

```
1  /* Problem 5--Tribute List
2     There is a simple formula for this one:  $(2n)! / (2^n)$ . Since the
3     numbers can get very large, I used Java's BigInteger class. */
4
5  import java.io.*;
6  import java.util.*;
7  import java.math.*;
8
9  public class prob5 {
10
11     private static Scanner in = null;
12     private static PrintWriter out = null;
13     private static int cs = 0;
14
15     public static void main (String[] args) throws Exception {
16
17         in = new Scanner (new File ("prob5.in"));
18         out = new PrintWriter ("prob5.out");
19         while (true) {
20             int sz = in.nextInt ();
21             if (sz==0) break;
22             Process (sz); //Process data case
23         }
24         in.close ();
25         out.close ();
26     }
27
28     /* Process computes the formula for each data case */
29     public static void Process (int sz) throws Exception {
30
31         BigInteger ans = BigInteger.ONE;
32         for (int i = 2; i <= 2*sz; i++)
33             ans = ans.multiply (new BigInteger (""+i));
34         ans = ans.divide (new BigInteger ("2").pow(sz));
35         out.println ("Case "+(++cs)+" : There are "+ans+
36                     " possible orderings.");
37     }
38 }
39
```