```java
import components.naturalnumber.NaturalNumber;

/**
 * Put a short phrase describing the program here.
 *
 * @author Vaishnavi Kasabwala
 *
 */
public final class Hailstone2 {
    /**
     * Private constructor so this utility class cannot be instantiated.
     */
    private Hailstone2() {
    }

    /**
     * Generates and outputs the Hailstone series starting with the given
     * {@code NaturalNumber}.
     *
     * @param n
     *            the starting natural number
     * @param out
     *            the output stream
     * @updates out.content
     * @requires n > 0 and out.is_open
     * @ensures out.content = #out.content * [the Hailstone series starting with
     *          n]
     */
    private static void generateSeries(NaturalNumber n, SimpleWriter out) {
        NaturalNumber x = new NaturalNumber2(n);
        NaturalNumber zero = new NaturalNumber2(0);
        NaturalNumber one = new NaturalNumber2(1);
        NaturalNumber two = new NaturalNumber2(2);
        NaturalNumber three = new NaturalNumber2(3);

        int count = 1;

        while (x.compareTo(one) != 0) {
            out.print(x + ", ");
            NaturalNumber temp = new NaturalNumber2(x);
            if (temp.divide(two).compareTo(zero) == 0) {// when even
                x.divide(two);
            } else { // when odd
                x.multiply(three);
                x.add(one);
            }
            count++;
        }
        out.println(x);
        out.println("Length of series: " + count);
    }

    /**
     * Main method.
     *
     * @param args
     *            the command line arguments
     */
    public static void main(String[] args) {
        SimpleReader in = new SimpleReader1L();
        SimpleWriter out = new SimpleWriter1L();

        /*
         * Put your main program code here; it may call myMethod as shown
         */

        out.println("Enter a positive integer: ");
        int input = in.nextInteger();
        NaturalNumber n = new NaturalNumber2(input);

        generateSeries(n, out);

        /*
         * Close input and output streams
         */
        in.close();
        out.close();
    }

}

```