

**READ THIS FIRST:**

Do your best to do every item on your own; if you cannot immediately do an item, go on to others and then come back to it later. Please check the resources section if you have any problems and talk with your professor if there are any further questions.

**Due: The following Sunday after this was posted by midnight.**

**Goals:**

- Practice getting around the command line compiling and running Java programs.
- Practice getting around in and using the lab submission site.
- Explain some key concepts we covered in class.
- Get some lab points.

***Part One. Description:***

In cryptography we often employ the Euclidean algorithm to perform certain calculations. The algorithm is designed to find the greatest common divisor between two integers  $a, b > 0$ , i.e.  $\gcd(a, b)$ . The algorithm assumes that  $a \geq b$ , otherwise their values are switched. Then, it applies the division algorithm to find  $a = q_1b + r_1$ . If the remainder  $r_1 = 0$  then the algorithm stops, meaning that the greatest common divisor is  $b$ . Otherwise, the algorithm continues dividing until the remainder is zero, as follows:

$$\begin{aligned}
 b &= q_2r_1 + r_2 \\
 r_1 &= q_3r_2 + r_3 \\
 r_2 &= q_4r_3 + r_4 \\
 &\dots \\
 r_{n-2} &= q_nr_{n-1} + r_n \\
 r_{n-1} &= q_{n+1}r_n + 0
 \end{aligned}$$

Once the algorithm terminates, the remainder  $r_n$  is the greatest common divider, i.e.  $r_n = \gcd(a, b)$ . Your task is to implement this algorithm.

**Input:**

The input has  $L$  lines; each line has two integers,  $a, b > 0$  separated by a single space.

**Output:**

The output is composed of  $L$  lines, each with an integer corresponding to  $\gcd(a, b) > 0$ .

Sample Input 1:

```
100 74
```

Sample Output 1:

```
2
```

Sample Input 2:

```
1071 462  
148 75
```

Sample Output 2:

```
21  
1
```

**Requirements:** You will create a file `Driver_lab2a.java` that will contain a class with the name `Driver_lab2a` which will contain the `main` method of the program. Inside your driver create a method with the following signature `long euclidAlg(long a, long b)` that receives a pair of positive integers, and returns the corresponding greatest common divisor as an integer. Note that for multiple lines of input you can call this function repeatedly. Also, make sure you follow the style guidelines that were given for this course (see iLearn).

**Part Two. Description:**

The Euclidean algorithm is widely used in cryptography to perform a number of different calculations. The original algorithm can be extended to be more efficient and to keep record of values. The extended algorithm still finds the greatest common divisor between two integers  $a, b > 0$ , i.e.  $\gcd(a, b)$ , under the assumption that  $a \geq b$ , otherwise their values are switched. Then, it computes the values for  $d$ ,  $x$ , and  $y$  that satisfy the equation:

$$d = ax + by \tag{1}$$

where  $d = \gcd(a, b)$ . Your task is to implement this extended algorithm efficiently.

**Input:**

The input has  $L$  lines; each line has two integers,  $a, b > 0$  separated by a single space.

**Output:**

The output is composed of  $L$  lines, each with three integers separated by a space corresponding to  $d = \gcd(a, b)$ ,  $x$ , and  $y$  that satisfy Eq (1).

Sample Input 1:

```
148 75
```

Sample Output 1:

```
1 37 -73
```

Sample Input 2:

```
1155 862
240 46
```

Sample Output 2:

```
1 203 -272
2 -9 47
```

**Requirements:** You will create a file `Driver_lab2b.java` that will contain a class with the name `Driver_lab2b` which will contain the `main` method of the program. Inside your driver create a method with the following signature `long[] euclidAlgExt(long a, long b)` that receives a pair of positive integers, and returns an array long integers of length three, corresponding to  $d$ ,  $x$ , and  $y$  in Eq (1). Note that for multiple lines of input you can call this function repeatedly. Also, make sure you follow the style guidelines that were given for this course (see iLearn).

**Resources:**

- Your textbook (Stanoyevitch)!
  - Project submission guidelines for this course (posted on iLearn)
  - Coding style guidelines for this course (posted on iLearn)
  - “How to” use the command line “shell” (posted on iLearn)
  - Piazza for asking questions to professor and classmates use the tag: `lab2`
  - The official Java reference: <http://docs.oracle.com/javase/tutorial/collections/TOC.html>
  - Stack Overflow Java Tag: <http://stackoverflow.com/questions/tagged/java>
- 

**Submission:**

- Upload your work to the submission site <https://car.rivas.ai> and submit your `Driver_lab2a.java` and `Driver_lab2b.java` before the due date and make sure they pass all the tests. If they do not pass all the tests, then it means that your programs are incorrect and you need to keep working on them.
- Once you pass all the tests, your professor will review your code for style and then you will receive a grade.