# Криптографија
## - други део -

доц. др Драган Ђокић

Математички факултет, Универзитет у Београду

dragan.djokic@matf.bg.ac.rs

29. фебруар 2024.

# Проблем дискретног логаритма

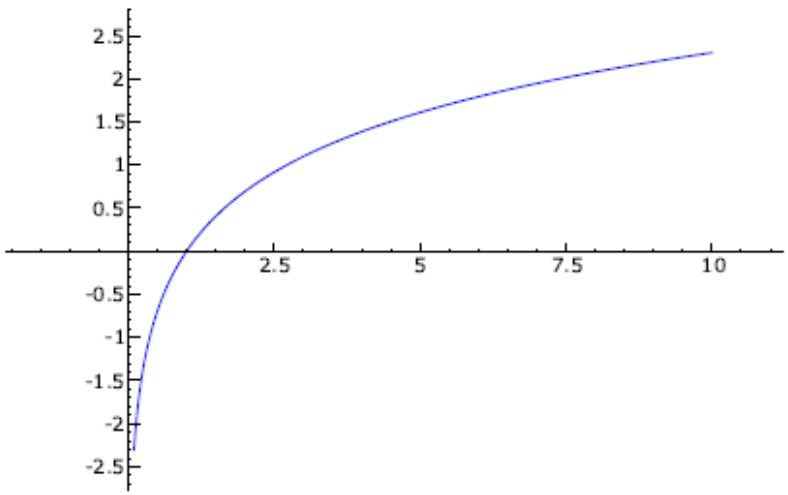
## Дефиниција

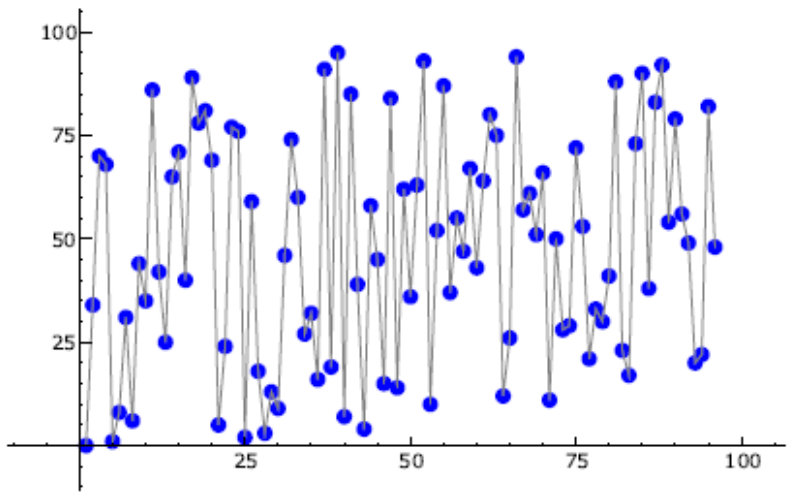
Нека је $G$ група нпр. $\mathbb{F}_q^*$, и нека су $a, g \in G$. Најмањи природан број $n$ (ако постоји) такав да је $a = g^n$ зовемо дискретни логаритам од $a$ у основи $g$ и означавамо са $\log_g a$.

# Проблем дискретног логаритма

## Дефиниција

Нека је $G$ група нпр. $\mathbb{F}_q^*$, и нека су $a, g \in G$. Најмањи природан број $n$ (ако постоји) такав да је $a = g^n$ зовемо дискретни логаритам од $a$ у основи $g$ и означавамо са $\log_g a$.




Класичан логаритам и дискретни $log_2$ у групи $\mathbb{Z}_{53}^*$ (други делује непредвидиво)

- Немамо формулу за израчунавање $n = \log_g a$ у $\mathbb{F}_q^*$

- Немамо формулу за израчунавање $n = \log_g a$ у $\mathbb{F}_q^*$
- Пример: $\log_2 6 = ?$ у групи $\mathbb{Z}_{23}^*$
  (рачунамо $2$, $2^2$, $2^3$, $\ldots$ и чекамо да се појави $6$)

| $n$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $2^n (mod\ 23)$ | 1 | 2 | 4 | 8 | 16 | 9 | 18 | 13 | 3 | 6 | 12 | 1 |
| $n$ | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | |
| $2^n (mod\ 23)$ | 2 | 4 | 8 | 16 | 9 | 18 | 13 | 3 | 6 | 12 | 1 | |

Добићемо да је $\log_2 6 = 20$ али траје предуго

- Немамо формулу за израчунавање $n = \log_g a$ у $\mathbb{F}_q^*$
- Пример: $\log_2 6 = ?$ у групи $\mathbb{Z}_{23}^*$
  (рачунамо $2$, $2^2$, $2^3$, $\ldots$ и чекамо да се појави $6$)

| $n$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $2^n (mod\ 23)$ | 1 | 2 | 4 | 8 | 16 | 9 | 18 | 13 | 3 | 6 | 12 | 1 |
| $n$ | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | |
| $2^n (mod\ 23)$ | 2 | 4 | 8 | 16 | 9 | 18 | 13 | 3 | 6 | 12 | 1 | |

Добићемо да је $\log_2 6 = 20$ али траје предуго

- Најгори случај за $n = \log_g a$ у $\mathbb{F}_q^*$: када је $g$ генератор $n$ велико - практично прођемо целу горњу табелу

- Не постоји брз алгоритам који решава проблем дискретног логаритма у $\mathbb{F}_q^*$

- Не постоји брз алгоритам који решава проблем дискретног логаритма у $\mathbb{F}_q^*$
- Постоје алгоритми који раде ефикасно за неке специфичне $q$ (да ли Дифи-Хелман може да се имплементира тако да не бира такве $q$-ове?)

- Не постоји брз алгоритам који решава проблем дискретног логаритма у $\mathbb{F}_q^*$
- Постоје алгоритми који раде ефикасно за неке специфичне $q$ (да ли Дифи-Хелман може да се имплементира тако да не бира такве $q$-ове?)
- Пример је Полиг-Хелманов алгоритам који предпоставља да су сви прости чиниоци броја $q-1$ „мали“ (детаљније у Живковић, глава 26.2)

# Меси-Омура криптосистем

- Користи се за размену кључева или порука. Ако је дужа порука дели се на блокове

# Меси-Омура криптосистем

- Користи се за размену кључева или порука. Ако је дужа порука дели се на блокове
- Алгоритам:
  - Фиксира се коначно поље $\mathbb{F}_q$ и то је свима познато ($q$ је јавни кључ)

# Меси-Омура криптосистем

- Користи се за размену кључева или порука. Ако је дужа порука дели се на блокове
- Алгоритам:
  - Фиксира се коначно поље $\mathbb{F}_q$ и то је свима познато ($q$ је јавни кључ)
  - И Алиса и Бобан бирају своје тајне кључеве $e_A$ и $e_B$ тд. НЗД $(e_A, q-1) =$ НЗД $(e_B, q-1) = 1$

# Меси-Омура криптосистем

- Користи се за размену кључева или порука. Ако је дужа порука дели се на блокове
- Алгоритам:
  - Фиксира се коначно поље $\mathbb{F}_q$ и то је свима познато ($q$ је јавни кључ)
  - И Алиса и Бобан бирају своје тајне кључеве $e_A$ и $e_B$ тд. НЗД $(e_A, q-1) =$ НЗД $(e_B, q-1) = 1$
  - Свако од њих рачуна свој тајни кључ $d_i \equiv e_i^{-1} \pmod{q-1}$, $i \in \{A, B\}$

# Меси-Омура криптосистем

- Користи се за размену кључева или порука. Ако је дужа порука дели се на блокове
- Алгоритам:
  - Фиксира се коначно поље $\mathbb{F}_q$ и то је свима познато ($q$ је јавни кључ)
  - И Алиса и Бобан бирају своје тајне кључеве $e_A$ и $e_B$ тд. НЗД $(e_A, q-1) =$ НЗД $(e_B, q-1) = 1$
  - Свако од њих рачуна свој тајни кључ $d_i \equiv e_i^{-1} \pmod{q-1}$, $i \in \{A, B\}$
  - Ако је $M$ блок поруке (кодиран елементом поља $\mathbb{F}_q$) коју треба послати Алиса рачуна $M^{e_A}$ и шаље Бобану

# Меси-Омура криптосистем

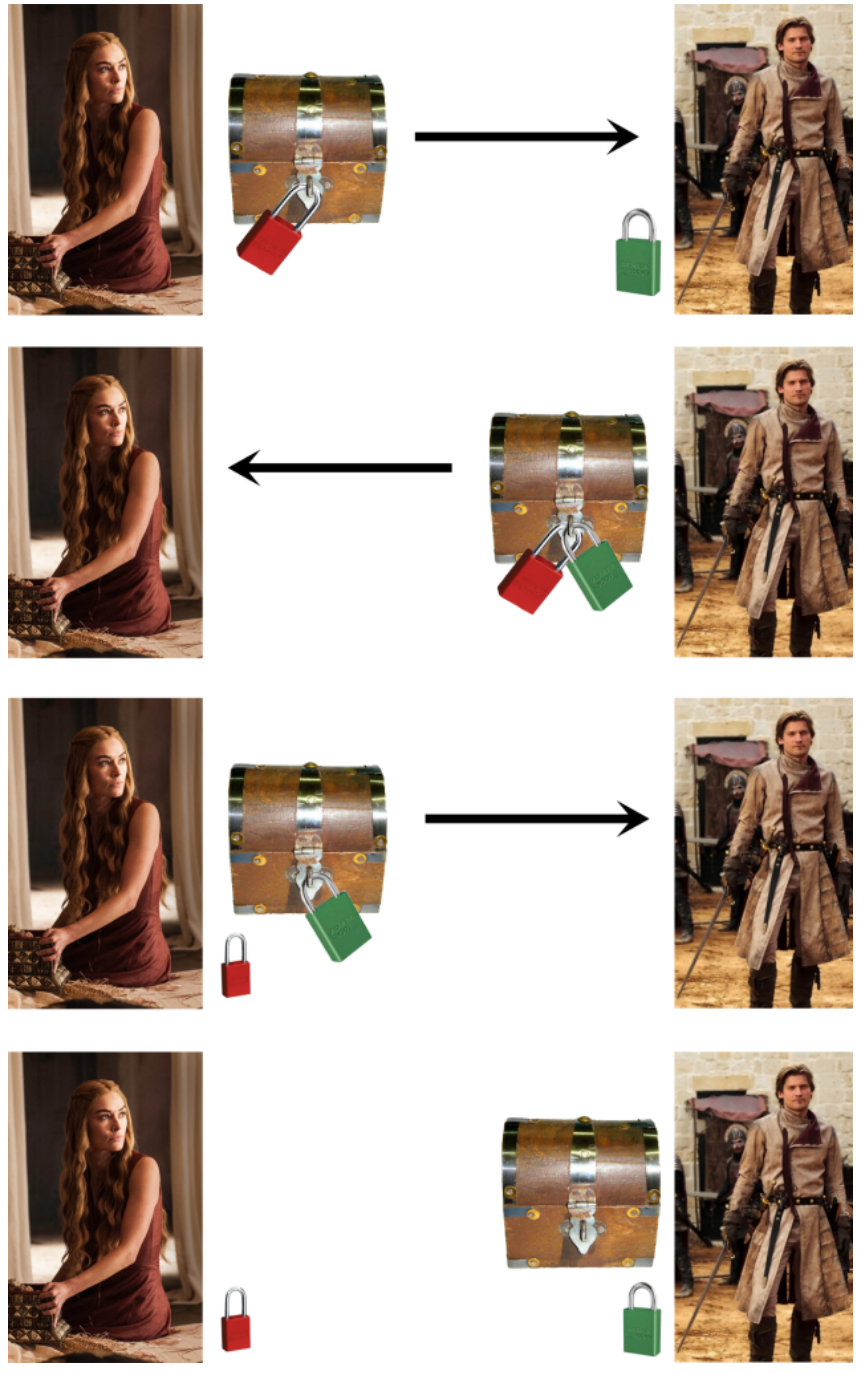
- Користи се за размену кључева или порука. Ако је дужа порука дели се на блокове
- Алгоритам:
  - Фиксира се коначно поље $\mathbb{F}_q$ и то је свима познато ($q$ је јавни кључ)
  - И Алиса и Бобан бирају своје тајне кључеве $e_A$ и $e_B$ тд. НЗД $(e_A, q-1) =$ НЗД $(e_B, q-1) = 1$
  - Свако од њих рачуна свој тајни кључ $d_i \equiv e_i^{-1} \pmod{q-1}$, $i \in \{A, B\}$
  - Ако је $M$ блок поруке (кодиран елементом поља $\mathbb{F}_q$) коју треба послати Алиса рачуна $M^{e_A}$ и шаље Бобану
  - Бобан не може да прочита $M^{e_A}$, али може да израчуна $(M^{e_A})^{e_B} = M^{e_A e_B}$ и пошаље назад Алиси

# Меси-Омура криптосистем

- Користи се за размену кључева или порука. Ако је дужа порука дели се на блокове
- Алгоритам:
  - Фиксира се коначно поље $\mathbb{F}_q$ и то је свима познато ($q$ је јавни кључ)
  - И Алиса и Бобан бирају своје тајне кључеве $e_A$ и $e_B$ тд. НЗД $(e_A, q-1) =$ НЗД $(e_B, q-1) = 1$
  - Свако од њих рачуна свој тајни кључ $d_i \equiv e_i^{-1} \pmod{q-1}$, $i \in \{A, B\}$
  - Ако је $M$ блок поруке (кодиран елементом поља $\mathbb{F}_q$) коју треба послати Алиса рачуна $M^{e_A}$ и шаље Бобану
  - Бобан не може да прочита $M^{e_A}$, али може да израчуна $(M^{e_A})^{e_B} = M^{e_A e_B}$ и пошаље назад Алиси
  - Сада Алиса рачуна $(M^{e_A e_B})^{d_A} = M^{e_B}$ и шаље опет Бобану. Овде се користи $e_A d_A \equiv 1 \pmod{q-1}$ што повлачи $M^{e_A d_A} = M$ у $\mathbb{F}_q$

# Меси-Омура криптосистем

- Користи се за размену кључева или порука. Ако је дужа порука дели се на блокове
- Алгоритам:
  - Фиксира се коначно поље $\mathbb{F}_q$ и то је свима познато ($q$ је јавни кључ)
  - И Алиса и Бобан бирају своје тајне кључеве $e_A$ и $e_B$ тд. НЗД $(e_A, q-1) =$ НЗД $(e_B, q-1) = 1$
  - Свако од њих рачуна свој тајни кључ $d_i \equiv e_i^{-1} \pmod{q-1}$, $i \in \{A, B\}$
  - Ако је $M$ блок поруке (кодиран елементом поља $\mathbb{F}_q$) коју треба послати Алиса рачуна $M^{e_A}$ и шаље Бобану
  - Бобан не може да прочита $M^{e_A}$, али може да израчуна $(M^{e_A})^{e_B} = M^{e_A e_B}$ и пошаље назад Алиси
  - Сада Алиса рачуна $(M^{e_A e_B})^{d_A} = M^{e_B}$ и шаље опет Бобану. Овде се користи $e_A d_A \equiv 1 \pmod{q-1}$ што повлачи $M^{e_A d_A} = M$ у $\mathbb{F}_q$
  - Бобан рачуна $(M^{e_B})^{d_B} = M$ и долази до почетне поруке.



- Ако Цица пресретне комуникацију највише што може да зна је $M^{e_A}$, $M^{e_B}$ и $M^{e_A e_B}$ и јавни кључ $q$.

- Ако Цица пресретне комуникацију највише што може да зна је $M^{e_A}$, $M^{e_B}$ и $M^{e_A e_B}$ и јавни кључ $q$.
- Да би дошла до информације $M$ мора да израчуна:
  - $e_A = \log_{M^{e_B}} \left(M^{e_A e_B}\right)$ дискретни логаритам
  - $d_A \equiv e_A^{-1} \pmod{q-1}$
  - $M = \left(M^{e_A}\right)^{d_A}$

Показаћемо како се рачуна мултипликативни инверз $a^{-1}$ по модулу $b$

- Еуклидов алгоритам (у $\mathbb{N}$) даје $ma + nb = 1 = \text{НЗД}(a, b)$ за неке $m, n \in \mathbb{Z}$
- Следи $ma \equiv 1 \pmod{b}$, тј. $a^{-1} = m \bmod b$

Показаћемо како се рачуна мултипликативни инверз $a^{-1}$ по модулу $b$

- Еуклидов алгоритам (у $\mathbb{N}$) даје $ma + nb = 1 = \text{НЗД}(a, b)$ за неке $m, n \in \mathbb{Z}$
- Следи $ma \equiv 1 \pmod{b}$, тј. $a^{-1} = m \bmod b$

Пример: $421^{-1} \equiv 281 \pmod{676}$

$$
\begin{aligned}
676 &= 1 \cdot 421 + 255 \\
421 &= 1 \cdot 255 + 166 \\
255 &= 1 \cdot 166 + 89 \\
166 &= 1 \cdot 89 + 77 \\
89 &= 1 \cdot 77 + 12 \\
77 &= 6 \cdot 12 + 5 \\
12 &= 2 \cdot 5 + 2 \\
5 &= 2 \cdot 2 + 1
\end{aligned}
\qquad
\begin{aligned}
1 &= 5 - 2 \cdot (12 - 2 \cdot 5) \\
&= -2 \cdot 12 + 5 \cdot (77 - 6 \cdot 12) \\
&= 5 \cdot 77 - 32 \cdot (89 - 77) \\
&= -32 \cdot 89 + 37 \cdot (166 - 89) \\
&= 37 \cdot 166 - 69 \cdot (255 - 166) \\
&= -69 \cdot 255 + 106 \cdot (421 - 255) \\
&= 106 \cdot 421 - 175 \cdot (676 - 421) \\
&= -175 \cdot 676 + \underline{281} \cdot 421
\end{aligned}
$$

Пример: $255^{-1} \equiv -281 \equiv 395 \pmod{676}$

$$
\begin{aligned}
676 &= 2 \cdot 255 + 166 \\
255 &= 1 \cdot 166 + 89 \\
166 &= 1 \cdot 89 + 77 \\
89 &= 1 \cdot 77 + 12 \\
77 &= 6 \cdot 12 + 5 \\
12 &= 2 \cdot 5 + 2 \\
5 &= 2 \cdot 2 + 1
\end{aligned}
$$

$$
\begin{aligned}
1 &= 5 - 2 \cdot (12 - 2 \cdot 5) \\
&= -2 \cdot 12 + 5 \cdot (77 - 6 \cdot 12) \\
&= 5 \cdot 77 - 32 \cdot (89 - 77) \\
&= -32 \cdot 89 + 37 \cdot (166 - 89) \\
&= 37 \cdot 166 - 69 \cdot (255 - 166) \\
&= -69 \cdot 255 + 106 \cdot (676 - 2 \cdot 255) \\
&= 106 \cdot 676 \underline{-281} \cdot 255
\end{aligned}
$$

Рекурзивна python функција која за целе бројеве $a$ и $b$ враћа $d = \text{НЗД}(a, b)$ и целе бројеве $m$ и $n$ такве да је $d = ma + nb$:

```
def extended_gcd(a, b):
    if b == 0:
        return (a, 1, 0)
    d, m, n = extended_gcd(b, a % b)
    return (d, n, m - a // b * n)
```

Рекурзивна python функција која за целе бројеве $a$ и $b$ враћа $d = \text{НЗД}(a, b)$ и целе бројеве $m$ и $n$ такве да је $d = ma + nb$:

```
def extended_gcd(a, b):
    if b == 0:
        return (a, 1, 0)
    d, m, n = extended_gcd(b, a % b)
    return (d, n, m - a // b * n)
```

Функција која за целе бројеве $a$ и $b$ враћа $a^{-1}$ по модулу $b$ (уколико постоји):

```
def mod_inv(a, b):
    d, m, n = extended_gcd(a, b)
    if d != 1:
        print("Vrednosti a i x nisu uzajamno proste!")
    else:
        return m % b
```

# Пример: Помоћу Меси-Омура криптосистема са јавним кључем $q = 677$ (прост) треба послати $M = 470$

Алиса бира тајни кључ $e_A = 255$ и $d_A = 395$. Бобан бира тајни кључ $e_B = 421$ и $d_B = 281$ (претх. примери)

$$M^{e_A} = 470^{255} (mod\ 677) = 470^{128} \cdot 470^{64} \cdot 470^{32} \cdot 470^{16} \cdot 470^{8} \cdot 470^{4} \cdot 470^{2} \cdot 470^{1} (mod\ 677)$$

$$470^2 \equiv 198$$

$$470^4 \equiv 198^2 \equiv 615 \equiv -62$$

$$470^8 \equiv (-62)^2 \equiv 459$$

$$470^{16} \equiv 459^2 \equiv 134$$

$$470^{32} \equiv 134^2 \equiv 354$$

$$470^{64} \equiv 354^2 \equiv 71$$

$$470^{128} \equiv 71^2 \equiv 302$$

$$M^{e_A} = 302 \cdot 71 \cdot 354 \cdot 134 \cdot 459 \cdot 615 \cdot 198 \cdot 470 (mod\ 677) = 292 (mod\ 677)$$

Алиса шаље Бобану 292

$$
\begin{aligned}
M^{e_A e_B} &= 292^{421}(mod\ 677) = 292^{256} \cdot 292^{128} \cdot 292^{32} \cdot 292^{4} \cdot 292^{1}(mod\ 677) \\
& 292^4 \equiv (292^2)^2 \equiv 639^2 \equiv (-38)^2 \equiv 90 \\
& 292^{32} \equiv (292^4)^8 \equiv (90^2)^4 \equiv 653^4 \equiv ((-24)^2)^2 \equiv 576^2 \equiv 46 \\
& 292^{128} \equiv (292^{32})^4 \equiv (46^2)^2 \equiv 85^2 \equiv 455 \\
& 292^{256} \equiv (292^{128})^2 \equiv 455^2 \equiv 540 \\
M^{e_A e_B} &= 540 \cdot 455 \cdot 46 \cdot 90 \cdot 292(mod\ 677) = 156
\end{aligned}
$$

Бобан шаље Алиси $156$

$$
\begin{aligned}
M^{e_A e_B d_A} &= 156^{395}(mod\ 677) = 156^{256} \cdot 156^{128} \cdot 156^{8} \cdot 156^{2} \cdot 156^{1}(mod\ 677) \\
& 156^2 \equiv 641 \equiv -36 \\
& 156^8 \equiv (156^2)^4 \equiv ((-36)^2)^2 = 619^2 \equiv 656 \equiv -21 \\
& 156^{128} \equiv (156^8)^{16} \equiv (441^2)^4 \equiv (182^2)^2 \equiv 628^2 \equiv 370 \\
& 156^{256} \equiv (156^{128})^2 \equiv 370^2 \equiv 146 \\
M^{e_A e_B d_A} &= 146 \cdot 370 \cdot (-21) \cdot (-36) \cdot 156(mod\ 677) = 313(mod\ 677)
\end{aligned}
$$

Алиса шаље Бобану $313$

$$
\begin{aligned}
M^{e_A e_B d_A d_B} &= 313^{281}(mod\ 677) = 313^{256} \cdot 313^{16} \cdot 313^{8} \cdot 313^{1}(mod\ 677) \\
& 313^{8} \equiv (313^{2})^{4} \equiv (481^{2})^{2} \equiv 504^{2} \equiv 141 \\
& 313^{16} \equiv (313^{8})^{2} \equiv 141^{2} \equiv 248 \\
& 313^{256} \equiv (313^{16})^{16} \equiv (248^{2})^{8} \equiv (574^{2})^{4} \equiv (454^{2})^{2} \equiv 308^{2} \equiv 84 \\
M^{e_A e_B d_A d_B} &= 84 \cdot 248 \cdot 141 \cdot 313(mod\ 677) = 470(mod\ 677)
\end{aligned}
$$

Бобан коначно добија поруку $470$

Класа која за дат прост број $q$ генерише тајне кључеве $e$ и $d$ (користи функције за степеновање и инверз од раније):

```
class Massey_Omura:

    def __init__(self, q):
        self.q = q

        while True:
            e = random.randrange(2, q-1)
            gcd, _, _ = extended_gcd(e, q-1)

            if gcd == 1:
                self.e = e
                break

        self.d = mod_inv(e, q-1)


    def encrypt(self, k):
        return mod_pow(k, self.e, self.q)


    def decrypt(self, k):
        return mod_pow(k, self.d, self.q)
```

# Тест програм:

```
def main():
    A = Massey_Omura(677)
    B = Massey_Omura(677)

    print(A.e, A.d)
    print(B.e, B.d)

    # Kljuc koji je odabrala osoba A
    k = 349

    # A sifruje
    k_ea = A.encrypt(k)

    # B sifruje
    k_ea_eb = B.encrypt(k_ea)

    # A desifruje
    k_eada_eb = A.decrypt(k_ea_eb)

    # B desifruje i dobija kljuc koji je osoba A odabrala
    k_eada_ebdb = B.decrypt(k_eada_eb)

    # Vrednost kljuca koju je B dobio
    print(k_eada_ebdb)
```

# ЕлГамалов криптосистем

- Јавни кључ $q$ (степен простог броја) и $g \in \mathbb{F}_q^*$ генератор

# ЕлГамалов криптосистем

- Јавни кључ $q$ (степен простог броја) и $g \in \mathbb{F}_q^*$ генератор
- Бобан бира свој тајни кључ $e_B$ и помоћу њега прави јавни кључ $g^{e_B}$ који шаље Алиси тј. објављује (као код Дифи-Хелмана)

# ЕлГамалов криптосистем

- Јавни кључ $q$ (степен простог броја) и $g \in \mathbb{F}_q^*$ генератор
- Бобан бира свој тајни кључ $e_B$ и помоћу њега прави јавни кључ $g^{e_B}$ који шаље Алиси тј. објављује (као код Дифи-Хелмана)
- $M \in \mathbb{F}_q$ кодирани блок (део поруке) коју Алиса жели да пошаље, она генерише случајан природан број $k < q$ који ће користити само једном за блок $M$ (за наредно $M$ бира ново $k$)

# ЕлГамалов криптосистем

- Јавни кључ $q$ (степен простог броја) и $g \in \mathbb{F}_q^*$ генератор
- Бобан бира свој тајни кључ $e_B$ и помоћу њега прави јавни кључ $g^{e_B}$ који шаље Алиси тј. објављује (као код Дифи-Хелмана)
- $M \in \mathbb{F}_q$ кодирани блок (део поруке) коју Алиса жели да пошаље, она генерише случајан природан број $k < q$ који ће користити само једном за блок $M$ (за наредно $M$ бира ново $k$)
- Алиса шаље Бобану пар информација $g^k$ и $Mg^{e_B k} = M\left(g^{e_B}\right)^k$ (ово лако рачуна јер зна $g$, $g^{e_B}$, $k$ и $M$)

# ЕлГамалов криптосистем

- Јавни кључ $q$ (степен простог броја) и $g \in \mathbb{F}_q^*$ генератор
- Бобан бира свој тајни кључ $e_B$ и помоћу њега прави јавни кључ $g^{e_B}$ који шаље Алиси тј. објављује (као код Дифи-Хелмана)
- $M \in \mathbb{F}_q$ кодирани блок (део поруке) коју Алиса жели да пошаље, она генерише случајан природан број $k < q$ који ће користити само једном за блок $M$ (за наредно $M$ бира ново $k$)
- Алиса шаље Бобану пар информација $g^k$ и $Mg^{e_B k} = M\left(g^{e_B}\right)^k$ (ово лако рачуна јер зна $g$, $g^{e_B}$, $k$ и $M$)
- Бобан рачуна $g^{e_B k} = \left(g^k\right)^{e_B}$, затим његов инверз и добија $M = Mg^{e_B k}\left(g^{e_B k}\right)^{-1}$

# ЕлГамалов криптосистем

- Јавни кључ $q$ (степен простог броја) и $g \in \mathbb{F}_q^*$ генератор
- Бобан бира свој тајни кључ $e_B$ и помоћу њега прави јавни кључ $g^{e_B}$ који шаље Алиси тј. објављује (као код Дифи-Хелмана)
- $M \in \mathbb{F}_q$ кодирани блок (део поруке) коју Алиса жели да пошаље, она генерише случајан природан број $k < q$ који ће користити само једном за блок $M$ (за наредно $M$ бира ново $k$)
- Алиса шаље Бобану пар информација $g^k$ и $Mg^{e_B k} = M\left(g^{e_B}\right)^k$ (ово лако рачуна јер зна $g$, $g^{e_B}$, $k$ и $M$)
- Бобан рачуна $g^{e_B k} = \left(g^k\right)^{e_B}$, затим његов инверз и добија $M = Mg^{e_B k}\left(g^{e_B k}\right)^{-1}$
- Цица мора да реши проблем дискретног логаритма да би урадила претходни корак

# ЕлГамалов криптосистем

- Јавни кључ $q$ (степен простог броја) и $g \in \mathbb{F}_q^*$ генератор
- Бобан бира свој тајни кључ $e_B$ и помоћу њега прави јавни кључ $g^{e_B}$ који шаље Алиси тј. објављује (као код Дифи-Хелмана)
- $M \in \mathbb{F}_q$ кодирани блок (део поруке) коју Алиса жели да пошаље, она генерише случајан природан број $k < q$ који ће користити само једном за блок $M$ (за наредно $M$ бира ново $k$)
- Алиса шаље Бобану пар информација $g^k$ и $Mg^{e_B k} = M\left(g^{e_B}\right)^k$ (ово лако рачуна јер зна $g$, $g^{e_B}$, $k$ и $M$)
- Бобан рачуна $g^{e_B k} = \left(g^k\right)^{e_B}$, затим његов инверз и добија $M = Mg^{e_B k}\left(g^{e_B k}\right)^{-1}$
- Цица мора да реши проблем дискретног логаритма да би урадила претходни корак

Напомена: Бобан само једном шаље $e_B$ Алиси (на почетку), код сваког блока имамо једну размену (Алиса шаље пар $\left(g^k, Mg^{e_B k}\right)$ Бобану)

# Генератор случајног простог броја

- У свим претходним алгоритмима: изабрати кључ = користити генератор случајних бројева

# Генератор случајног простог броја

- У свим претходним алгоритмима: изабрати кључ = користити генератор случајних бројева
- Знамо како ради генератор (псеудо)случајних природних бројева

# Генератор случајног простог броја

- У свим претходним алгоритмима: изабрати кључ = користити генератор случајних бројева
- Знамо како ради генератор (псеудо)случајних природних бројева
- Али како се генерише случајан прост број?

# Генератор случајног простог броја

- У свим претходним алгоритмима: изабрати кључ = користити генератор случајних бројева
- Знамо како ради генератор (псеудо)случајних природних бројева
- Али како се генерише случајан прост број?
- Насумично изабран број вероватно није прост

# Генератор случајног простог броја

- У свим претходним алгоритмима: изабрати кључ = користити генератор случајних бројева
- Знамо како ради генератор (псеудо)случајних природних бројева
- Али како се генерише случајан прост број?
- Насумично изабран број вероватно није прост
- Не постоји формула за $n$-ти прост број $p_n = \ldots$

# Генератор случајног простог броја

- У свим претходним алгоритмима: изабрати кључ = користити генератор случајних бројева
- Знамо како ради генератор (псеудо)случајних природних бројева
- Али како се генерише случајан прост број?
- Насумично изабран број вероватно није прост
- Не постоји формула за $n$-ти прост број $p_n = \ldots$
- Принцип рада генератора:
  - изабере се непаран (велики) псеудослучајан број $n$
  - затим се прост број тражи у низу $n$, $n+2$, $n+4$, $n+6$, $\ldots$

# Генератор случајног простог броја

- У свим претходним алгоритмима: изабрати кључ = користити генератор случајних бројева
- Знамо како ради генератор (псеудо)случајних природних бројева
- Али како се генерише случајан прост број?
- Насумично изабран број вероватно није прост
- Не постоји формула за $n$-ти прост број $p_n = \ldots$
- Принцип рада генератора:
  - изабере се непаран (велики) псеудослучајан број $n$
  - затим се прост број тражи у низу $n$, $n+2$, $n+4$, $n+6$, $\ldots$
  - треба нам ефикасан начин да проверимо да ли је неки број прост. Елементарно решето је споро - временска сложеност $O(\sqrt{n})$.

# Тестови прималности

Направљени тако да

- ако број $n$ падне на тесту онда је $n$ сложен
- ако број $n$ прође тест он може (али не мора) да буде прост

# Тестови прималности

Направљени тако да

- ако број $n$ падне на тесту онда је $n$ сложен
- ако број $n$ прође тест он може (али не мора) да буде прост
- $v = \frac{\text{card}\{n \in [1,N] \mid n \text{ је прост}\}}{\text{card}\{n \in [1,N] \mid n \text{ је прошао тест}\}}$ је вероватноћа да је број који прошао тест прост
- веће $v$ - бољи тест.

# Тестови прималности

Направљени тако да

- ако број $n$ падне на тесту онда је $n$ сложен
- ако број $n$ прође тест он може (али не мора) да буде прост
- $v = \frac{\text{card}\{n \in [1,N] \mid n \text{ је прост}\}}{\text{card}\{n \in [1,N] \mid n \text{ је прошао тест}\}}$ је вероватноћа да је број који прошао тест прост
- веће $v$ - бољи тест.
- Тест обично зависи од неких параметара. Понављање теста за разне (независне) параметре повећава повећава вероватноћу да је број који је преживео сва тестирања заиста прост

# Кармајклови бројеви

- Мала Фермаова теорема: $(\bigstar)$ $a^{n-1} \equiv 1 \pmod{n}$, за $n$ прост и НЗД$(a, n) = 1$

# Кармајклови бројеви

- Мала Фермаова теорема: $(\bigstar)$ $a^{n-1} \equiv 1 \pmod{n}$, за $n$ прост и НЗД$(a, n) = 1$
- Али није немогуће да $(\bigstar)$ важи и уколико $n$ није прост

# Кармајклови бројеви

- Мала Фермаова теорема: $(\bigstar)$ $a^{n-1} \equiv 1 \pmod{n}$, за $n$ прост и НЗД$(a, n) = 1$
- Али није немогуће да $(\bigstar)$ важи и уколико $n$ није прост

## Дефиниција

Ако за пародан број $a$ и сложен број $n$ тд. НЗД$(a, n) = 1$ важи $(\bigstar)$ кажемо да је $n$ псеудопрост број у бази $a$.

# Кармајклови бројеви

- Мала Фермаова теорема: $(\bigstar)$ $a^{n-1} \equiv 1 \pmod{n}$, за $n$ прост и НЗД$(a, n) = 1$
- Али није немогуће да $(\bigstar)$ важи и уколико $n$ није прост

## Дефиниција

Ако за пародан број $a$ и сложен број $n$ тд. НЗД$(a, n) = 1$ важи $(\bigstar)$ кажемо да је $n$ псеудопрост број у бази $a$.

Пример: Број $n = 91 = 7 \cdot 13$ је псеудопрост у бази $3$ јер је $3^{90} \equiv 1 \pmod{91}$, али није псеудопрост у бази $2$ јер је $2^{90} \equiv 64 \pmod{91}$

# Кармајклови бројеви

- Мала Фермаова теорема: $(\bigstar)$ $a^{n-1} \equiv 1 \pmod{n}$, за $n$ прост и НЗД$(a, n) = 1$
- Али није немогуће да $(\bigstar)$ важи и уколико $n$ није прост

## Дефиниција

Ако за пародан број $a$ и сложен број $n$ тд. НЗД$(a, n) = 1$ важи $(\bigstar)$ кажемо да је $n$ псеудопрост број у бази $a$.

Пример: Број $n = 91 = 7 \cdot 13$ је псеудопрост у бази $3$ јер је $3^{90} \equiv 1 \pmod{91}$, али није псеудопрост у бази $2$ јер је $2^{90} \equiv 64 \pmod{91}$

- Ојлерова теорема: $a^{\varphi(n)} \equiv 1 \pmod{n}$.

# Кармајклови бројеви

- Мала Фермаова теорема: $(\bigstar)$ $a^{n-1} \equiv 1 \pmod{n}$, за $n$ прост и НЗД$(a, n) = 1$
- Али није немогуће да $(\bigstar)$ важи и уколико $n$ није прост

## Дефиниција

Ако за пародан број $a$ и сложен број $n$ тд. НЗД$(a, n) = 1$ важи $(\bigstar)$ кажемо да је $n$ псеудопрост број у бази $a$.

Пример: Број $n = 91 = 7 \cdot 13$ је псеудопрост у бази $3$ јер је $3^{90} \equiv 1 \pmod{91}$, али није псеудопрост у бази $2$ јер је $2^{90} \equiv 64 \pmod{91}$

- Ојлерова теорема: $a^{\varphi(n)} \equiv 1 \pmod{n}$.
- Ако је $n$ псеудопрост у бази $a$ мора бити $a^{n-\varphi(n)-1} \equiv 1 \pmod{n}$

# Кармајклови бројеви

- Мала Фермаова теорема: $(\bigstar)$ $a^{n-1} \equiv 1 \pmod{n}$, за $n$ прост и НЗД$(a, n) = 1$
- Али није немогуће да $(\bigstar)$ важи и уколико $n$ није прост

## Дефиниција

Ако за пародан број $a$ и сложен број $n$ тд. НЗД$(a, n) = 1$ важи $(\bigstar)$ кажемо да је $n$ псеудопрост број у бази $a$.

Пример: Број $n = 91 = 7 \cdot 13$ је псеудопрост у бази $3$ јер је $3^{90} \equiv 1 \pmod{91}$, али није псеудопрост у бази $2$ јер је $2^{90} \equiv 64 \pmod{91}$

- Ојлерова теорема: $a^{\varphi(n)} \equiv 1 \pmod{n}$.
- Ако је $n$ псеудопрост у бази $a$ мора бити $a^{n-\varphi(n)-1} \equiv 1 \pmod{n}$
- $n - \varphi(n) - 1$ је обично много мањи од $n - 1$

# Кармајклови бројеви

- Мала Фермаова теорема: $(\bigstar)$ $a^{n-1} \equiv 1 \pmod{n}$, за $n$ прост и НЗД$(a, n) = 1$
- Али није немогуће да $(\bigstar)$ важи и уколико $n$ није прост

## Дефиниција

Ако за пародан број $a$ и сложен број $n$ тд. НЗД$(a, n) = 1$ важи $(\bigstar)$ кажемо да је $n$ псеудопрост број у бази $a$.

Пример: Број $n = 91 = 7 \cdot 13$ је псеудопрост у бази $3$ јер је $3^{90} \equiv 1 \pmod{91}$, али није псеудопрост у бази $2$ јер је $2^{90} \equiv 64 \pmod{91}$

- Ојлерова теорема: $a^{\varphi(n)} \equiv 1 \pmod{n}$.
- Ако је $n$ псеудопрост у бази $a$ мора бити $a^{n-\varphi(n)-1} \equiv 1 \pmod{n}$
- $n - \varphi(n) - 1$ је обично много мањи од $n - 1$
- У нашем примеру $91 - \varphi(91) - 1 = 18$, па базе $a$ у којима је $91$ псеудопрост треба тражити међу $a^{18} \equiv 1 \pmod{91}$

## Теорема

1. Ако је $n$ псеудопрост и у бази $a$ и у $b$ онда је псеудопрост и у бази $ab$
2. Ако је $n$ псеудопрост у бази $a$, али није псеудопрост у $b$ онда није псеудопрост ни у бази $ab$
3. Ако је $n$ није псеудопрост у бази $a$ онда није псеудопрост ни у бази $b$, за бар пола $b$-ова из $\mathbb{Z}_n^* = \{b \in \mathbb{Z}_n \mid \text{НЗД}(b, n) = 1\}$

## Теорема

1. Ако је $n$ псеудопрост и у бази $a$ и у $b$ онда је псеудопрост и у бази $ab$
2. Ако је $n$ псеудопрост у бази $a$, али није псеудопрост у $b$ онда није псеудопрост ни у бази $ab$
3. Ако је $n$ није псеудопрост у бази $a$ онда није псеудопрост ни у бази $b$, за бар пола $b$-ова из $\mathbb{Z}_n^* = \{b \in \mathbb{Z}_n \mid \text{НЗД}(b, n) = 1\}$

Доказ:

1. $(ab)^{n-1} = a^{n-1}b^{n-1} \equiv 1 \cdot 1 \pmod{n}$
2. $(ab)^{n-1} = a^{n-1}b^{n-1} \equiv b^{n-1} \not\equiv 1 \pmod{n}$
3. За сваки базу $c$ у којој је $n$ псеудопрост постоји база $b = ca$ у којој $n$ није псеудопрост (према 2.)

## Дефиниција

Кармајклов број $n$ је сложен број који је псеудопрост у свакој бази $a \in \mathbb{Z}_n^*$

## Дефиниција

Кармајклов број $n$ је сложен број који је псеудопрост у свакој бази $a \in \mathbb{Z}_n^*$

- По претх. теореми вероватноћа да број $n$ који није ни прост ни Кармајклов прође тест $(\bigstar)$
  - у једном тестирању са случајно изабраном базом је највише $\frac{1}{2}$

## Дефиниција

Кармајклов број $n$ је сложен број који је псеудопрост у свакој бази $a \in \mathbb{Z}_n^*$

- По претх. теореми вероватноћа да број $n$ који није ни прост ни Кармајклов прође тест ($\bigstar$)
  - у једном тестирању са случајно изабраном базом је највише $\frac{1}{2}$
  - у $k$ тестирања са случајно и независно изабраним базама је највише $\frac{1}{2^k}$

## Дефиниција

Кармајклов број $n$ је сложен број који је псеудопрост у свакој бази $a \in \mathbb{Z}_n^*$

- По претх. теореми вероватноћа да број $n$ који није ни прост ни Кармајклов прође тест ($\bigstar$)
  - у једном тестирању са случајно изабраном базом је највише $\frac{1}{2}$
  - у $k$ тестирања са случајно и независно изабраним базама је највише $\frac{1}{2^k}$
- Врло ефикасан тест, али не одваја просте од Кармајклових бројева

## Дефиниција

Кармајклов број $n$ је сложен број који је псеудопрост у свакој бази $a \in \mathbb{Z}_n^*$

- По претх. теореми вероватноћа да број $n$ који није ни прост ни Кармајклов прође тест ($\bigstar$)
  - у једном тестирању са случајно изабраном базом је највише $\frac{1}{2}$
  - у $k$ тестирања са случајно и независно изабраним базама је највише $\frac{1}{2^k}$
- Врло ефикасан тест, али не одваја просте од Кармајклових бројева
- Користи само степеновање (поновљеним квадрирањем)

Коблиц, глава $V.1$:

- Сваки Кармајклов број је облика $p_1 p_2 \dots p_k$, где је $k \geqslant 3$ и $p_i$-ови су међусобно различити прости бројеви

Коблиц, глава $V.1$:

- Сваки Кармајклов број је облика $p_1 p_2 \dots p_k$, где је $k \geqslant 3$ и $p_i$-ови су међусобно различити прости бројеви
- Ако је $n$ Кармајклов и $p$ прост важи

$$p | n \implies p - 1 | n - 1$$

- Пример:

$$\begin{array}{llll}
1105 = 5 \cdot 13 \cdot 17 & (4 \mid 1104; & 12 \mid 1104; & 16 \mid 1104) \\
1729 = 7 \cdot 13 \cdot 19 & (6 \mid 1728; & 12 \mid 1728; & 18 \mid 1728) \\
2465 = 5 \cdot 17 \cdot 29 & (4 \mid 2464; & 16 \mid 2464; & 28 \mid 2464) \\
2821 = 7 \cdot 13 \cdot 31 & (6 \mid 2820; & 12 \mid 2820; & 30 \mid 2820) \\
6601 = 7 \cdot 23 \cdot 41 & (6 \mid 6600; & 22 \mid 6600; & 40 \mid 6600) \\
8911 = 7 \cdot 19 \cdot 67 & (6 \mid 8910; & 18 \mid 8910; & 66 \mid 8910).
\end{array}$$