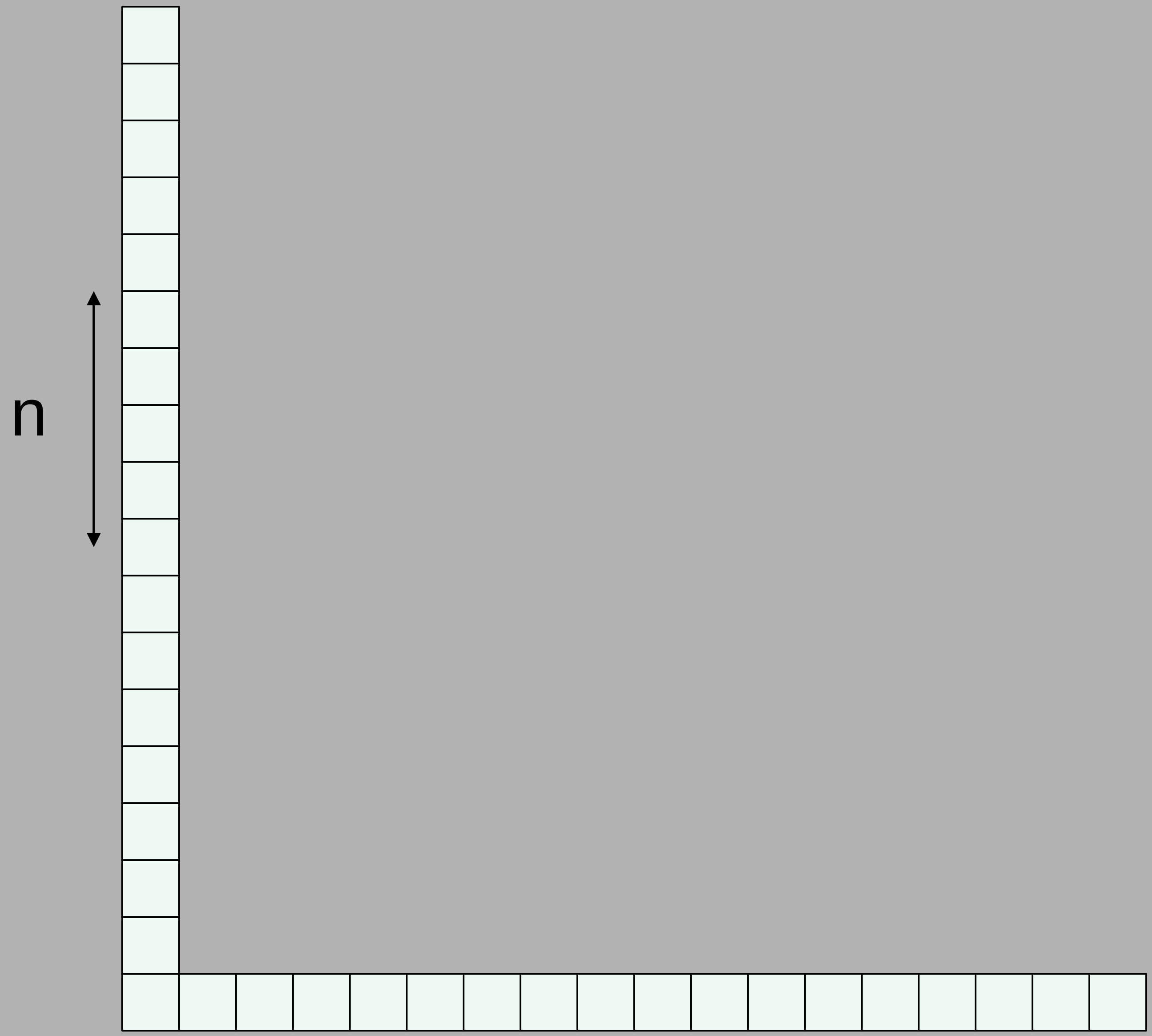
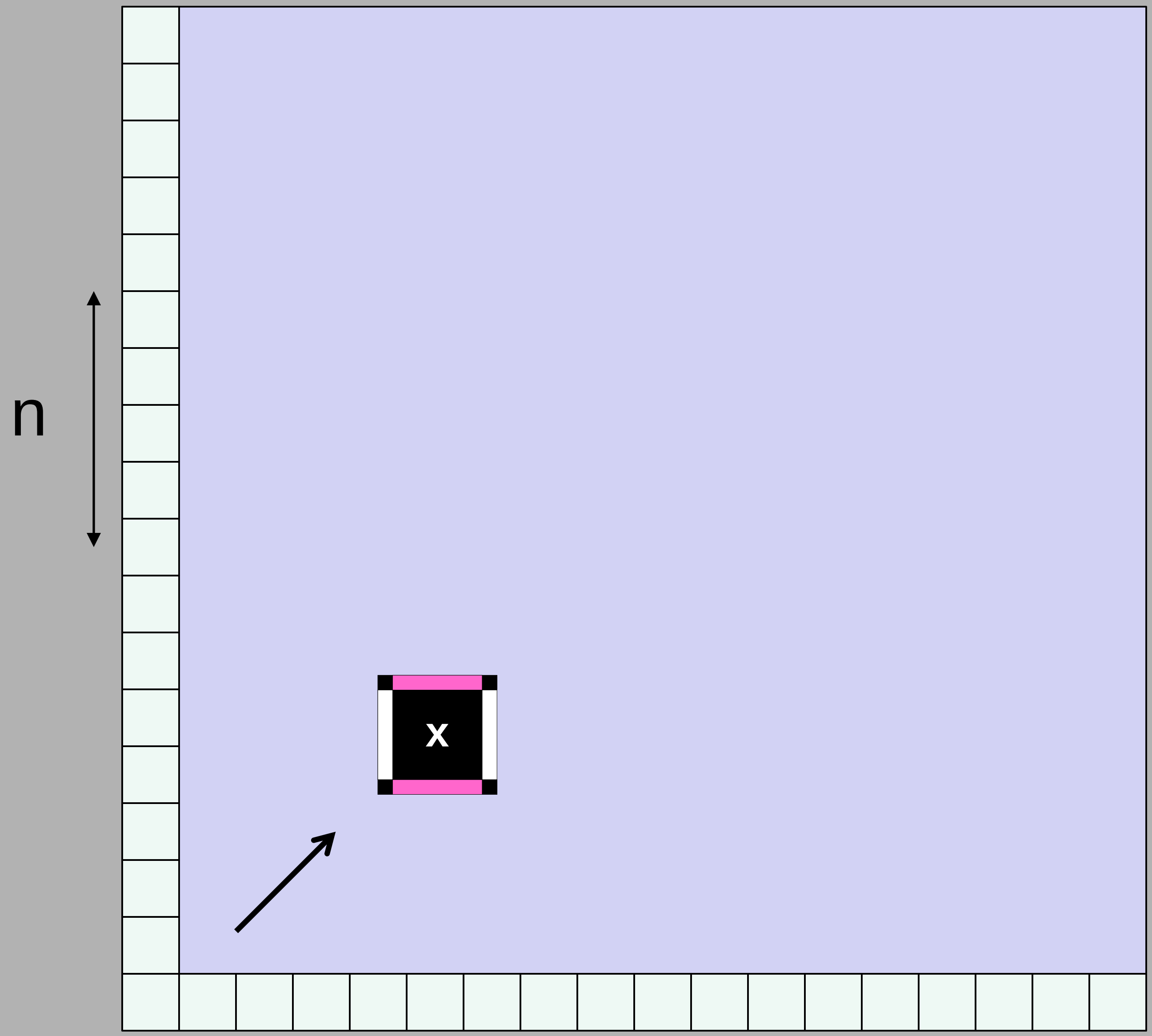
How efficiently can you build an n x n square?



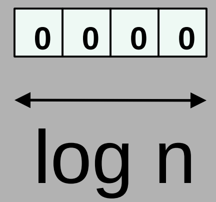
How efficiently can you build an n x n square?



Tile Complexity:
2n

How efficiently can you build an n x n square?

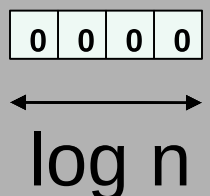
-Use log n tile types to seed counter:



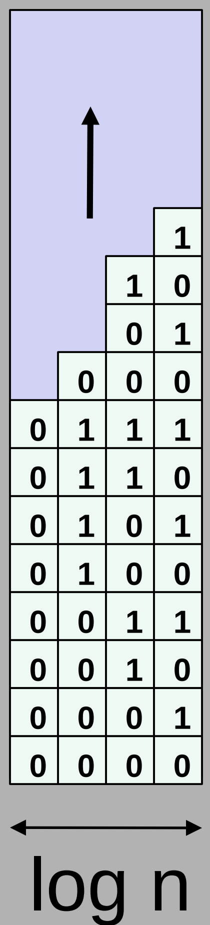
How efficiently can you build an n x n square?

-Use **log n** tile types capable of Binary counting:

-Use **8** additional tile types capable of binary counting:



How efficiently can you build an n x n square?



-Use **log n** tile types capable of Binary counting:

-Use **8** additional tile types capable of binary counting:

How efficiently can you build an n x n square?

-Use **log n** tile types capable of Binary counting:

-Use **8** additional tile types capable of binary counting:

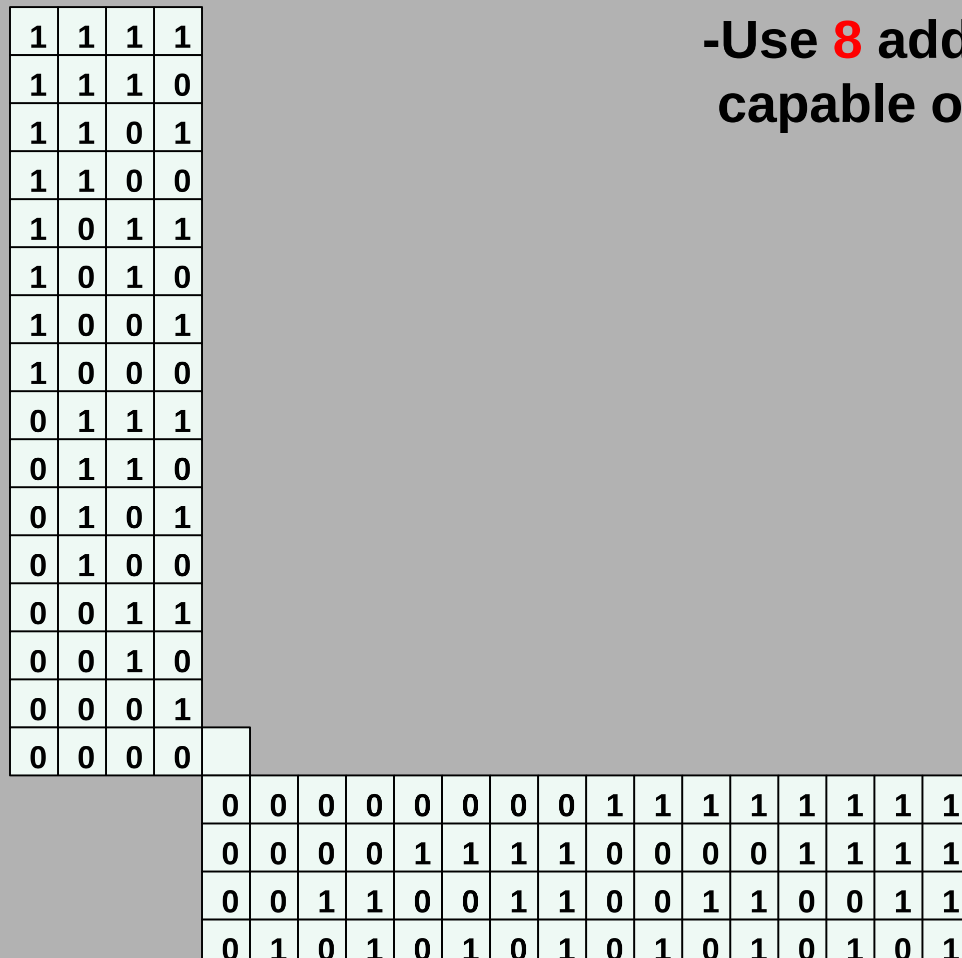
1	1	1	1
1	1	1	0
1	1	0	1
1	1	0	0
1	0	1	1
1	0	1	0
1	0	0	1
1	0	0	0
0	1	1	1
0	1	1	0
0	1	0	1
0	1	0	0
0	0	1	1
0	0	1	0
0	0	0	1
0	0	0	0

log n

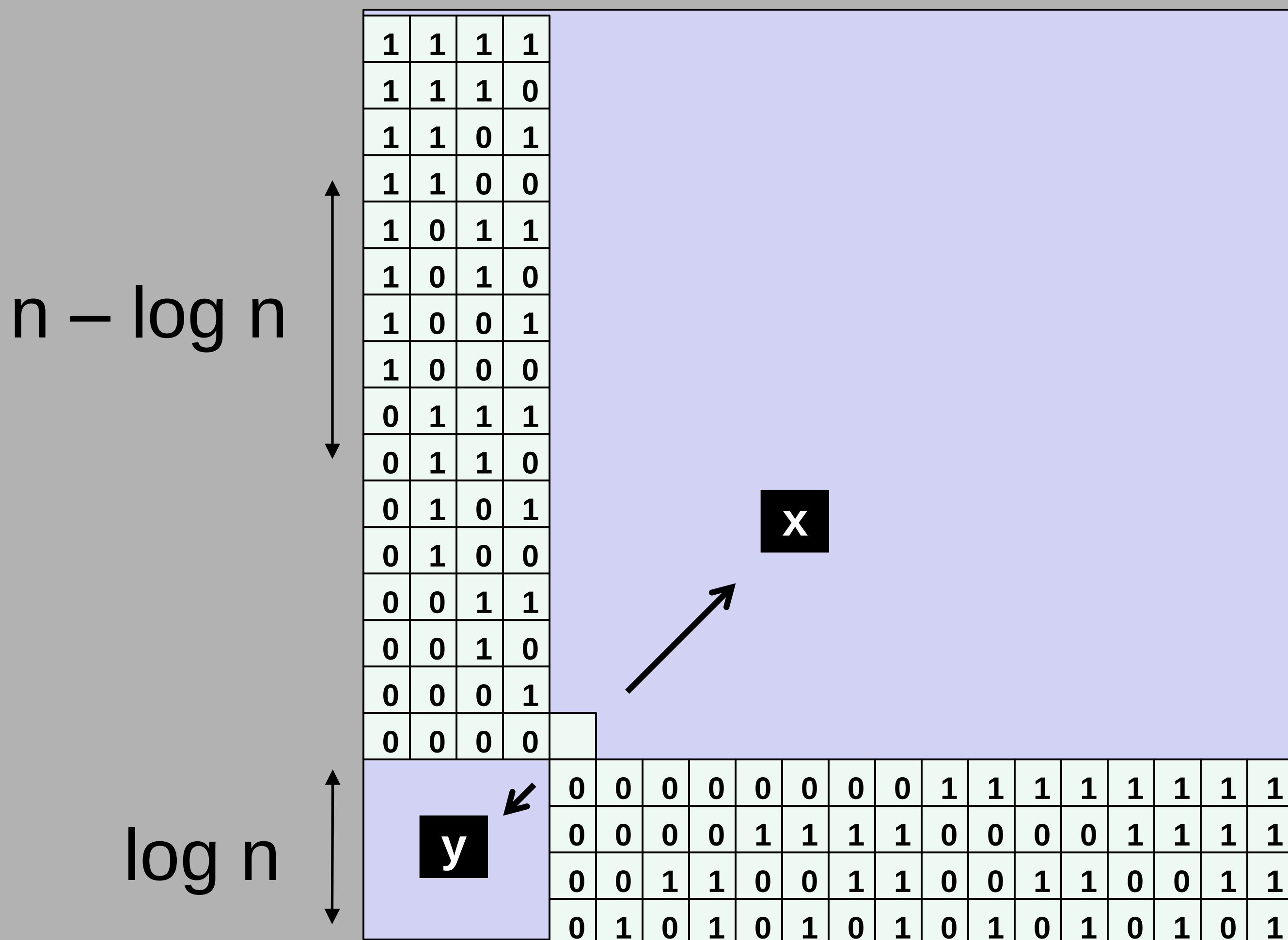
How efficiently can you build an n x n square?

-Use **log n** tile types capable of Binary counting:

-Use **8** additional tile types capable of binary counting:



How efficiently can you build an n x n square?



Tile Complexity:

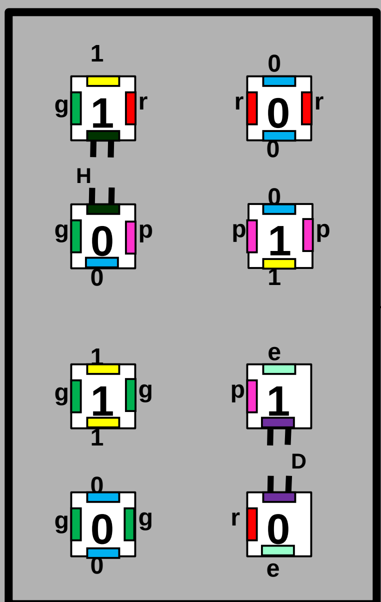
O(log n)

(Rothemund, Winfree 2000)

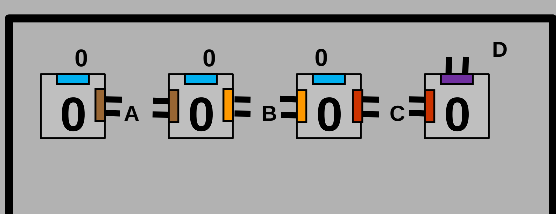
Binary Counter

(Rothemund, Winfree 2000)

Tile set:

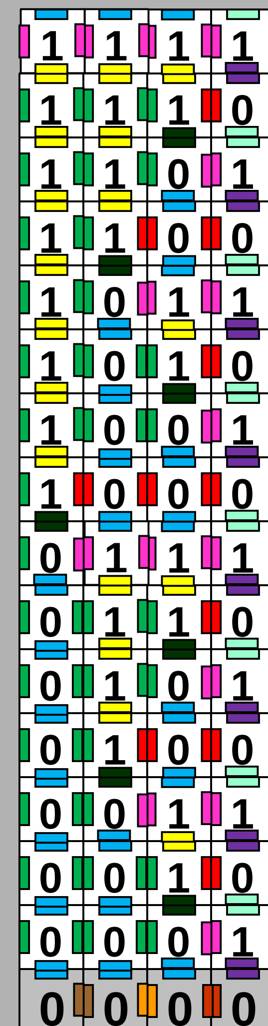


8 tile types



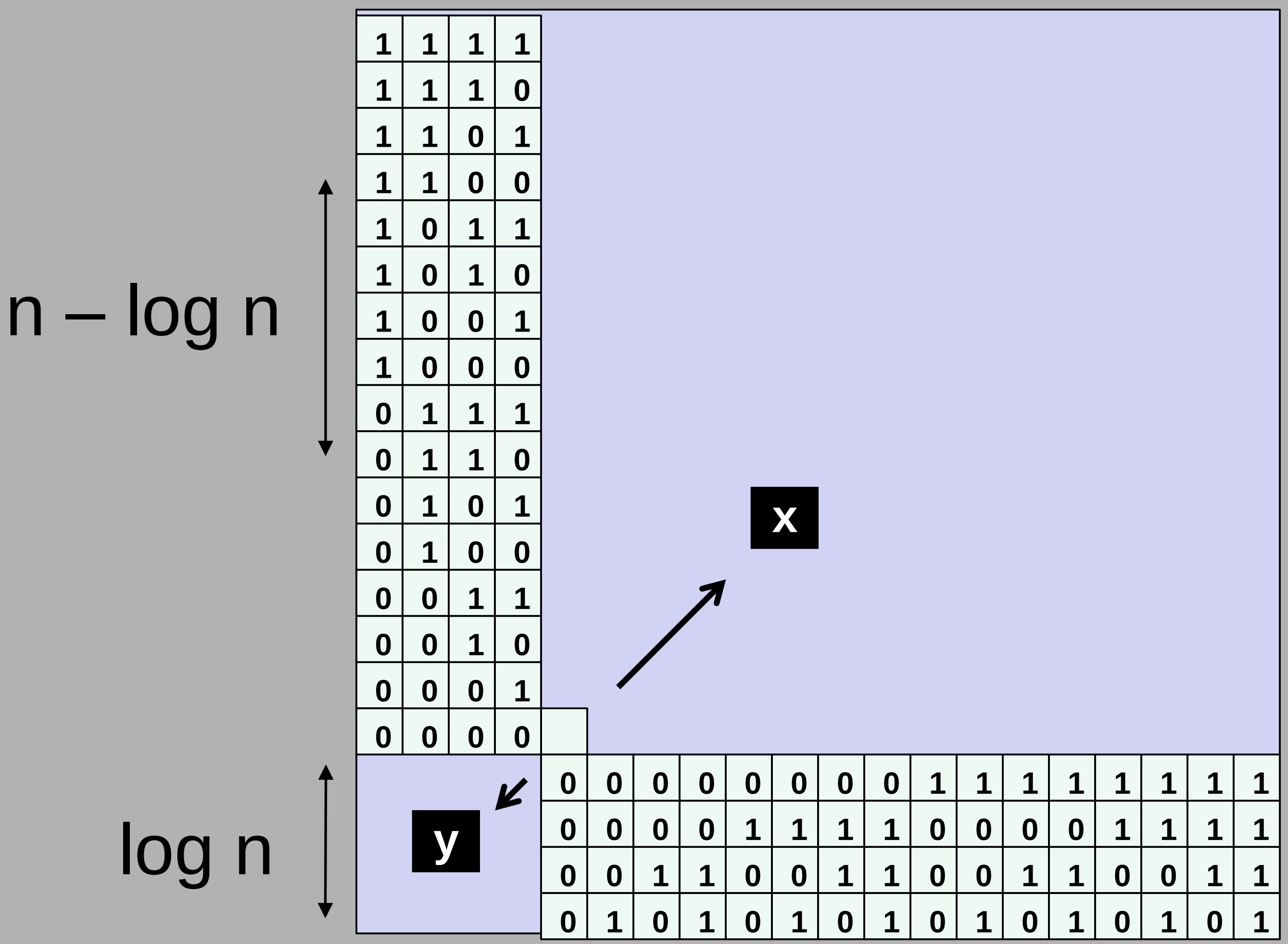
log n
tile types

Assembly:



n x log n rectangle:
O(log n)
tile complexity

How efficiently can you build an n x n square?



Tile Complexity:

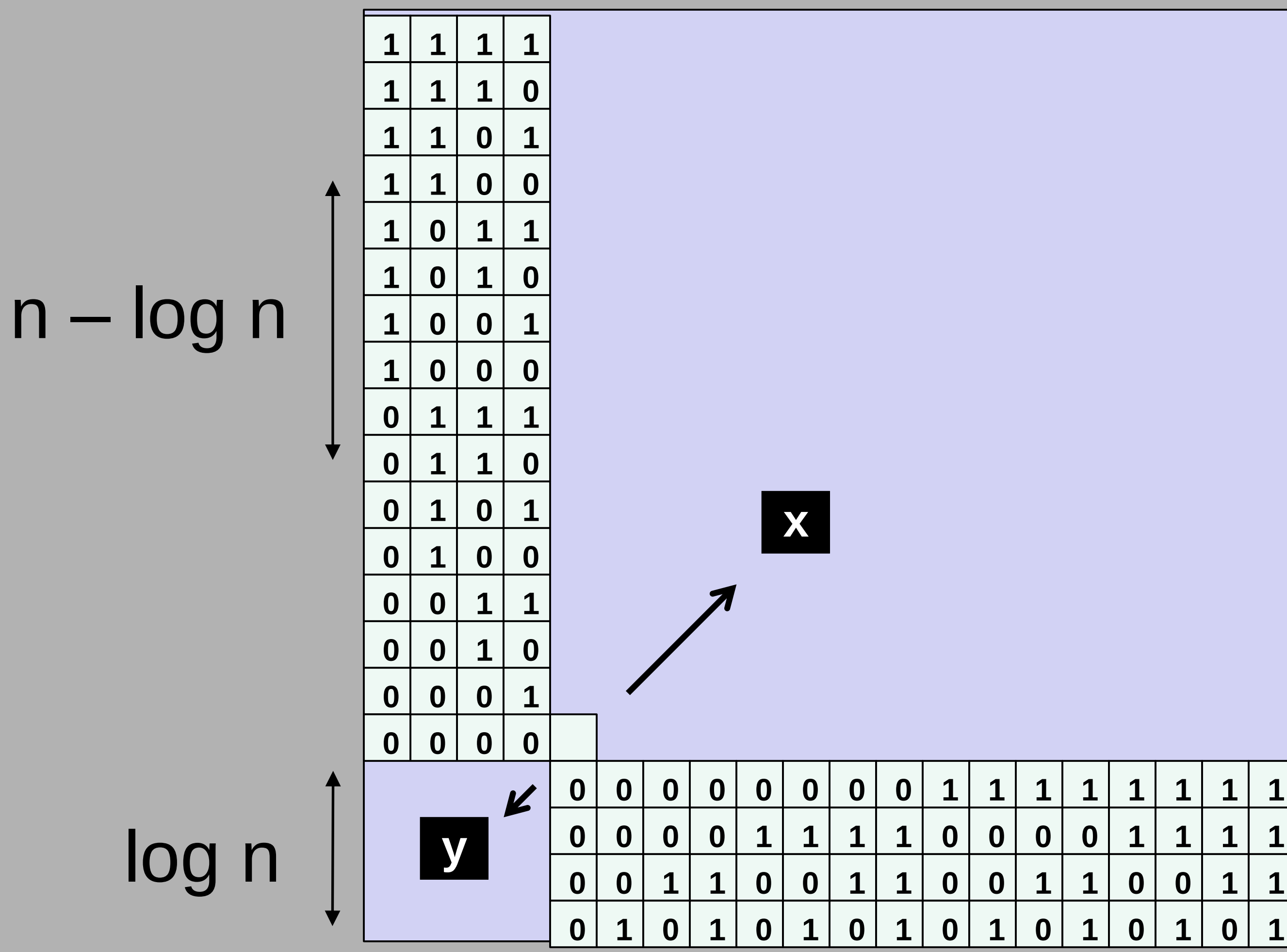
O(log n)

(Rothemund, Winfree 2000)

Can this be beat?

What is a lower bound?

How efficiently can you build an n x n square?



Tile Complexity:
O(log n)
(Rothemund, Winfree 2000)

Can this be beat?

What is a lower bound?

Challenge:
Derive information theoretic lower bound

Wow. Can we meet that?

Lower Bound:

(almost all n)

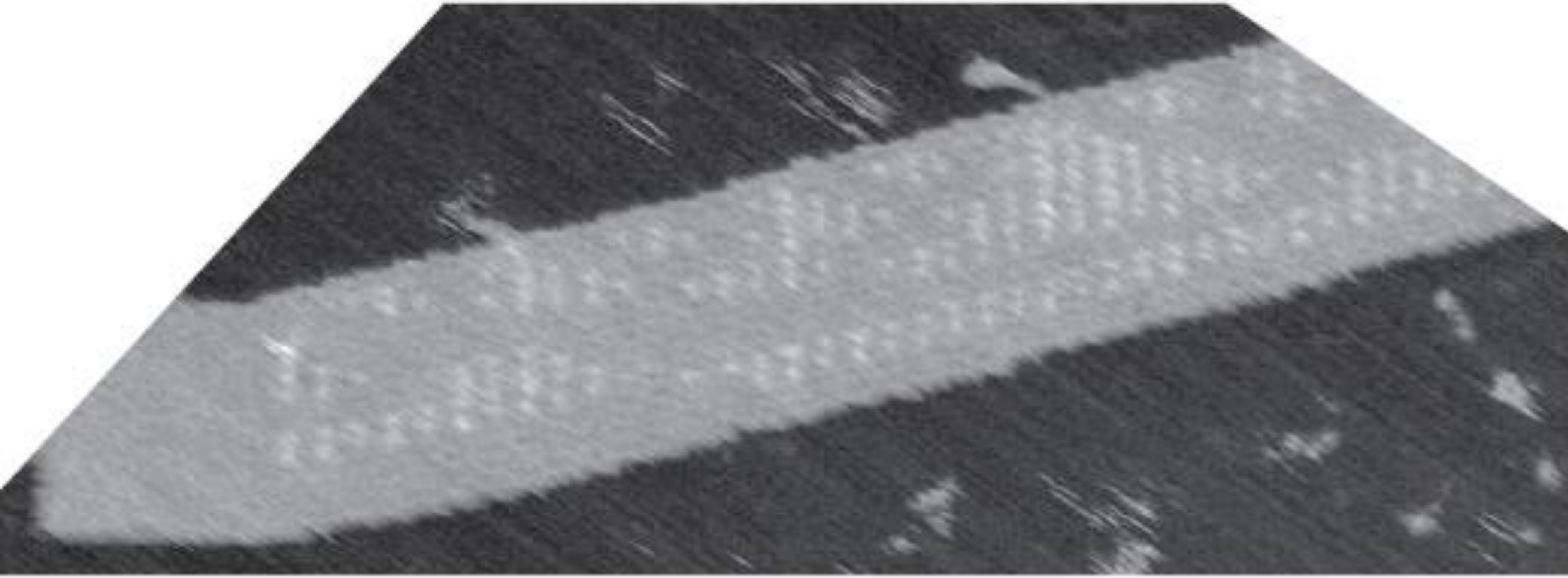
$$\Omega(\frac{\log n}{\log \log n})$$

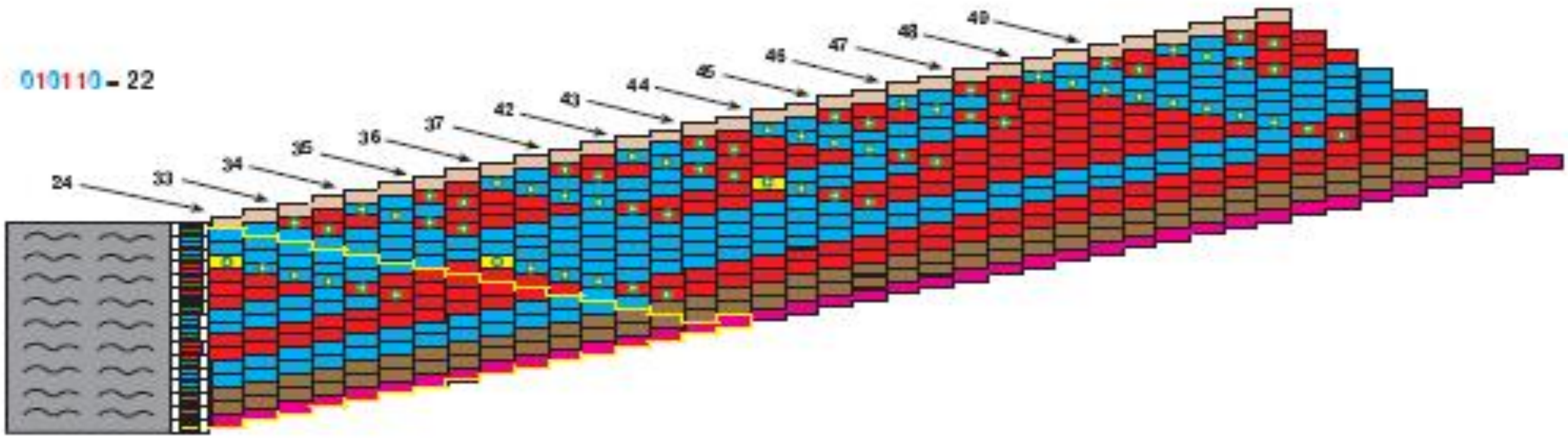
(Rothemund, Winfree 2000)

Current Upper Bound:

O(log n)

(Rothemund, Winfree 2000)





Barish, Shulman, Rothemund, Winfree, 2009

Build a 2 x 16 rectangle:

t = 2

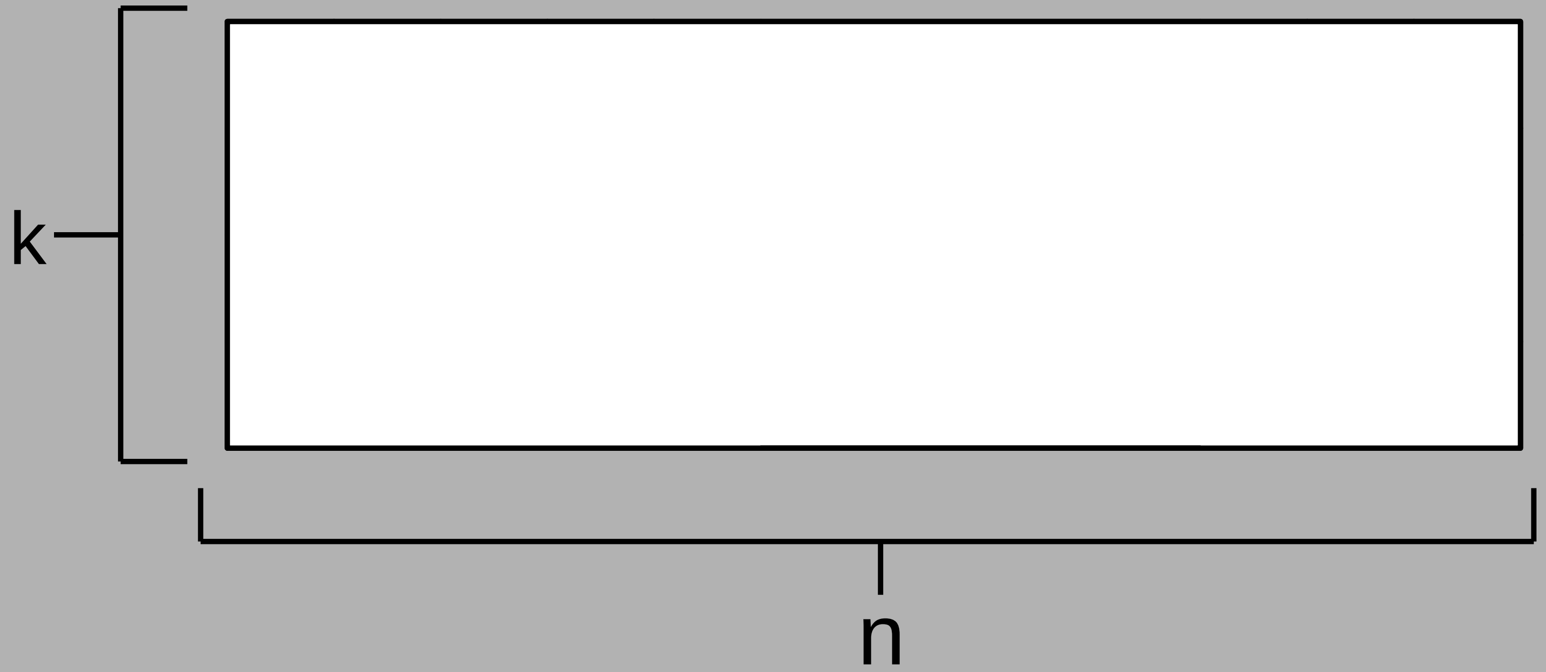
S_1	0	0	0	1	1	1	1	2	2	2	2	3	3	3	P
S	C_1	C_2	C_3	C_0	C_1	C_2	C_3	C_0	C_1	C_2	C_3	C_0	C_1	C_2	C_3

2 x n lines

Lower Bound	Upper Bound
$\Omega(\sqrt{n})$	$O(\sqrt{n})$

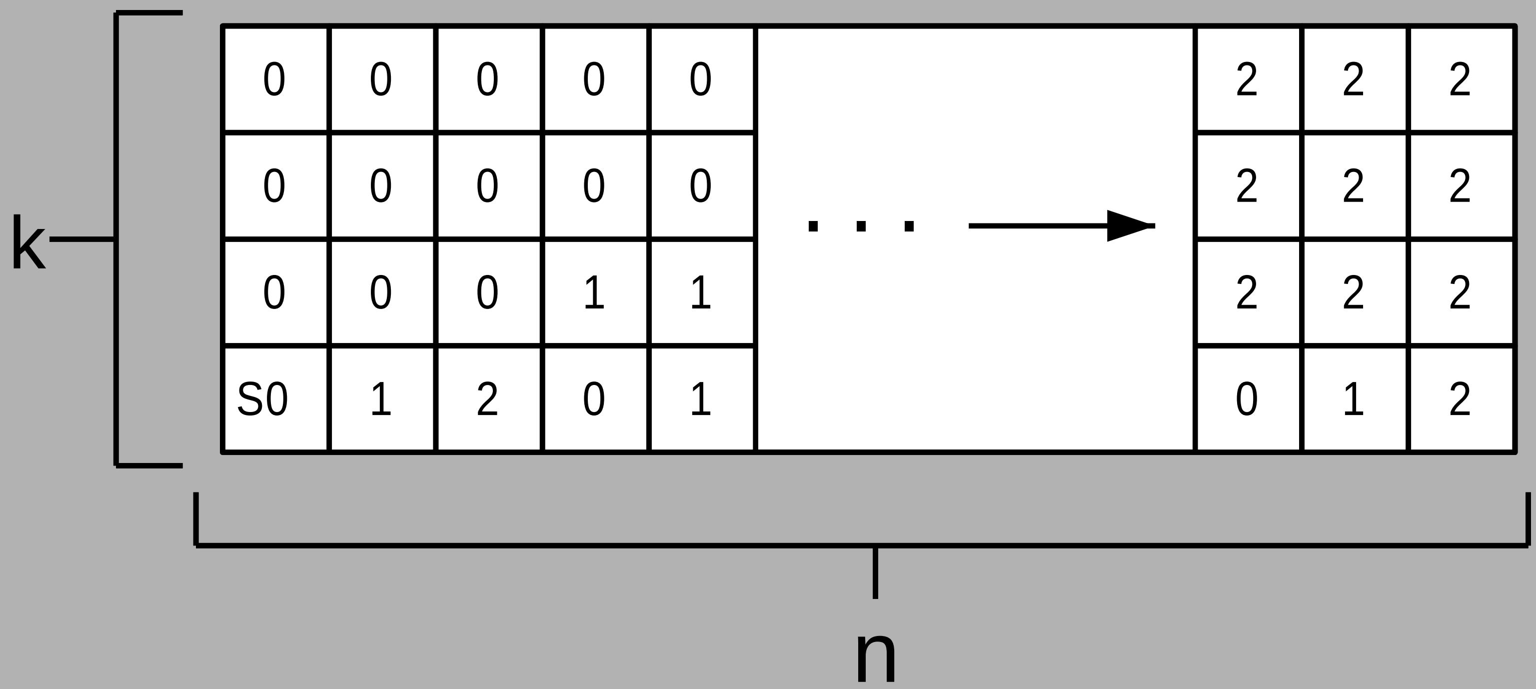
Building k x n Rectangles

k-digit, base $n^{(1/k)}$ counter:

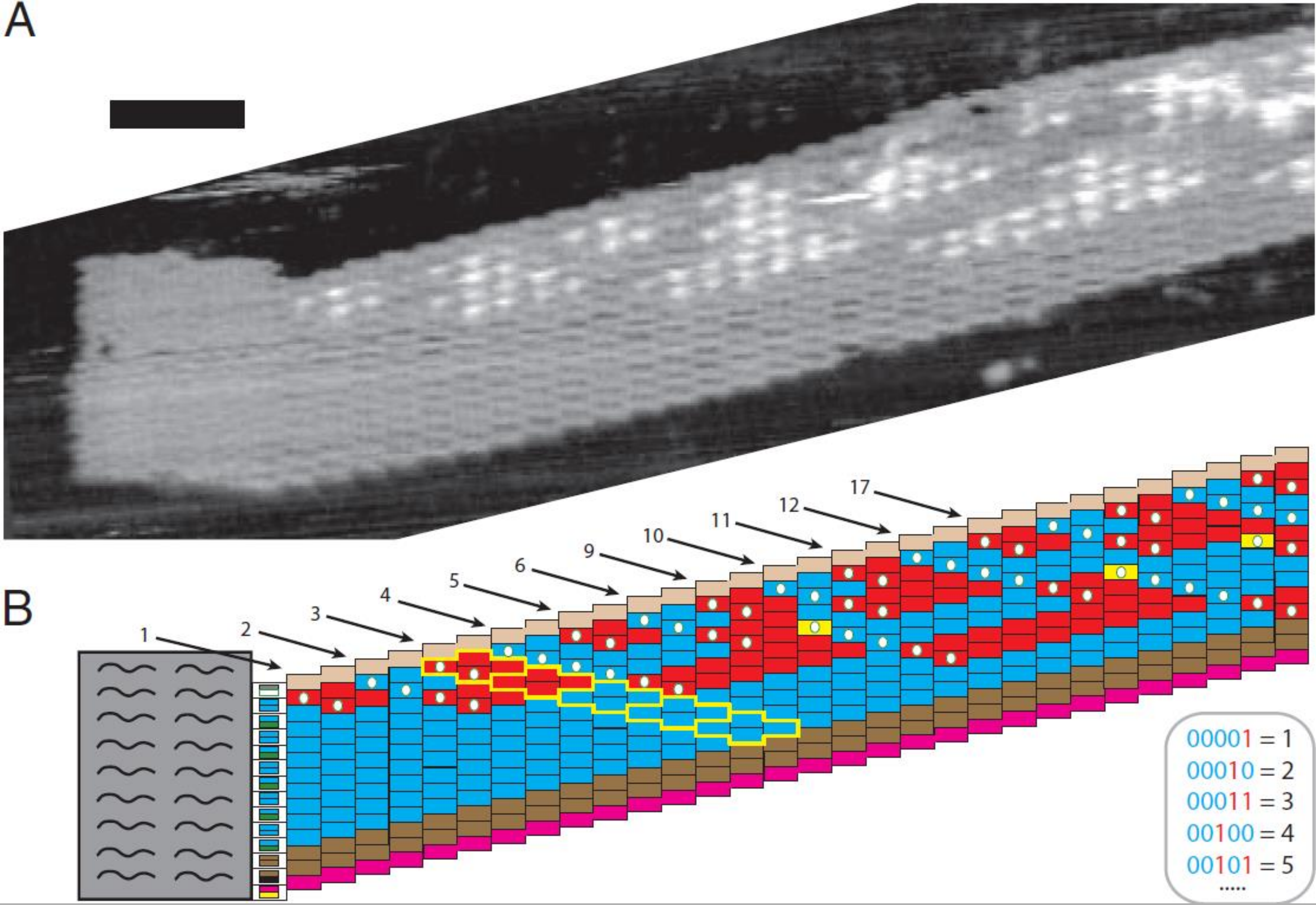


Building k x n Rectangles

k-digit, base $n^{(1/k)}$ counter:



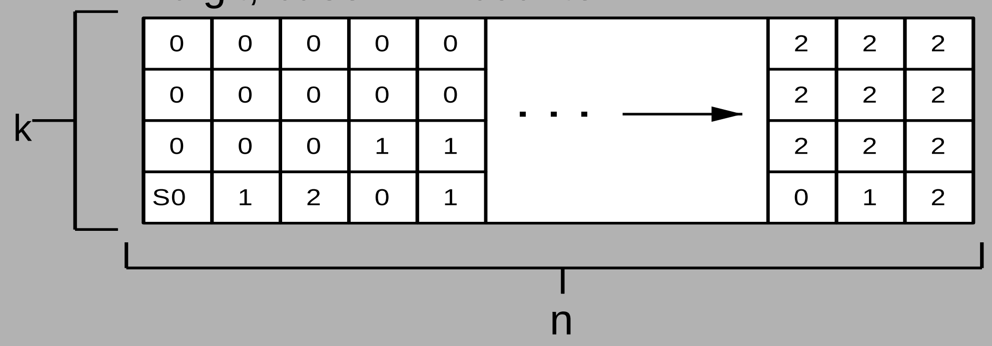
Tile Complexity: $O(k + n^{1/k})$



Barish, Shulman, Rothemund, Winfree, 2009

Building k x n Rectangles

k-digit, base $n^{(1/k)}$ counter:



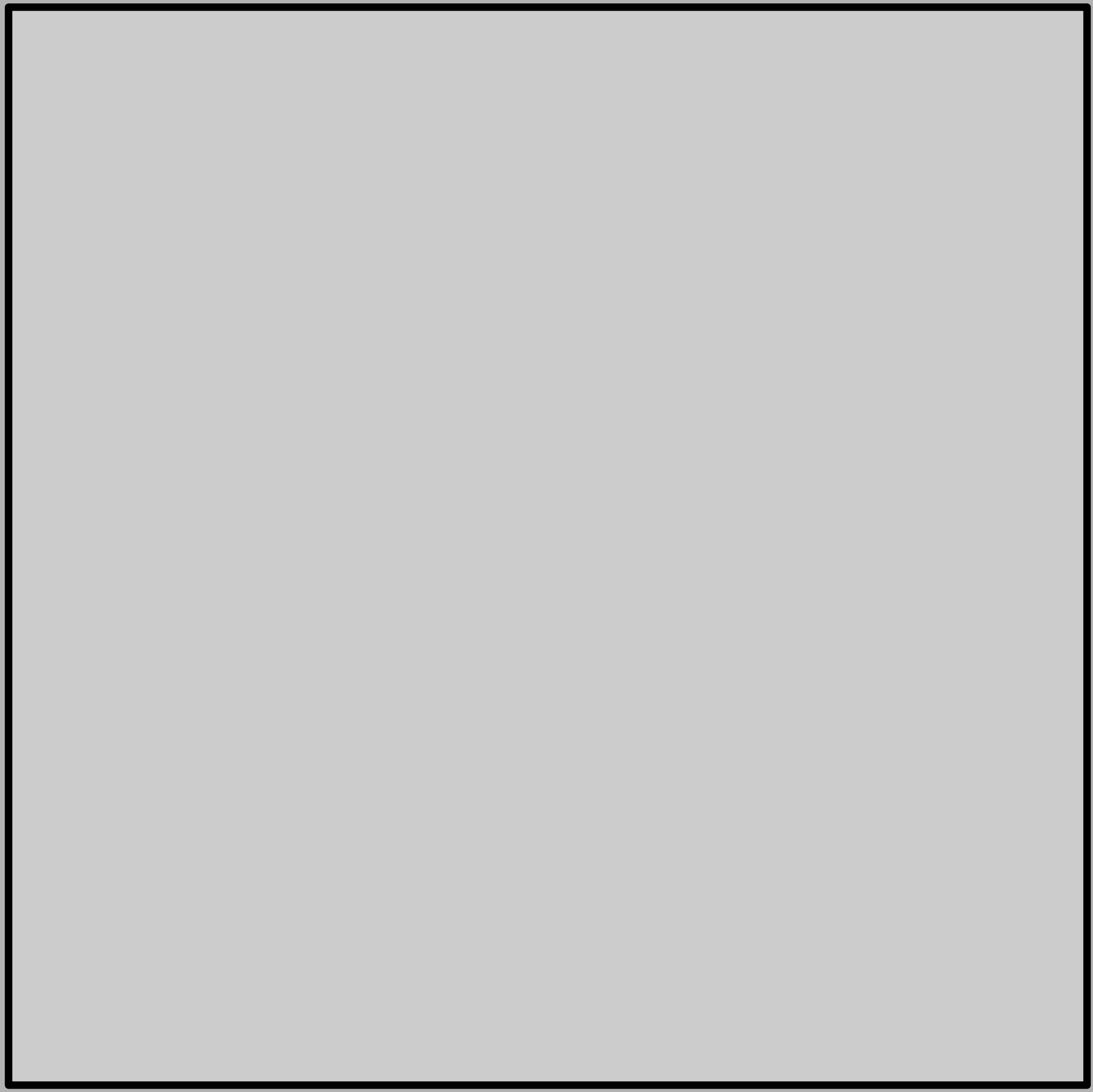
k x n rectanlges

Lower Bound	Upper Bound
?	$O(k+n^{1/k})$

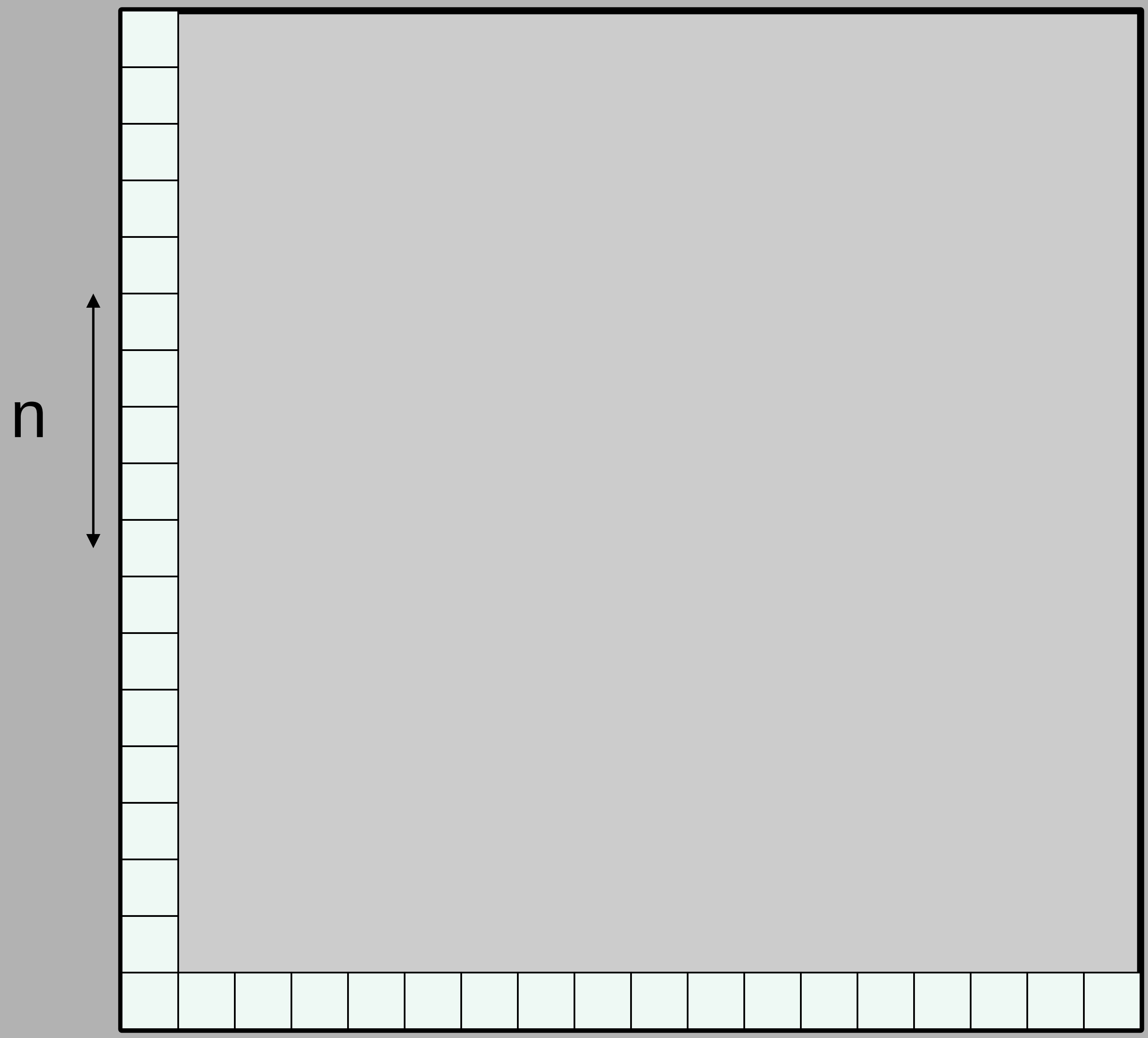
Outline

- Background, Motivation
- Model
- Rectangle and Squares

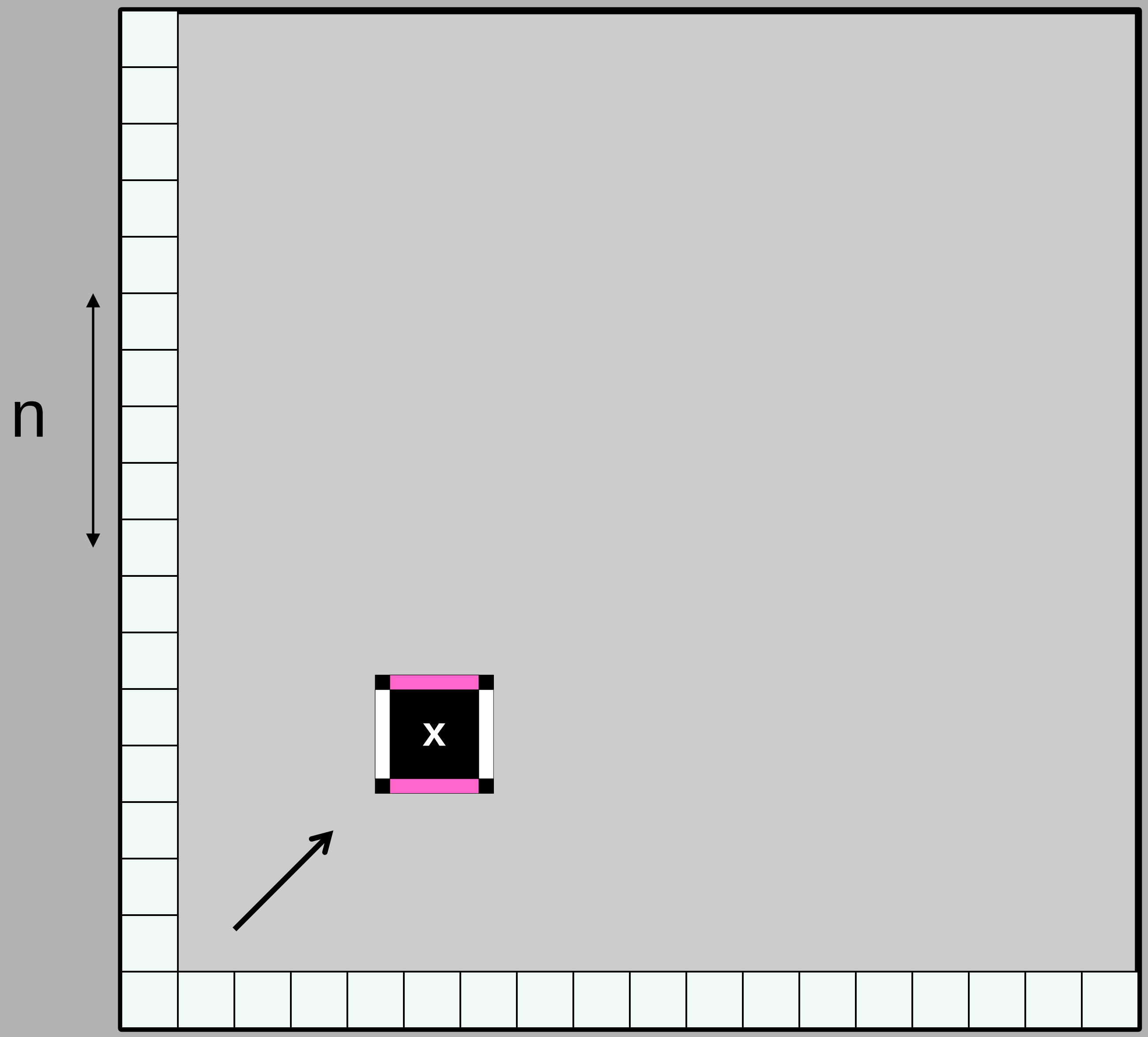
How efficiently can you build an n x n square?



How efficiently can you build an n x n square?

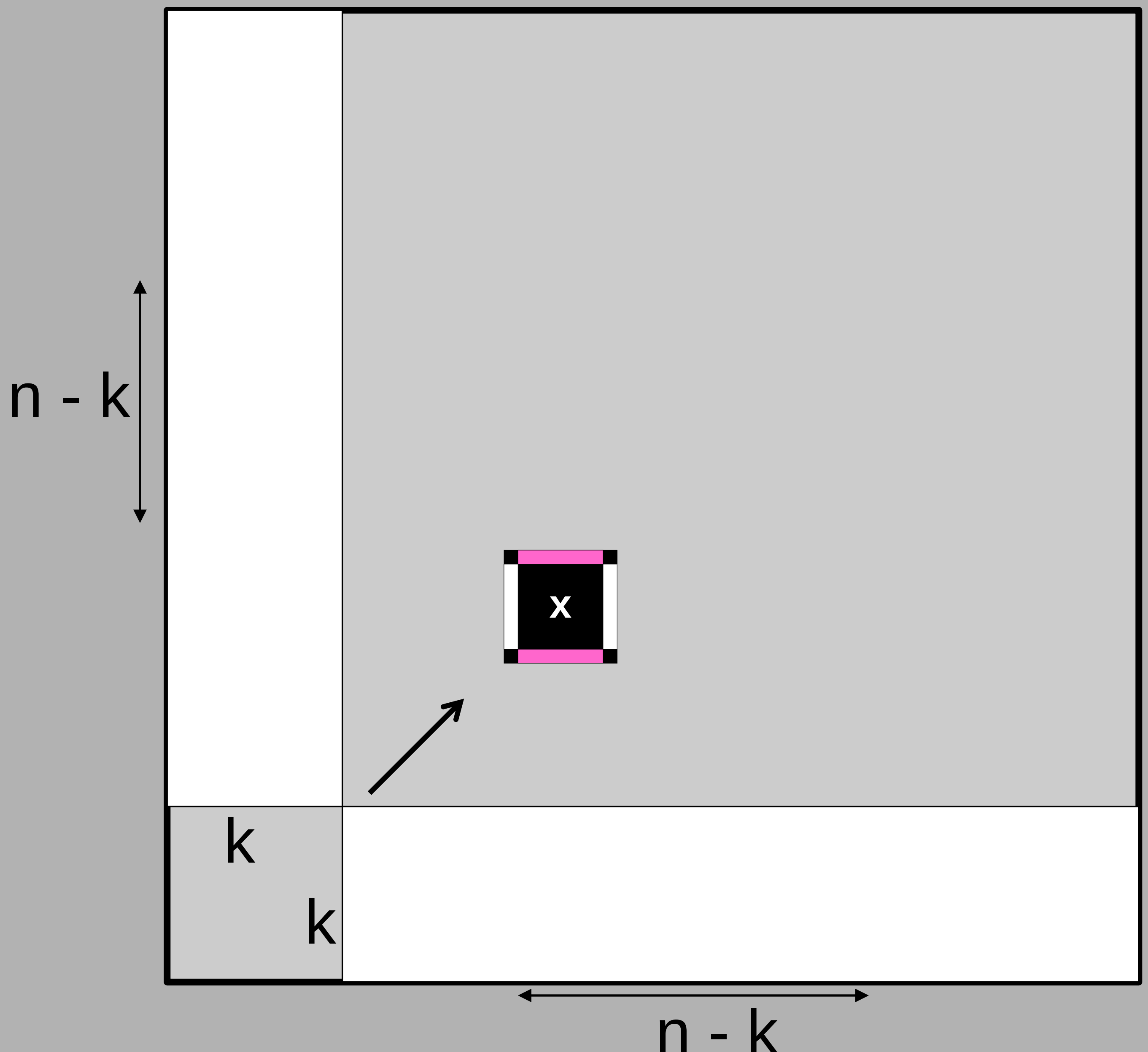


How efficiently can you build an n x n square?



Tile Complexity:
2n

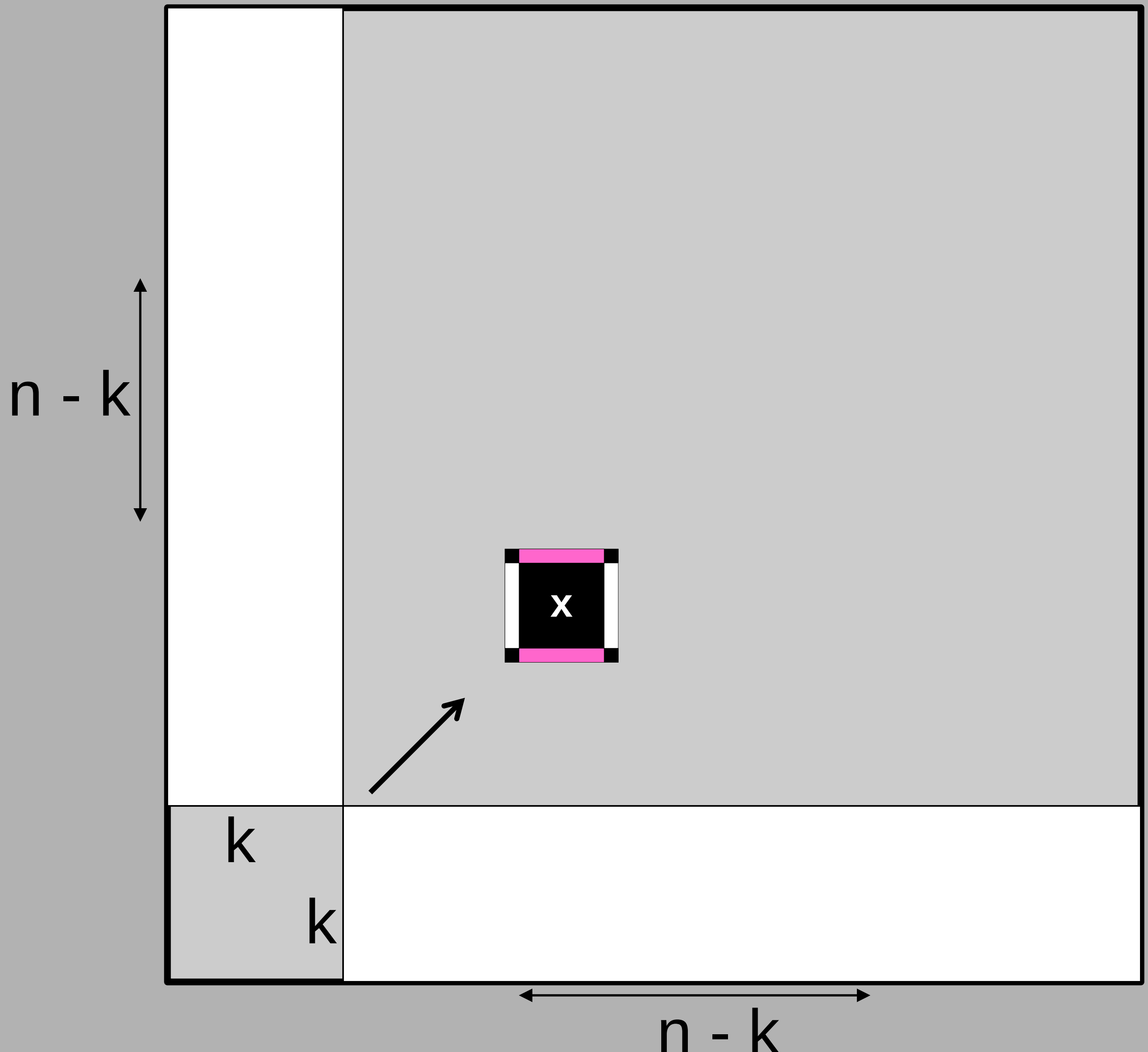
How efficiently can you build an n x n square?



Tile Complexity:

$$O(k + n^{1/k})$$

How efficiently can you build an n x n square?



Tile Complexity:

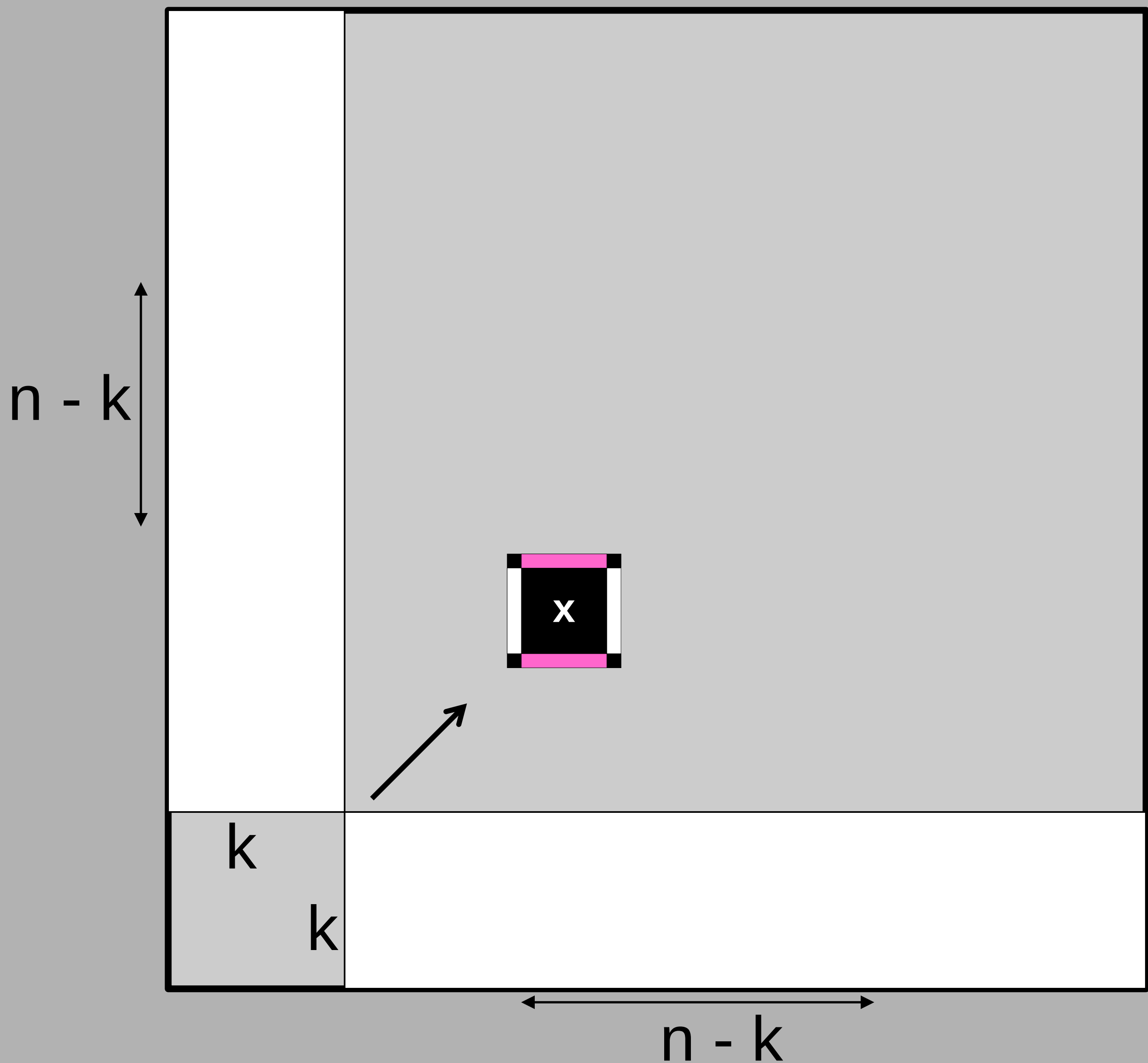
$$O(k + n^{1/k})$$

Tile Complexity:

$$O(\frac{\log n}{\log \log n})$$

(Adleman, Cheng, Goel, Huang STOC 2001)

How efficiently can you build an n x n square?



Tile Complexity:

$$O(k + n^{1/k})$$

Tile Complexity:

$$O\left(\frac{\log n}{\log \log n}\right)$$

(Adleman, Cheng, Goel, Huang STOC 2001)

Lower Bound:

$$\Omega\left(\frac{\log n}{\log \log n}\right)$$

(Rothemund, Winfree 2000)