L28

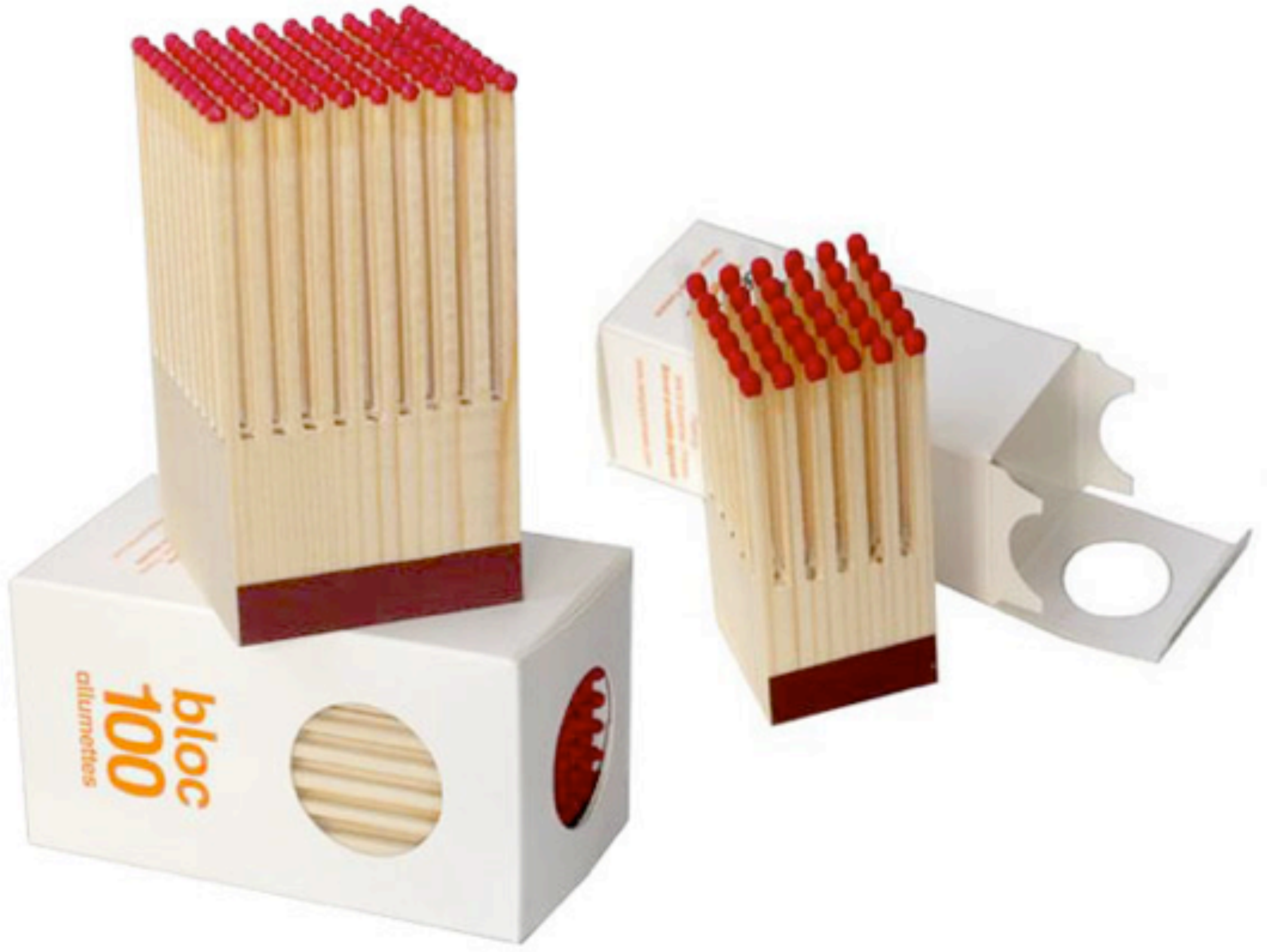
4102

12.05.2013

review,
crypto

abhi


VIRGINIA
1788
JAMESTOWN
1607-2007
QUADRICENTENNIAL
2000
E PLURIBUS UNUM



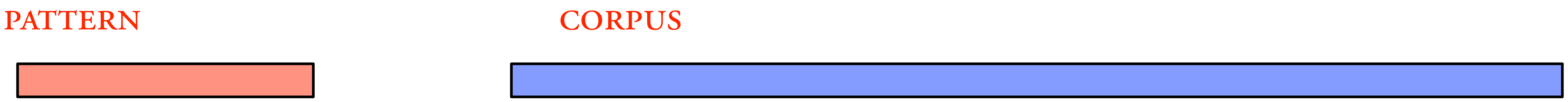
http://kitsunenoir.com/blogimages/bloc-matches.jpg

FINGERPRINTING

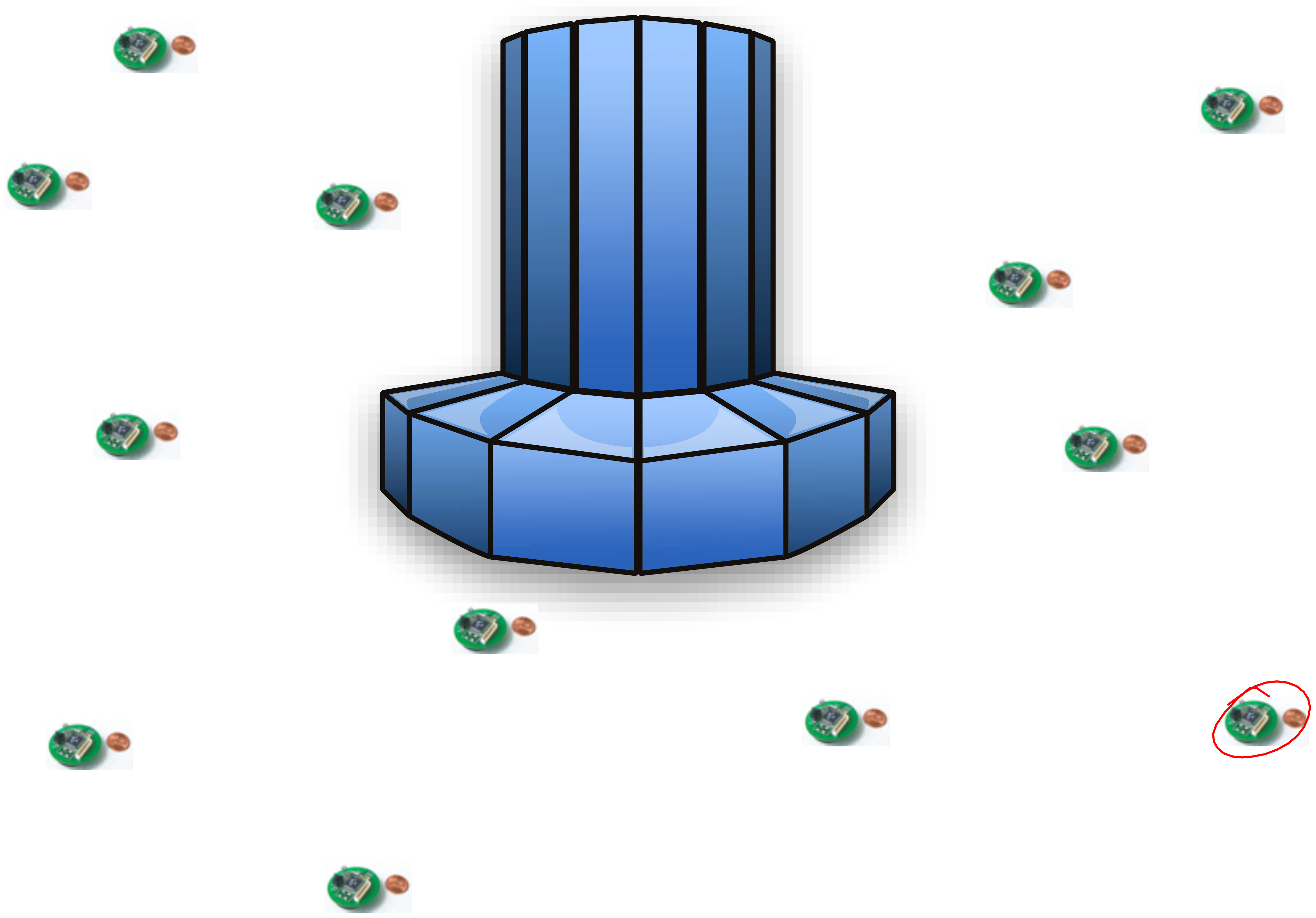
Alice

Bob

STRING MATCHING



RELIABLE COMMUNICATION



GOAL:

DEVISE A RELIABLE METHOD FOR NODES TO SEND MESSAGE
TO THE SERVER WITH AS LITTLE COORDINATION AS POSSIBLE.

SIMPLE ALGORITHM

AT TIME T, FLIP A COIN THAT IS HEADS WITH PR $\frac{1}{n}$

IF HEADS, THEN BROADCAST. IF SUCCESS, THEN STOP.

ELSE WAIT AND TRY AGAIN.

REPEAT $cn \log n$ TIMES

ANALYZE THE SIMPLE ALGORITHM

$S_{i,t}$ = event that node i succeeds in sending its message at time t

$$\Pr[S_{i,t} = 1] = \left(\frac{1}{n}\right)\left[1 - \frac{1}{n}\right]^{n-1}$$

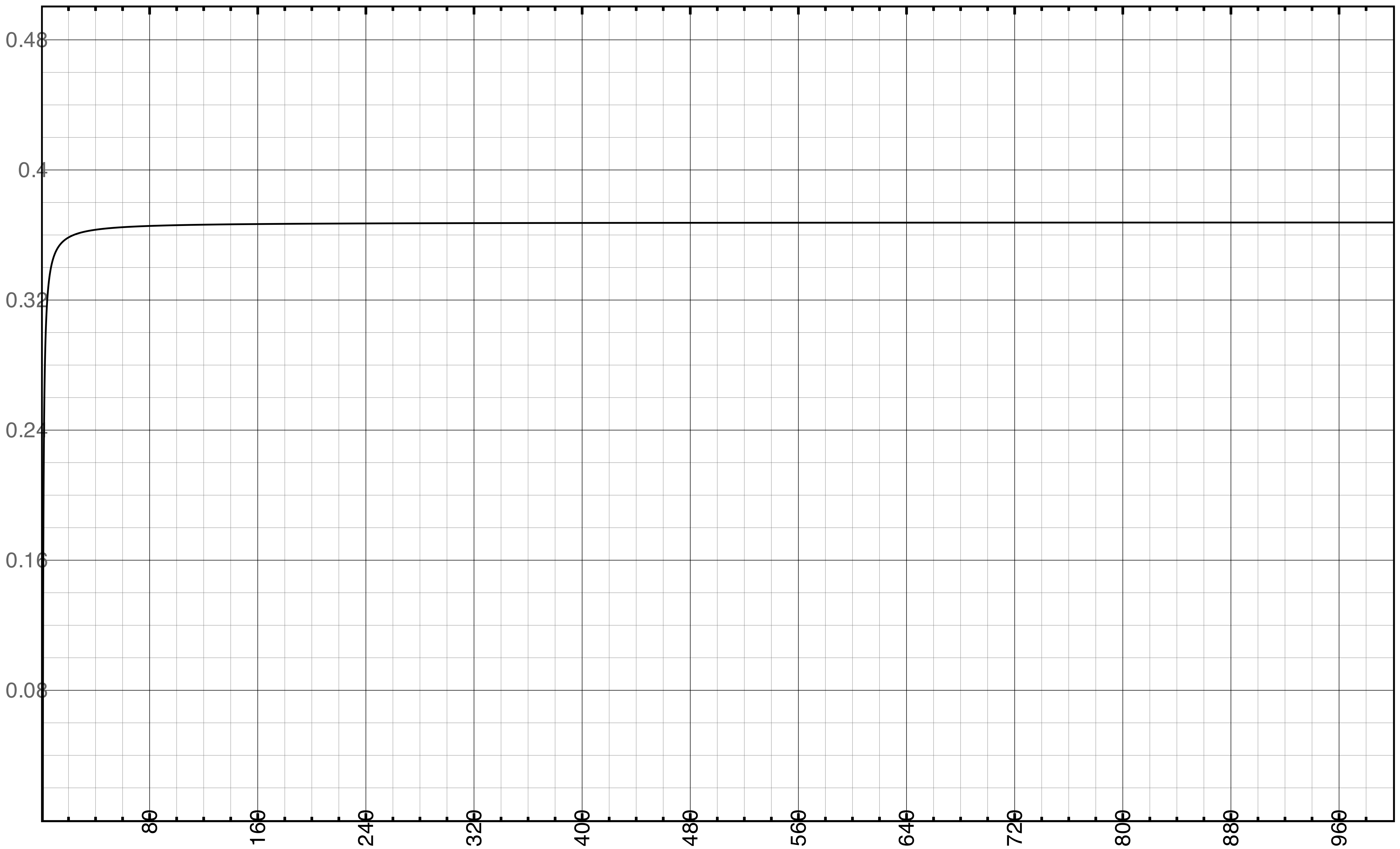
heads for i

$$\Pr[S_{i,t} = 1] = \frac{1}{n}\left(1 - \frac{1}{n}\right)^{n-1} \sim \frac{1}{n \cdot e}$$

$\frac{1}{e}$

FACT: IF $f(n) = \left(1 - \frac{1}{n}\right)^n$ THEN

FACT: IF $f(n) = \left(1 - \frac{1}{n}\right)^n$ THEN $\sim \frac{1}{e}$



APMA

$S_{i,t} =$ NODE i SUCCEEDS IN SENDING AT TIME t

$$\frac{1}{en} \leq \Pr[S_{i,t} = 1] \leq \frac{1}{2n}$$

FAILURE

$F_{i,t}$ = probability that i fails @ times 1, 2, 3 ... t

FAILURE

$F_{i,t} =$ NODE i FAILS TO SEND AT TIMES 1,2,...,t

$$\Pr[F_{i,t}] = \bigwedge_{j=1}^{t} \Pr[\overline{S_{i,j}}]$$

FAILURE

$F_{i,t} =$ NODE i FAILS TO SEND AT TIMES 1,2,...,t

$$\Pr[F_{i,t}] = \bigwedge_{j=1}^{t} \Pr[\overline{S_{i,j}}] = \Pi_{j=1}^{t} \Pr[\overline{S_{i,j}}]$$

$$\Pr[F_{i,t}] = \bigwedge_{j=1}^{t} \Pr[\overline{S_{i,j}}] = \prod_{j=1}^{t} \Pr[\overline{S_{i,j}}]$$

FAILURE

$F_{i,t} =$ NODE i FAILS TO SEND AT TIMES $1,2,\ldots,t$

$$\Pr[F_{i,t}] = \bigwedge_{j=1}^{t} \Pr[\overline{S_{i,j}}] = \Pi_{j=1}^{t} \Pr[\overline{S_{i,j}}] < \left(1 - \frac{1}{2n}\right)^{t}$$

$$\left(1 - \frac{1}{2n}\right)$$

FOR $t = O(n \ln n)$

$$\Pr[F_{i,t}] = n^{-c}$$

ALL FAIL

$$F_t =$$

$$\Pr[F_t] =$$

ALL FAIL

$F_t =$ SOME NODE i FAILS TO SEND AT TIMES 1,2,...,t

$$\Pr[F_t] = \bigvee_{i=1}^{n} \Pr[F_{i,t}]$$

ALL FAIL

$F_t =$ SOME NODE i FAILS TO SEND AT TIMES 1,2,...,t

$$\Pr[F_t] = \bigvee_{i=1}^{n} \Pr[F_{i,t}] \leq \sum_{i=1}^{n} \Pr[F_{i,t}] \leq \sum_{i=1}^{n} n^{-c}$$

SUMMARY

AT TIME T, FLIP A COIN THAT IS HEADS WITH PR $\frac{1}{n}$

IF HEADS, THEN BROADCAST. IF SUCCESS, THEN STOP.

ELSE WAIT AND TRY AGAIN.

REPEAT $O(n \ln n)$ TIMES

WITH PROBABILITY

EVERY NODE SUCCEEDS IN SENDING MESSAGE.

TOOLS WE USED

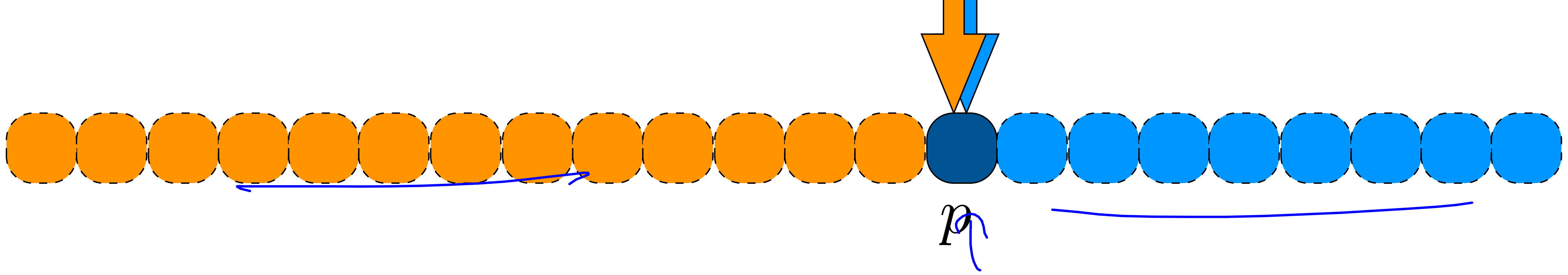
ANALYSIS OF $\left(1 - \frac{1}{n}\right)^n$

PROBABILITY THAT MANY INDEPENDENT EVENTS ALL OCCUR:

PROBABILITY THAT ONE OUT OF N EVENTS OCCURS:

SECOND EXAMPLE:

MEDIAN



SELECT $(i, A[1, \ldots, n])$

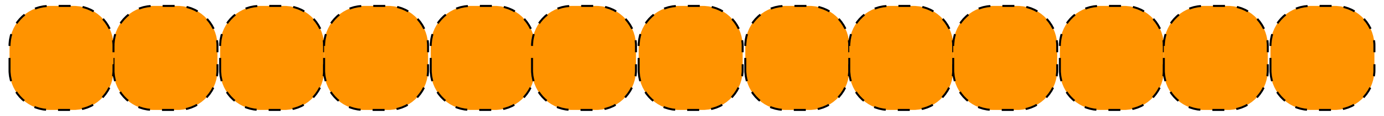
PICK FIRST ELEMENT

PARTITION LIST ABOUT THIS ONE

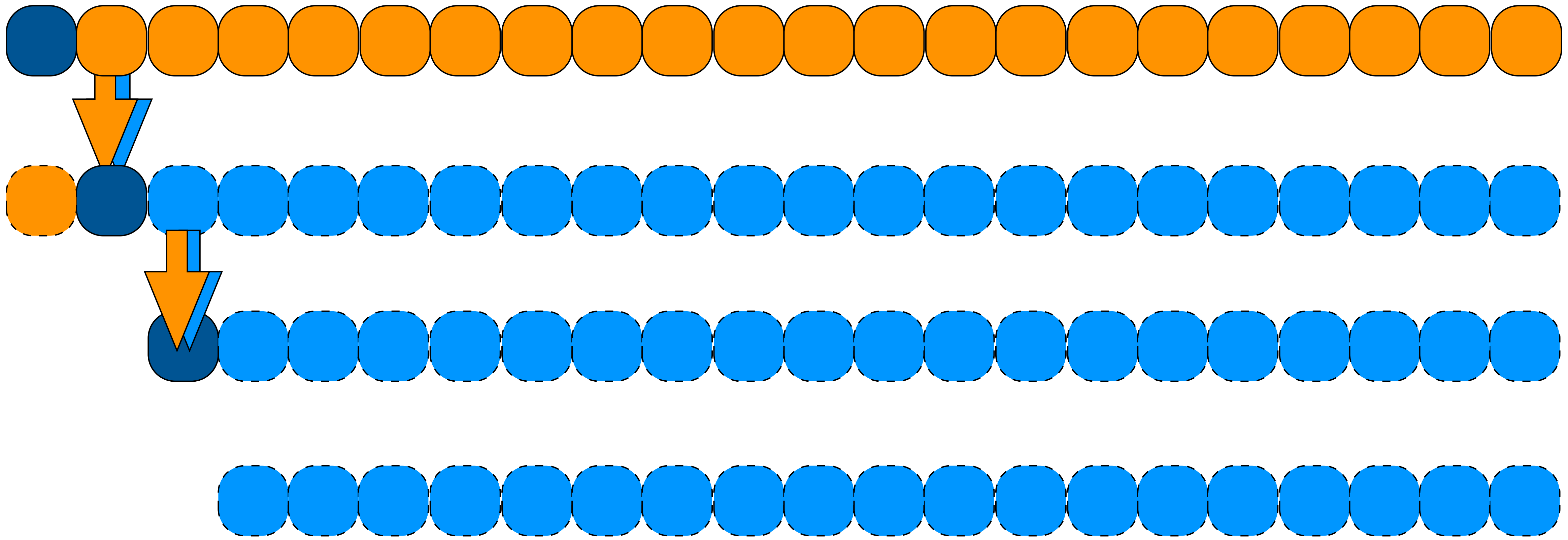
IF PIVOT IS POSITION i, RETURN PIVOT

ELSE IF PIVOT IS IN POSITION > i SELECT $(i, A[1, \ldots, p-1])$

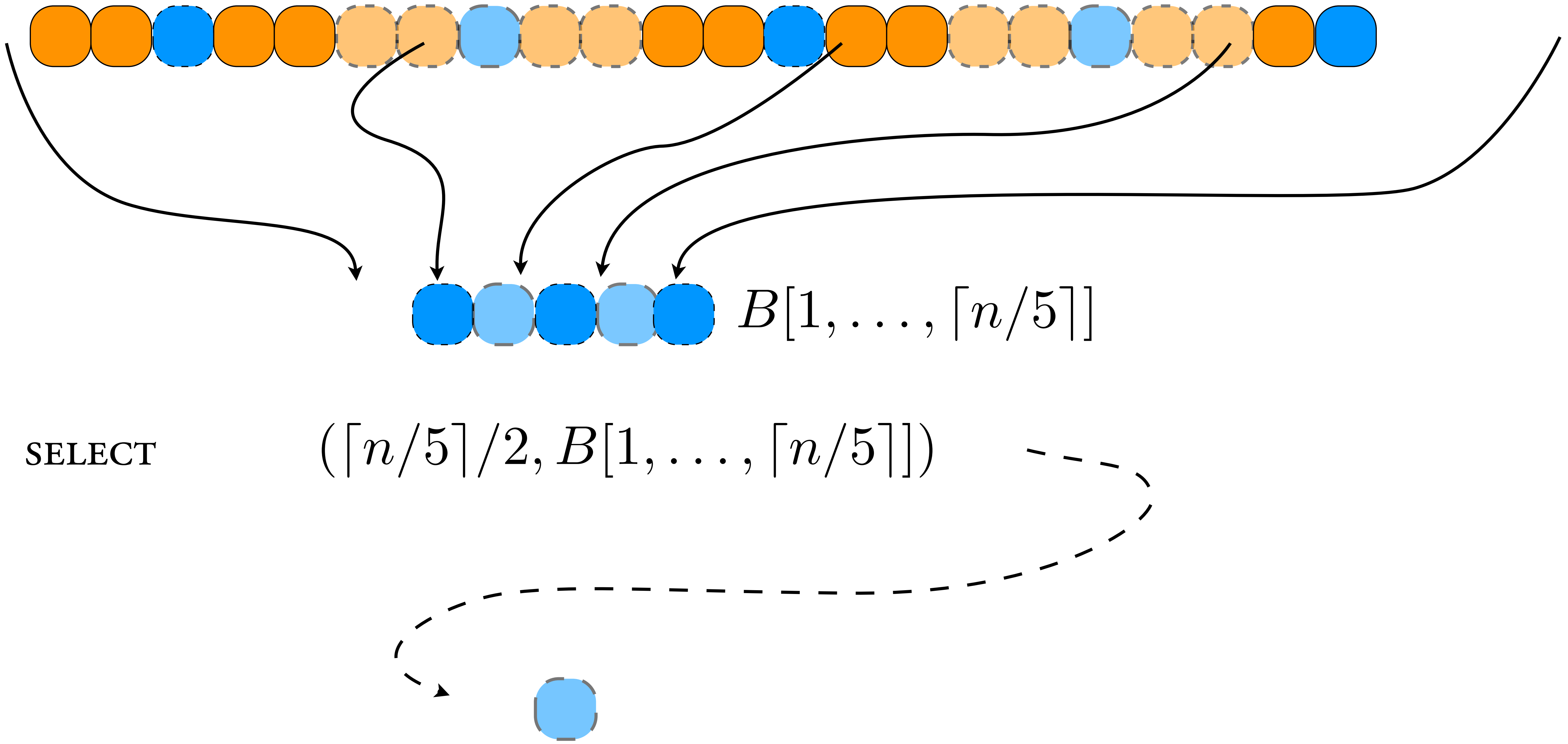
ELSE SELECT $((i-p-1), A[p+1, \ldots, n])$



PROBLEM: WHAT IF WE ALWAYS PICK BAD PARTITIONS?



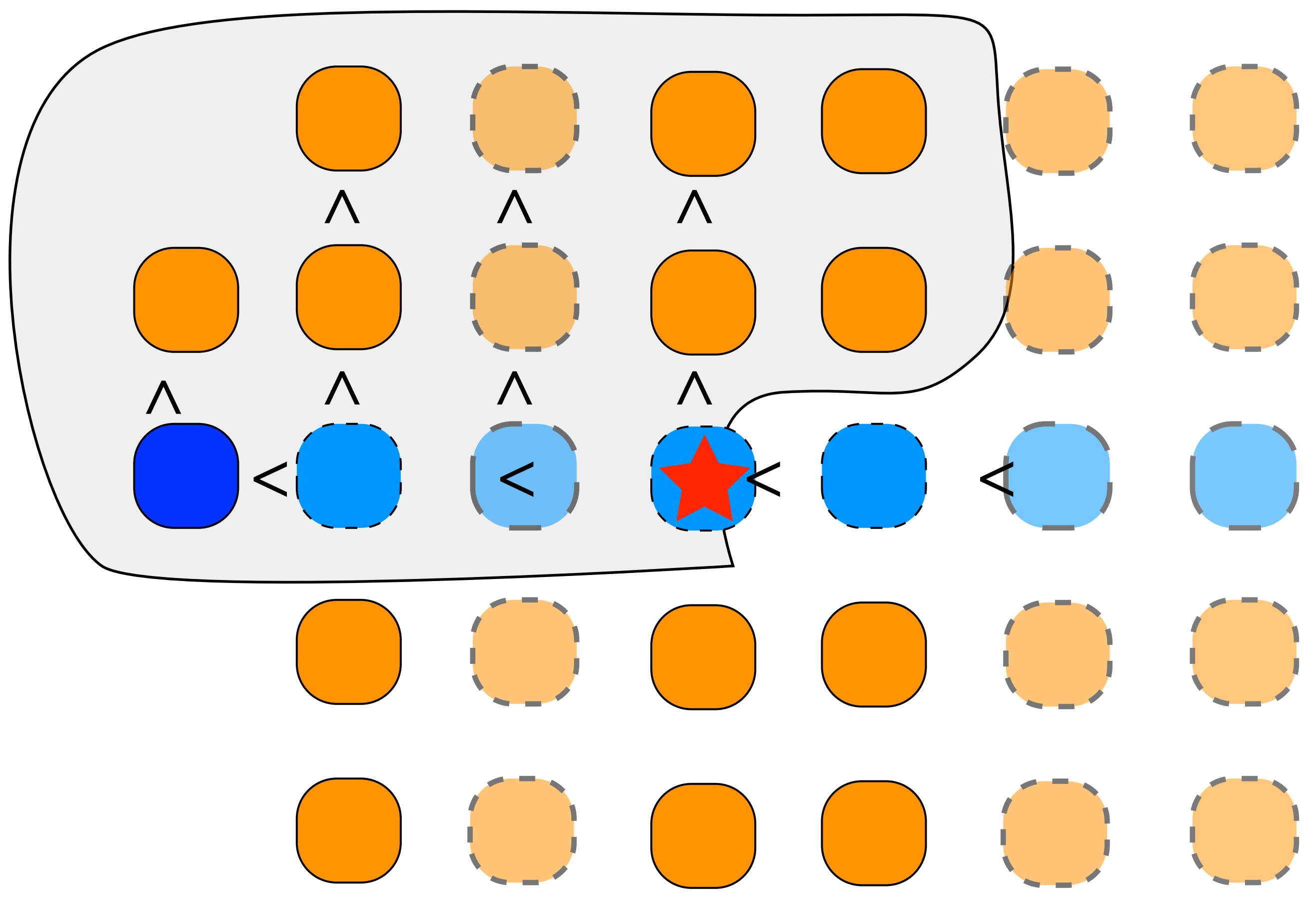
PARTITION $(A[1, \ldots, n])$



A NICE PROPERTY OF OUR PARTITION

$$3\left(\left\lceil \frac{1}{2}\lceil n/5\rceil \right\rceil - 2\right)$$

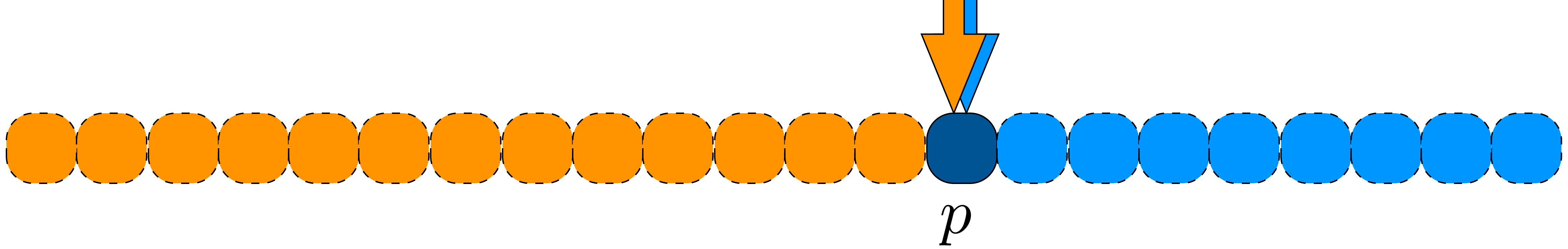
$$\geq \frac{3n}{10} - 6$$



THIS IMPLIES THERE ARE

AT MOST $\dfrac{7n}{10} + 6$ NUMBERS

LARGER THAN ★
/SMALLER



SELECT $(i, A[1, \ldots, n])$

~~PICK FIRST ELEMENT~~

PIVOT = PARTITION $(A[1, \ldots, n])$

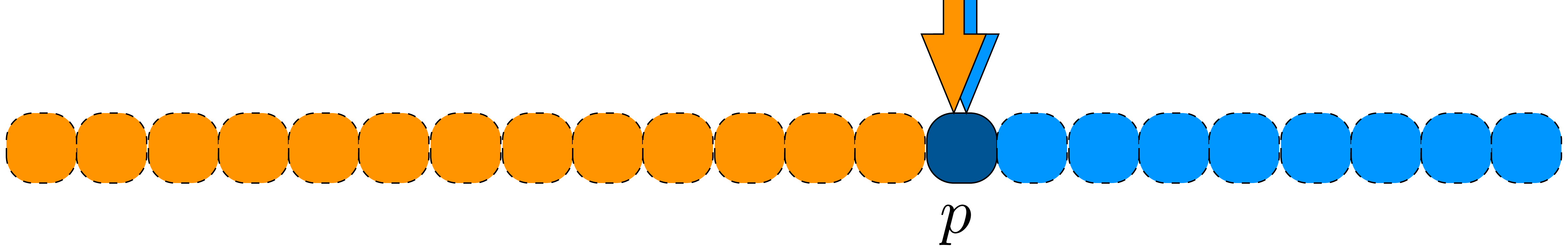
IF PIVOT IS POSITION i, RETURN PIVOT

ELSE IF PIVOT IS IN POSITION > i SELECT $(i, A[1, \ldots, p-1])$

ELSE SELECT $((i-p-1), A[p+1, \ldots, n])$

$$S(n) = S(\lceil n/5 \rceil) + O(n) + S(7n/10 + 6)$$

$$\Theta(n)$$



RandomizedSelect $(i, A[1, \ldots, n])$

PICK RANDOM PARTITION ELEMENT

PARTITION LIST ABOUT THIS ONE

IF PIVOT IS POSITION i, RETURN PIVOT

ELSE IF PIVOT IS IN POSITION > i SELECT $(i, A[1, \ldots, p-1])$

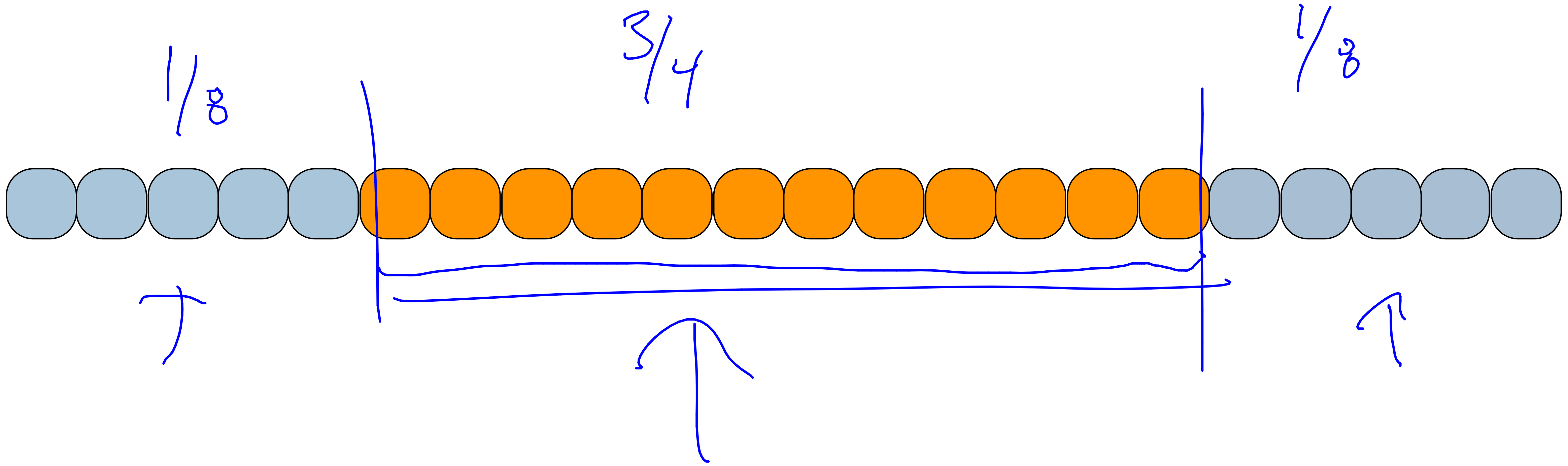
ELSE SELECT $((i-p-1), A[p+1, \ldots, n])$

RandomizedSelect $(i, A[1, \ldots, n])$

PICK RANDOM PARTITION ELEMENT

PARTITION LIST ABOUT THIS ONE

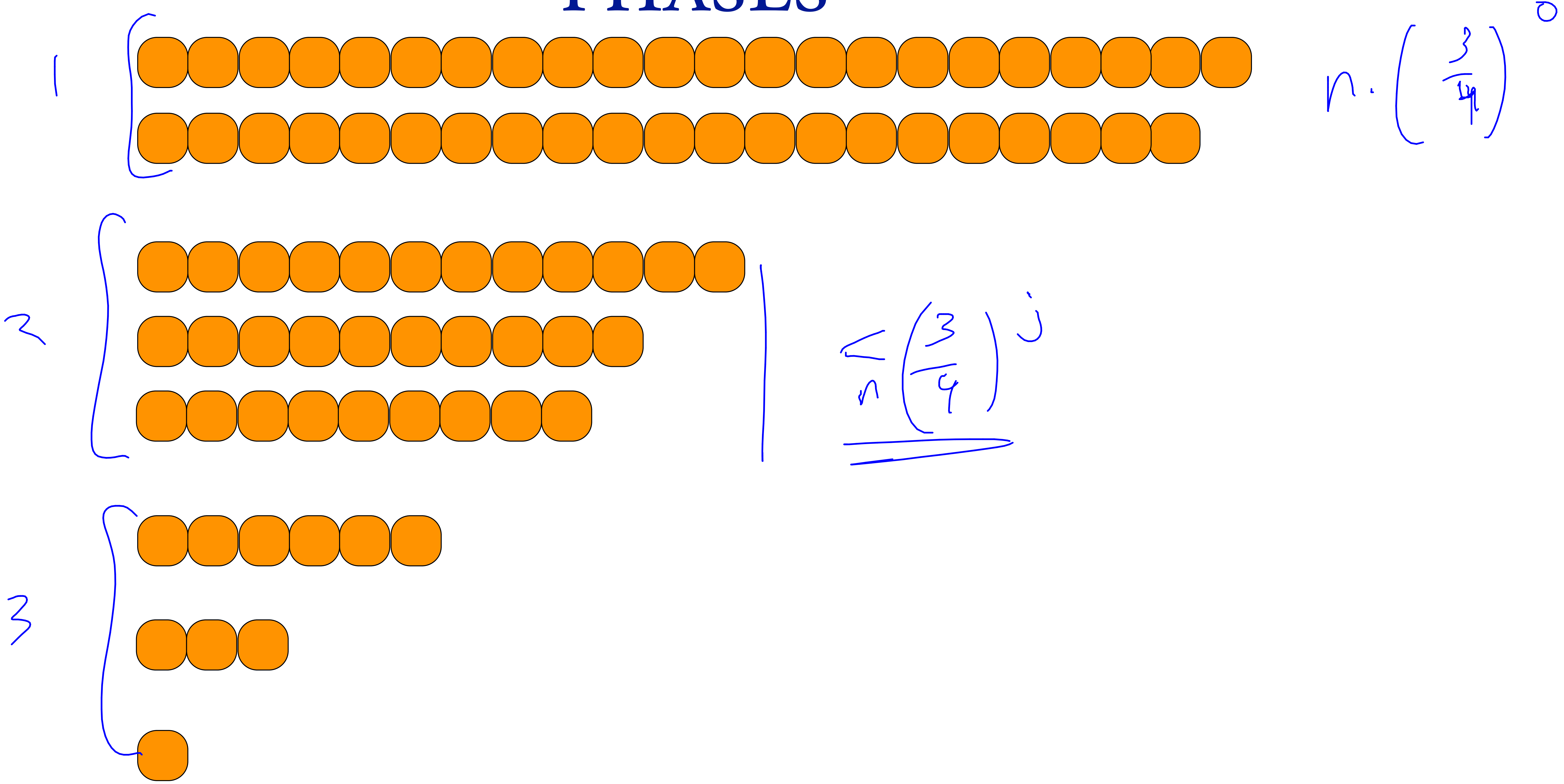
....



RUNNING TIME ANALYSIS

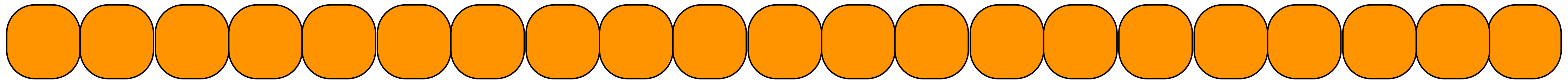
RECURSIVE CALLS

PHASES



PHASES

ALGORITHM IS IN PHASE J IF



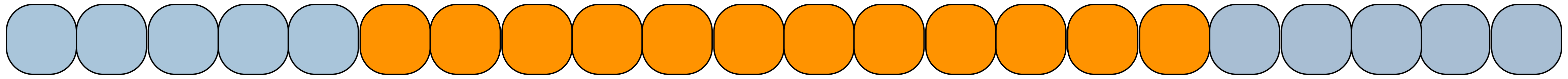
SIZE OF INPUT LIST IS $< \left(\frac{3}{4}\right)^{j} n$

RandomizedSelect $(i, A[1, \ldots, n])$

PICK RANDOM PARTITION ELEMENT

PARTITION LIST ABOUT THIS ONE

....



X_j = NUMBER OF STEPS IN PHASE J

$E[X_j] =$

X_j = NUMBER OF STEPS IN PHASE J

$$E[X_j] = \sum_{j=0}^{\infty} j \cdot \Pr[X_j = j]$$

$$\left(\frac{4}{3}\right) n \left(\frac{3}{4}\right)^j$$

$\Pr[X_j = 1] =$

$\Pr[X_j = 2] =$

$\Pr[X_j = j] =$

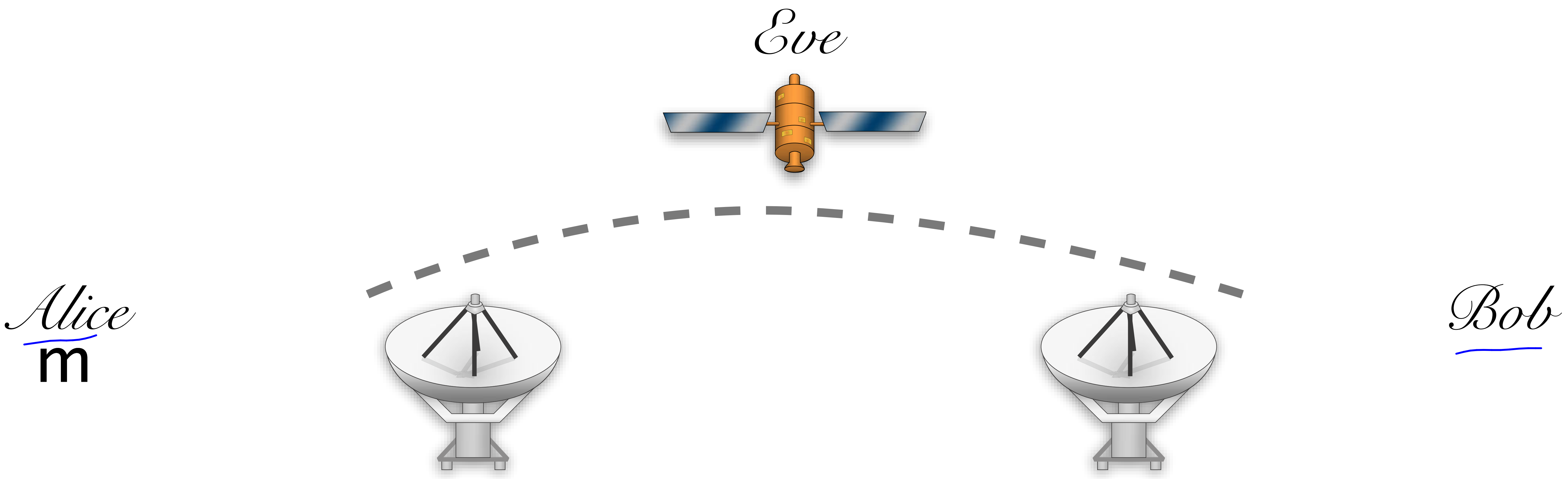
LINEARITY OF EXPECTATION

$$\forall X, Y, \quad E[X + Y] = E[X] + E[Y]$$

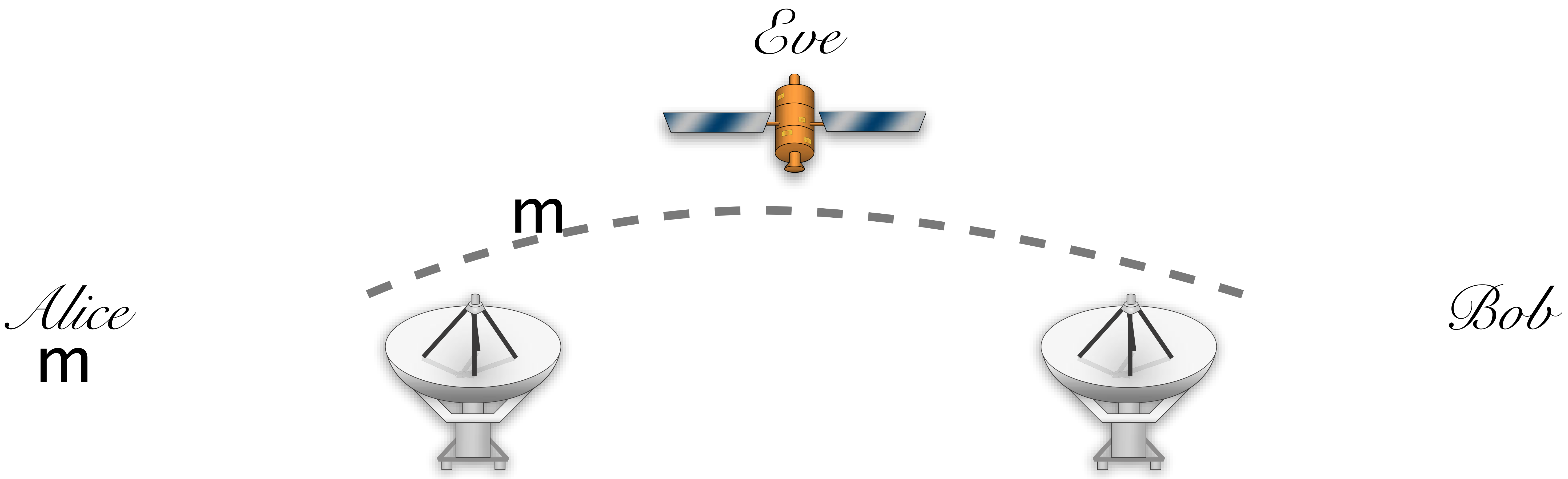
EXPECTED RUNNING TIME

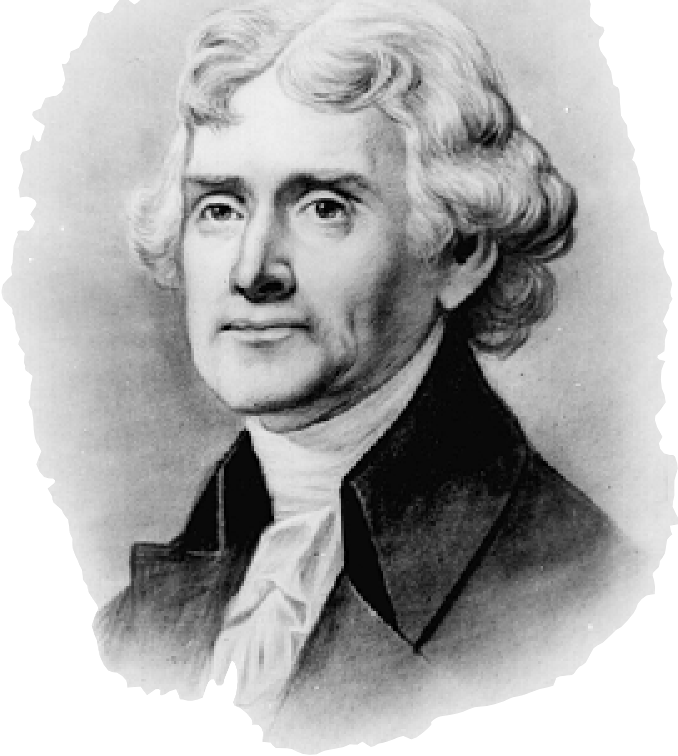
$$E[X] = \Theta(n)$$

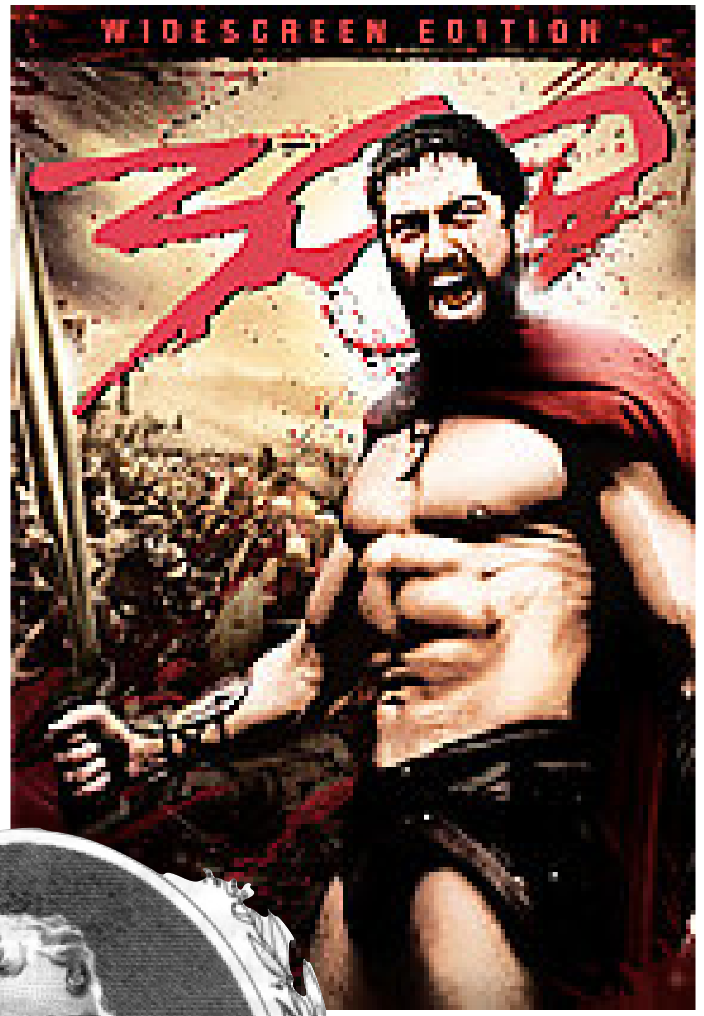
PRIVATE COMMUNICATION

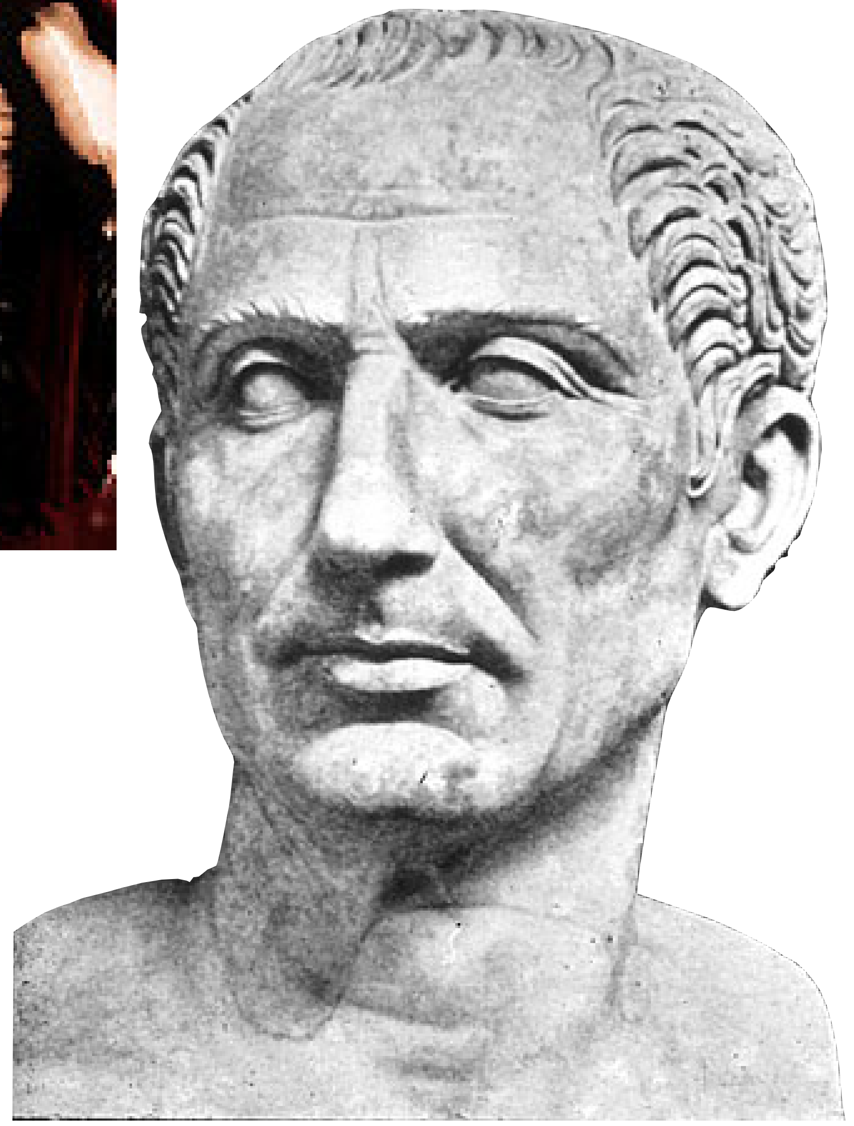


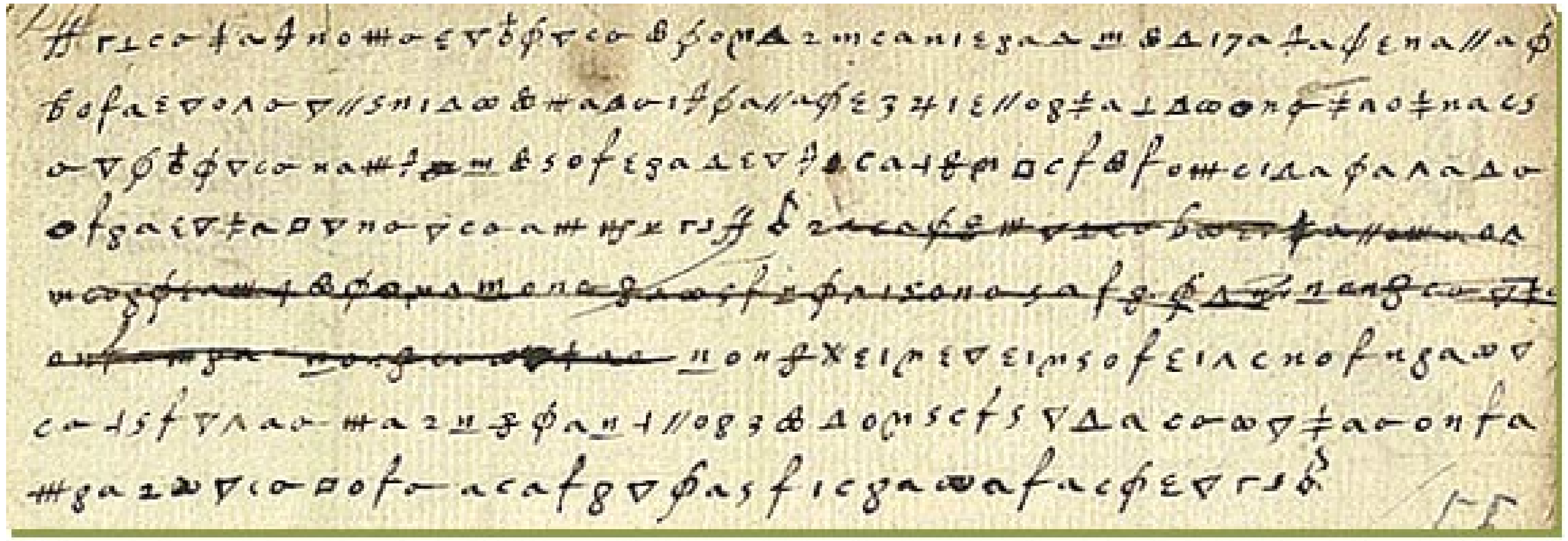
PRIVATE COMMUNICATION

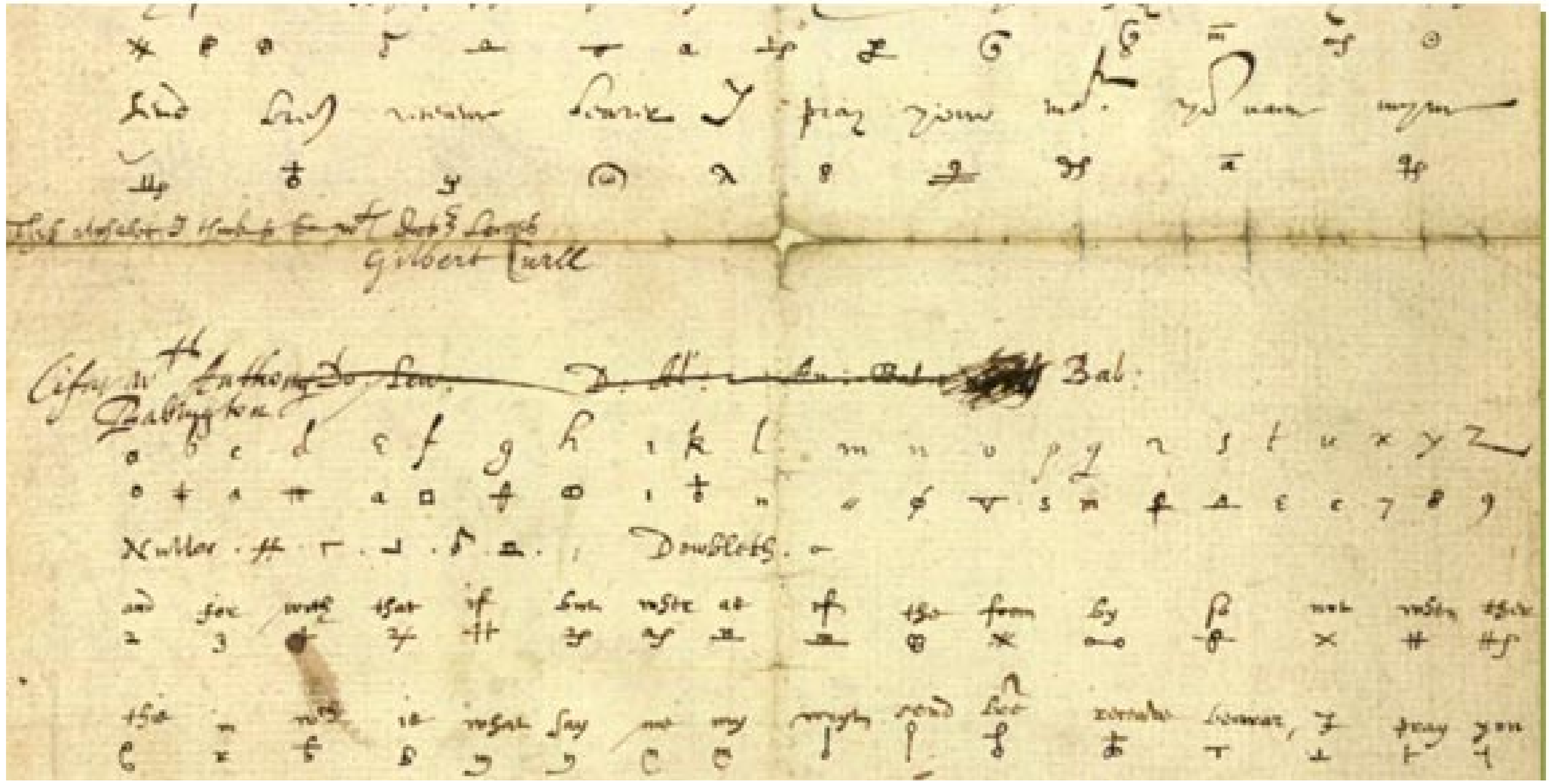





WIDESCREEN EDITION
300






Gilbert Curll

SUBSTITUTION CIPHER

$$
\begin{aligned}
\mathcal{M} &= \{A, B, \ldots, Z\}^* \\
\mathcal{K} &= \text{the set of permutations over } \{A, B, \ldots, Z\} \\
\mathsf{Gen} &= k \text{ where } k \xleftarrow{r} \mathcal{K}. \\
Enc_k(m_1 m_2 \ldots m_n) &= c_1 c_2 \ldots c_n \text{ where } c_i = k(m_i) \\
Dec_k(c_1 c_2 \ldots c_n) &= m_1 m_2 \ldots m_n \text{ where } m_i = k^{-1}(c_i)
\end{aligned}
$$

SIZE OF KEYSPACE IS

$26! = 403291461126605635584000000$

EOR TZSRWF XEASG ZV DWGYEZPWQYOG NFKRXENPQERX ERDOFNIARX VZW VQDNHNEQENFP NFERWQDENZFX
UREJRRF SNXEWAXEVAH RFENENRX NF ZAW DZFFRDERS XZDNREG XADO ERDOFNIARX OQKR URDZTR
NFSNXYRFXQUHR RFQUHNFP VZW NFXEQFDR QAEZTQERS ERHHRW TQDONFRX XRDAWR JNWRHRXX FREJZWLX
NFERWFRE UQFLNFP XQERHHNER WQSNZERHRKNXNZF QFS TZWR NF EONX DZAWXR JR NFEWZSADR XZTR ZV
EOR VAFSQTRFEQH DZFDRYEX ZV EONX XEASG RTYOQXNX JNHH UR YHQDRS ZF WNPZWZAX YWZZVX ZV
XRDAWNEG UQXRS ZF YWRDNXR SRVNFNENZFX QFS QXXATYENZFX

EOR TZSRWF XEASG ZV DWGYEZPWQYOG NFKRXENPQERX ERDOFNIARX VZW VQDNHNEQENFP NFERWQDENZFX
UREJRRF SNXEWAXEVAH RFENENRX NF ZAW DZFFRDERS XZDNREG XADO ERDOFNIARX OQKR URDZTR
NFSNXYRFXQUHR RFQUHNFP VZW NFXEQFDR QAEZTQERS ERHHRW TQDONFRX XRDAWR JNWRHRXX FREJZWLX
NFERWFRE UQFLNFP XQERHHNER WQSNZERHRKNXNZF QFS TZWR NF EONX DZAWXR JR NFEWZSADR XZTR ZV
EOR VAFSQTRFEQH DZFDRYEX ZV EONX XEASG RTYOQXNX JNHH UR YHQDRS ZF WNPZWZAX YWZZVX ZV
XRDAWNEG UQXRS ZF YWRDNXR SRVNFNENZFX QFS QXXATYENZFX

FREQUENCY ANALYSIS

RSA

'78

1. modular exponentiation
2. greatest common divisor
3. Picking large primes $
4. Euler's theorem

P/C

public key encryption

Amz PK

Enc(pk^{Amz}, m) → c

Dec(sk^{Amz}, c) → m

MOD-EXP

1000 bits !!

$(a, x, n) \to a^x \bmod n$

a mod N

a^{12}

$a^{n\ldots}$

MOD-EXP

$$(a, x, n) \rightarrow a^x \bmod n$$

$$a^x \bmod n = \prod_{i=0}^{\ell} x_i a^{2^i} \bmod n$$

$17^2 \bmod 11 =$

26
11) 289
22
69
66
3

MOD-EXP

$$(a, x, n) \to a^x \bmod n$$

Algorithm 2: ModularExponentiation(a, x, n)

Input: $a, x \in [1, n]$

$r \leftarrow 1$
while $x > 0$ **do**
  if x *is odd* **then**
    $r \leftarrow r \cdot a \bmod n$
  $x \leftarrow \lfloor x/2 \rfloor$
  $a \leftarrow a^2 \bmod n$
Return r

$$a^x \to \left(a^{x/2}\right)^2$$

MOD-EXP

$$(a, x, n) \rightarrow a^x \bmod n$$

$$a^x \bmod n = \prod_{i=0}^{\ell} x_i a^{2^i} \bmod n$$

Algorithm 2: ModularExponentiation(a, x, n)

```
Input: a, x ∈ [1, n]
r ← 1
while x > 0 do
    if x is odd then
        r ← r · a mod n
    x ← ⌊x/2⌋
    a ← a^2 mod n
Return r
```

EUCLID

greatest common divisor of

gcd(35 and 14) = 7

5·7 2·7

EUCLID AND THE GCD

12379182789373

14210416062275 4

what is the GCD?

Algorithm 1: ExtendedEuclid(a, b)

Input: (a, b) s.t $a > b \geq 0$

Output: (x, y) s.t. $ax + by = \gcd(a, b)$

```
if a mod b = 0 then
    Return (0, 1)
else
    (x, y) ← ExtendedEuclid (b, a mod b)
    Return (y, x − y(⌊a/b⌋))
```

\# of bits in a = n

$\Theta(\log a)$

GIVEN (A,B):

FINDS (X,Y) S.T. AX + BY = GCD(A,B)

35 and 14

$\gcd(35, 14) = 7$

$(1)35 + (14)(-2) = 7$

13 and 73

$\gcd(13, 73) = 1$

$(13)(-28) + (73)5 = 1$

$$\begin{array}{r} 365 \\ -364 \\ \hline 1 \end{array}$$

CRYPTOGRAPHY

3296403179432394481965339349045974732228635O
31500646399521148595996590847768392238771217
69252874938669758963521262177684757622917354
10764395167469005450386721087598087995167019
51260209070780169584330401159403323161691626
51931932385937935848982371478700671595968131
07098610562722922433990122345442992245859824
74364293651925019779584845838833700838150940
56504167483874319231730153624474523841938831
33113697736378643670286581890300666191500953
329742364829

LARGE PRIME NUMBER

```
import java.io.*;
import java.math.*;
import java.util.*;

public class pr {
   public static void main(String args[]) {

      BigInteger prime = new BigInteger(1500,80,new Random());
      System.out.println("prime is " +prime);

   }
}
```

2^{-80}

RABIN-MILLER

$$L_N = \{\alpha \in \mathbb{Z}_N \mid \alpha^{N-1} = 1 \text{ and if } \alpha^{u2^{j+1}} = 1 \text{ then } \alpha^{u2^j} = 1\}$$

RABIN-MILLER

$$L_N = \{\alpha \in \mathbb{Z}_N \mid \alpha^{N-1} = 1 \text{ and if } \alpha^{u2^{j+1}} = 1 \text{ then } \alpha^{u2^j} = 1\}$$

Algorithm 3: Miller-Rabin Primality Test

```
Handle base case N = 2
for t times do
    Pick a random α ∈ Z_N
    if α ∉ L_N then Output "composite"
Output "prime"
```

RABIN-MILLER

$$L_N = \{\alpha \in \mathbb{Z}_N \mid \alpha^{N-1} = 1 \text{ and if } \alpha^{u2^{j+1}} = 1 \text{ then } \alpha^{u2^j} = 1\}$$

Algorithm 3: Miller-Rabin Primality Test

```
Handle base case N = 2
for t times do
    Pick a random α ∈ Z_N
    if α ∉ L_N then Output "composite"
Output "prime"
```

Theorem 38.1. *If N is composite, then the Miller-Rabin test outputs "composite" with probability $1 - 2^{-t}$. If N is prime, then the test outputs "prime."*

EULER TOTIENT

phi

$\Phi(n)$: # of positive integers that are smaller & relatively prime to n

$\phi(7) =$ 1 2 3 4 5 6 7

$\phi(p) = p-1$

$\gcd(7,1)=1$

EULER TOTIENT

$$\Phi(n)$$

$\phi(p) = p-1$ (p → prime)

$\phi(n) = \phi(p) \cdot \ldots \cdot \phi(q)$

$\phi(15) = 8 = 2 \cdot 4 = \phi(3) \cdot \phi(5)$

15 ↳ $3 \cdot 5$

1 2 3 4 5 6 7 8 9 10 11 12 13 14 → Z_n^*

EULER TOTIENT

$$|\mathbb{Z}_n^\star| = \Phi(n)$$

prime $\Phi(p) = p - 1$

product of 2 primes $\Phi(n) = (p-1)(q-1)$

EULER THEOREM

if $\gcd(a, n) = 1$

$$a^{\Phi(n)} = 1 \bmod n$$

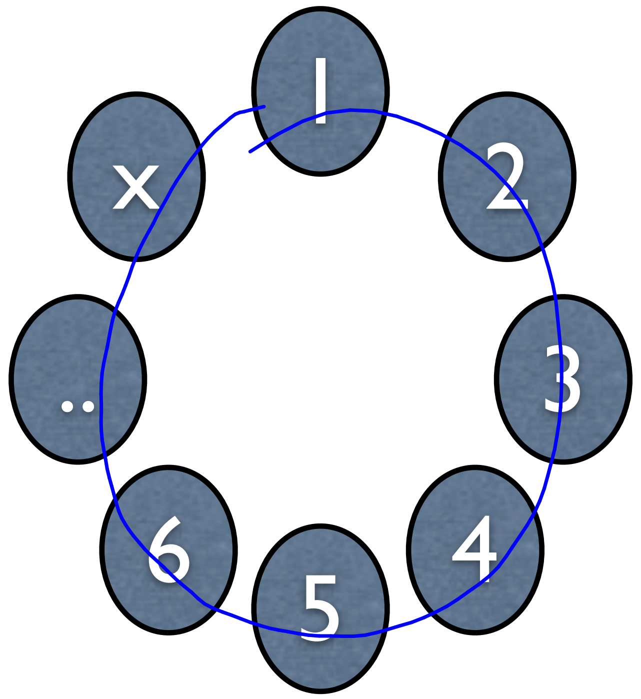
(Why RSA works)

EULER THEOREM

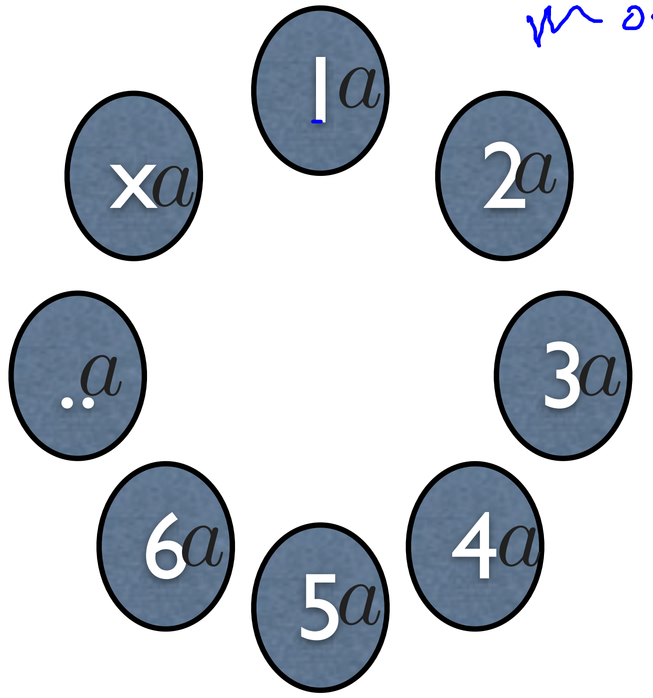
$\phi(n)$

$1 \ldots$

$$\forall a \in \mathbb{Z}_N^\star,\ a^{\Phi(N)} = 1 \bmod N$$

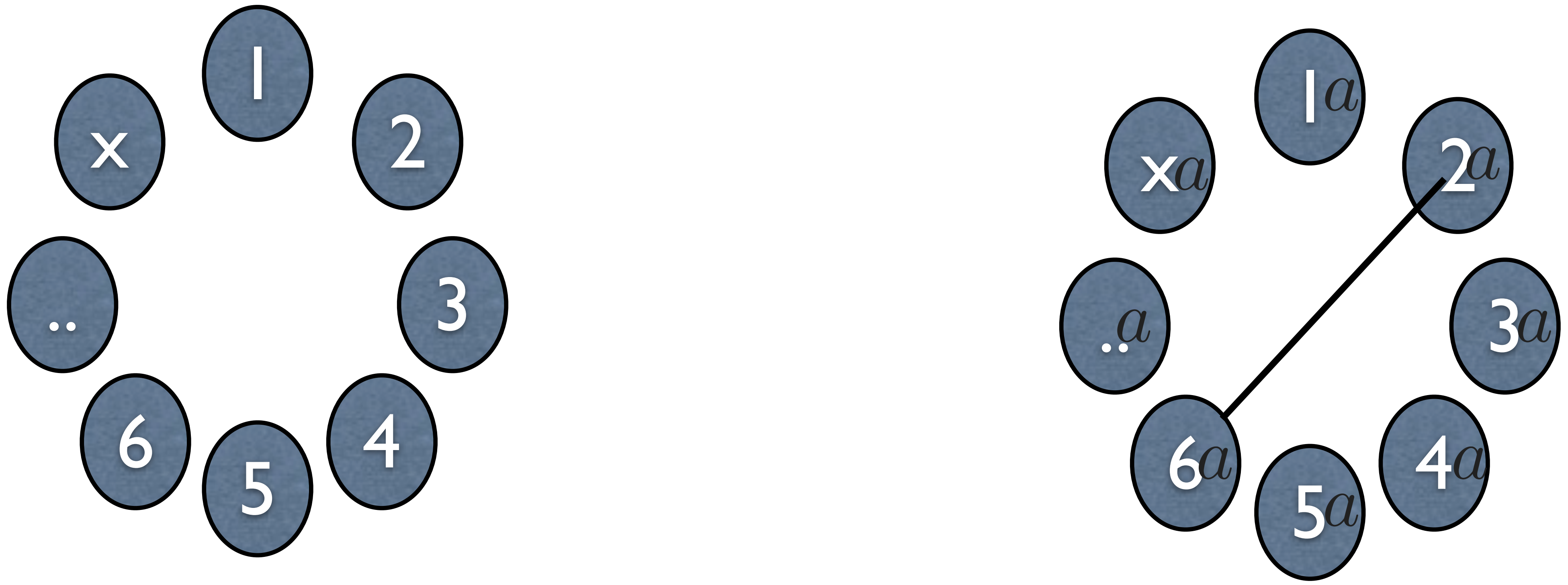


mod N.



EULER THEOREM

$$\forall a \in \mathbb{Z}_N^\star,\ a^{\Phi(N)} = 1 \bmod N$$



argue: all are distinct

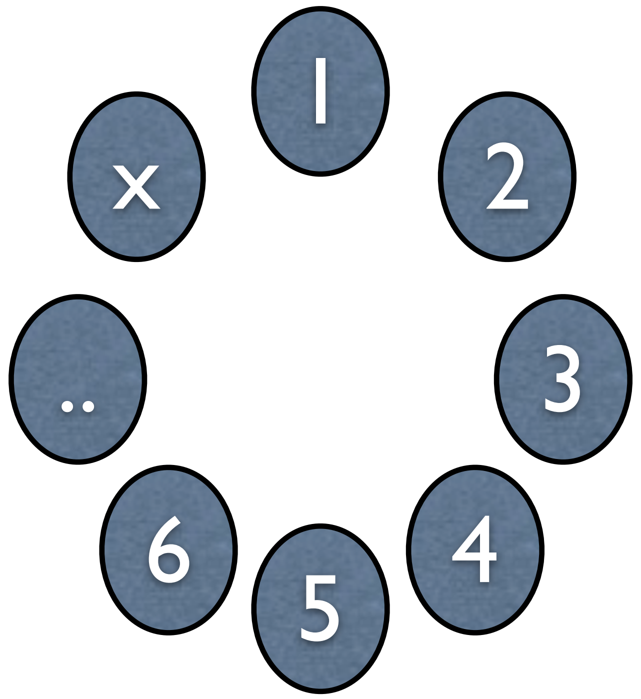
spse two are equal.

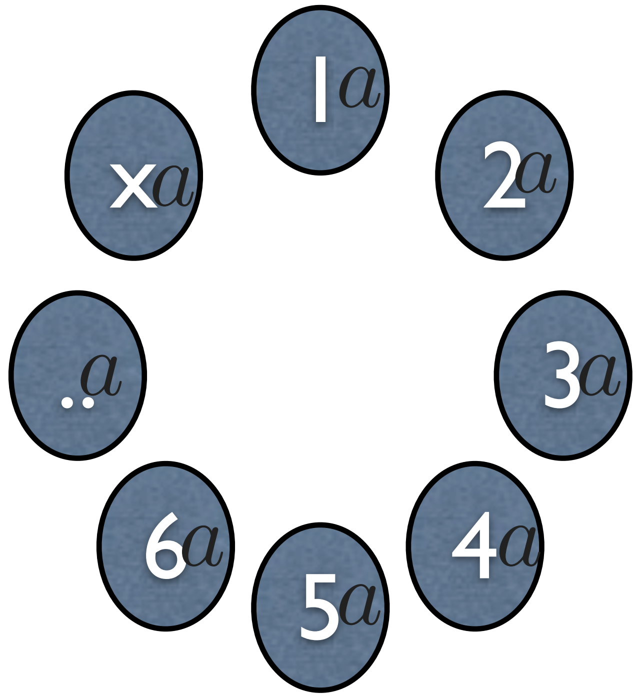
multiply by a^{-1}

this implies 2=6!

EULER THEOREM

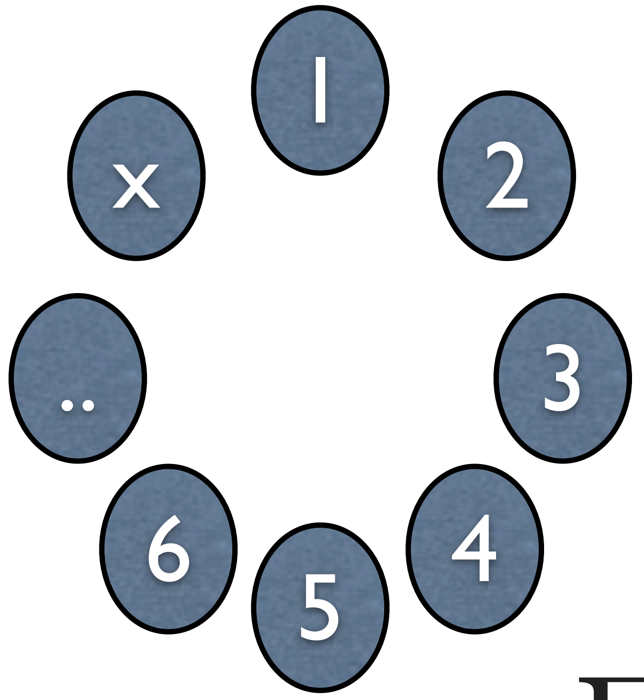
$$\forall a \in \mathbb{Z}_N^\star,\ a^{\Phi(N)} = 1 \bmod N$$

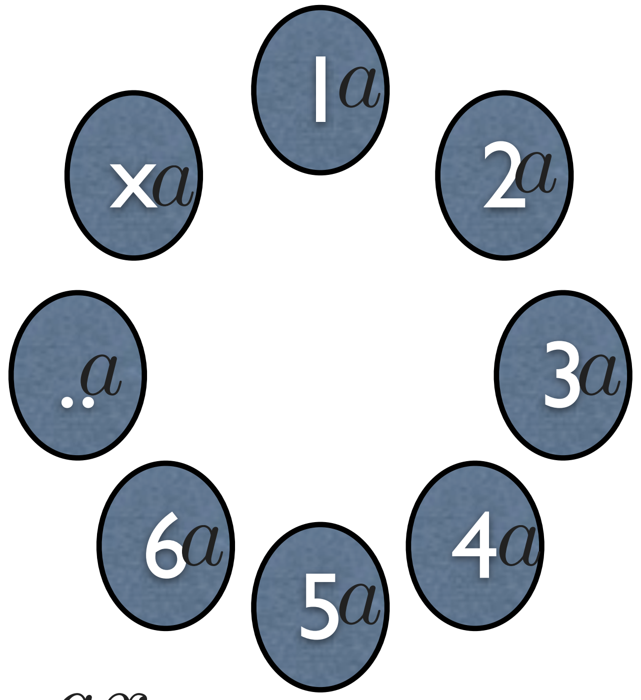




EULER THEOREM

$$\forall a \in \mathbb{Z}_N^\star,\ a^{\Phi(N)} = 1 \bmod N$$

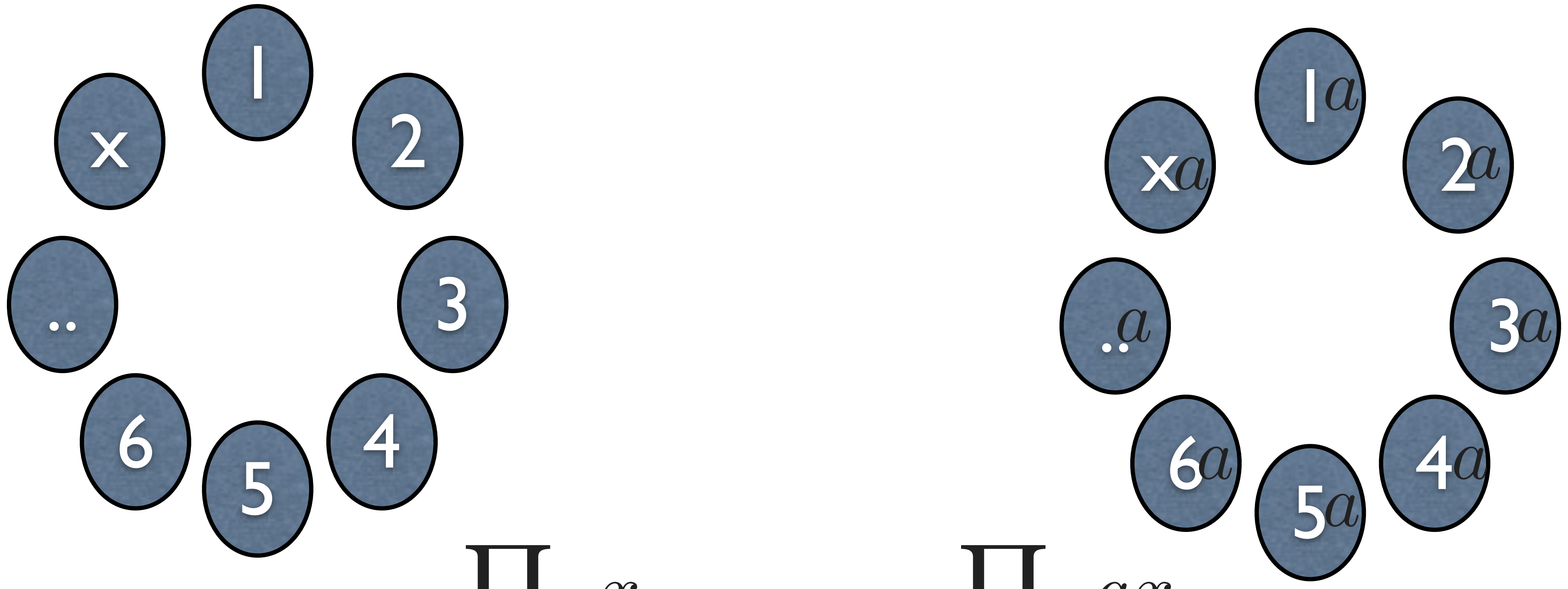




$$\prod_{x \in Z_N^\star} x \qquad \prod_{x \in Z_N^\star} ax$$

EULER THEOREM

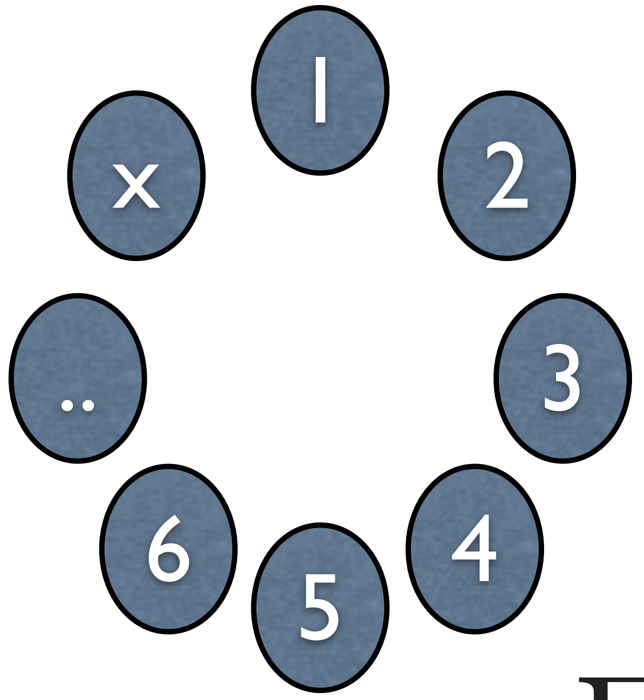
$$\forall a \in \mathbb{Z}_N^\star,\ a^{\Phi(N)} = 1 \bmod N$$

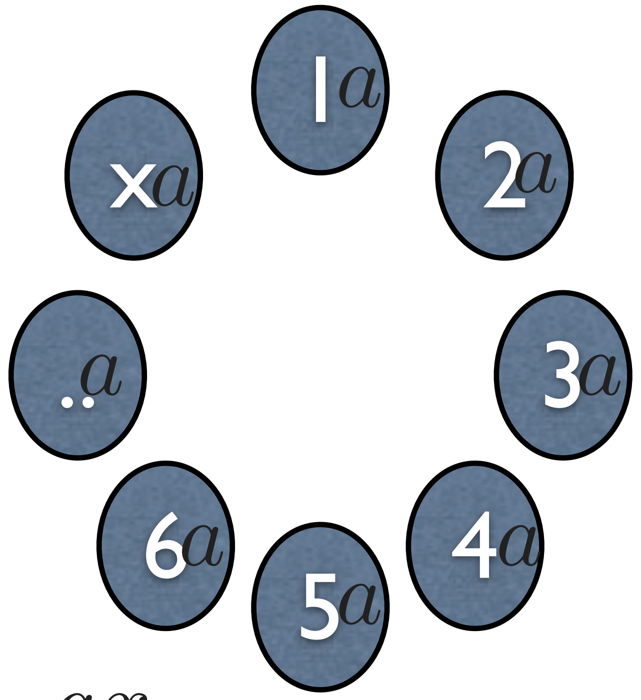


$$\prod_{x \in Z_N^\star} x = \prod_{x \in Z_N^\star} ax$$

EULER THEOREM

$$\forall a \in \mathbb{Z}_N^\star,\ a^{\Phi(N)} = 1 \bmod N$$



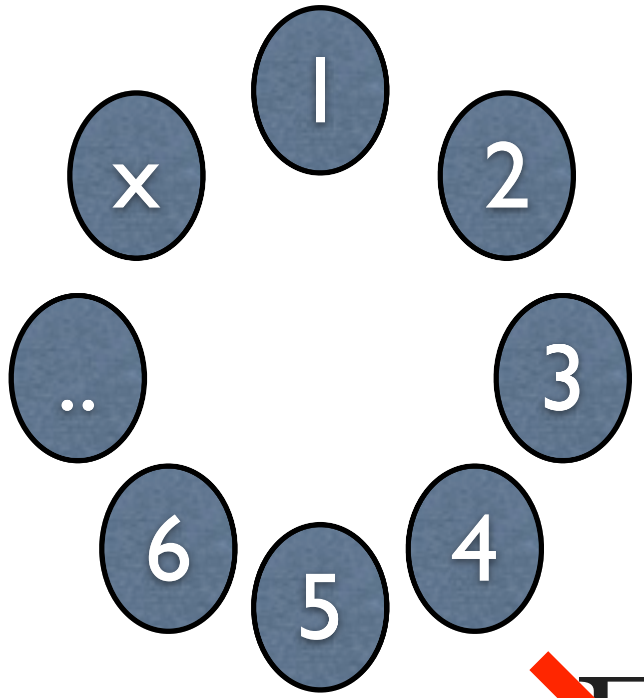


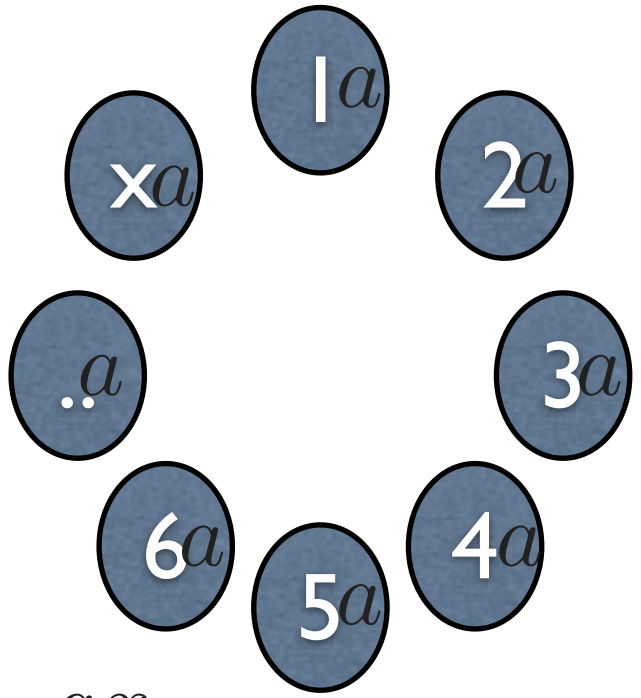
$$\prod_{x \in Z_N^\star} x = \prod_{x \in Z_N^\star} ax$$

$$a^{\Phi(N)} \prod_{x \in Z_N^\star} x$$

EULER THEOREM

$$\forall a \in \mathbb{Z}_N^\star,\ a^{\Phi(N)} = 1 \bmod N$$





$$\prod_{x \in Z_N^\star} x = \prod_{x \in Z_N^\star} ax$$

$$a^{\Phi(N)} \prod_{x \in Z_N^\star} x$$

$\prod x$ $\prod ax$

$\prod x = \prod a \cdot \prod x$

$1 = a^{\phi(n)}$

TEXTBOOK RSA

GEN(1^n)
① pick 2 primes p, q ~ 1000 bit #s

② $N = p \cdot q$ ③ $\phi(N) = (p-1)(q-1)$

④ pick e s.t. $\gcd(e, \phi(N)) = 1$ $e = 17, 65537$

⑤ use euclid to compute d s.t. $e \cdot d = 1 \bmod \phi(N)$

$e \cdot d + k \cdot \phi(N) = 1$

TEXTBOOK RSA

GEN(1^n)

N = pq $\quad \Phi(N) = (p-1)(q-1)$

e is a number such that $\gcd(e, \Phi(N)) = 1$

d is such that $e \cdot d = 1 \bmod \Phi(N)$

$pk = (N, e)$

$sk = (N, d)$

ENC$_{pk}$(m) : $m^e \bmod N$ (pk: e, N)

$Dec_{sk}(c) = c^d \bmod N$ (sk: d, N)

$Dec(Enc(m)) =$

$Dec(m^e) = m^{ed} \bmod N$

$= m^{k \cdot \Phi(N)+1} \bmod N$

$= (m^{\Phi(n)})^k \cdot m \bmod N$

$= 1^k \cdot m \bmod N$

N = 949 E=11 D=707

TEXTBOOK RSA

GEN(1^n)

$$N \leftarrow pq, p, q \in \Pi_n, e \in \mathbb{Z}^*_{\phi(n)}$$

$$pk \leftarrow (N, e) \qquad sk \leftarrow (N, d)$$

ENC$_{pk}$(m)

$$c \leftarrow m^e \bmod N$$

DEC$_{sk}$(c)

$$m \leftarrow c^d \bmod N$$

JUNE 1942

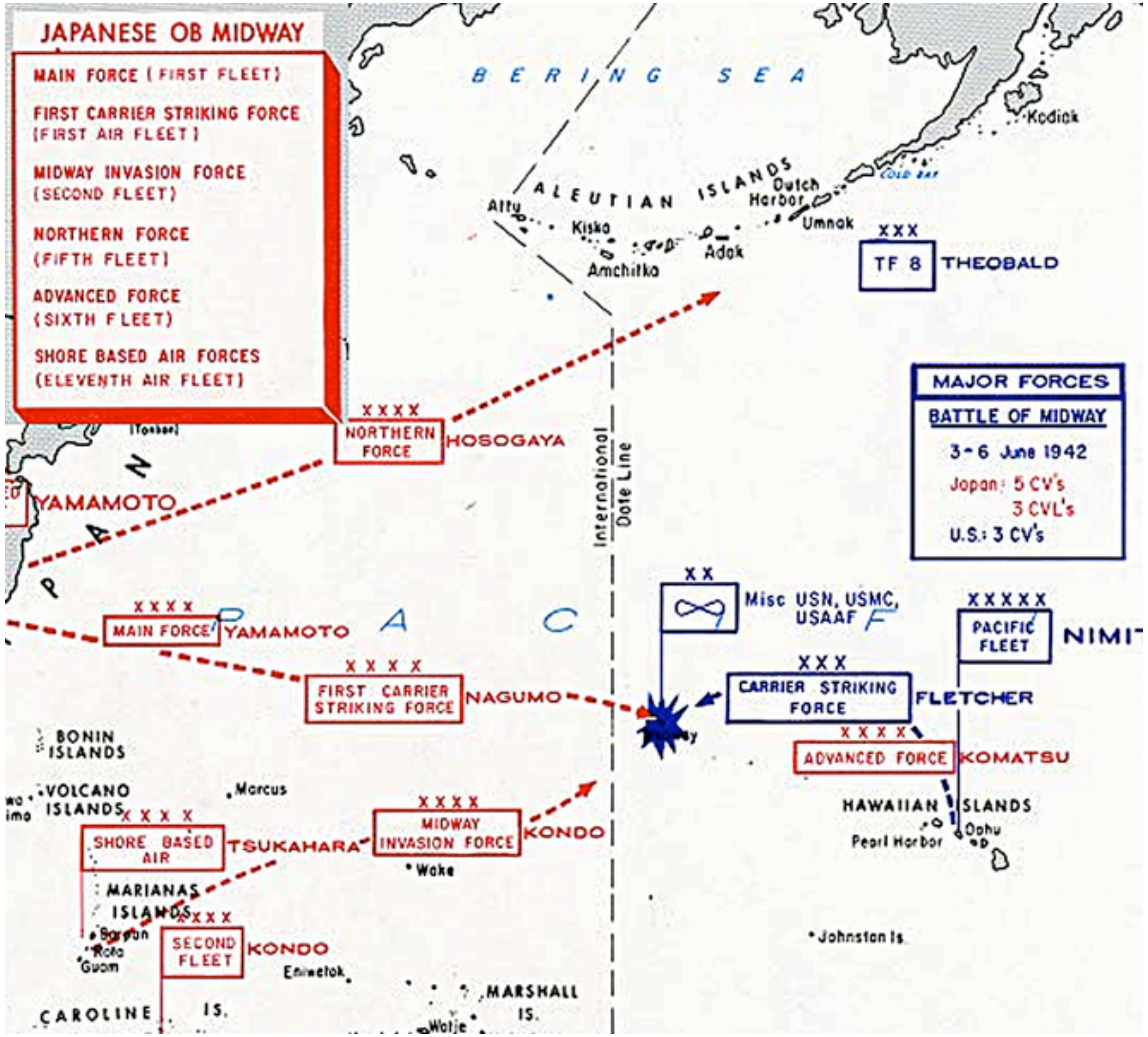
JN-25B

CMDR EDWARD T LAYTON

(FLEET INTELLIGENCE OFFICER)

LT CMDR JOSEPH ROCHEFORT

(COMBAT INTELLIGENCE UNIT)


JAPANESE OB MIDWAY
MAIN FORCE (FIRST FLEET)
FIRST CARRIER STRIKING FORCE (FIRST AIR FLEET)
MIDWAY INVASION FORCE (SECOND FLEET)
NORTHERN FORCE (FIFTH FLEET)
ADVANCED FORCE (SIXTH FLEET)
SHORE BASED AIR FORCES (ELEVENTH AIR FLEET)
BERING SEA
ALEUTIAN ISLANDS
Kodiak
Dutch Harbor
Umnak
Attu
Kiska
Adak
Amchitka
XXX
TF 8
THEOBALD
MAJOR FORCES
BATTLE OF MIDWAY
3-6 June 1942
Japan: 5 CV's
3 CVL's
U.S.: 3 CV's
XXXX
NORTHERN FORCE
HOSOGAYA
YAMAMOTO
International Date Line
XX
Misc USN, USMC, USAAF
XXXXX
PACIFIC FLEET
XXXX
MAIN FORCE
YAMAMOTO
XXXX
FIRST CARRIER STRIKING FORCE
NAGUMO
XXX
CARRIER STRIKING FORCE
FLETCHER
XXXX
ADVANCED FORCE
KOMATSU
BONIN ISLANDS
VOLCANO ISLANDS
Marcus
XXXX
SHORE BASED AIR
TSUKAHARA
XXXX
MIDWAY INVASION FORCE
KONDO
Wake
HAWAIIAN ISLANDS
Oahu
Pearl Harbor
MARIANAS ISLANDS
XXXX
SECOND FLEET
KONDO
Saipan
Rota
Guam
Eniwetok
Johnston Is.
MARSHALL IS.
CAROLINE IS.
Wotje

secure encryption schemes need to use randomness!

PKCS1.5

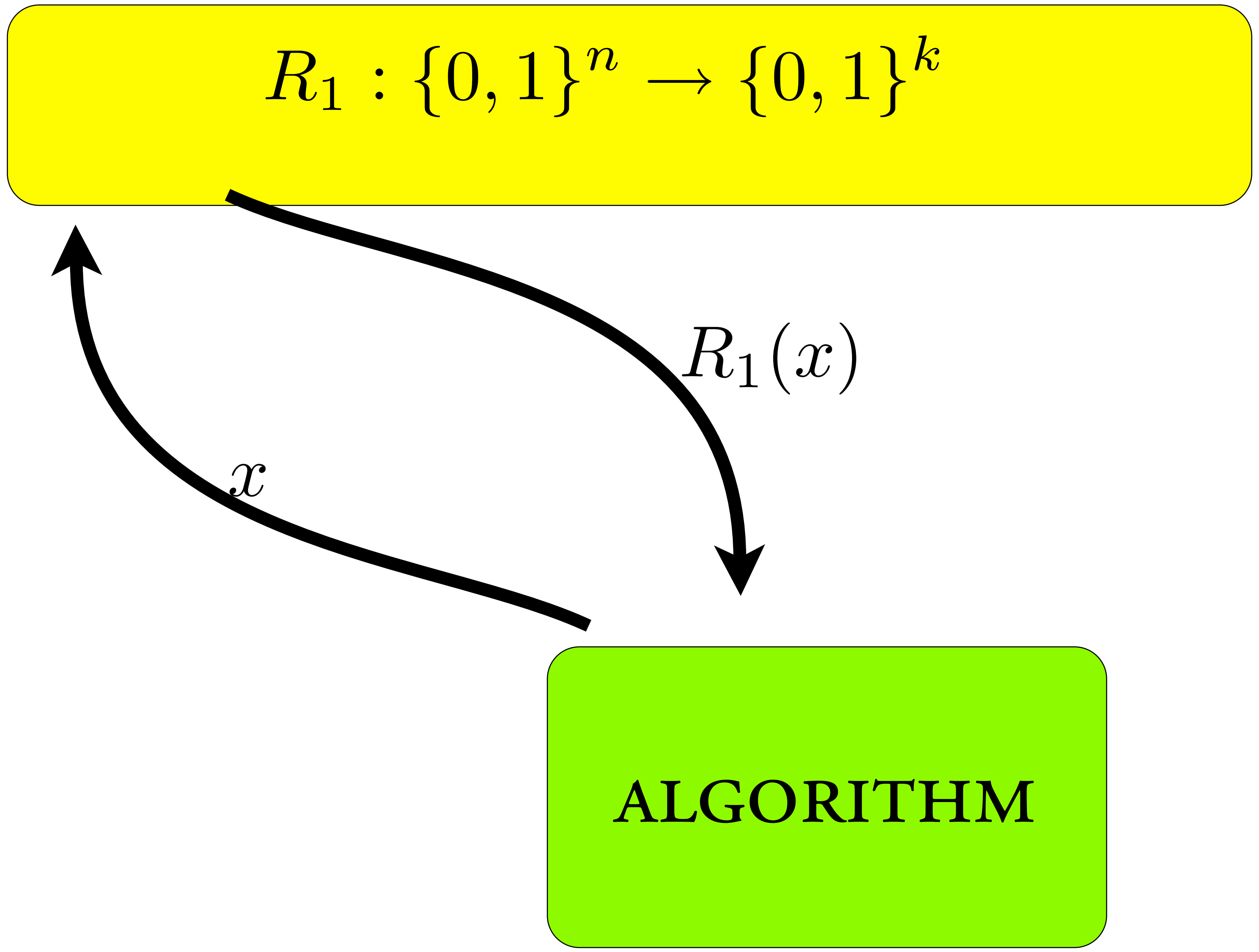
ENC$_{pk}$(m)

PICK r AS A RANDOM STRING WITH NO 0s (TYPICALLY 8 BYTES)

$$c \leftarrow (0||2||r||0||m)^e \bmod N$$

CCA2 ATTACK AGAINST THIS SCHEME

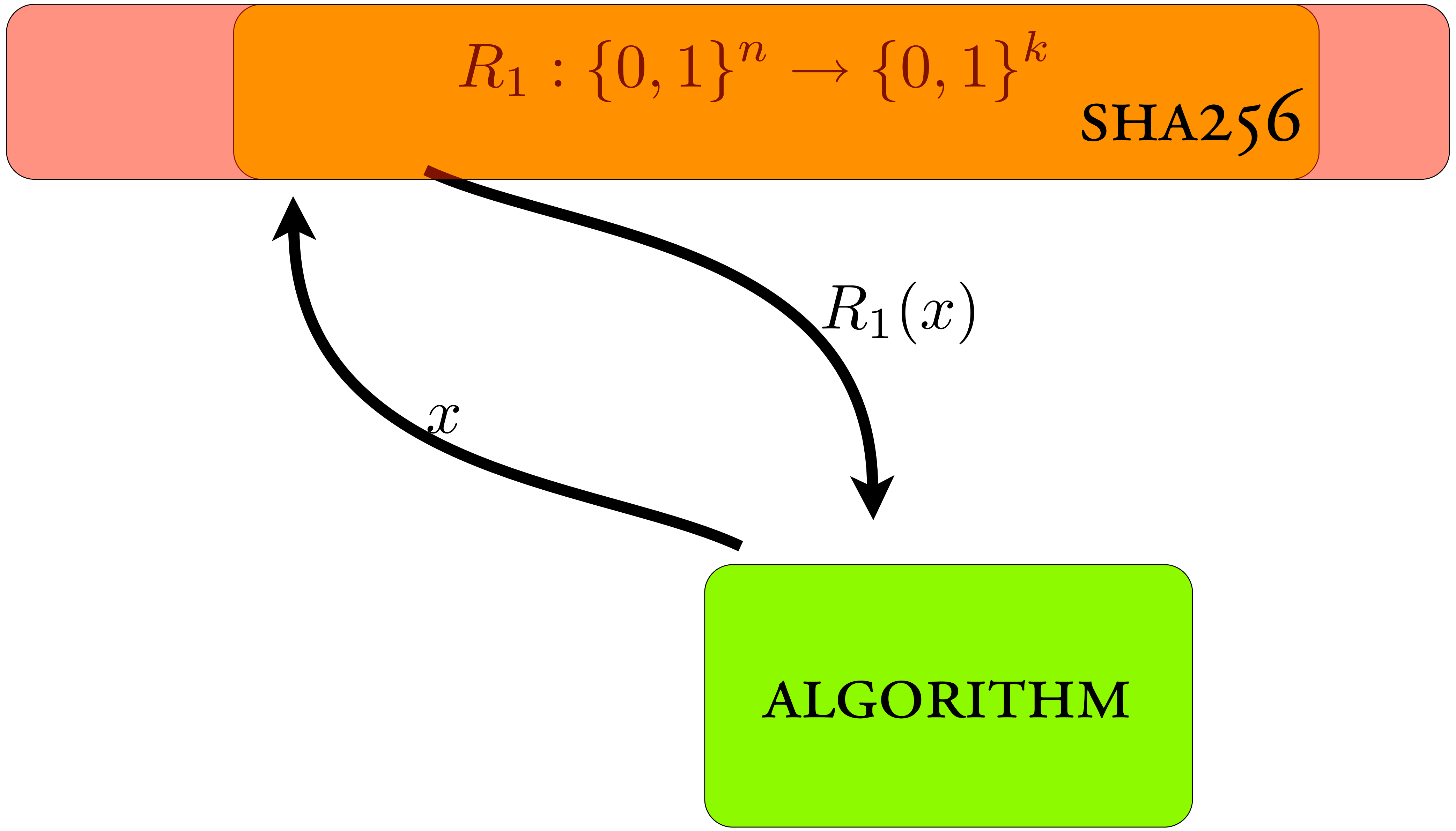
RANDOM ORACLE MODEL



PUBLIC FUNCTION. NOT KEYED.

ANYONE CAN EVALUATE, OUTPUT IS UNPREDICTABLE.

RANDOM ORACLE MODEL



HEURISTIC SECURITY ONLY

CANNOT BE ALWAYS BE SECURELY INSTANTIATED

OAEP+

GEN(1^n)

$$f, f^{-1} \leftarrow \text{TRAPDOOR OWP}()$$

ENCpk(m)

$$r \leftarrow U_n$$
$$s \leftarrow R_1(r) \oplus m \;||\; R_2(r||m)$$
$$t \leftarrow R_3(s) \oplus r$$
$$c \leftarrow f(s||t)$$

$$R_1 : \{0,1\}^{k_0} \rightarrow \{0,1\}^n$$
$$R_2 : \{0,1\}^{n+k_0} \rightarrow \{0,1\}^{k_1}$$
$$R_3 : \{0,1\}^{n+k_1} \rightarrow \{0,1\}^{k_0}$$

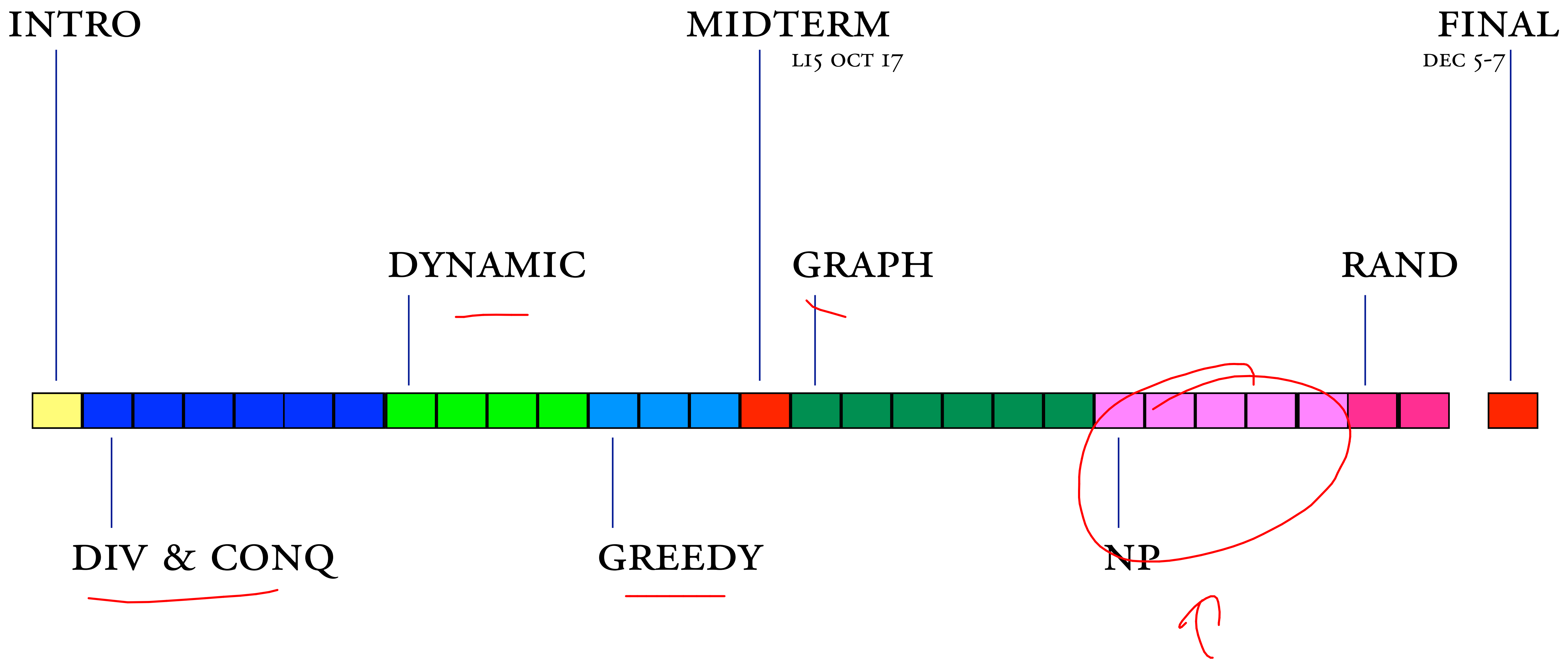
DECsk(c)

$$(s = (s_1, s_2), t) \leftarrow f^{-1}(c)$$
$$r \leftarrow R_3(s) \oplus t$$
$$m \leftarrow R_1(r) \oplus s_1$$
$$R_2(r||m) \stackrel{?}{=} s_2 \quad \text{OUTPUT m ELSE FAIL}$$

Theme

“small problems are easy to solve.”

“solve big problems by making them into smaller ones.”


INTRO
MIDTERM
L15 OCT 17
FINAL
DEC 5-7
DYNAMIC
GRAPH
RAND
DIV & CONQ
GREEDY
NP

TOPICS

D&C	DP	GREEDY	GRAPH	RAND	NPC	OTHER
MULT	LOG	SCHED	MST	MATCH	RED	GPU
QUICK	CHAIN	HUFF	BFS	FINGER	IND	DISSENT
CLOSE	TYPESET	ESPRESSO	DIJKSTRA	STRING	VC	PQ
MEDIAN	GERRY	CACHING	BELL-FORD	ENC	3COL	
FFT	ZAP		ALLSHORT		SUBSET	
MATMUL	POSTER		MAXFLOW		SET	
MASTERS	TUG		BIPARTITE			
BUS			EDGE-DISJ			
NIFTY			BASEBALL			
			ASSIGNMENT			
			STABLE			

first goal: create an amazing learning experience second goal: instill my enthusiasm for this area third goal: enjoy every second of this semester