Δημήτριος Κυριακίδης
el17077

Κρυπτογραφία
2º Φυλλάδιο Ασκήσεων

Άσκηση 1:

Αρχικά θα αποδειχθεί ότι, εάν κάποιος αριθμός έχει αντίστροφο ως προς modulo, τότε αυτός είναι μοναδικός στον αντίστοιχο δακτύλιο. Συγκεκριμένα:
Έστων $m, a \in \mathbb{N}$. Αν $\exists a^{-1} \in \mathbb{N}: aa^{-1} \equiv 1 \; mod \; m$, τότε ο a^{-1} είναι μοναδικός $mod \; m$
$\forall \, b: ab \equiv 1 \; mod \; m \Rightarrow b \equiv b(aa^{-1}) \Rightarrow b \equiv (ba)a^{-1} \Rightarrow b \equiv a^{-1}$

Από το μικρό θεώρημα Fermat προκύπτει ότι, για κάθε πρώτο p, ακέραιο α με $p \nmid a$ ισχύει $a^{p-1} \equiv 1 \; mod \; p \Leftrightarrow a^{p-2}a \equiv 1 \; mod \; p$, δηλαδή κάθε ακέραιος α αμοιβαία πρώτος με τον p έχει αντίστροφο ως προς mod p. Επιπλέον, αν $a < p$, η ισοτιμία $a^2 \equiv 1 \; mod \; p$ επαληθεύεται μόνο για $a = 1$ και $a = p - 1$, αφού $a^2 \equiv 1 \; mod \; p \Rightarrow (a^2 - 1)|p \Rightarrow (a - 1)|p \vee (a + 1)|p$.

Επομένως, συνδυάζοντας τα δύο παραπάνω συμπεραίνουμε ότι για κάθε πρώτο p, οι αριθμοί 2 έως και p - 2 μπορούν να διαταχθούν με μοναδικό τρόπο σε ζεύγη αβ τέτοια, ώστε $\alpha\beta \equiv 1 \; mod \; p$. Άρα $1 \cdot 2 \cdot \ldots \cdot (p-2)(p-1) \equiv 1 \cdot (p-1) \equiv -1 \; mod \; p$. Ειδικά για τους 2 και 3 δείχνουμε ότι $(2-1)! = 1 \equiv -1 \; mod \; 2$ και $(3-1)! = 2 \equiv -1 \; mod \; 3$.

Άσκηση 3:

Αναζητούμε λύση στην ισοτιμία $25x \equiv 1 \; mod \; 77$
$\begin{cases} 25x \equiv 1 \; mod \; 7 \\ 25x \equiv 1 \; mod \; 11 \end{cases} \Leftrightarrow \begin{cases} 4x \equiv 1 \; mod \; 7 \\ 3x \equiv 1 \; mod \; 11 \end{cases} \Leftrightarrow \begin{cases} x \equiv 2 \; mod \; 7 \\ x \equiv 4 \; mod \; 11 \end{cases}$
Οι απλοποιήσεις των ισοτιμιών γίνονται μέσω απλούστατων παρατηρήσεων. Στο παραπάνω σύστημα μπορεί να εφαρμοστεί το CRT αφού $(7, 11) = 1$. Άρα έχει μοναδική λύση στον δακτύλιο $\mathbb{Z}_{77}$.
Θεωρούμε $M_1 = \frac{M}{m_1} = \frac{77}{7} = 11$ και αντιστοίχως $M_2 = 7$
$\exists N_1 \in \mathbb{Z}_{m_1}: N_1M_1 \equiv 1 \; mod \; m_1 \Rightarrow 11N_1 \equiv 1 \; mod \; 7 \Rightarrow N_1 = 2$,
αντιστοίχως $7N_2 \equiv 1 \; mod \; 11 \Rightarrow N_2 = 8$
Μία λύση είναι η $y = \sum_{i=1}^{2} N_iM_ia_i = 2 \cdot 11 \cdot 2 + 8 \cdot 7 \cdot 4 = 44 + 224 = 268$, άρα η αντίστοιχη στον δακτύλιο είναι $268 \; mod \; 77 = 37$
$25 \cdot 37 = 925 \equiv 1 \; mod \; 77$

Άσκηση 4:

α.) Έστω G κυκλική ομάδα με γεννήτορα α, και έστω H υποομάδα της G. Αν $H = \{e\}$ τότε $H = \langle e \rangle$ είναι κυκλική. Αν $H \neq \{e\}$ τότε $a^n \in H$ για κάποιον $n \in \mathbb{N}$. Έστω m ο μικρότερος θετικός

ακέραιος για τον οποίο ισχύει $a^m \in H$. Θα αποδειχθεί ότι ο $c = a^m$ είναι γεννήτωρ της H. Πρέπει, συνεπώς, να αποδειχθεί ότι κάθε $b \in H$ είναι δύναμη του c. Αφού $b \in H$ και $H \leq G$, ισχύει $b = a^n$ για κάποιον n. Εφαρμόζοντας ευκλείδεια διαίρεση έχουμε $n = qm + r \Rightarrow$
$a^n = a^{qm+r} \Rightarrow a^n = (a^m)^q \cdot a^r \Rightarrow a^r = (a^m)^{-q} \cdot a^n$
Εφ' όσον $a^n \in H, a^m \in H$, λόγω ιδιοτήτων ομάδας αληθεύουν $(a^m)^{-q} \in H$ και $a^n \in H$.
Άρα $(a^m)^{-q} \cdot a^n \in H$, δηλαδή $a^r \in H$. Όμως, από υπόθεση $m = \min\{i \in \mathbb{Z}^+ : a^i \in H\}$ και επίσης $r < m$, άρα θα πρέπει $r = 0$. Τότε $n = qm$ και $b = a^{qm} = (a^m)^q = c^q$.

<u>Άσκηση 5:</u>

Ο έλεγχος πρώτων αριθμών Fermat για ακέραιο n γίνεται επιλέγοντας τυχαίο $a \in \mathbb{Z}_n$ και εξετάζοντας αν $a^{n-1} \not\equiv 1 \, mod \, n$, οπότε ο n είναι οπωσδήποτε σύνθετος, αλλιώς ίσως είναι πρώτος.

Για την εκτέλεση του ελέγχου πρέπει να γίνονται υπολογισμοί της μορφής $x^y \, mod \, m$ αποδοτικά για μεγάλους αριθμούς. Προς τον σκοπό αυτό θα μπορούσαμε να χρησιμοποιήσουμε την παρακάτω μέθοδο Python που εκτελεί την -γνωστή- απαιτούμενη αυτήν διαδικασία,

```
## (x ^ y) % p
def modexp(x, y, p):
    res = 1

    x %= p
    if(x == 0): return 0

    while(y > 0):
        if(y % 2 == 1): res = (res * x) % p

        y /= 2
        x = (x ** 2) % p

    return res
```

όμως αυτό δεν χρειάζεται, καθώς υπάρχει η προκατασκευασμένη μέθοδος pow, η οποία έχει την ίδια δυνατότητα: https://docs.python.org/3/library/functions.html#pow ("if the third argument mod is present, return base to the power exp, modulo mod, computed more efficiently than pow(base, exp) % m"). Άρα υλοποιούμε τον έλεγχο ως εξής:

```
def singleTest(a, n):
    return pow(a, n - 1, n) == 1

def fermat(n):
    print(f"{n} {('might be' if singleTest(2, n) else 'is not')} prime")
```

Και στην συνέχεια λαμβάνουμε τα αποτελέσματα με διαδοχικές χρήσεις της μεθόδου:

```
fermat(67280421310721)
fermat(170141183460469231731687303715884105721)
fermat(pow(2, 2281) - 1)
fermat(pow(2, 9941) - 1)
fermat(pow(2, 19939) - 1)
```

```
67280421310721 might be prime
170141183460469231731687303715884105721 is not prime
```
2^{2281} - 1 might be prime
2^{9941} - 1 might be prime
2^{19939} - 1 is not prime

Το πρόγραμμα ως έχει θα εκτύπωνε όλους τους αριθμούς ολογράφως. Πρέπει να επισημανθεί ότι για τον έλεγχο έχει χρησιμοποιηθεί ως βάση μόνο το 2 χάριν ταχύτητας. Για πιό αξιόπιστα αποτελέσματα, στις περιπτώσεις όπου το υπόλοιπο προκύπτει 1, μπορεί να επαναληφθεί ο έλεγχος για άλλες τιμές του α, λαμβάνοντας N τυχαία διακριτά δείγματα του $\mathbb{Z}_n$ ως εξής:

```
import random
random.sample(range(1, n - 1), N)
```

Άσκηση 7:

Τα ζητούμενα ψηφία προκύπτουν από την πράξη $1707 \uparrow\uparrow 1783\ mod\ 10^{17}$. Ο υπολογισμός αυτός μπορεί να γίνει με χρήση του θεωρήματος υπολοίπων Euler: αν $(a, m) = 1$ τότε $a^b\ mod\ m = a^{b\ mod\ \varphi(m)}\ mod\ m$. Υλοποιώντας αναδρομικά αυτόν τον τύπο σε Python λαμβάνουμε το αποτέλεσμα 70080500540924243.

```
import sys
from sympy.ntheory.factor_ import totient as phi

sys.setrecursionlimit(4000)

## a^^b mod m
def hyperexp(a, b, m):
    if(b == 1): return a % m
    return pow(a, hyperexp(a, b - 1, phi(m)), m)

print(hyperexp(1707, 1782, 10**17))
```

Ο υπολογισμός των τιμών της συνάρτησης φ του Euler γίνεται με χρήση έτοιμης βιβλιοθήκης, ωστόσο μπορεί να υπολογιστεί ως εξής: $\varphi(n) = n \prod_{i=1}^{k} \left(1 - \frac{1}{p_i}\right)$, όπου $p_1, \dots, p_k$ οι διαφορετικοί πρώτοι παράγοντες του n.

<u>Άσκηση 6:</u>

Χρησιμοποιώντας το ίδιο θεώρημα με την άσκηση 7 προκύπτει:

$$548^{1998000^{100^{10}}} \; mod\; 10^3 = 548^{1998000^{100^{10}} \; mod\; \varphi(10^3)} \; mod\; 10^3 =$$
$$= 548^{1998000^{100^{10}\; mod\; \varphi(\varphi(10^3))} \; mod\; \varphi(10^3)} \; mod\; 10^3 =$$
$$= 548^{1998000^{100^{10}\; mod\; \varphi(40)} \; mod\; 400} \; mod\; 10^3$$
$$= 548^{1998000^{100^{10}\; mod\; 160} \; mod\; 400} \; mod\; 10^3$$

$\varphi(10^3) = 10^3\left(1-\frac{1}{2}\right)\left(1-\frac{1}{5}\right) = 400$ και $\varphi(400) = 400\left(1-\frac{1}{2}\right)\left(1-\frac{1}{5}\right) = 160$

$100^{10} = 100^4 \cdot 100^6 = (2 \cdot 5^2)^4 \cdot 100^6 = 2^4 \cdot 5^8 \cdot 100^6$

Άρα $160|100^{10}$, επομένως

$$548^{1998000^{100^{10}\; mod\; 160} \; mod\; 400} \; mod\; 10^3 = 548^{1998000^0 \; mod\; 400} \; mod\; 10^3 = 548^{1\; mod\; 40} \; mod\; 10^3$$
$$= 548 \; mod\; 10^3 = 548$$

<u>Άσκηση 10:</u>

Εξετάζουμε τις τιμές των μεταβλητών κατά τις διαδοχικές εκτελέσεις του βρόχου παραγωγής ψευδοτυχαίων αριθμών (PRGA):

```
i = 0; j = 0
while next key needed:
      i = (i + 1) mod 256; j = (j + P[i]) mod 256
      swap(P[i], P[j])
      Kₒ = P[(P[i] + P[j]) mod 256]
      return Kₒ
```

Πρώτη εκτέλεση
i = 0, j = 0
i = 1, j = P[1]
swap(P[1], P[P[1]])
K_o = …

Δεύτερη
i = 2, j = (1 + P[2]) mod 256 = 1
swap(P[1], P[2])
K_o = P[(P[1] + P[2]) mod 256] = P[P[2] mod 256] *(αφού P[1] = 0)* = P[P[2]] ≠ 0 γιατί στο P[2] έχει ανατεθεί η προηγούμενη τιμή P[P[1]] όπου P[1] ≠ 2 άρα P[P[2]] ≠ P[2].