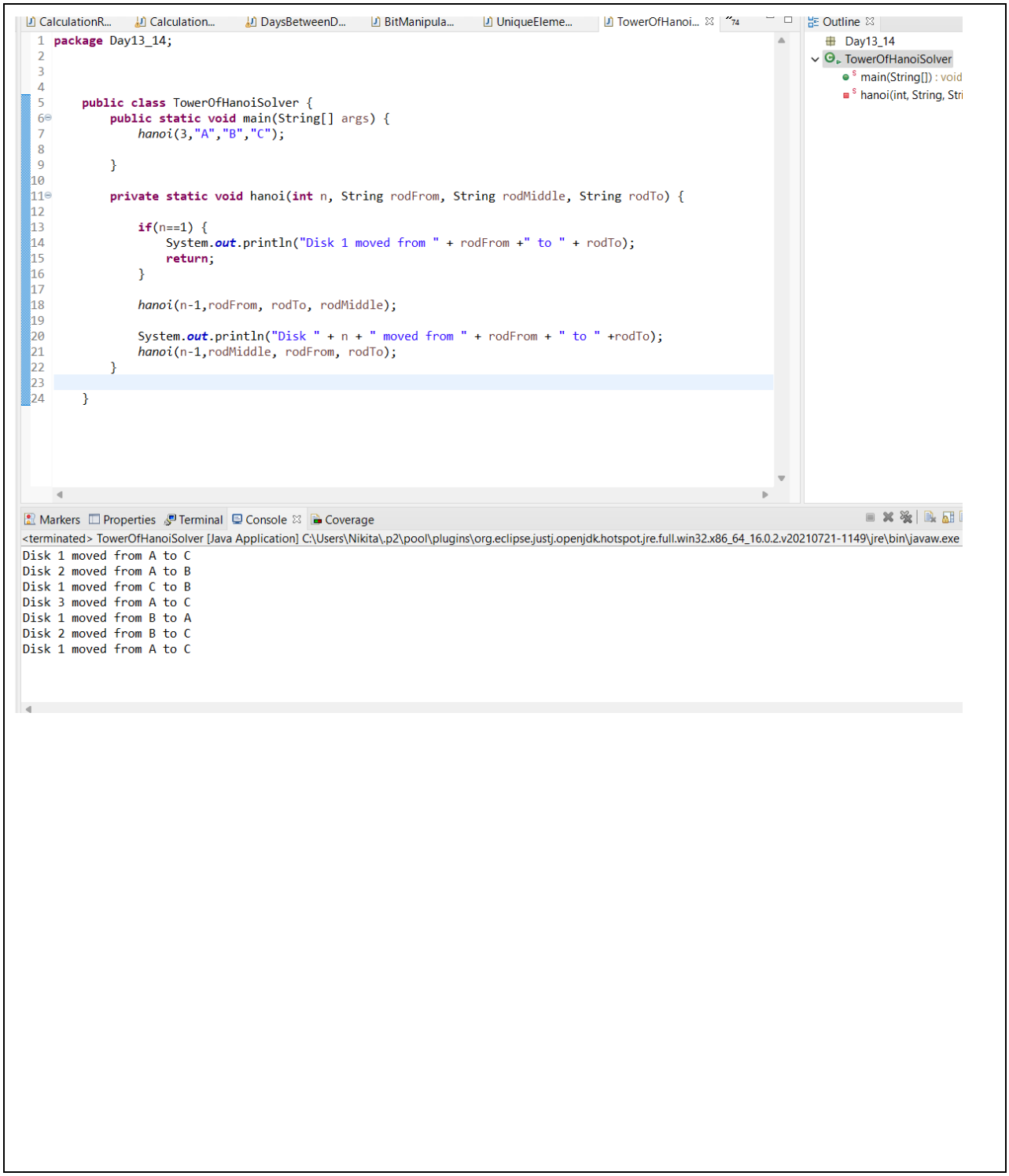
Tower of Hanoi Solver

Create a program that solves the Tower of Hanoi puzzle for n disks. The solution should use recursion to move disks between three pegs (source, auxiliary, and destination) according to the game's rules. The program should print out each move required to solve the puzzle.

```
public class TowerOfHanoiSolver {
		public static void main(String[] args) {
			hanoi(3,"A","B","C");

		}

		private static void hanoi(int n, String rodFrom, String rodMiddle,
String rodTo) {

			if(n==1) {
				System.out.println("Disk 1 moved from " + rodFrom +" to "
+ rodTo);
				return;
			}

			hanoi(n-1,rodFrom, rodTo, rodMiddle);

			System.out.println("Disk " + n + " moved from " + rodFrom + " to
" +rodTo);
			hanoi(n-1,rodMiddle, rodFrom, rodTo);
		}

	}
```




```java
package Day13_14;



    public class TowerOfHanoiSolver {
        public static void main(String[] args) {
            hanoi(3,"A","B","C");

        }

        private static void hanoi(int n, String rodFrom, String rodMiddle, String rodTo) {

            if(n==1) {
                System.out.println("Disk 1 moved from " + rodFrom +" to " + rodTo);
                return;
            }

            hanoi(n-1,rodFrom, rodTo, rodMiddle);

            System.out.println("Disk " + n + " moved from " + rodFrom + " to " +rodTo);
            hanoi(n-1,rodMiddle, rodFrom, rodTo);
        }

    }
```