

CMPE 224-343
Fall 2023
Programming Homework 2

This assignment is due by 23:59 on Friday 19, April 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:

- **08 April 2024 18.00-20.00**
https://teams.microsoft.com/l/meetup-join/19%3ameeting_OTY0MzJmNmItMTMzOS00YTViLTkwMmEtYjA5YmRiZGFIZDZl%40thread.v2/0?context=%7b%22Tid%22%3a%22e5714f9e-09d8-4b08-9d5a-4bf08b29b891%22%2c%22Oid%22%3a%2256be7a8b-fbed-4a49-a35e-b8e8bb77968d%22%7d
- **09 April 2024 18.00-20.00**
https://teams.microsoft.com/l/meetup-join/19%3ameeting_ZGIzNDk2NmUtMWZhMC00ZmNhLTk1ZjAtYzUwNTk0ZDJINDAx%40thread.v2/0?context=%7b%22Tid%22%3a%22e5714f9e-09d8-4b08-9d5a-4bf08b29b891%22%2c%22Oid%22%3a%2256be7a8b-fbed-4a49-a35e-b8e8bb77968d%22%7d

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

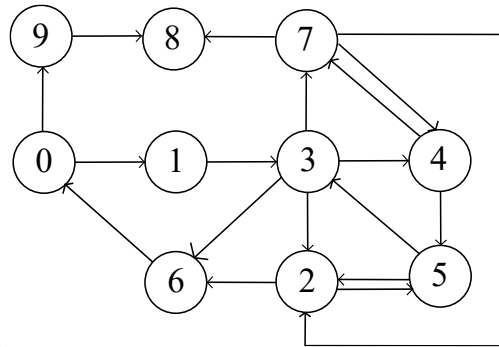
In this part, you must implement your own graph data structure by taking inspiration from your textbook and use it to help to solve problem. You are not allowed to use any external library or .jar file. Any solutions without using graph data structure are not evaluated!

Question 1(25 points):

The Directed graph shown below. From using this graph and HW2_Q1_text text file, write a code that take the starting point, and show two step forward like sample output. In the question test the starting point as 0,2, and 4. In the question you create 4 java document and 1 text file:

- DirectGraph: Create direct graph

- In the question your explanations are very important.

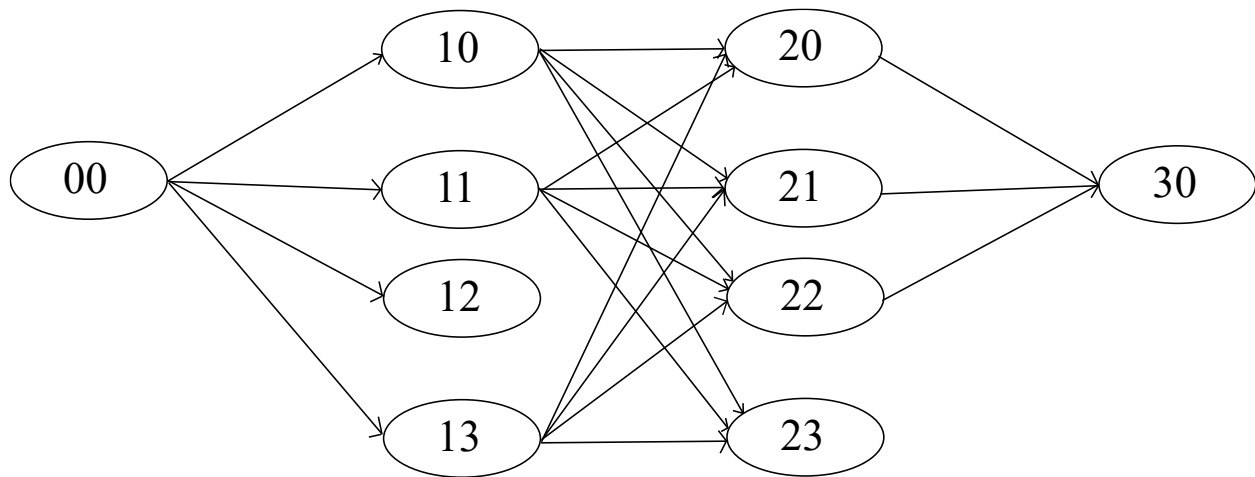


```
V=9
E=18
0 1
0 9
1 3
2 5
2 6
3 2
3 4
3 6
3 7
4 5
4 7
5 2
5 3
6 0
7 2
7 4
7 8
9 8

Start point:
3 2 5
3 2 6
3 4 5
3 4 7
3 6 0
3 7 2
3 7 4
3 7 8
```

Question 2(25 points):

In the below, the neural network of the artificial intelligence is given; however, some neurons do not connected to the next neurons. From using the graph and HW2_Q2_text find these neurons. The sample output is given, but the result part has been erased. In your program result must be written. The result part example must be like this:



If the neuron is 11:

0 11

If the neuron is 20:

0 10 20, however all the possible other connections for 11,12... must be added

The result:

0 11

0 10 20

...

In this question there are three program and 1 text file:

- Valuefinder: take the value of txt file and convert the integer
- FileRead: read the text file.

- HW2_Q2_solution: Create your graph, find the nonconnected neurons and write the path to the neurons which is started in 0 and end in the number of neuron. Write all possible paths.

Sample Output:

```
V=10
E=19
0 10
0 11
0 12
0 13
10 20
10 21
10 22
10 23
11 20
11 21
11 22
11 23
13 20
13 21
13 22
13 23
20 30
21 30
22 30

Result is
.....
```

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 Friday 19, April 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.