List of tasks (all programs must be compiled using Gradle (or Maven). There must be tests for tasks, jUnit should be used for testing. Compliance with the Java Code Convention is mandatory)

1. Using the serialization mechanism, the client creates a server program: the client creates an object, serializes it, sends it to the server, the server deserializes it and loads it into memory.

2. Using flows, solve a system of linear equations with a tridiagonal matrix.

3. Write a method that takes as a parameter a reference to the thread group object and creates a thread that periodically displays data about the thread hierarchy and thread subgroups within the given group. Test the method as part of an application that uses several short-running threads belonging to different groups.

4. Using the reflection mechanism, write a program that displays a complete description of the class with the given name, including all information about the class except for import statements, comments, and code for initializers, constructors, and methods. Use your own loader to load the class.

5. Write your own implementation of the SkipList Free lock algorithm.

6. Write an implementation of the Non-blocking Michael-Scott queue algorithm

7. Using the synchronized wait notify mechanism, implement Cyclic Barrier.

8. Using the synchronized wait notify mechanism, implement ReentrantLock

9. Using the synchronized wait notify mechanism, implement Phaser.

10. Implement the ThreadPool pattern.