Euclid's Algorithm

Euclid's algorithm to find the greatest common divisor of two integers is based on the following lemma.

Lemma If $a = bq + r$ where a, b, q, and r are integers, then $\gcd(a, b) = \gcd(b, r)$.

Proof. Since $\gcd(a, b)$ divides both a and b, then it divides $r = a - bq$. Therefore,

$$\gcd(a, b) \leq \gcd(b, r). \tag{1}$$

Since $\gcd(b, r)$ divides both b and r, then it divides $a = bq + r$. Therefore,

$$\gcd(a, b) \geq \gcd(b, r). \tag{2}$$

Combining inequalities (1) and (2), we conclude that $\gcd(a, b) = \gcd(b, r)$. □

Let's find $\gcd(1124, 84)$ using Euclid's algorithm.

$$\begin{aligned}
1124 &= 84(13) + 32 \\
84 &= 32(2) + 20 \\
32 &= 20(1) + 12 \\
20 &= 12(1) + 8 \\
12 &= 8(1) + \boxed{4} \longleftarrow \text{(the last nonzero remainder is the answer)} \\
8 &= 4(2) + 0
\end{aligned}$$

Applying the above lemma, we conclude that

$$\gcd(1124, 84) = \gcd(84, 32) = \gcd(32, 20) = \gcd(20, 12) = \gcd(12, 8) = \gcd(8, 4) = \gcd(4, 0) = 4.$$

The following is a Python version of Euclid's algorithm.

```
def gcd(a,b):
  while b:  # any number except zero is True in Python
     a, b = b, a%b
  return a
```

Let's test it by typing this code and saving it in a file called `gcd.py`.

```
>>> from gcd import *
>>> gcd(1124, 84)
4
```

We can also code Euclid's algorithm using recursion as follows.

```
def gcd(a,b):
  if b == 0:
    return a
  else:
    return gcd(b, a%b)
```