

```
1  /* Problem 2--Pointless Riffle
2   It's not necessary to riffle the cards many times and check to see if
3   the deck is in order again. I just shuffle the deck once and count
4   cycles of cards. */
5
6  import java.io.*;
7  import java.util.*;
8
9  public class prob2 {
10
11     private static int cs = 0;
12     private static Scanner in = null;
13     private static PrintWriter out = null;
14
15     public static void main (String[] args) throws Exception {
16
17         in = new Scanner (new File ("prob2.in"));
18         out = new PrintWriter ("prob2.out");
19         while (true) {
20             int ds = in.nextInt ();
21             if (ds==0) break;
22             Process (ds); //Process each case
23         }
24         in.close ();
25         out.close ();
26     }
27
28     /* Process riffles the deck once and counts cycles in the shuffling */
29     public static void Process (int ds) throws Exception {
30
31         int[] once = new int[ds];
32         for (int i=ds-1,j=(ds+1)/2-1;i >= 0; j--,i-=2)
33             once[i] = j; //riffles deck
34         for (int i=ds-2,j=ds-1; i >=0; j--,i-=2)
35             once[i] = j;
36         int sh = 1;
37         for (int i=0; i < ds; i++) {
38             if (once[i] ==-1) continue;
39             int ct=0, j = i;
40             while (once[j] != -1) { //counting cards in this cycle
41                 ct++;
42                 int k = j;
43                 j = once[j];
44                 once[k] = -1;
45             }
46             sh = LCM (sh,ct); //find the least common multiple of cycle lengths
47         }
48         out.print ("Case "+(++cs)+
49             "\n": President Snow should shuffle the deck "+sh+" times.\r\n\r\n");
50     }
51
52     /* computes the greatest common factor of a and b */
53     public static int GCF (int a, int b) {
54
55         int r = a%b;
56         if (r==0) return b;
57         if (r > b-r) r = b-r;
58         return GCF (b,r);
59     }
60
61     /* computes the least common multiple of a and b */
62     public static int LCM (int a, int b) {
63
64         return a/GCF(a,b)*b;
65     }
66 }
67
```