Assignment 1

Implement a program in Java to solve the problem assigned to you with your assignment partner.

Step 1:

Choose an assignment partner. Add your partner to your Gitlab repository for this assignment.

Step 2:

During the lecture, the instructor will run a script to assign two programming problems to each group randomly.

Step 3 (Phase 1):

- One of the assignment partners (P1) will write a test for the given problem. Once P1 is done writing the test, he/she will push the test code to their repository. It is possible that the code with the test case does not compile.
- The other partner (P2) will write the code that passes the test case written by P1. P2 will push the code to the repository once he/she ensures that the code works (and passes the test).
- Repeat the above process at least three times. Depending on the problem, you may need to write more test cases and corresponding code.
- Your code must satisfy the requirements given in the problem.
- Your test cases must test the code to uncover potential issues.

Step 4 (Phase 2):

- In this part, the role of the partners will be reversed. P2 will first write the test cases, and P1 will write code to pass the test cases.
- Each team will write their code to solve the second problem given to them in this phase.
- The team must create a separate folder and project for this problem.

Constraints

- Both the problems must be implemented in Java, and tests must be written using JUnit tests. If the need be, teams may use any mocking library/framework.
- The project must be a Maven project.
- The implementation cannot use in-built libraries for the data structure in question.
- If the type is not specified and you need to choose a type (of a node, for example), you may choose any type (even a simple integer is fine).

- The projects must be compilable and testable using maven. If the final deliverable does not compile with simple maven command, i.e., *'mvn compile'* , the group will get zero in that phase. Similarly, *'mvn test'* must run all the tests.

Delivery

- The code of the assignment must be pushed to the GitLab repository of one of the assignment partners.
- On Brightspace, you must provide the link to the GitLab repository against Assignment-1.

Rubric

- The final deliverable compiles with maven - **prerequisite** (if not done, the evaluator will not proceed further)
- Tests are written first and pushed to the repository - **prerequisite** (if not done, the evaluator will not proceed further)
- Total marks = 2*4.5 = 9 (4.5 for each phase)
- Effective tests - 1*3 = 3 (for each phase); total 6 marks
    - a test tests functional requirement or edge case with required setup = 1
    - a test tests trivial aspects when the other critical aspects are not tested = 0.5
- The given problem is implemented as per given requirements = 1.5 for each phase; total 3 marks
    - the implementation works perfectly = 1.5
    - the implementation works most of the time = 1.0
    - the implementation works sometimes/some cases = 0.5
    - the implementation doesn't work = 0