INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR
Mid-Spring Semester 2019-20

Date of Examination : 25-02-2020 **Session(FN/AN)** AN **Duration** 2 hrs **Total Marks** 60
Subject No : CS60094 **Subject Name :** COMPUTATIONAL NUMBER THEORY
Department/Centre/School : Computer Science and Engineering
Specific charts, graph paper,log book etc. required No
Special Instructions (if any) Answer five questions. If you make any assumptions, state them upfront. Provide concise answers.

1. Let $a, b \in \mathbb{Z}$ with $a \geq b \geq 0$, let $d = \gcd(a, b)$, and assume $d > 0$. Suppose that on input a, b Euclid's algorithm performs k division steps, and computes the remainder sequence $\{r_i\}_0^k$ and the quotient sequence $\{q_i\}_{i=2}^{k+1}$ (such that $r_0 = a$, $r_1 = b$ and $r_{i-2} = q_i r_{i-1} + r_i$). Now suppose we run Euclid's algorithm on input $\frac{a}{d}, \frac{b}{d}$. Show that on these inputs, the number of divisions steps is also k, the remainder sequence is $\left\{\frac{r_i}{d}\right\}_0^k$ and the quotient sequence is $\{q_i\}_{i=2}^{k+1}$. **[12]**

   Solution: Let $r'_0, r'_1, \ldots q'_2, q'_3, \ldots$ denote the remainder sequence of Euclidean algorithm on input $a/d, b/d$. In the first iteration, we have $r'_0 = a/d$, $r'_1 = b/d$ and $r'_2 = r'_0 \pmod{r'_1}$. Since $d \mid a, d \mid b$, we have $d \mid a - q_2 b$ i.e., $d \mid r_2$. Dividing both sides of $a = q_2 b + r_2$ by d, we have $a/d = q_2(b/d) + (r_2/d)$ where r_2/d is an integer and $0 \leq r_2/d < b/d$. Therefore, $r'_2 = r_2/d$ and $q'_2 = q_2$.

   Assume that upto $i-1$ division steps, the remainder sequence is $\{r_j/d\}_{j=0}^{i-1}$ and the quotient sequence is $\{q_j\}_{j=2}^{i}$. Let $r'_{i+1} = (r_{i-1}/d) \pmod{(r_i/d)}$. On input a, b, we have after i division steps $r_{i-1} = q_{i+1} r_i + r_{i+1}$ with $0 \leq r_{i+1} < r_i$. Since $d \mid r_{i-1}, d \mid r_i$, we have $d \mid r_{i-1} - q_{i+1} r_i$ i.e., $d \mid r_{i+1}$. Dividing both sides of $r_{i-1} = q_{i+1} r_i + r_{i+1}$ by d, we have $(r_{i-1}/d) = q_{i+1}(r_i/d) + (r'_{i+1}/d)$ where r'_{i+1}/d is an integer and $0 \leq r'_{i+1}/d < r_i/d$. Therefore, $r'_{i+1} = r_{i+1}/d$ and $q'_{i+1} = q_{i+1}$. As a result, we have $r'_{k+1} = r_{k+1}/d = d/d = 1$ as required (since $\gcd(a/d, b/d) = 1$) i.e., after k divisions we get $\gcd(a/d, b/d)$ as the remainder.

2. Let $n \in \mathbb{N}$, $x_1, x_2, \ldots, x_k \in \mathbb{Z}_n$ and $e_1, e_2, \ldots, e_k \in \mathbb{N}$ such that each e_i for $i = 1, \ldots, k$ has a binary representation of at most ℓ bits. Devise an algorithm to compute $x_1^{e_1} x_2^{e_2} \cdots x_k^{e_k} \pmod{n}$ using at most ℓ squarings modulo n and $\ell + 2^k$ additional multiplications modulo n. **[12]**

   Solution: Let $e_i = (a_{i,\ell-1}, \ldots, a_{i,1}, a_{i,0})_2$ for each $i \in \{1, \ldots, k\}$. Described below is an algorithm for computing $x_1^{e_1} x_2^{e_2} \cdots x_k^{e_k} \pmod{n}$.

   Precompute the 2^k values $x_1^{b_1} x_2^{b_2} \cdots x_k^{b_k} \pmod{n}$ for all $(b_1, b_2, \ldots, b_k) \in \{0,1\}^k$
   Initialise $y \leftarrow 1$
   For $i \leftarrow \ell - 1, \ldots, 0$, do
   $y \leftarrow y^2 \pmod{n}$
   $y \leftarrow y \cdot x_1^{a_{1,i}} x_2^{a_{2,i}} \cdots x_k^{a_{k,i}} \pmod{n}$
   Return y

   The precomputation step takes atmost 2^k multiplications modulo n (each product with exactly j x_i's can be computed using one modular multiplicaiton of products of smaller than j x_i's). The first step within the loop

requires one squaring modulo n. Since $x_1^{a_{1,i}} x_2^{a_{2,i}} \cdots x_k^{a_{k,i}} \pmod n$ is already precomputed, the second step requires (at most) one multiplication modulo n. (No multiplication is required in the second step when all $a_{j,i}$'s are 0.) Therefore the algorithm requires ℓ modular squarings plus $\ell + 2^k$ modular multiplications.

We prove correctness by induction. We claim that after the i-th iteration ($i = \ell - 1, \ldots, 0$), the value of y will be $\prod_{j=1}^{k} x_j^{\sum_{d=0}^{\ell-1-i} 2^d a_{j,i+d}} \pmod n$. When $i = \ell - 1$ (i.e., in the first iteration), y is squared and then multiplied with $x_1^{a_{1,\ell-1}} x_2^{a_{2,\ell-1}} \cdots x_k^{a_{k,\ell-1}} \pmod n$. Since $y = 1$ initially, squaring does not change its value. The multiplication step results in

$$y = \prod_{j=1}^{k} x_j^{a_{j,\ell-1}} \pmod n = \prod_{j=1}^{k} x_j^{\sum_{d=0}^{\ell-1-i} 2^d a_{j,i+d}} \pmod n.$$

Suppose that after m-th iteration ($i = m$), $y = \prod_{j=1}^{k} x_j^{\sum_{d=0}^{\ell-1-m} 2^d a_{j,m+d}} \pmod n$. In the $m-1$-st iteration, following the squaring step, we have

$$y = \prod_{j=1}^{k} x_j^{\sum_{d=0}^{\ell-1-m} 2^{d+1} a_{j,m+d}} \pmod n = \prod_{j=1}^{k} x_j^{\sum_{d=1}^{\ell-m} 2^d a_{j,m+d-1}} \pmod n.$$

The multiplication step changes the value of y to

$$\begin{aligned} y &= \prod_{j=1}^{k} x_j^{\sum_{d=1}^{\ell-m} 2^d a_{j,m+d-1}} \cdot x_1^{a_{1,m-1}} x_2^{a_{2,m-1}} \cdots x_k^{a_{k,m-1}} \pmod n \\ &= \prod_{j=1}^{k} x_j^{\sum_{d=0}^{\ell-m} 2^d a_{j,m+d-1}} \pmod n \\ &= \prod_{j=1}^{k} x_j^{\sum_{d=0}^{\ell-1-(m-1)} 2^d a_{j,(m-1)+d}} \pmod n, \end{aligned}$$

as required. Therefore after the last iteration, we have

$$y = \prod_{j=1}^{k} x_j^{\sum_{d=0}^{\ell-1} 2^d a_{j,d}} \pmod n = \prod_{j=1}^{k} x_j^{e_j} \pmod n.$$

3. (a) Let m_1, m_2 be coprime moduli and let $a_1, a_2 \in \mathbb{Z}$. By the extended gcd algorithm, one can compute $u, v \in \mathbb{Z}$ such that $um_1 + vm_2 = 1$. Prove that $x = um_1a_2 + vm_2a_1 \pmod{m_1m_2}$ is the simultaneous solution of the congruences $x \equiv a_i \pmod{m_i}$ for $i = 1, 2$. **[6]**

Solution: Given that $um_1 + vm_2 = 1$, we have

$$um_1a_2 + vm_2a_1 \equiv um_1a_2 \pmod{m_2} \equiv (1 - vm_2)a_2 \pmod{m_2} \equiv a_2 \pmod{m_2}.$$

Similarly, we can show that $um_1a_2 + vm_2a_1 \equiv a_1 \pmod{m_1}$. Hence $um_1a_2 + vm_2a_1 \pmod{m_1m_2}$ is a solution to the congruences $x \equiv a_i \pmod{m_i}$, $i = 1, 2$.

(b) Let $m_1, m_2, \ldots, m_t$ be pairwise coprime moduli, and $a_1, a_2, \ldots, a_t \in \mathbb{Z}$. Write an incremental procedure for the Chinese remainder theorem that starts with the solution $x \equiv a_1 \pmod{m_1}$ and then runs a loop (for $i = 2, 3, \ldots, t$ in that order), the i-th iteration of which computes the simultaneous solution of $x \equiv a_j \pmod{m_j}$ for $j = 1, 2, \ldots, i$. **[6]**

Solution: Let ExtendedEuclid denote the algorithm that on input a, b returns d, u, v such that $d = \gcd(a, b)$ and $d = ua + vb$. When the inputs are coprime, the first output (i.e., d) is always 1. We now describe an algorithm CRT-V that on input $m[1, \ldots, t]$ and $a[1, \ldots, t]$ with $m[i]$'s being pairwise coprime, outputs a

solution $x \pmod M$ with $M = \prod_{i=1}^{t} m[i]$ simultaneously satisfying the congruences $x \equiv a[i] \pmod{m[i]}$ for $i = 1, \ldots, t$.

CRT-V $(m[1,2,\ldots,t], a[1,2,\ldots,t])$

$y \leftarrow a[1] \pmod{m[1]}$
$M \leftarrow m[1]$
for $i \leftarrow 2$ to t, do
    $(d, u, v) \leftarrow \mathsf{ExtendedEuclid}(M, m[i])$
    $y \leftarrow u \cdot M \cdot a[i] + v \cdot m[i] \cdot y \pmod{M \cdot m[i]}$
    $M \leftarrow M \cdot m[i]$
return y

For convenience we will denote $m[i], a[i]$ as m_i, a_i in what follows.
Claim. *The value of y after k iterations is a simultaneous solution (modulo $\prod_{i=1}^{k} m_i$) to the congruences $x \equiv a_i \pmod{m_i}$ for $i = 1, \ldots, k$.*

Proof. We will prove the claim by induction on k. For $k = 1$, $a[1]$ itself is a solution to $x \equiv a_1 \pmod{m_1}$. Since y is initialised to a_1, the claim is true for $k = 1$. When $k = 2$, the algorithm returns $y = um_1a_2 + v \cdot m_2 \cdot a_1 \pmod{m_1 \cdot m_2}$ where u, v are output by $\mathsf{ExtendedEuclid}(m_1, m_2)$. From part(a), we know that y is a solution and hence the claim holds for $k = 2$. Suppose that after k iterations, y simultaneously satisfies $x \equiv a_i \pmod{m_i}$ for $i = 1, \ldots, k$. At this point $M = \prod_{i=1}^{k} m_i$. In the $(k+1)$-st iteration, u, v returned by $\mathsf{ExtendedEuclid}(M, m_{k+1})$ satisfy $uM + vm_{k+1} = 1$. Also, y is updated to $uMa_{k+1} + vm_{k+1}y \pmod{Mm_{k+1}}$. We now show that this updated value satisfies the congruences $x \equiv a_i \pmod{m_i}$ for $i = 1, \ldots, k+1$. For $1 \le i \le k$, $m_i | M$ and we have

$$\begin{aligned} uMa_{k+1} + vm_{k+1}y \pmod{m_i} &\equiv vm_{k+1}y \pmod{m_i} \\ &\equiv (1 - uM)y \pmod{m_i} \\ &\equiv y \pmod{m_i} \\ &\equiv a_i \pmod{m_i}. \end{aligned}$$

So the updated y still satisfies the congruences $x \equiv a_i \pmod{m_i}$ for $i = 1, \ldots, k$. Also, we have

$$\begin{aligned} uMa_{k+1} + vm_{k+1}y \pmod{m_{k+1}} &\equiv uMa_{k+1} \pmod{m_{k+1}} \\ &\equiv (1 - vm_{k+1})a_{k+1} \pmod{m_{k+1}} \\ &\equiv a_{k+1} \pmod{m_{k+1}} \end{aligned}$$

thus showing that the updated y is a solution to $x \equiv a_{k+1} \pmod{m_{k+1}}$. This concludes the proof of the claim. □

4. Let $f(x)$ be a polynomial with integer coefficients and let $e \in \mathbb{N}$. We have seen how to lift solutions to congruences of the form $f(x) \equiv 0 \pmod{p^e}$ to solutions of $f(x) \equiv 0 \pmod{p^{e+1}}$.
   (a) Let z be a solution of $f(x) \equiv 0 \pmod{p^e}$. Write $z' = z + kp^e$. Show how we can compute all values of k for which z' satisfies $f(z') \equiv 0 \pmod{p^{2e}}$. **[6]**

   Solution: Let $f(x) = a_d x^d + a_{d-1}x^{d-1} + \cdots + a_1 x + a_0$ with $a_i \in \mathbb{Z}$ for $0 \le i \le d$. Then $f'(x) =$

$da_d x^{d-1} + (d-1)a_{d-1}x^{d-2} + \cdots + 2a_2 x + a_1$. For $z' = z + kp^e$, we have

$$\begin{aligned} f(z') &= f(z + kp^e) \\ &= a_d(z+kp^e)^d + a_{d-1}(z+kp^e)^{d-1} + \cdots + a_1(z+kp^e) + a_0 \\ &= f(z) + kp^e(da_d z^{d-1} + (d-1)a_{d-1}z^{d-2} + \cdots + 2a_2 z + a_1) + p^{2e}g(z) \\ &= f(z) + kp^e f'(z) + p^{2e}g(z) \end{aligned}$$

where $g(z)$ is some polynomial in z. Modulo p^{2e}, the above equation reduces to $f(z') \equiv f(z) + kp^e f'(z) \pmod{p^{2e}}$. We need to find values of k for which $f(z') \equiv \pmod{p^{2e}}$, which are given by solutions of

$$kp^e f'(z) \equiv -f(z) \pmod{p^{2e}}.$$

Since $p^e | f(z)$, we can divide by p^e leading to the congruence

$$kf'(z) \equiv -\frac{f(z)}{p^e} \pmod{p^e}.$$

(b) Given that the only solution to $2x^3+4x^2+3 \equiv 0 \pmod{25}$ is $14 \pmod{25}$, use the lifting procedure of Part (a) to compute all the solutions of $2x^3 + 4x^2 + 3 \equiv 0 \pmod{625}$. **[6]**

Solution: Set $f(x) = 2x^3+4x^2+3$, $p = 5$ and $e = 2$ to apply the lifting procedure. We have $f'(x) = 6x^2+8x$ and it is given that $f(14) \equiv 0 \pmod{25}$. In order to find roots of $f(x)$ modulo 625, we need to solve for k in the following congruence: $kf'(z) \equiv -\frac{f(z)}{p^e} \pmod{p^e}$. We have $f'(z) = f'(14) = 1288 \equiv 13 \pmod{25}$ and $f(z)/p^e = 6275/25 = 251 \equiv 1 \pmod{25}$. We now need to solve the congruence $13k \equiv -1 \pmod{25}$. Multiplying by $13^{-1} = 2 \pmod{25}$ yields $k \equiv 2 \cdot (-1) \pmod{25}$ resulting in a unique solution for k given by $23 \pmod{25}$. Substituting for k, we get a unique solution $z + kp^e = 14 + 23 \cdot 25 \equiv 589 \pmod{625}$ for the congruence $f(x) \equiv 0 \pmod{625}$.

5. Let p be an odd prime and $e \in N$.

(a) For $a, b \in \mathbb{Z}_{p^e}^*$, show that $a^2 \equiv b^2 \pmod{p^e}$ if and only if $a \equiv \pm b \pmod{p^e}$. **[6]**

Solution: Suppose $a^2 \equiv b^2 \pmod{p^e}$. Then $p^e | (a^2 - b^2)$ i.e., $p^e | (a+b)(a-b)$. It must be the case that $p^e | (a+b)$ or $p^e | (a-b)$. For otherwise, p divides both $a+b$ and $a-b$, implying that $p | 2a$ which is impossible since $a \in \mathbb{Z}_{p^e}^*$ and p is odd. Hence, $a \equiv \pm b \pmod{p^e}$.

(b) Prove that $|\mathsf{QR}_{p^e}| = \phi(p^e)/2$.
(Here, QR_m denotes the set of quadratic residues modulo m). **[6]**

Solution: Let $a_1, a_2, \ldots, a_{\phi(p^e)}$ be the elements of $\mathbb{Z}_{p^e}^*$ arranged in increasing order so that $a_i \equiv p^e - a_j \pmod{p^e}$ for all $i \in [1, \phi(p^e)/2]$ and $j \in [\phi(p^e)/2 + 1, \phi(p^e)]$. We have $\mathsf{QR}_{p^e} = \{a_1^2, a_2^2, \ldots, a_{\phi(p^e)}^2\}$. From the previous part, we know that $a_i^2 \equiv a_j^2 \pmod{p^e}$ if and only if $a_i \equiv \pm a_j \pmod{p^e}$ for $i, j \in [1, \phi(p^e)]$ and so $|\mathsf{QR}_{p^e}| \leq p^e/2$. Clearly $a_i \neq \pm a_j \pmod{p^e}$, thus implying that $a_i^2 \neq a_j^2 \pmod{p^e}$ for $i \neq j$ and $i, j \leq \phi(p^e)/2$. In other words, the first $\phi(p^e)/2$ squares are distinct, implying that $|\mathsf{QR}_{p^e}| = p^e/2$.

6. (a) Prove that the polynomial $f(x) = x^3 + 2x + 1$ is irreducible over $\mathbb{F}_3[x]$. **[4]**

Solution: We have $f(0) = 1 \equiv 1 \pmod 3$, $f(1) = 4 \equiv 1 \pmod 3$, $f(2) = 13 \equiv 1 \pmod 3$. So $f(x)$ has no roots in $\mathbb{F}_3$ and is hence irreducible in $\mathbb{F}_3[x]$.

(b) Represent $\mathbb{F}_{27} = \mathbb{F}_{3^2}$ by adjoining a root θ of $f(x)$. Let $\alpha = \theta + 1 \in \mathbb{F}_{27}$. Determine whether or not α is primitive or normal. **[8]**

Solution: We have

$$\alpha^2 = \theta^2 + 2\theta + 1$$
$$\alpha^3 = (\theta + 1)(\theta^2 + 2\theta + 1) = \theta^3 + 1 = -2\theta = \theta$$
$$\alpha^{3^2} = (\alpha^3)^3 = \theta^3 = -2\theta - 1 = \theta + 2$$

We have

$$\begin{pmatrix} \alpha \\ \alpha^3 \\ \alpha^{3^2} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 2 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ \theta \\ \theta^2 \end{pmatrix}$$

The matrix above has determinant 0 making $\alpha, \alpha^3, \alpha^{3^2}$ are linearly dependent. Therefore $\alpha = \theta + 1$ is not a normal element.

To determine whether or not α is primitive, we need to check whether or not its order in $\mathbb{F}_{27}$ is 26. Any element in $\mathbb{F}_{27}$ has order dividing 26 i.e., the only possible orders are 1, 2, 13, 26. We know $\alpha^2 = \theta^2 + 2\theta + 1 \neq 1$. It remains to check whether $\operatorname{ord} \alpha = 13$. We have

$$\alpha^{12} = \alpha^3 \alpha^9 = \theta(\theta + 2) = \theta^2 + 2\theta$$
$$\alpha^{13} = (\theta + 1)(\theta^2 + 2\theta) = \theta^3 + 2\theta = -1 = 2 \neq 1$$

We have $\operatorname{ord} \alpha \neq 2$, $\operatorname{ord} \alpha \neq 13$. So α must be a primitive element in $\mathbb{F}_{27}$.