Exercise 1:
68 x 139 (by hand):

| | | | |
|---|---|---|---|
| 68 | x | 139 | |
| 34 | x | 278 | |
| 17 | x | 556 (remainder) -----→ | 556 |
| 8 | x | 1112 | |
| 4 | x | 2224 | |
| 2 | x | 4448 | |
| 1 | x | 8896 ----------→ | +8896 |
| | | | 9542 |

```
CianO@LAPTOP-G35GKLVJ MINGW64 ~/git/algorithms20290-2021-repository-Cian2001/wk2
-algorithm-analysis (main)
$ java RussianPeasantsAlgorithm data/8ints.txt
9452
```

**2. Implement the Russian Peasant's algorithm in Java and verify its correctness.**

I tested it on the example above and it returned the correct result.
I also tested it on several handpicked numbers and some of the inputs form last weeks data folder. All returned the correct results.

```
1  public class RussianPeasantsAlgorithm {
2      public static void main(String[] args) {
3
4          In in = new In(args[0]);// to take input from a file
5          int[] a = in.readAllInts();
6
7          long x, y, res = 0; // i changed it to long because ints were too small
8
9          x = a[0];//input first two digits in a file
10         y = a[1];
11
12         while(x>1){
13             if(x%2 == 1) {// checks if it is an odd number (remainder)
14                 res += y;// if it is it adds on to the total
15             }
16             x=x/2;// x is halved until it is less than 1
17             y=y*2;// y is doubled until x is less than 1
18         }
19         res += y;//adds up the results
20
21         Stopwatch timer = new Stopwatch();// records the length of time in ms
22         StdOut.println("elapsed time = " + timer.elapsedTime());
23         System.out.println(res);
24     }
25 }
26
```

604,242 * 686,904
Answer = 415,056,246,768
elapsed time = 0.001

```
CianO@LAPTOP-G35GKLVJ MINGW64 ~/git/algorithms20290-2021-repository-Cian2001/wk2
-algorithm-analysis (main)
$ java RussianPeasantsAlgorithm data/8Kints.txt
elapsed time = 0.001
415056246768
```

1985477589 * 1723372036
Answer = 3,421,716,554,987,301,204

elapsed time = 0.0

```
CianO@LAPTOP-G35GKLVJ MINGW64 ~/git/algorithms20290-2021-repository-Cian2001/wk2
-algorithm-analysis (main)
$ java RussianPeasantsAlgorithm
elapsed time = 0.0
3421716554987301204
```

I'm not sure how to graph the data as the answer of 415 billion got a result of 0.001 ms and the answer of 3.4 quintillion got a result of 0.0 ms!

First of all, the smaller number took 0.001 ms and the larger one took 0.0 ms (I understand that rounding came into play here) however it makes no sense.

Also I don't know how to graph these results effectively as the difference in time to run the program is marginal and is hardly enough to make any logical conclusions from the data.

### What do you think is the complexity of your algorithm and why?

I think it would be log(n) as the results didn't seem to change given the bigger inputs. (And in my case the time efficiency got better as the input got larger.) I'm not sure how though and my only reasoning is that it can't possibly be (1) and it is much faster than n, so it must be log(n).