I highly encourage you to ask your question in the discussion board. Maybe many students might have same question like you. Also, other students can reply and you might get your answer **faster**. Also, you can write an email to the TAs and put the course teacher in the cc for clarification on the requirements.

How to get help if you are stuck?

According to the course policy, all the helps should be taken during office hours. Occasionally, we might reply in email.

Problem Description: Treasure Hunts

Monster land has n cities and one capital. The cities are numbered from 1 to n . Some of the cities are connected by bidirectional roads and each road has a length. It is possible to go from a city to any other city by these roads. Yeti Monster found an ancient book that has a secret information about the Monster land. The book said that the land has some very precious treasures. Finding them will make the finder the king of the land. However, the exact position of the treasures are kept secret. Further reading the document, Yeti, found some clue. The clue says that all the treasures are located exactly at a distance L from the capital. The capital is the S_{th} city in the city list.

As part of more details of the clue: A treasures can be located in a place and the place is either a city or at a point on a road, if and only if the shortest distance from the place to the capital along the roads of the country equals exactly L .

Yeti became impatience to know how many treasures are located in the Monster land so that Yeti can become the king and rule that country. Yeti knows that you are a genius programmer, and you can help! So, help Yeti!

Input specification (Must be read from standard input (no file i/o))

The first line of input contains 3 space separated integers, C ($2 \leq C \leq 10^5$), R ($C-1 \leq R \leq \min(10^5, n(n-1)/2)$), S ($1 \leq S \leq N$), representing the number of cities, number of roads, and the city number that represents the capital, respectively.

Then the next R lines contain the descriptions of the roads. Each of them contains 3 space separated integers v_i, u_i, w_i ($1 \leq v_i, u_i \leq n, v_i \neq u_i, 1 \leq w_i \leq 1000$), where v_i, u_i are numbers of the cities connected by this road and w_i is its length. The last input line contains integer L ($0 \leq L \leq 10^9$) — the distance from the capital to the places where the treasures are located. It is guaranteed that:

- between any two cities no more than one road exists.
- each road connects two different cities;
- from each city there is at least one way to any other city by the roads.

The Output (standard output):

Print two numbers — the number of treasures in the cities and the number of treasures on the roads in Monster land.

Partial Credit Options and some hints:

If your code can produce output for the number of treasures in the cities properly, you can get 70% credit for each test case. We have learned couple of graph algorithms. You need to decide which algorithm is appropriate to get this information. Also, make sure to test your code properly. The second number that shows the number of treasures in the road could be a bit tricky to find. So, carefully plan. That part will require a couple of if-else conditions based on the source and destination of each edge, with the min distance available in the vertices of the edge, the weight of the edge, and the value of L .

Sample Input**Sample Output**

4 6 1 1 2 1 1 3 5 2 3 1 2 4 1 3 4 1 1 4 2 2	In city: 2 On the road: 1	Explanation: 1 treasure is at city 3 1 treasure is at city 4, 1 treasure is on the road (1,3)
5 6 3 3 1 1 3 2 1 3 4 1 3 5 1 1 2 6 4 5 8 4	In city: 0 On the road: 3	Explanation: 1 treasure is on the road (1, 2) 1 treasure is on the road (4,5) at a distance 3 from city 4 (towards city 5) 1 treasure is on the road (4,5) at a distance 3 from city 5 (towards city 4)

Some Restrictions:

- You have to use a known algorithm we have learned in the class as part of this process and need to implement that algorithm based on the lecture (see lecture note and watch the recording. We have discussed this code in the class or recording).
- The function/method that implements the algorithm must be named based on the algorithm name
- The lecture version of the code has some extra lines to store and print the result. Your code should not contain any of those extra lines as those lines are not relevant for this assignment.

What To Submit

See the first page of the assignment about what to submit.

Hints:

1. What algorithm you have learned to find the shortest paths to all the nodes from a single node?
2. Some hints are provided above in the partial credit options
3. Of-course draw the sample graph and see how the output is calculated from the sample graph.

Good Luck!