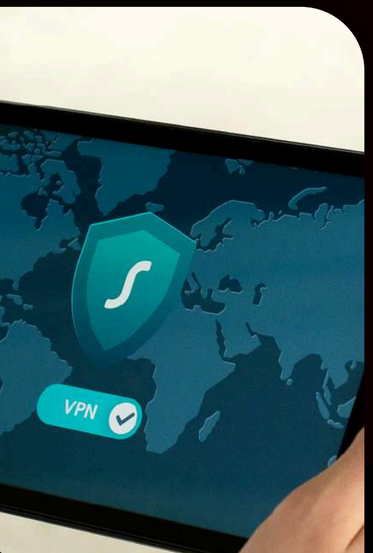
# ВООТН АЛГОРИТМІ ЖӘНЕ ҚАЙТА КОДТАУ ӘДІСТЕРІН ҚОЛДАНА ОТЫРЫП МОДУЛЬ БОЙЫНША ДӘРЕЖЕ ЖӘНЕ КӨБЕЙТУ ЕСЕПТЕУЛЕРІ

*10-ЛЕКЦИЯ*




# ЛЕКЦИЯ МАҚСАТЫ ЖӘНЕ МАЗМҰНЫ

*МАҚСАТЫ:*

Модульдік дәрежеге шығару ($M^e$ mod{n}) операциясын оңтайландырудың кеңейтілген әдістерін зерттеу[1]

**Негізгі сұрақтар:**

Booth алгоритмі және Канондық қайта кодтау әдісінің айырмашылығы.
Қайта кодтау әдістерінің негізгі кемшілігі (мысалы, RSA үшін).
Модульдік көбейтудің негізгі тәсілдері.
Карацуба-Офман алгоритмінің артықшылығы.

*ИДЕЯ: КӨРСЕТКІШТІ Е ЕҢ АЗ НӨЛДІК ЕМЕС РАЗРЯДТАРМЕН ЖАЗУ.*

## Цифрлар жиынтығы:

{-1, 0, 1}(немесе $\bar{1}, 0, 1) қолданылады.

## Басты ереже:

Нәтижеде екі нөлдік емес цифр (мысалы, '1' мен '1 немесе '1' мен '1') қатар орналаспайды[7].

## Нәтиже:

Егер е саны нөлмен толықтырылса, ол үшін жалғыз канондық вектор бар[8]. Бұл көрсеткішті "сиретудің" (sparse) ең тиімді жолы.

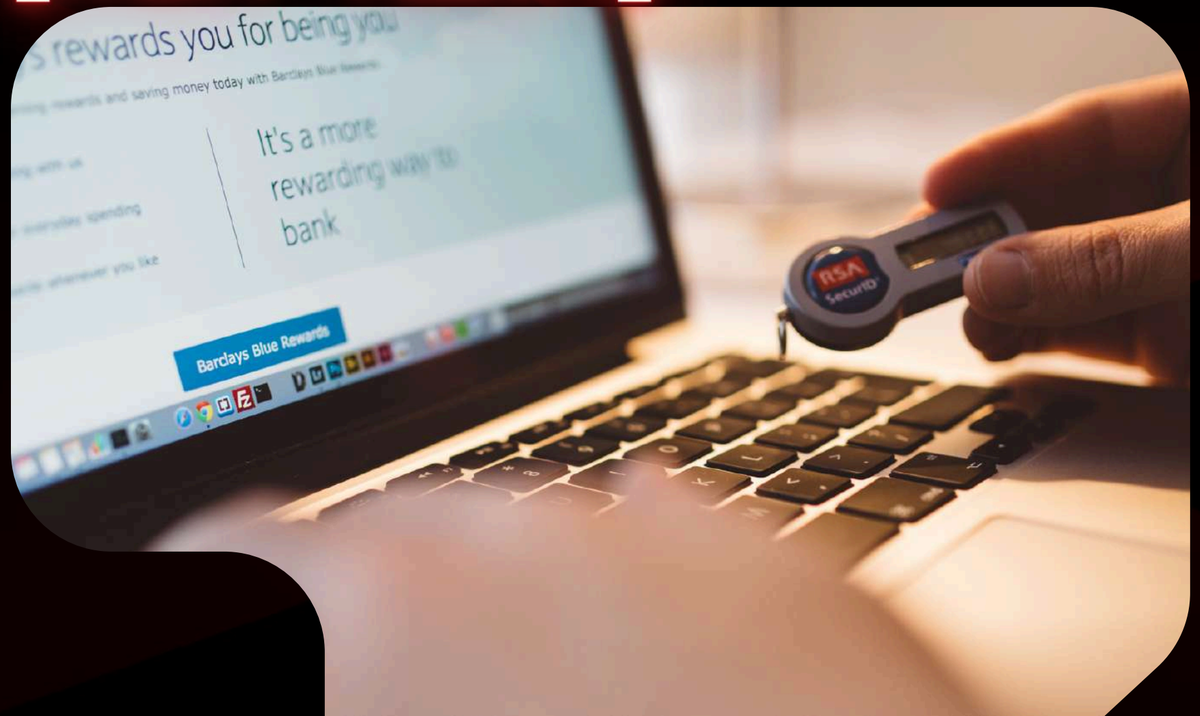
# КАНОНДЫҚ ҚАЙТА КОДТАУ ӘДІСІ

## *КАНОНДЫҚ VS. BOOTH (E = 3038 МЫСАЛЫ)*

Канондық әдіс Booth алгоритміне қарағанда көбінесе жақсырақ (сирек) нәтиже береді.

# КАНОНДЫҚ VS. BOOTH E = 3038 МЫСАЛЫ)

# ҚАЙТА КОДТАУДЫҢ НЕГІЗГІ КЕМШІЛІГІ [RSA ҮШІН]





*ҚАЙТА КОДТАУ ӘДІСТЕР ҚОЛДАНУ) $M^E \PMOD{N}$ ЕСЕПТЕУ ҮШІН $M^{-1}$ (КЕРІ МӘН) ҚАЖЕТ ЕТЕДІ*

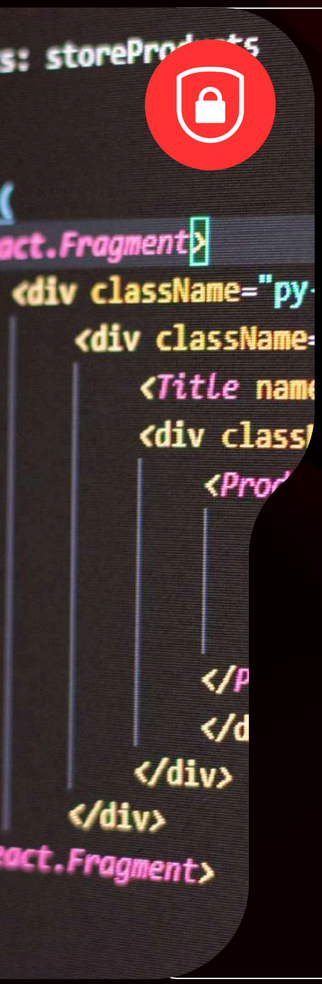
## Мәселе:

$M^{-1}$ мәнін (мысалы, Эвклид алгоритмі арқылы) есептеуге кететін уақыт, қайта кодтау арқылы үнемделген уақыттан асып түседі.

## Салдары:

Бұл әдістер $M^{-1}$ мәні басқа себептермен алдын ала белгілі болмаса, RSA криптожүйелерінде кеңінен қолданылмайды.

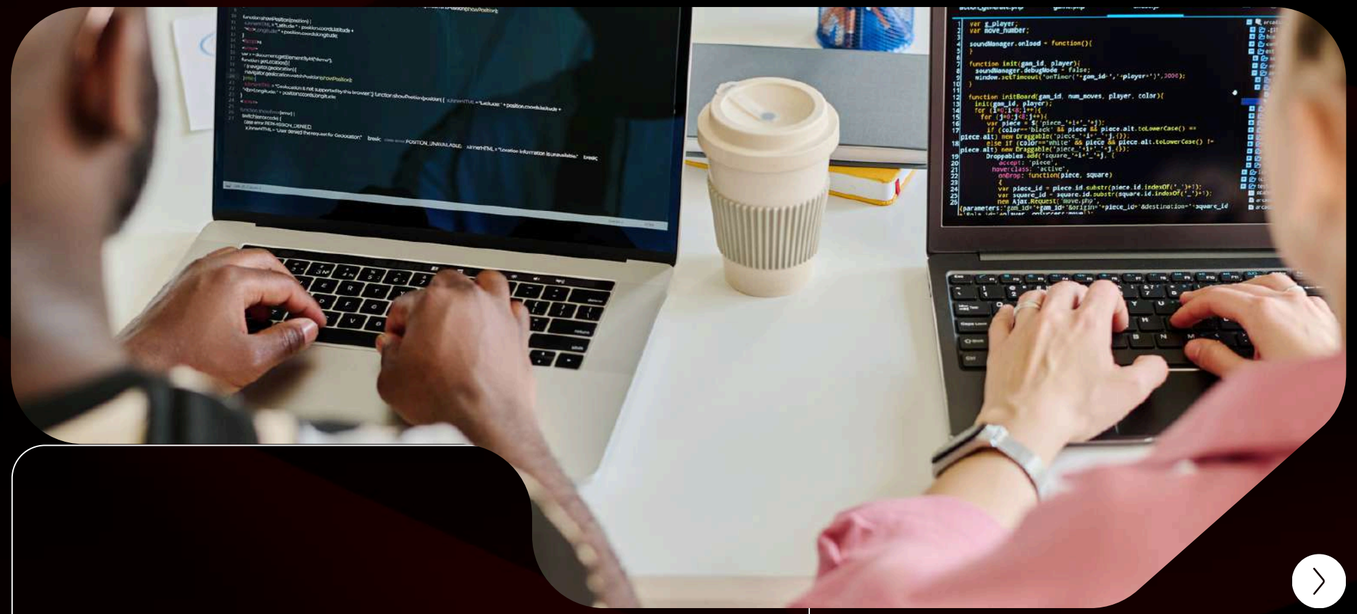
# МОДУЛЬ БОЙЫНША КӨБЕЙТУ [ЕСЕПТЕУ НЕГІЗІ]




*ДӘРЕЖЕГЕ ШЫҒАРУДЫҢ (МЫСАЛЫ, БИНАРЛЫҚ ӘДІС) ӘРБІР ҚАДАМЫ МОДУЛЬ БОЙЫНША КӨБЕЙТУДЕН ТҰРАДЫ.*


# КӨБЕЙТУДІҢ СТАНДАРТТЫ АЛГОРИТМІ

Бұл "мектеп" әдісі: a көбейгішін $b$ көбейткішінің әр цифрына көбейтіп, нәтижелерді қосу.
Алгоритм $s^2$ ішкі туынды амалын орындайды (мұндағы s - машиналық сөздер саны).
Есептеу күрделілігі: $O(k^2)$ биттік амалдар.
Бағалау: Қолдану қарапайым, бірақ үлкен k үшін асимптотикалық түрде баяу.




*СТАНДАРТТЫ $O(K^2)$АЛГОРИТМІНЕН ЖЫЛДАМЫРАҚ ЖҰМЫС ІСТЕЙТІН РЕКУРСИВТІ ӘДІС[27].*

# КАРАЦУБА-ОФМАН АЛГОРИТМІ [1962]

# ҚОРЫТЫНДЫ

## Канондық қайта кодтау

көрсеткішті "сиретудің" (нөлдік емес биттерді азайту) ең тиімді жолы, көбіне Booth әдісінен де жақсы.

## Қайта кодтаудың кемшілігі:

$M^{(-1)}$мәнін есептеуді қажет етеді, бұл оны RSA үшін тиімсіз етеді

## Стандартты көбейту $O(k^2)$ күрделілігіне ие.

Карацуба-Офман 4 көбейтуді 3-ке (рекурсивті) азайту арқылы $O(k^2)$-тан жылдам нәтиже береді.

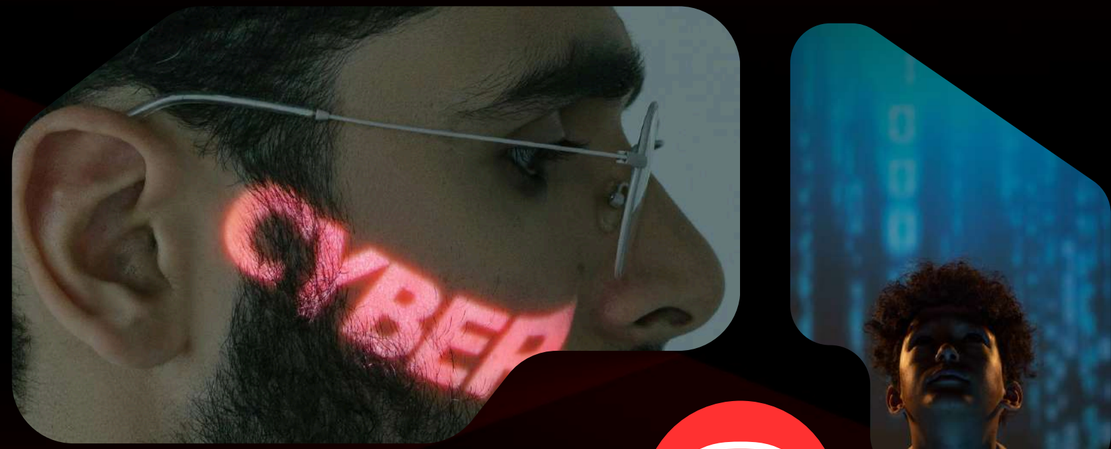
# БАҚЫЛАУ СҰРАҚТАРЫ

Адаптивті m-арлы әдіс алдын ала есептеу қадамында көбейту санын қалай азайтады?

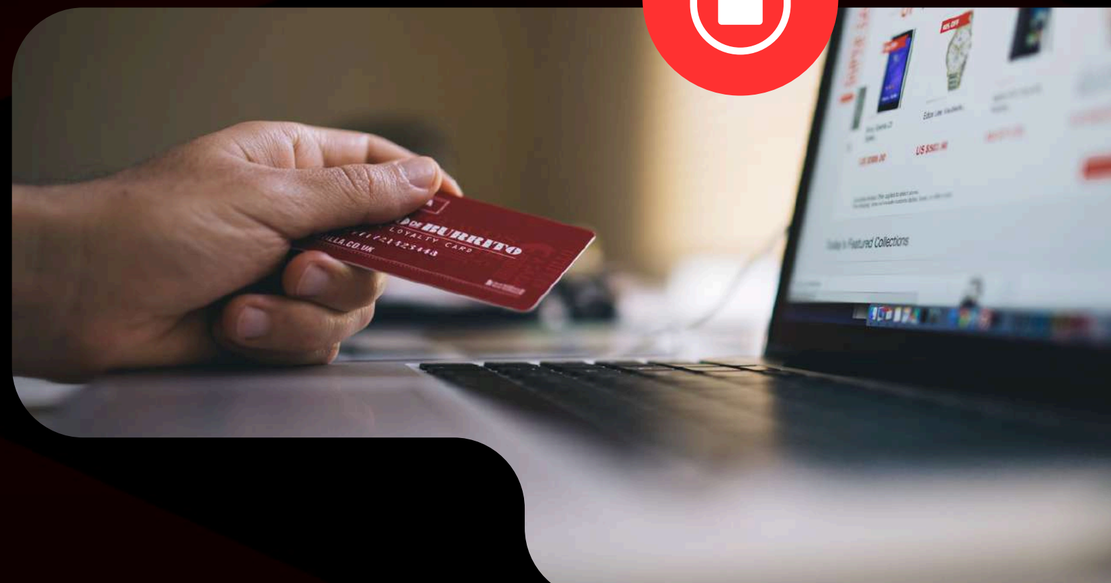
Канондық қайта кодтау (e=3038$ мысалында) Booth алгоритміне қарағанда нелiктен тиімдірек болды?

Қайта кодтау әдістерін ($M^{[-1}$ ) қажет) RSA-да қолданудың негізгі кемшілігі неде?

Карацуба-Офман алгоритмі $O(k^2$ ) стандартты көбейтуді қалай жақсартады (негізгі идеясы)?





# THANK YOU

## *ӘДЕБИЕТТЕР ТІЗІМІ:*


1. Knuth, D. E. (1997). The Art of Computer Programming, Volume 2: Seminumerical Algorithms (3rd ed.). Addison-Wesley Professional. (Мәтінде [6] және [15] ретінде сілтеме жасалған негізгі дерекқөз).
2. Menezes, A. J., van Oorschot, P. C., & Vanstone, S. A. (1996). Handbook of Applied Cryptography. CRC Press. (14-тарау, "Efficient Exponentiation" & "Efficient Multiplication").
3. Bos, J., & Coster, M. (1990). Addition chain heuristics. In Advances in Cryptology — CRYPTO ’89 Proceedings (pp. 400-407). Springer-Verlag. (VLNW әдісін ұсынған авторлар).
4. Karatsuba, A., & Ofman, Y. (1962). Multiplication of many-digital numbers by automatic computers. Proceedings of the USSR Academy of Sciences, 145, 293–294. (Карацуба-Офман алгоритмі).