CS 235: Algebraic Algorithms, Spring 2021

Midterm Exam

Date: Wednesday, March 10, 2021.   primality test .

Problem 1. Find integers $a, b, c > 1$ satisfying the system of equations: $a \cdot c = 647701$, $b \cdot c = 690497$. Describe the method used.

Solution. We have: $a \cdot c = 647701$, $b \cdot c = 690497$, then c is a common divisor of 647701 and 690497, so let it be the greatest common divisor.

To find $\gcd(647701, 690497)$, we run the Euclidean Algorithm on input $a = 690497$ and $b = 647701$. The steps are as follows:

$$690497 = 647701 \cdot 1 + 42796 \longrightarrow q_1 = 1,\ r_1 = 42796$$

$$647701 = 42796 \cdot 15 + 5761 \longrightarrow q_2 = 15,\ r_2 = 5761$$

$$42796 = 5761 \cdot 7 + 2469 \longrightarrow q_3 = 7,\ r_3 = 2469$$

$$5761 = 2469 \cdot 2 + 823 \longrightarrow q_4 = 2,\ r_4 = 823$$

$$2469 = 823 \cdot 3 + 0 \longrightarrow q_5 = 3,\ r_5 = 0$$

Since $r_5 = 0$, $\gcd(690497, 647701) = r_4 = 823$. Hence, $c = \gcd(690497, 647701) = 823$, $a = 647701/823 = 787$ and $b = 690497/823 = 839$

Euclidicen Algorithm

gcd of $(ac, bc) = C$

$690497 = 1 \times 647701 + 42796$

$647701 = 15 \times 42796 + 5761$

$42796 = 7 \times 5761 + 2469$

$5761 = 2 \times 2469 + 823$

$2469 = 3 \times \boxed{823} + 0$ 6L   6

2469

GCD

$A = 839$

$C = 823$

$B = 767$

Extended Euclidicen Algorithm

$823 = 5761 - 2 \times 2469$

$823 = 5761 - 2(42796 - 7 \times 5761)$

$823 = 15 \times 5761 - 2 \times 42796$

$823 = 15 \times (647701 - 15 \times 42796) - 2 \times 42796$

$823 = 15 \times 647701 - 227 \times 42796$

$823 = 15 \times 647701 - 227 \times (690497 - 647701)$

$= 242 \times 647701 - 227\ 690497$

2.

Problem 2. The Extended Euclidean Algorithm expresses $\gcd(a, b)$ as $d = as - bt$. Can these s, t be both odd? Both even? Explain.

Solution. s and t can be both odd. Proof of existence: $gcd(3, 2) = 1$ and running EEA on inputs $a = 3$ and $b = 2$ gives the linear combination $3 \cdot 1 - 2 \cdot 1 = 1$ where $s = 1$ and $t = 1$ which are both odd.

However, s and t cannot be both even. Assume, for the sake of contradiction, that s and t are even, then we can express $s = 2s'$ and $t = 2t'$ for some integers s', t'. This means that $\gcd(s, t) > 1$ as it is at least 2, which contradicts Theorem 4.3 (iii) which says $\gcd(s, t) = 1$.

$s = 2s'$ $t = 2t'$

$2as' - 2bt'$

$\times 3 \times 5$ 5×7

lcm of 30 35 = 2 × 3 × 5 × 7

210

2×3×5 5×7 = 0 = 0

Problem 3. Is the pair of congruences $x \equiv a \pmod{30}$, $x \equiv b \pmod{35}$ solvable for every a, b? Explain. No since 30 and 35 are not prime

NO

Solution. Observe that the prime factorisation of 30 is $2 \cdot 3 \cdot 5 = 30$. Therefore, by CRT, the congruence $x \equiv a \pmod{30}$ can be expressed as the following system:

$$x \equiv a \pmod{2}$$

$$x \equiv a \pmod{3}$$

$$x \equiv a \pmod{5}$$

Similarly, we can express $b \equiv a \pmod{35}$ as:

$$x \equiv b \pmod{5}$$

$$x \equiv b \pmod{7}$$

This means that if the given system is solvable, then it must be the case that $a \equiv b \pmod{5}$ (by CRT), or simply $5|(a-b)$.

Hence, the system is **not** solvable for every arbitrary a and b, **unless** $5|(a-b)$.

Let's say $x \equiv a \pmod{30}$ and $x \equiv b \pmod{35}$ they are both qualified.

then

$A_1 \equiv a \pmod{30}$ $\quad x = A_1 + B_1$

$B_1 \equiv b \pmod{35}$ $\quad 35 A_1 \pmod{30}$

Problem 4. Describe a polynomial time algorithm to decide for prime p and integers $a \in [0, p)$ if the equation $(x^2 \mod p) = a$ has solution. Explain fully.

Solution. Observe that asking whether the equation $(x^2 \mod p) = a$ has a solution is equivalent to asking whether $a \in (\mathbb{Z}_p^*)^2$. By Euler's Criterion, if $a \in (\mathbb{Z}_p^*)^2$, then $a^{(p-1)/2} = 1$ and if $a \notin (\mathbb{Z}_p^*)^2$, then $a^{(p-1)/2} = -1$.

Thus, we can design an algorithm as follow: calculate $a^{(p-1)/2}$ in $\mathbb{Z}_p$ then check if the result equals to -1; if not, return a yes answer; else, return a no answer. By section 3.4, evaluating some a^e in $\mathbb{Z}_n$ for any integer n takes time $O(||e|| \cdot ||n||^2)$. In our case, evaluating $a^{(p-1)/2}$ in $\mathbb{Z}_p$ takes time $O(||(p-1)/2|| \cdot ||p||^2) \sim O(||p||^2)$ which is polynomial time.

For all integer 0 ~ p-1

$x^2 \pmod p$ 2   1차   2차

7     $\underline{4}$ 16 $\longrightarrow$

2   3^3 $4^3 \equiv 1$

16

64