CMPE 343
Fall 2023
Programming Homework 3

This assignment is due by 23.59 on Sunday 19, May 2024

You are welcome to ask your HW related questions. You should use only one of these options:

There will be two Q&A Office Hours on the following days:

- **11 May 2024 18.00-20.00**

https://tedu.zoom.us/j/95363590973?pwd=T01XTHYzeHJPdmdOZXdyV1h2M1NpUT09

- **12 May 2024 18.00-20.00**

https://tedu.zoom.us/j/97065468238?pwd=cC9RSFM5YWsxQThUSnVIRE42T1RBUT09

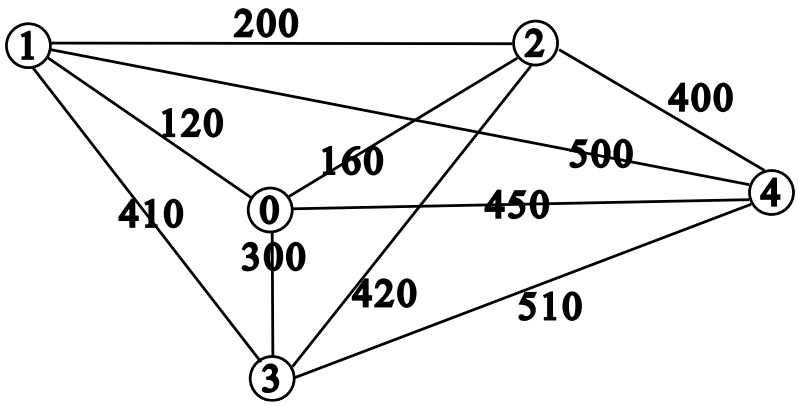
Note: Please make sure that you have read the HW document well before participating. However, no HW related questions will be accepted except from the above options.

PROGRAMMING TASK

In this part, you must implement your own graph data structure by taking inspiration from your textbook and use it to help to solve problem. You are not allowed to use any external library or .jar file. Any solutions without using graph data structure are not evaluated!

Question 1(25 points):

In the defined system there is one source (0), and 4 load point (1,2,3,4). In the graph the locations of the source and load is given find the smallest path of connection to loads and source. (Minimum Spanning Tree of the undirected graph)



Sample Output:

```
V=5
E=10
0 1 120
0 2 160
0 3 300
0 4 450
1 0 120
1 2 200
1 3 410
1 4 500
2 0 160
2 1 200
2 3 420
2 4 400
3 0 300
3 1 410
3 2 420
3 4 510
4 0 450
4 1 500
4 2 400
4 3 510

The Minimum Spanning Tree Path
… … …
… … …
… … …
… … …
… … …

The Minimum Spanning Tree value= …
```

The Minimum Spanning Tree Path part shows the path for example:
0 3 300
0 4 450
1 2 200
1 4 500 (this is only example not the result)

Important Note: If the path contain 0. The answer must be:
0 4 450
Not 4 0 450

If the connection is not contain zero the lowest number is written in the path.
3 2 420 is not correct.
2 3 420 correct.

The Minimum Spanning Tree value for above example
The Minimum Spanning Tree value=1450 (**this is only example not the result**)
In the question you create 3 java document and one text file:

- **HW3_Q1_solution: Main body and the solution of Minimum Spanning Tree question**

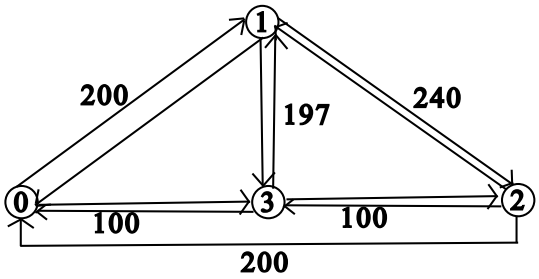
- **FileRead: read the text file.**

- **Valuefinder: take the value of txt file and convert the integer and create the undirected graph**

- **HW3_Q1: text file (I uploaded to the system)**

Question 2(25 points):

Find the shortest path for starting point is 0 and assume that finish points are 1 , 2 , 3 (find each points shortest path). In the question the output must shown in order. The graph is directed graph.



Sample Output:

```
V=4
E=11
0 1 200
0 3 100
1 0 200
1 2 240
1 3 197
2 0 200
2 1 240
```

```
2 3 100
3 0 100
3 1 197
3 2 100

The result
… … … …
… … … …
… … … …
```

The example output is:

0 1 200

0 1 2 440

0 3 100 (**this is only example not the result**)

Important Note: Please do not forget. The result will be contain 2 point and 1 value or 3 point and value.

The starting point is 0 and the final point is firstly 1, secondly 2, thirdly 3. The shown graph is directed graph.

In the question you create 3 java document and one text file:

- **HW3_Q2_solution: Main body, find the solution of Shortest Path.**
- **FileRead: read the text file.**
- **Valuefinder: take the value of txt file and convert the integer and create the directed graph.**
- **HW3_Q2: text file (I uploaded to the system)**

In the questions, your explanations are very important.

WHAT TO HAND IN

- **You need to upload your code into VPL on LMS for each question.** If you do not upload your code into VPL on LMS, your homework will **not be graded.**
- The Java sources should be WELL DOCUMENTED as comments, as part of your grade will be based on the level of your comments.

- You need to upload **maximum-3 pages** PDF report document that explains your own answers for programming task in a clearly readable PA report format (refer to **PA REPORT FORMAT** section).

PA REPORT FORMAT

A programming assignment report is a self-description of a programming assignment and your solution. The report must not be hand-written. You may use a word processor or the on-line editor of your choice and prepare as a PDF document. The report must be grammatically correct and use complete English sentences. Each report should include the following sections, in the order given:

Information (%2.5): This section includes your ID, name, section, assignment number information properly.

Problem Statement and Code Design (%15): Include a brief summary of the problem and/or your sub-tasks to be completed in this assignment. You should show your modular design rationale by creating a structure chart that indicates your top-down, stepwise refinement of the problem solution. You may create the structure chart using available graphical tools like MS PowerPoint, SmartDraw etc.

Implementation and Functionality (%20): Since you have modular source code, you should describe each sub-module (program) in this section. Each sub-module should include names and types of any input/output parameters as well as the pseudocode algorithm that used for completing its task. By this way, you give meaning to each chart boxes from the previous section.

Testing (%7.5): You should provide a tester class that is able to identify key test points of your program. This class should be able to generate additional (apart from the given sample input/output) test data for the purpose of being clear on what aspects of the solution are being tested with each set. This section should also include a description of any program *bugs* that is, tests which has incorrect results. You should write these to describe your tests, summarize your results, and argue that they cover all types of program behavior.

Final Assessments (%5): In this final section, you should briefly answer the following questions:

- What were the trouble points in completing this assignment?
- Which parts were the most challenging for you?
- What did you like about the assignment? What did you learn from it?

GRADING:

- Codes (%50: %25 for Q1 and %25 for Q2)
  - Available test cases evaluation on VPL: %15
  - Hidden test cases evaluation: %15
  - Approach to the problem: %20
- Report (%50: %25 for Q1 and %25 for Q2)
  - Information: %2.5
  - Problem Statement and Code design: %15
  - Implementation, Functionality: %20
  - Testing: %7.5
  - Final Assessments: %5

IMPORTANT

IMPORTANT NOTES: Do not start your homework before reading these notes!!!

1. This assignment is due by 23.59 on Sunday 19, May 2024.

2. You should upload your homework to LMS before the deadline. No hardcopy submission is needed. You should upload your codes into VPL and your report into submission place on LMS.

3. The standard rules about late homework submissions apply (20 points will be deducted for each late day). Please see the course syllabus for further discussion of the late homework policy as well as academic integrity.

4. You ARE NOT ALLOWED to modify the given method names. However, if necessary, you may define additional data members and member functions.

5. Your classes' name MUST BE as shown in the homework description.

6. The submissions that do not obey these rules will not be graded.

7. To increase the efficiency of the grading process as well as the readability of your code, you have to follow the following instructions about the format and general layout of your program.

8. Do not forget to write down your id, name, section, assignment number or any other information relevant to your program in the beginning of your Java files. Example:

```
//-------------------------------------------------
```

```
// Title: Scheduler tester class
// Author: Name/Surname
// ID: 0000000000
// Section: 1
// Assignment: 1
// Description: This class tests the …
//-------------------------------------------------
```

9. Since your codes will be checked without your observation, you should report everything about your implementation. Add detailed comments to your classes, functions, declarations etc. Make sure that you explain each function in the beginning of your function structure. Example:

```
void setVariable(char varName, int varValue)
//-------------------------------------------------------
// Summary: Assigns a value to the variable whose
// name is given.
// Precondition: varName is a char and varValue is an
// integer
// Postcondition: The value of the variable is set.
//-------------------------------------------------------
{
      // Body of the function
}
```

10. Indentation, indentation, indentation...

11. This homework will be graded by your TAs, Deniz Merve Gündüz, Ruhi Zafer Çağlayan. You are also welcome to ask your course instructors Ulaş Güleç for help.