Euler–Fermat-tétel és RSA

(feladatsor)

Diszkrét modellek alkalmazásai

1. Adott $m \in \mathbb{N}^+$ modulusra számítsuk ki $\mathbb{Z}_m^*$ elemeit. Írassuk ki ezeket $m = 1$-től 20-ig.

2. Írjunk függvényt, amely kiszámítja $\varphi(n)$-et
   a) egyszerű számlálással; b) a prímtényezős felbontás segítségével.

3. Számítsuk ki kézzel:
   a) $\varphi(25)$; b) $\varphi(27)$; c) $\varphi(30)$; d) $\varphi(36)$; e) $\varphi(45)$; f) $\varphi(81)$; g) $\varphi(100)$; h) $\varphi(600)$.

4. Írjunk függvényt, amely kiszámítja az $a^n \bmod m$ moduláris hatványt az Euler–Fermat-tétel segítségével, ahol $a \in \mathbb{Z}$, $n, m \in \mathbb{N}^+$ és $\text{lnko}(a, m) = 1$.

5. a) Generáljunk egy RSA-kulcspárt. Az egyszerűség kedvéért használhatjuk a következő (egyébként nem biztonságos) beállításokat: prímszámot a `random_prime()` függvénnyel generálhatunk (pl. `random_prime(1000)`), a nyilvános kitevő pedig lehet az első olyan páratlan szám, amely megfelel a feltételeknek.
   b) Titkosítsunk egy bemenő $0 \leq m < n$ üzenetet a nyilvános kulccsal.
   c) Állítsuk vissza a titkosított c értékből az m üzenetet a titkos kulccsal.

6. Írjunk programot, amely megpróbálja feltörni az RSA-t a `factor()` függvény segítségével, azaz csak a nyilvános kulcs ismeretében megpróbálja megoldani az $m^e \equiv c \pmod{n}$ egyenletet m-re, ahol c a feltörendő titkosított üzenet. Törjük fel az előző feladatban titkosított üzenetet.

7. Gyorsítsuk az RSA implementációját a kínai maradéktétel segítségével.

8. a) Írjunk függvényt, amely a Fermat-prímteszt segítségével eldönti, hogy egy adott $n \in \mathbb{N}^+$ szám prímszám-e. (Egyetlen véletlen a-t használjunk.)
   b) Keressünk olyan n-et, amelyre a teszt hibás eredményt ad.
   c) Javítsuk a tesztet úgy, hogy megismételjük k-szor (külön-külön véletlen a-kra), ahol k szintén bemenő paraméter.

9. Keressük meg 1-től 10000-ig az $a = 2$ alaphoz tartozó összes Fermat-álprímet.

10. Keressük meg 1-től 10000-ig az összes Carmichael-számot.

11. Implementáljuk a Miller–Rabin-prímtesztet.

12. Keressük meg 1-től 10000-ig az $a = 2$ alaphoz tartozó összes erős álprímet.