National Yang Ming Chiao Tung University
Department of Electrical and Computer Engineering
Data Structures, Fall 2023

September 26, 2023
Instructor: H.-M. Chen
TA: H.-J. Luo

Programming Assignment #1

Arrays

1. Problem Description

The greatest common divisor $gcd(a, b)$ between two positive integers a and b is defined as the largest positive integer that divides both a and b without a remainder.

GCD is a very powerful tool in modern cryptography, and when the target integers to be calculated are small (less than 10^8), GCD can be calculated in a few seconds with a naïve method. However, the numbers in modern cryptography requires at least 512 digits to prevent attackers from using a brute-force method to derive the secret key. This required number is too large for the naïve methods to calculate GCD in a reasonable time and the numbers exceeds the limit of even long long in the C language. In this problem, you will need to calculate the GCD of two big integers efficiently.

2. Input Format

One line containing two integers, a and b, where $0 < a, b < 10^{256}$.

3. Output Format

An integer representing $gcd(a, b)$ with a single end-of-line (endl).

4. Sample Input / Output

Sample Input 1
20230925 52903202

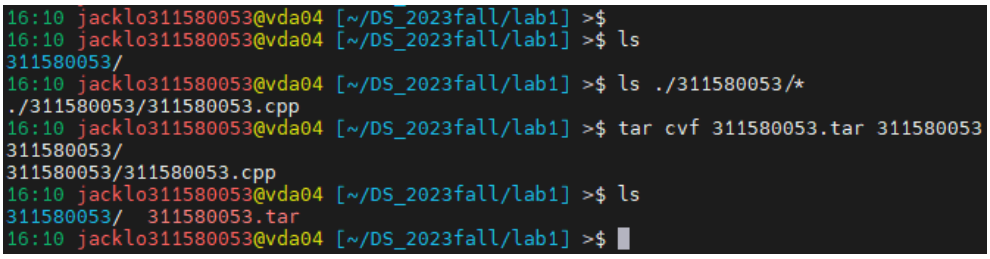
Sample Output 1
11

Sample Input 2
11111122222233333344444455555 666666777777888888999999

Sample Output 2
333333

5. Submission Information

1. Your program must be written in C/C++ language and can be compiled on the Linux platform.
2. Please put the required files in a folder named with your Student_ID and the required files should also be named with your Student_ID (.cpp, .c).
3. To submit your program, please use the command below to compress the folder named with "[Student_ID].tar" in the Linux environment and upload it to E3.
   tar cvf Student_ID.tar Student_ID



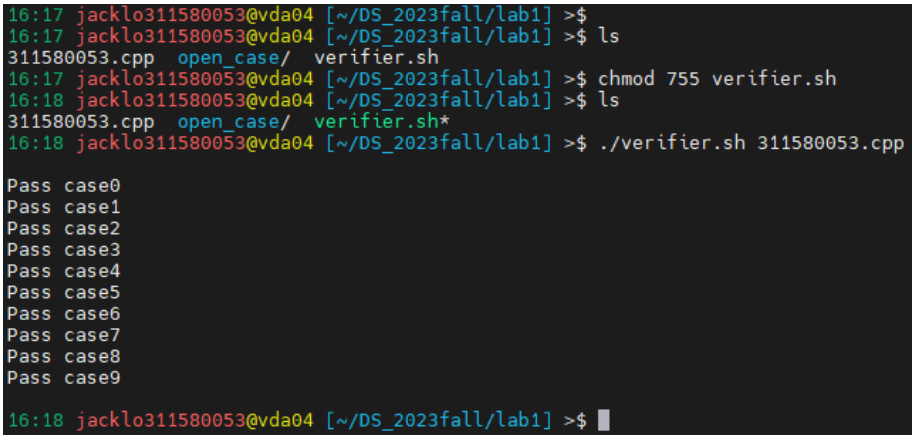
6. Due Date

Be sure to upload the tar file by "October 10, 2023". There will be a 10% penalty per day for the first four days (weekend included) and will not be accepted afterwards.

7. Grading Policy

The programming assignment will be graded based on the following rules:

- Pass the open cases with compilable source code (60%)



- Pass the hidden cases with compilable source code (40%)
- -10% of your total score if any file occurs naming error or not compress
- ***No credits for plagiarism***

Hint

To deal with the big integers, we need a "data structure", such as an integer array in C to represent larger values. For instance, you can use an integer array where each element represents one (decimal) digit, like representing 202309 by the following code snippet.

```
vector<int> digits = {9, 0, 3, 2, 0, 2};
```

It is not required to use the representation above, though. You can use any representation that facilitates your implementation.

Algorithm: Binary Algorithm for Greatest Common Divisor

Input: Two positive integers a and b.

Output: A positive integer ans representing greatest common divisor of a and b.

$n \leftarrow min(a,b), m \leftarrow max(a,b), ans \leftarrow 1$

while $n \neq 0$ and $m \neq 0$ **do**

  if n is even and m is even **then**

    $ans \leftarrow ans \times 2, n \leftarrow n/2, m \leftarrow m/2$

  else if n is even **then**

    $n \leftarrow n/2$

  else if m is even **then**

    $m \leftarrow m/2$

  end

  if $n > m$ **then**

    $swap(n,m)$

  $m \leftarrow (m-n)$

end

return $n \times ans$
