Computational Algebra. Lecture 4

M.Eulàlia Montoro

10 de marzo de 2022

Traditional Euclidean Algorithm

1. $r_0 \longleftarrow f$, $r_2 \longleftarrow g$.
2. $i \longleftarrow 1$
   while $r_i \neq 0$ **do** $r_{i+1} \longleftarrow r_{i+1}\, rem\, r_i$, $i \longleftarrow i+1$
3. **return** r_{i-1}

MCD

- What does "MCD" compute?

MCD

- What does "MCD" compute? It computes the greater common divisor of two given numbers using the Euclidean algorithm.

MCD

- What does ''MCD'' compute? It computes the greater common divisor of two given numbers using the Euclidean algorithm.
- Which is the algebraic complexity of ''MCD'' ?

MCD

- What does "MCD" compute? It computes the greater common divisor of two given numbers using the Euclidean algorithm.
- Which is the algebraic complexity of "MCD" ?

$$c(MCD) = c(While) = c(s[N]) = Fib(N) - 1,$$

where N is the number of steps (ie. divisions).

$$c(s[0]) = c(s[1]) = 0$$

$$c(s[i]) = c(s[i-2]) + c(s[i-1]) + 1$$

Writing a table of values, we conclude that:

$$c(s[i]) = Fib(i) - 1,$$

where $Fib(i)$ is the i-th Fibonacci number. Moreover, if $a > b > 0$, it can be seen that $N < 5log_{10}(b) + 1$ (where $\log_{10}(b)$ is the number of digits of b in basis 10).

Homework: Stein Algorithm

This algorithm (1962, Silver and Tersian) (1967, Stein) does not involve any division in $\mathbb{Q}$, only substractions, parity tests and divisions by two of even numbers. So, it is well adapted to binary calculus.

a) Show that if a and b are nonegative integers, then
   - $gcd(2a, 2b) = 2gcd(a, b)$
   - $gcd(2a+1, 2b) = gcd(2a+1, b)$
   - $gcd(2a+1, 2b+1) = gcd(min(2a+1, 2b+1), |a-b|)$.

Homework: Stein Algorithm

b) Design and implement a recursive algorithm which allows the computation of the gcd of any pair of integers by using the three properties mentioned above. Show that the algorithm concludes in a finite number of steps.

Homework: Stein Algorithm

c) Show that the number of bits operations of this algorithm is at most $O((logN)^2)$, with $N = max(a, b)$.

Homework: Stein Algorithm

c) Design and implement an iterative version of this algorithm.

Algorithm 1 *Iterbin*

procedure Iterbin(a, b)
    $d \leftarrow 1, t$;
    while $EvenQ[a]$**and**$EvenQ[b]$ **do**
        $a \leftarrow \frac{a}{2}$; $b \leftarrow \frac{b}{2}$; $d \leftarrow 2d$;
    while $a \neq 0$ **do**
        while $EvenQ[a]$ **do** $a \leftarrow \frac{a}{2}$
        while $EvenQ[b]$ **do** $b \leftarrow \frac{b}{2}$
        $t \leftarrow \frac{|a-b|}{2}$
        if $a \geq b$ **then** $a \leftarrow t$
        else$b \leftarrow t$
    return $d \cdot b$

Homework: Stein Algorithm

e) Modify either of the versions (recursive or iterative) of your code in such a way that the output is a relation of the form r, s, u, with $r.a + s.b = u = gcd(a, b)$.

Hint: Encyclopedia of Cryptography and Security