The luagcd Package in LaTeX

Chetan Shirore and Dr. Ajit Kumar

December 29, 2022

1 Introduction

Using Lua, the `luagcd` package is developed to find the **greatest common divisor (gcd)** of integers in LaTeX. It provides an easy way to find gcd of two or more integers inside LaTeX documents. The package provides commands to obtain step-by-step computation of gcd of two integers by using Euclidean algorithm. In addition, the package has the command to express gcd of two integers as a linear combination. The Bezout's Identity can be verified for any two integers using commands in the package. No particular environment is required for the use of commands in the package. It is written in Lua, and the TeX file has to be compiled with the LuaLaTeX engine.

In order to find the gcd of two integers, a and b, the package uses a repeated division algorithm. This technique is basically based on the fact $\gcd(a,b) = \gcd(a, b - \lambda a)$, where λ is an integer. This technique to find the gcd of integers is called as Euclidean algorithm. For finding gcd of more than two numbers, the associativity of the gcd operation is used.

$$\gcd(\gcd(a,b),c) = \gcd(a,b,c)$$

This associativity of the gcd operation allows the recursion of functions to find the gcd of more than two integers. The recursion technique (where the function calls itself) in Lua is effectively used in the package for finding the gcd of integers. The package also makes use of the following result.

$$\gcd\{a,b\} = \gcd\{|a|,b\} = \gcd\{a,|b|\} = \gcd\{|a|,|b|\}$$

Algorithms in the package convert non-zero integers to positive integers, and then the Euclidean algorithm is used to find their gcd. If the gcd of a and b is d and it is expressed as a linear combination $ax + by = d$, then we have

$$ax + by = d \iff (-a)(-x) + by = d \iff ax + (-b)(-y) = d \iff (-a)(-x) + (-b)(-y) = d$$

So in order to find a solution of $(-a)(-x) + by = d$, $ax + (-b)(-y) = d$ or $(-a)(-x) + (-b)(-y) = d$ it suffices to find a solution of $ax + by = d$. This fact is used in an algorithm of the package to express the gcd of two integers as their linear combination.

2 Installation and License

The installation of `luagcd` package is similar to plain latex package, where the `.sty` file is in LaTeX directory of texmf tree. The package can be included with `\usepackage{luagcd}` command in the preamble of the LaTeX document. The TeX file is to be compiled using the LuaLaTeX engine.

The `luagcd` package is released under the LaTeX Project Public License v1.3c or later. The complete license text is available at http://www.latex-project.org/lppl.txt. It is developed in Lua. Lua is available as a certified open-source software. Its license is simple and liberal, which is compatible with GPL.

3   Commands in the luagcd package

Table 1 lists operations in the `luagcd` package.

Command	Syntax	Description
`\luagcd`	`\luagcd{x_1, x_2, ..., x_n}`	Gives gcd of integers $x_1, x_2, x_3, \ldots, x_n$.
`\luagcdwithsteps`	`\luagcdwithsteps{a}{b}`	Gives the gcd of two integers, a and b, in a step-by-step manner by repeated application of the Euclidean algorithm.
`\luagcdlincomb`	`\luagcdlincomb{a}{b}`	Gives the gcd of two integers a and b and expresses it as a linear combination of two integers a and b.
`\luagcdlincombwithsteps`	`\luagcdlincombwithsteps{a}{b}`	Expresses the gcd of two integers a and b as their linear combination in a step-by-step manner.

Table 1: Operations in the luagcd package

4   Examples and usage

The command `\luagcd{20,30,60,70}` outputs to $\boxed{10}$.
The command `\luagcdwithsteps{-20}{-6008}` outputs the following.

Step 1: Apply the division algorithm to 6008 and 20.
$6008 = 20(300) + 8$
Step 2: Apply the division algorithm to 20 and 8.
$20 = 8(2) + 4$
Step 3: Apply the division algorithm to 8 and 4.

$8 = 4(2) + 0$
The gcd of -20 and -6008 is the last non-zero remainder and it is 4.

The command `\luagcdwithsteps{-20}{-6008}` outputs the following LaTeX code.

```
Step 1: Apply the division algorithm to 6008 and 20.\\
$6008 = 20(300) + 8$\\
Step 2: Apply the division algorithm to 20 and 8.\\
$20 = 8(2) + 4$\\
Step 3: Apply the division algorithm to 8 and 4.\\
$8 = 4(2) + 0$ \\
The gcd of -20 and -6008 is the last non-zero remainder and it is 4.
```

The command `\luagcdlincomb{10011}{210}` outputs the following.

The gcd of 10011 and 210 is 3 and the equation $-10011x+210y = 3$ and has a solution $(x, y) = (3, -143)$.

The command `\luagcdlincombwithsteps{-10011}{210}` outputs the following.

Step 1:10011 is written as a linear combination of 10011 and 210.
$10011 = (1)(10011) + (0)(210)$
Step 2:210 is written as a linear combination of 10011 and 210.
$210 = (0)(10011) + (1)(210)$
Step 3: The equation in Step 2 is multiplied by 47 and subtracted from the equation in Step 1.
$141 = (1)(10011) + (-47)(210)$
Step 4: The equation in Step 3 is multiplied by 1 and subtracted from the equation in Step 2.
$69 = (-1)(10011) + (48)(210)$
Step 5: The equation in Step 4 is multiplied by 2 and subtracted from the equation in Step 3.
$3 = (3)(10011) + (-143)(210)$
The gcd of -10011 and 210 is 3 and the equation $-10011x + 210y = 3$ and has a solution $(x, y) = (-3, -143)$.

The command `\luagcdlincombwithsteps{-10011}{210}` outputs the following LaTeX code.

```
Step 1:10011 is written as a linear combination of 10011 and 210.\\
$10011 = (1)(10011) + (0)(210)$\\
Step 2:210 is written as a linear combination of 10011 and 210.\\
$210 = (0)(10011) + (1)(210)$\\
Step 3: The equation in Step 2 is multiplied by 47 and
subtracted from the equation in Step 1.\\
$141 = (1)(10011) + (-47)(210)$\\
Step 4: The equation in Step 3 is multiplied by 1 and
 subtracted from the equation in Step 2.\\
$69 = (-1)(10011) + (48)(210)$\\
```

Step 5: The equation in Step 4 is multiplied by 2 and
subtracted from the equation in Step 3.\\
$3 = (3)(10011) + (-143)(210)$\\
The gcd of -10011 and 210 is 3 and the equation $-10011x + 210y = 3$
and has a solution $(x,y) = (-3,-143)$.