

Computational tools for problem solving
Lab list 1

Division, Euclidean and Extended Euclidean Algorithms

Problem 1.

- i) Write a program to perform the division algorithm. Compute the quotient and remainder q, r for $(a, b) = (103, 11)$.
- ii) Write a program to perform trial division to find the prime factors of a given number n .
- iii) Find the smallest prime number larger than 10010.

The Euclidean Algorithm to compute $\gcd(a, b)$ works because

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

where $r = a - qb$, so that the iteration of the division algorithm produces a strictly descending sequence of remainders $0 \leq r_{i+1} < r_i$. As you all know, the iterations are written in a table

	q_1	q_2	q_3	\dots	q_{n-1}	q_n	q_{n+1}	
a	b	r_1	r_2	\dots	r_{n-2}	r_{n-1}	r_n	
r_1	r_2	r_3	r_4	\dots	r_n	0		$\Rightarrow \gcd(a, b) = r_n.$

where, if we let $r_{-1} = a$ and $r_0 = b$ then

$$r_{i+1} = r_{i-1} - q_i r_i \tag{1}$$

for $i = 0, 1, \dots$

Problem 2.

- i) Write a program to perform the Euclidean algorithm. Return as an output also the number n of steps needed for the computation.
- ii) Use your program to compute $\gcd(a, b)$ for the following pairs of integers a, b (all of them have comparable size):

- 1) $(a, b) = (487669403177, 28736540321)$.
- 2) $(a, b) = (20365011074, 12586269025)$.
- 3) $(a, b) = (2^{35} - 1, 2^{34} - 1)$.

The Extended Euclidean Algorithm “XGCD” finds not only $\gcd(a, b)$ but also the integers s_n, t_n such that Bézout’s identity holds

$$\gcd(a, b) = r_n = s_n a + t_n b. \quad (2)$$

Bézout’s coefficients s_n, t_n are also computed inductively. Let

$$\begin{aligned} s_{-1} &= 1, s_0 = 0, \\ t_{-1} &= 0, t_0 = 1, \end{aligned}$$

and like in (1), define

$$\begin{aligned} s_{i+1} &= s_{i-1} - q_i s_i, \\ t_{i+1} &= t_{i-1} - q_i t_i \end{aligned}$$

for all $i = 0, 1, \dots$

It is easy to show

$$s_i a + t_i b = r_i, \quad \forall i = -1, 0, 1, \dots \quad (3)$$

Problem 3.

- i) Show (3) is true (use induction on i).
- ii) Write a program to perform the Extended Euclidean Algorithm. Return as an output also the number n of steps needed for the computation.
- iii) Use your program to compute s, t such that

$$sa + tb = \gcd(a, b)$$

for the following pairs of integers a, b :

- 1) $(a, b) = (487669403177, 28736540321)$.
- 2) $(a, b) = (20365011074, 12586269025)$.
- 3) $(a, b) = (2^{35} - 1, 2^{34} - 1)$.

In the particular case of two values a, b such that $\gcd(a, b) = 1$, the XGCD algorithm is very useful to compute $a^{-1} \pmod{b}$ or $b^{-1} \pmod{a}$. This follows by taking remainders \pmod{b} or \pmod{a} respectively in both sides of the identity (2).

Problem 4.

- i) Use the XGCD to compute $28736540321^{-1} \pmod{487669403177}$.
- ii) Use the XGCD to compute $12586269025^{-1} \pmod{20365011074}$.

Problem 5.

- i) Can you guess the values of a, b so that n is the largest possible? Do some research to show that the Euclidean Algorithm to compute $\gcd(a, b)$ with $a > b$ takes at most $5 \log_{10}(b) + 1$ steps.