1. Recall this problem from HW 2: Using a loop, write code that sends the following sequence of strings to a server: “0”, “01”, “012”, “0123”, “01234”, “012345”, “0123456”, “01234567”. Assume that toServer is already declared and is assigned to a valid NetConnection. You should have a solution that uses a while loop to accomplish this. Rewrite this solution to use the new loop form we have recently discussed, the “for” loop. Here’s how a “for” loop works. The basic form is:
   
   ```
   for (<initialization>; <condition>; <increment>) {
     <body statements>
   }
   ```

   <initialization> is an expression that initialized the loop counter. The traditional loop counter is an integer variable called i, j, or k, and it can be declared either right in the initialization area or outside the loop. If declared in the initialization area, the scope of the variable is the loop header and body. <condition> is a Boolean expression that is evaluated before the loop executes for the first time and then after each subsequent iteration. If it evaluates to true, the body is executed, otherwise it is skipped. Finally, <increment> is an expression that modifies the loop control variable so that progress is made toward termination. As an example, the following for loop calculates the sum of the integers from 1 to 10.

   ```java
   int sum = 0;
   for (int i = 1; i <= 10; i = i + 1) {
     sum = sum + i;
   }
   ```

   a) Rewrite your solution to this problem using a for loop.

2. Write a for loop that creates ten buttons labeled 1 through 10 and adds them to a JPanel called keypad. (Remember that you can get “5” with the expression “" + 5)
3. Write a while loop that receives messages on an existing NetConnection called fromServer until a message that contains the pattern "XOXO" is received. The idea is to skip the lines sent before the one containing the desired pattern.

4. Write a while loop that receives messages on an existing NetConnection called fromServer and keeps a count of lines that start with either the string "<table>" or the string "<form>". You only need to keep one count.

5. Write a recursive method using the following skeleton that sends the message “SOS” on a NetConnection a specified number of times. For example, assuming that to911 is a NetConnection instance that is already connected to some server, to send “SOS” five times, you would invoke the method as follows: sendSOS (to911, 5);

```java
public void sendSOS ( NetConnection toSomewhere, int n ) {
}
```
6. Write a recursive method for the StringList class that would change all of the strings in the list to all lower case.