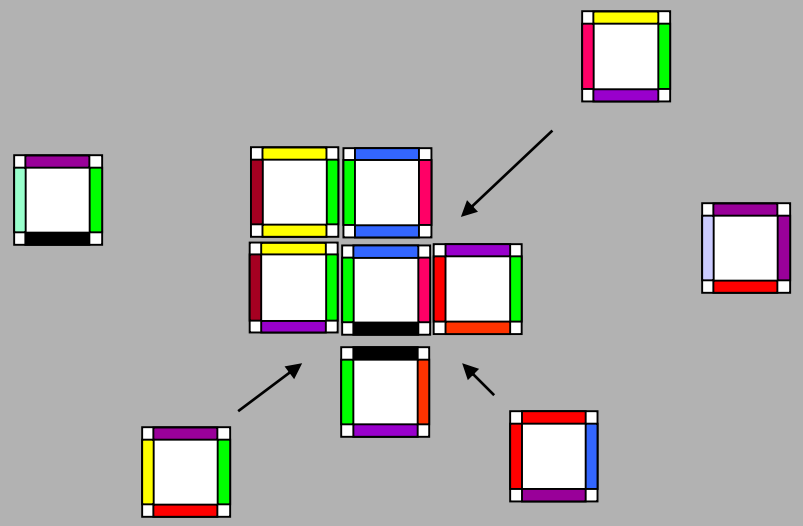
Concentration Programming:

Randomized Self-Assembly for Approximate Shapes



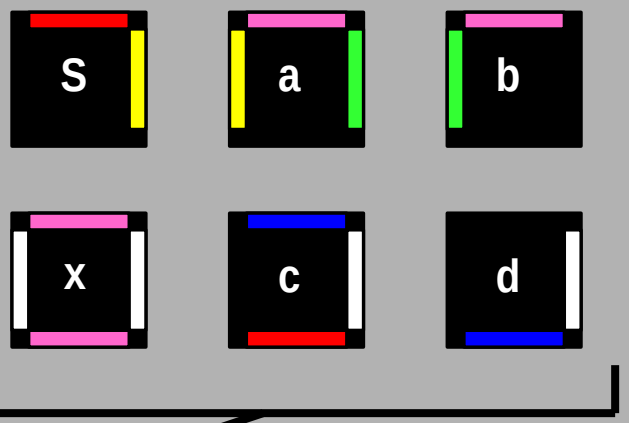
Outline

- Assembly Model
- Basic Constructions
- Probabilistic Assembly Model
- Main Result

Tile Assembly Model

(Rothemund, Winfree, Adleman)

T =



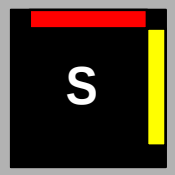
Tile Set:

Glue Function:

- G(y) = 2
- G(g) = 2
- G(r) = 2
- G(b) = 2
- G(p) = 1
- G(w) = 1

Temperature: t = 2

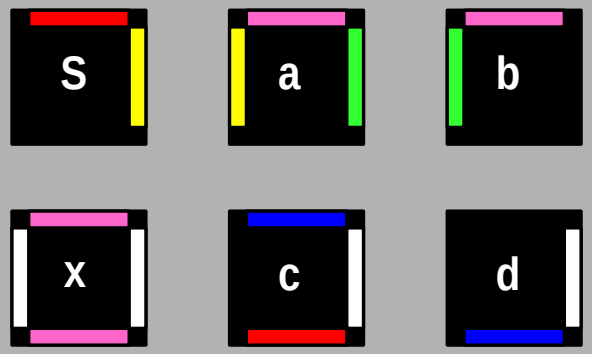
Seed Tile:



Tile Assembly Model

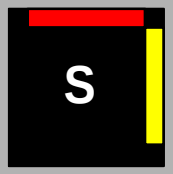
(Rothemund, Winfree, Adleman)

T =



G(y) = 2
G(g) = 2
G(r) = 2
G(b) = 2
G(p) = 1
G(w) = 1

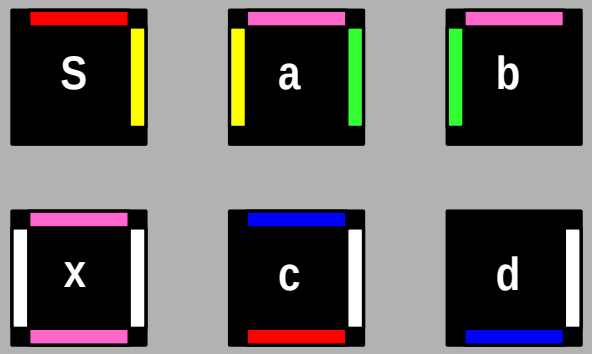
t = 2



Tile Assembly Model

(Rothemund, Winfree, Adleman)

T =



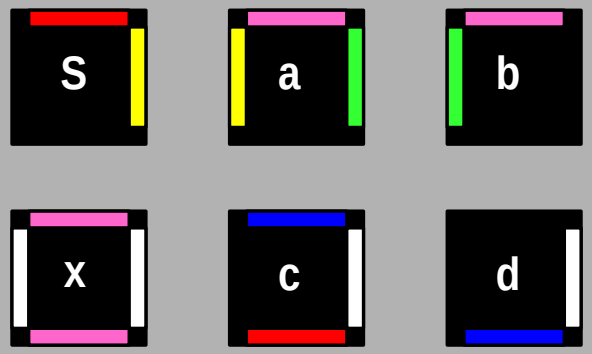
G(y) = 2
G(g) = 2
G(r) = 2
G(b) = 2
G(p) = 1
G(w) = 1

t = 2

Tile Assembly Model

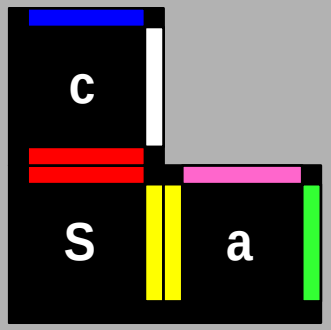
(Rothemund, Winfree, Adleman)

T =



G(y) = 2
G(g) = 2
G(r) = 2
G(b) = 2
G(p) = 1
G(w) = 1

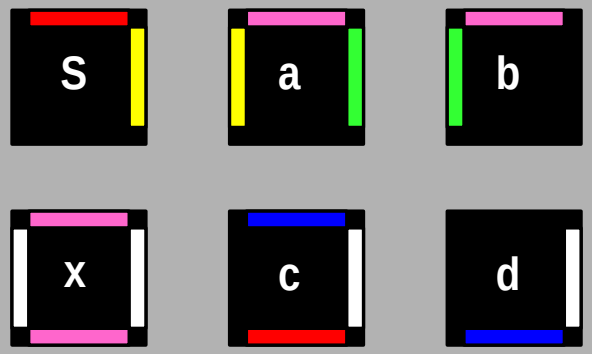
t = 2



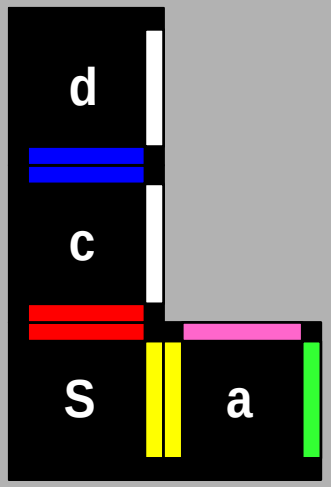
Tile Assembly Model

(Rothemund, Winfree, Adleman)

T =



G(y) = 2
G(g) = 2
G(r) = 2
G(b) = 2
G(p) = 1
G(w) = 1

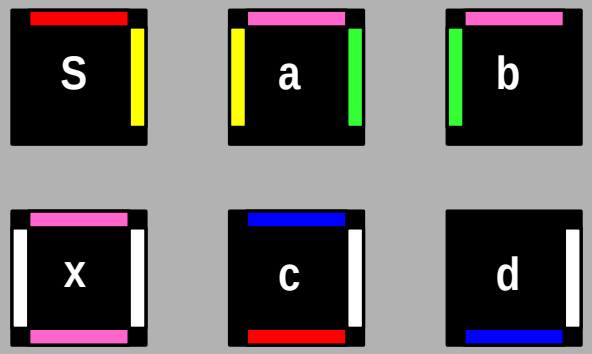


t = 2

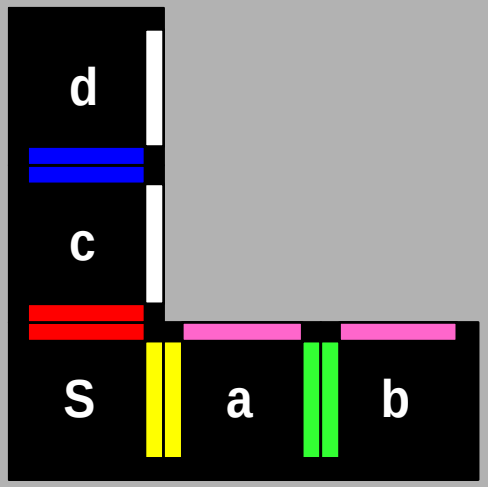
Tile Assembly Model

(Rothemund, Winfree, Adleman)

T =



G(y) = 2
G(g) = 2
G(r) = 2
G(b) = 2
G(p) = 1
G(w) = 1

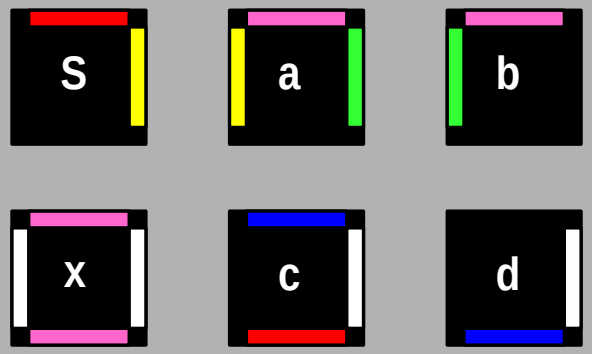


t = 2

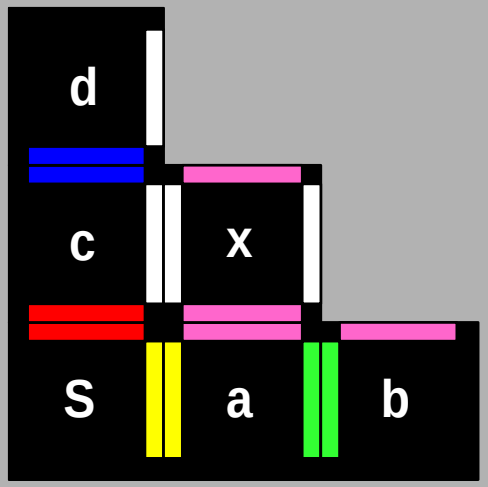
Tile Assembly Model

(Rothemund, Winfree, Adleman)

T =



G(y) = 2
G(g) = 2
G(r) = 2
G(b) = 2
G(p) = 1
G(w) = 1

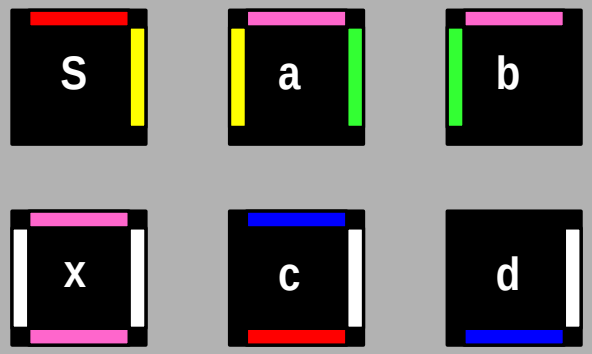


t = 2

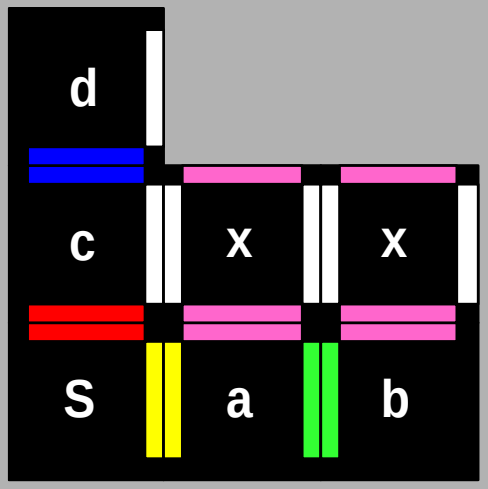
Tile Assembly Model

(Rothemund, Winfree, Adleman)

T =



G(y) = 2
G(g) = 2
G(r) = 2
G(b) = 2
G(p) = 1
G(w) = 1

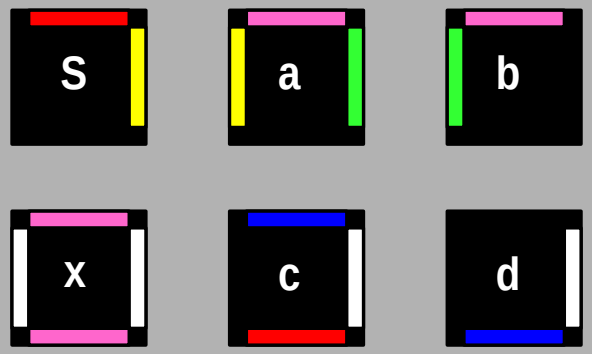


t = 2

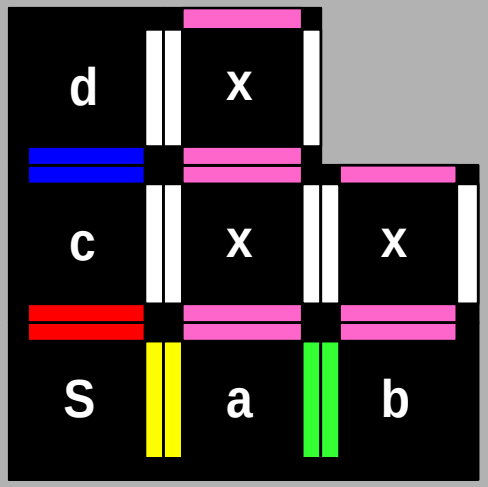
Tile Assembly Model

(Rothemund, Winfree, Adleman)

T =



G(y) = 2
G(g) = 2
G(r) = 2
G(b) = 2
G(p) = 1
G(w) = 1

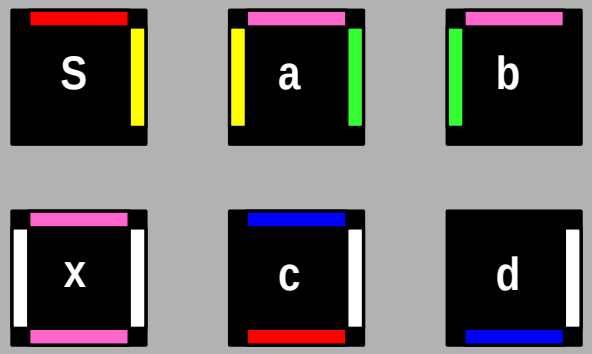


t = 2

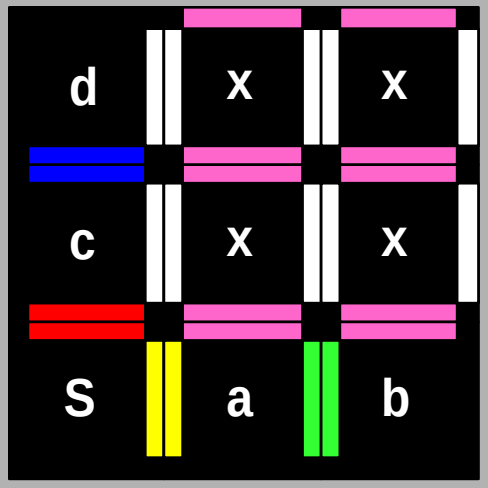
Tile Assembly Model

(Rothemund, Winfree, Adleman)

T =



G(y) = 2
G(g) = 2
G(r) = 2
G(b) = 2
G(p) = 1
G(w) = 1

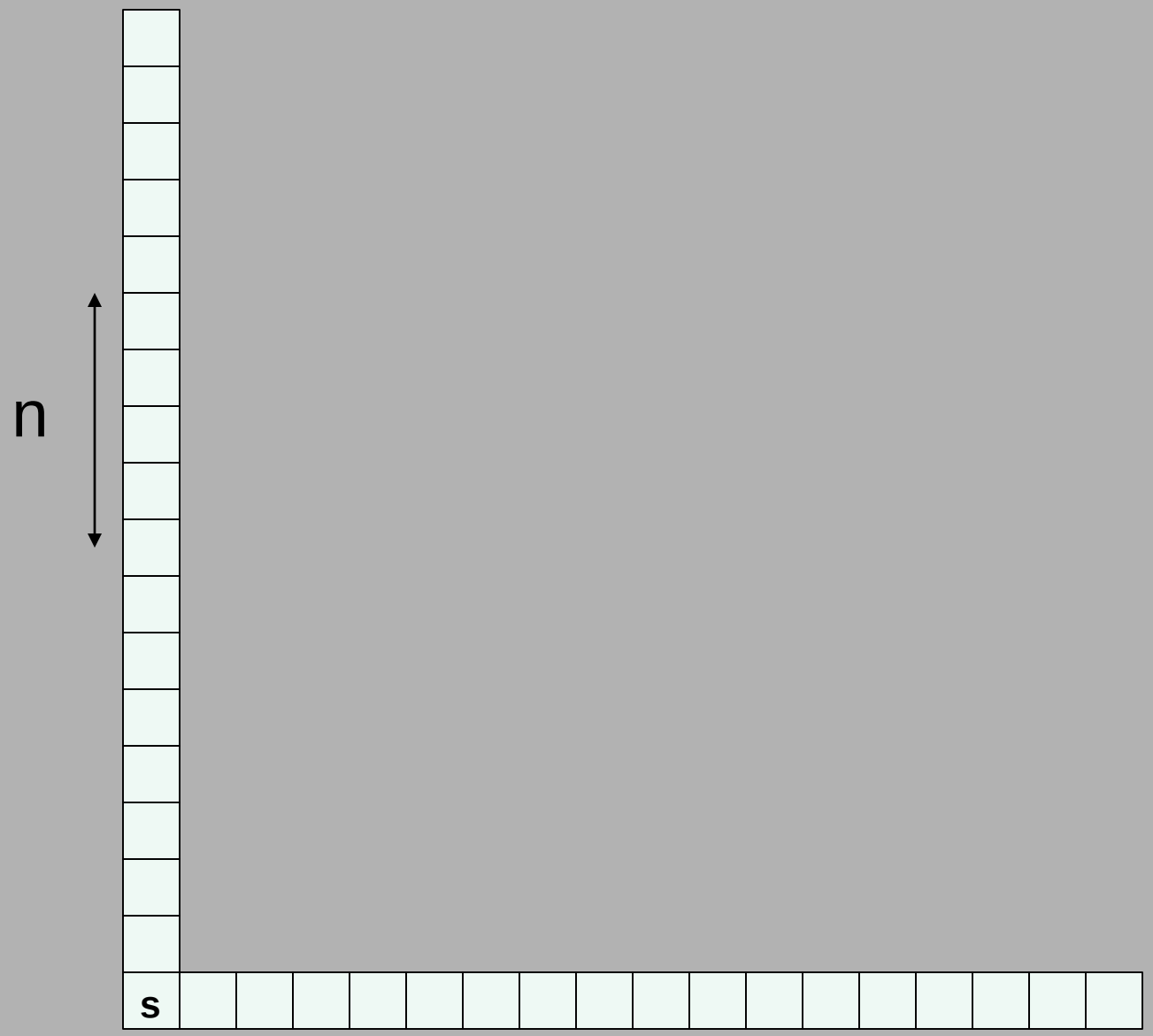


t = 2

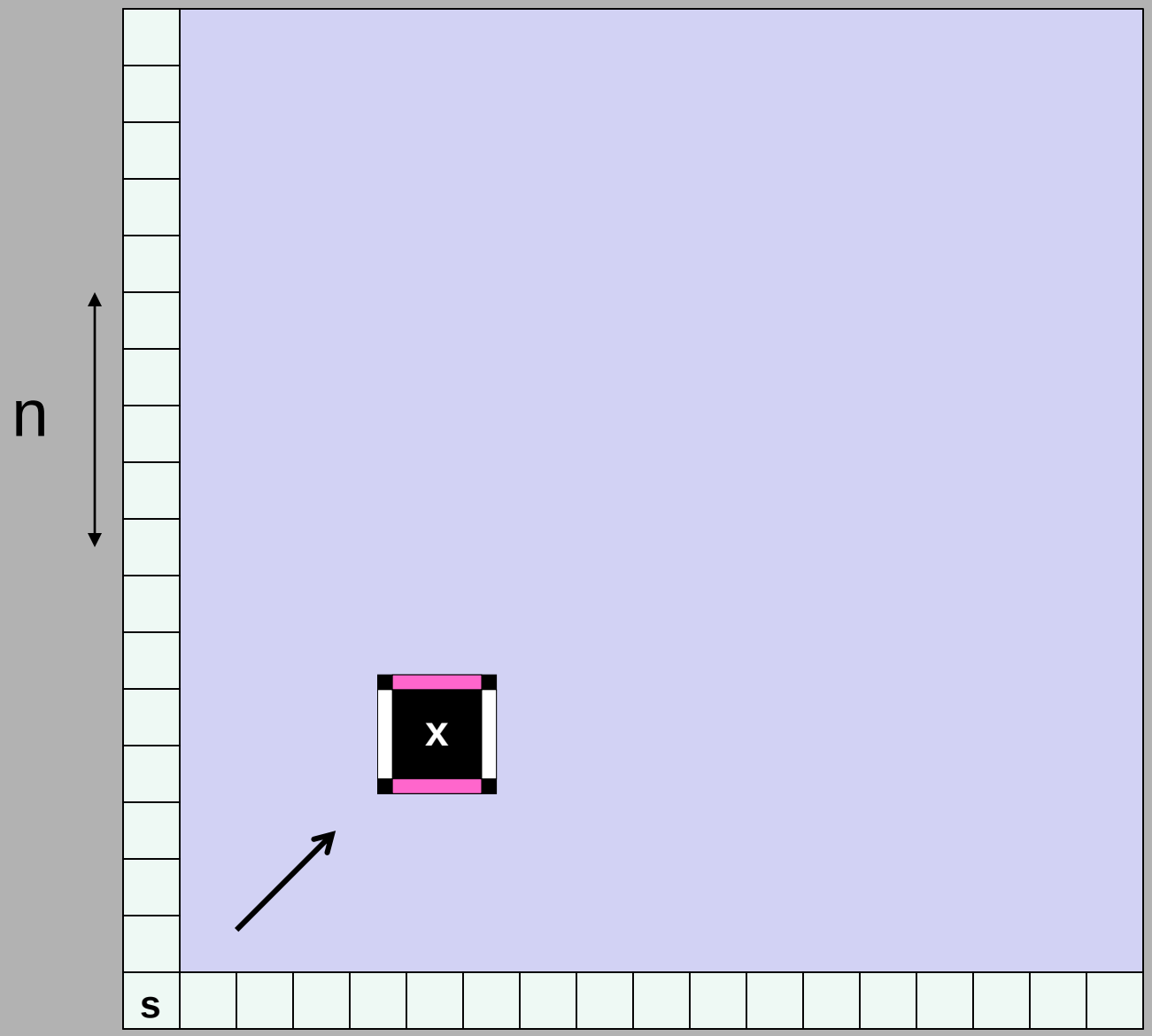
How efficiently can you build an n x n square?

s

How efficiently can you build an n x n square?



How efficiently can you build an n x n square?



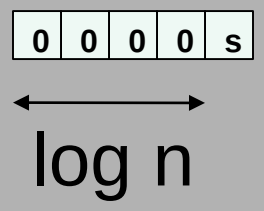
Tile Complexity:
2n

How efficiently can you build an n x n square?

s

How efficiently can you build an n x n square?

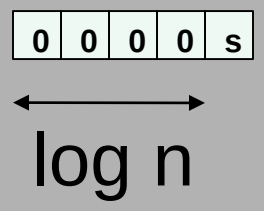
-Use log n tile types to seed counter:



How efficiently can you build an n x n square?

-Use log n tile types capable of Binary counting:

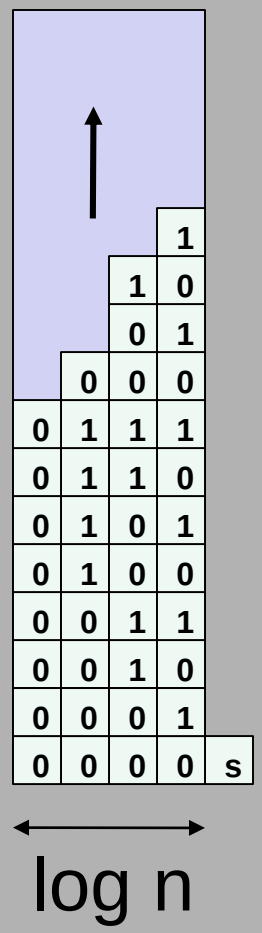
-Use 8 additional tile types capable of binary counting:



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	s

log n

How efficiently can you build an n x n square?

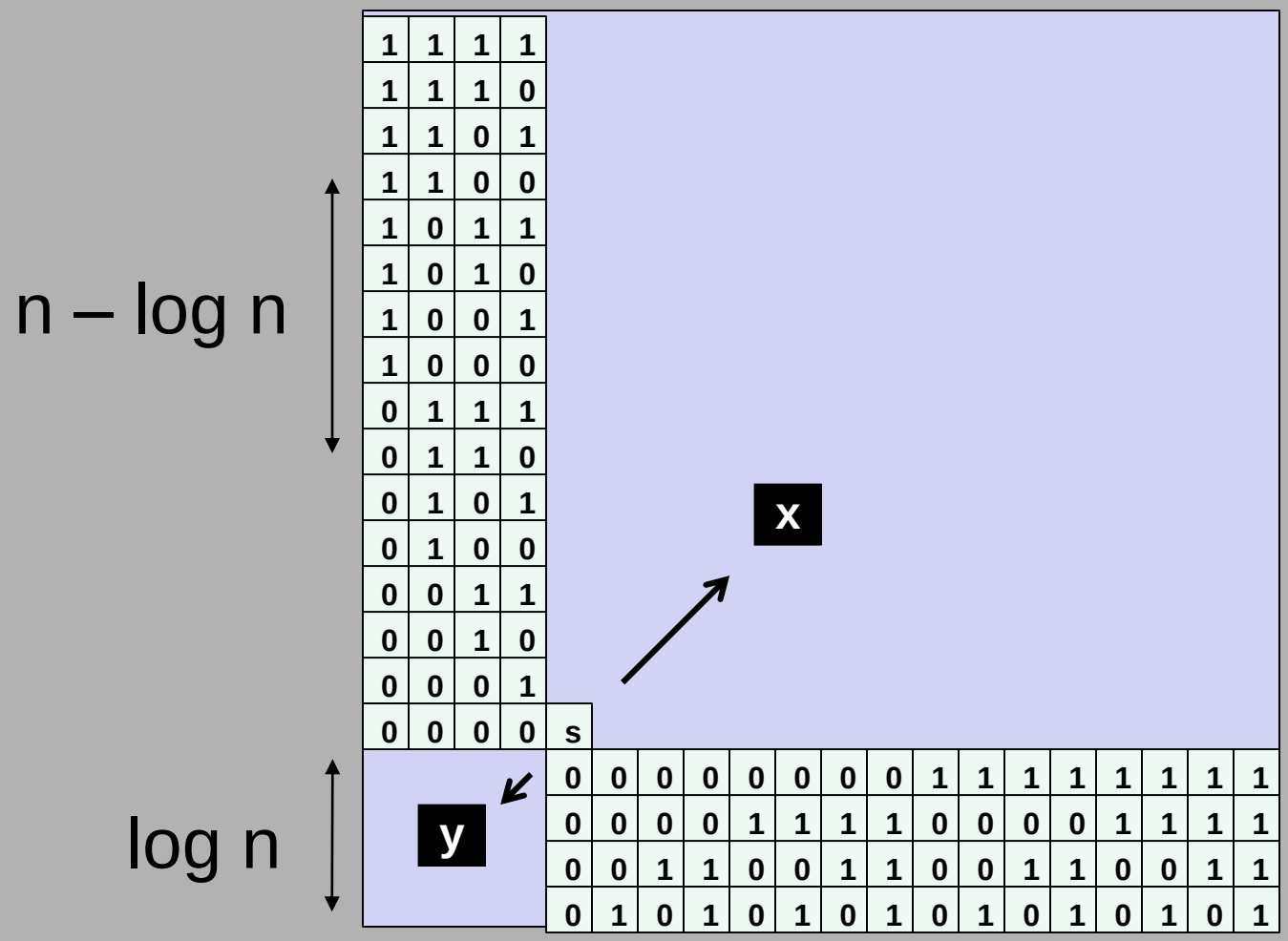
-Use log n tile types capable of Binary counting:

-Use 8 additional tile types capable of binary counting:

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

s															
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

How efficiently can you build an n x n square?



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

With optimal counter:
Tile Complexity:
O(log n / loglog n)
(Adleman, Cheng, Goel, Huang 2001)

Meets lower bound:
Ω(log n / loglog n)
(Rothemund, Winfree 2000)

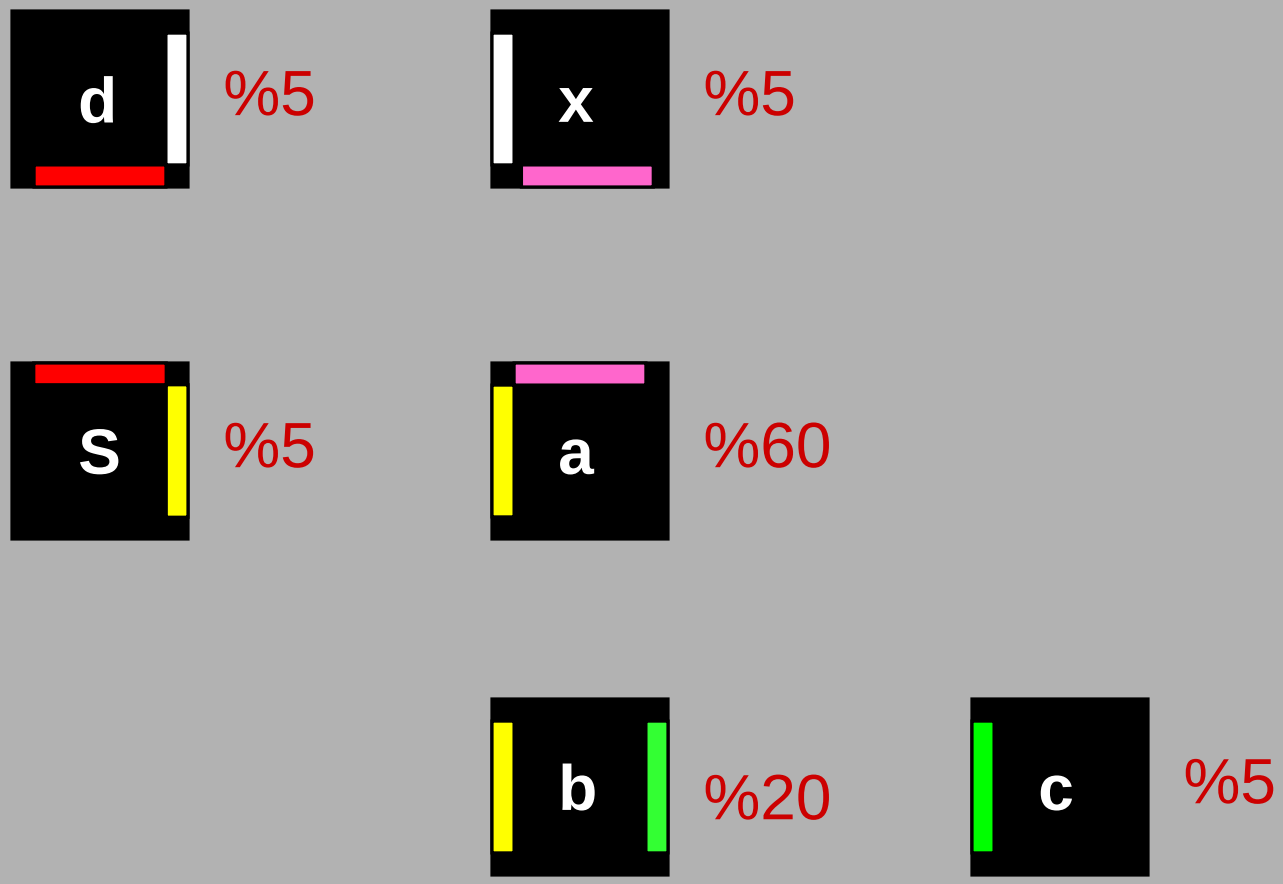
Outline

- Assembly Model
- Basic Constructions
- Probabilistic Assembly Model
- Main Result

Probabilistic Assembly Model

(Becker, Remila, Rapaport)

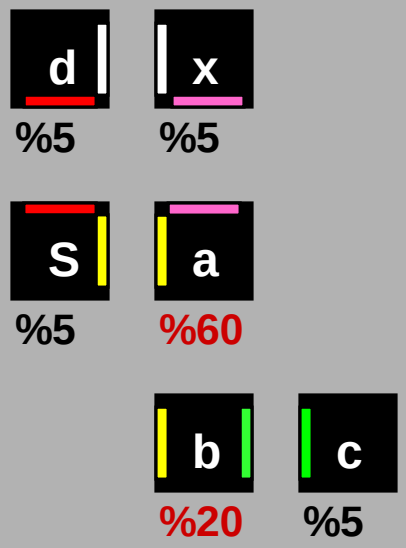
Assign Relative Concentrations:



Probabilistic Assembly Model

(Becker, Remila, Rapaport)

Tileset =



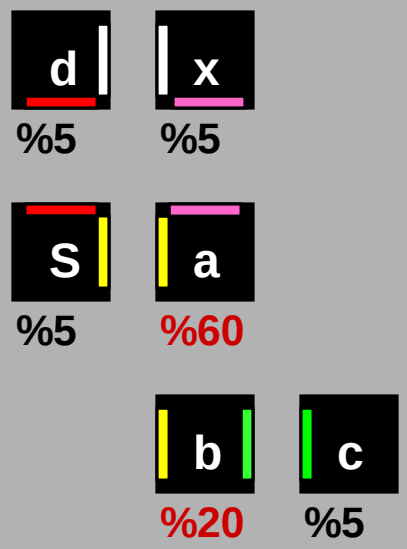
G(y) = 2
G(g) = 2
G(r) = 2
G(p) = 1
G(w) = 1

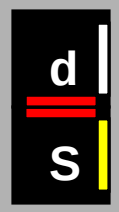
t = 2

Probabilistic Assembly Model

(Becker, Remila, Rapaport)

Tileset =





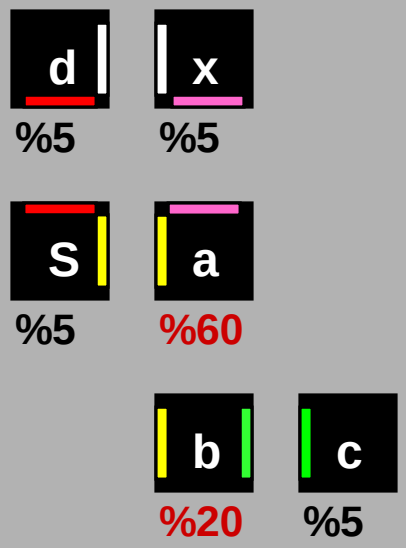
G(y) = 2
G(g) = 2
G(r) = 2
G(p) = 1
G(w) = 1

t = 2

Probabilistic Assembly Model

(Becker, Remila, Rapaport)

Tileset =



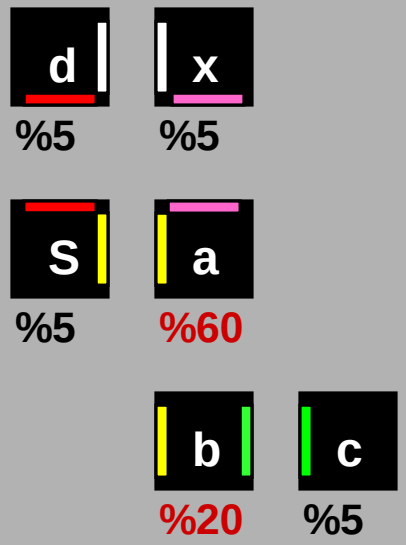
G(y) = 2
G(g) = 2
G(r) = 2
G(p) = 1
G(w) = 1

t = 2

Probabilistic Assembly Model

(Becker, Remila, Rapaport)

Tileset =



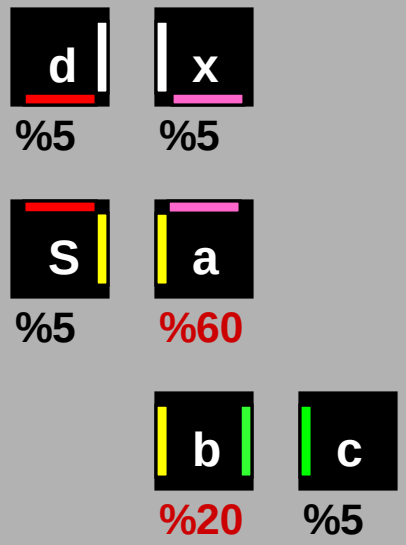
G(y) = 2
G(g) = 2
G(r) = 2
G(p) = 1
G(w) = 1

t = 2

Probabilistic Assembly Model

(Becker, Remila, Rapaport)

Tileset =



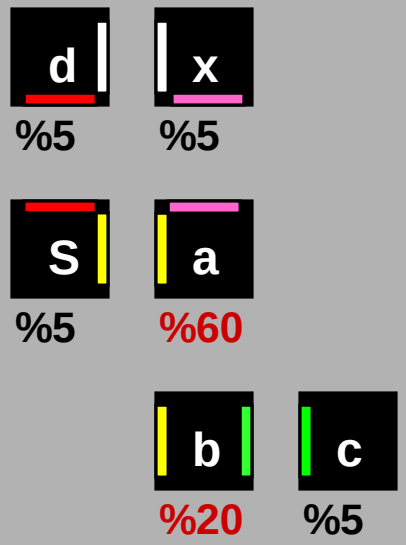
G(y) = 2
G(g) = 2
G(r) = 2
G(p) = 1
G(w) = 1

t = 2

Probabilistic Assembly Model

(Becker, Remila, Rapaport)

Tileset =



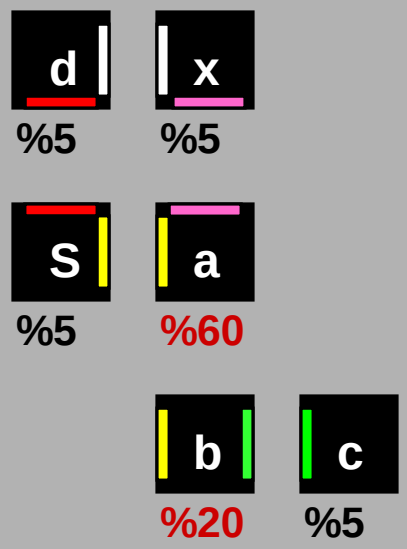
G(y) = 2
G(g) = 2
G(r) = 2
G(p) = 1
G(w) = 1

t = 2

Probabilistic Assembly Model

(Becker, Remila, Rapaport)

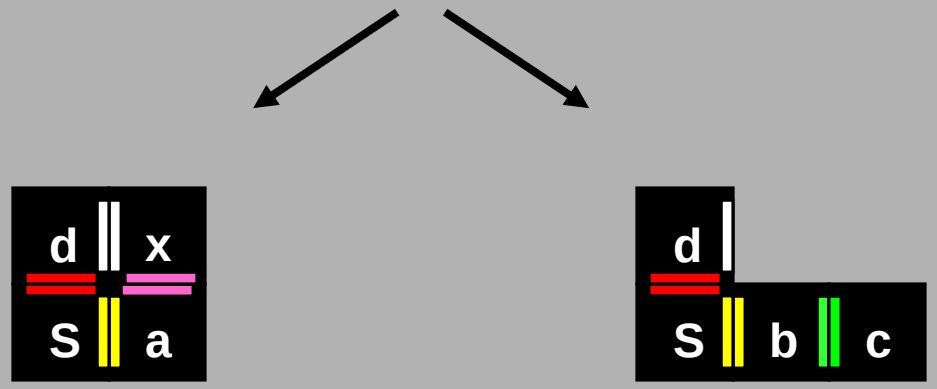
Tileset =



G(y) = 2
G(g) = 2
G(r) = 2
G(p) = 1
G(w) = 1

t = 2

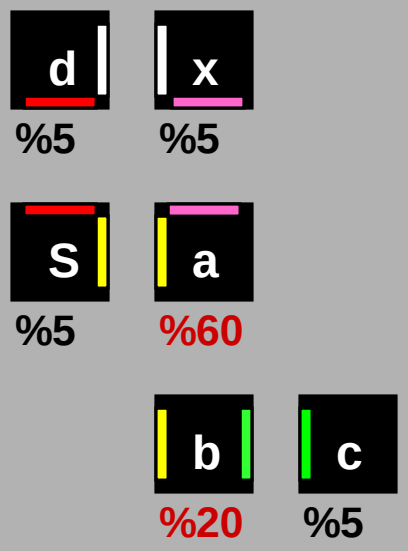
Two Terminal Shapes Produced

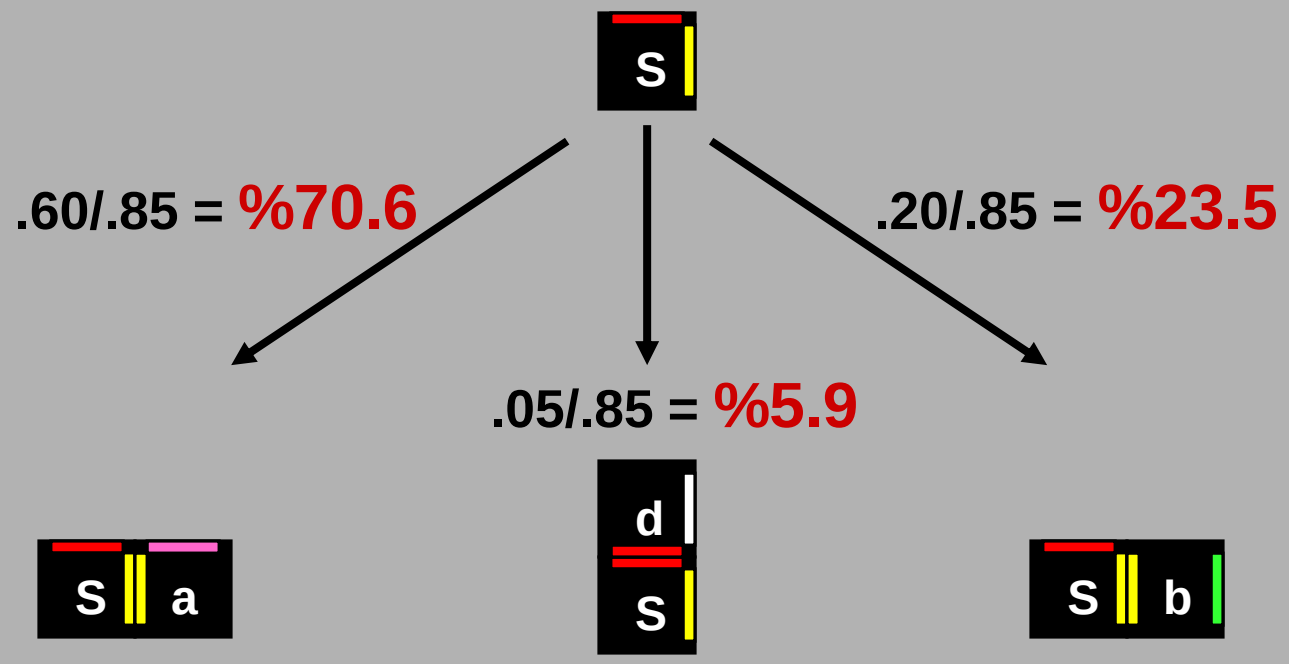


Probabilistic Assembly Model

(Becker, Remila, Rapaport)

Tileset =





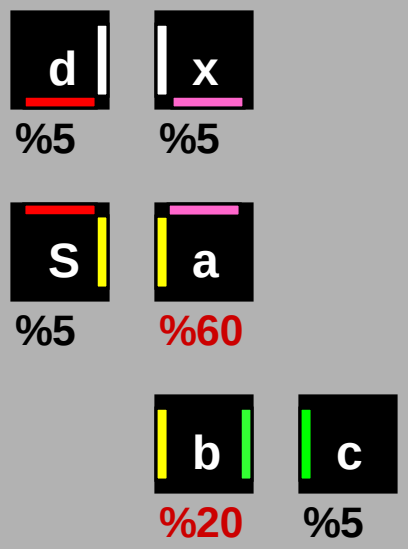
G(y) = 2
G(g) = 2
G(r) = 2
G(p) = 1
G(w) = 1

t = 2

Probabilistic Assembly Model

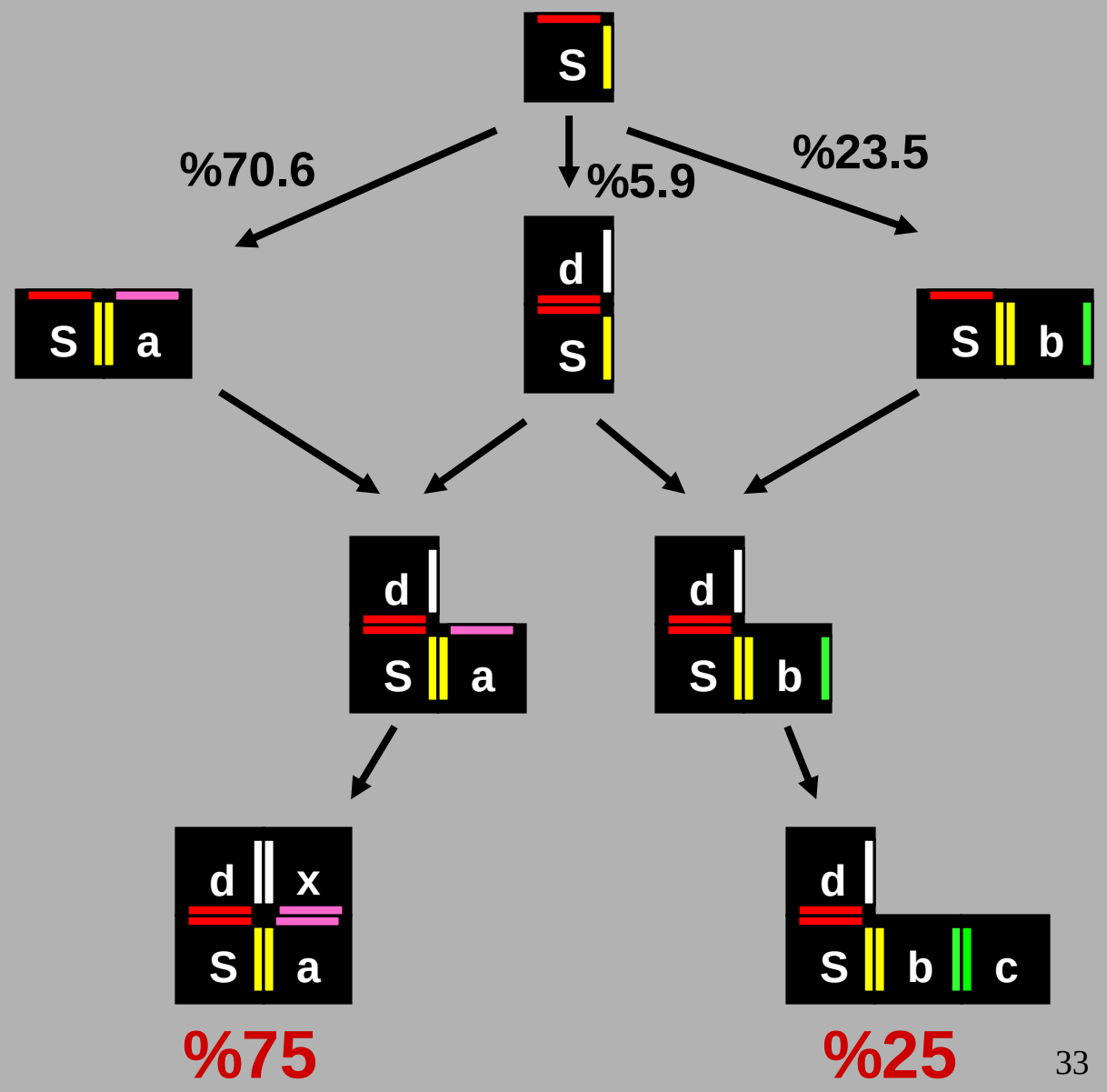
(Becker, Remila, Rapaport)

Tileset =

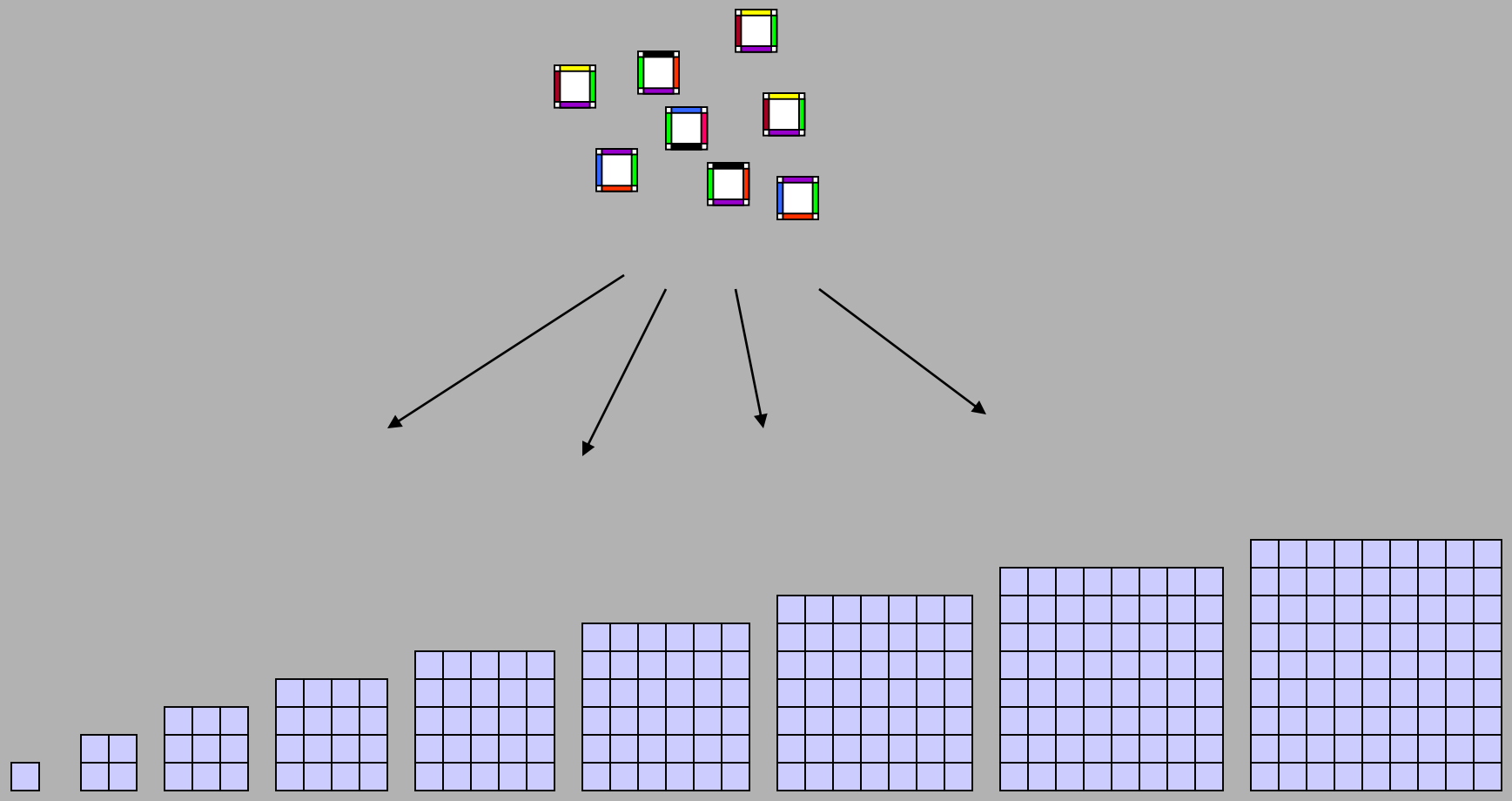


G(y) = 2
G(g) = 2
G(r) = 2
G(p) = 1
G(w) = 1

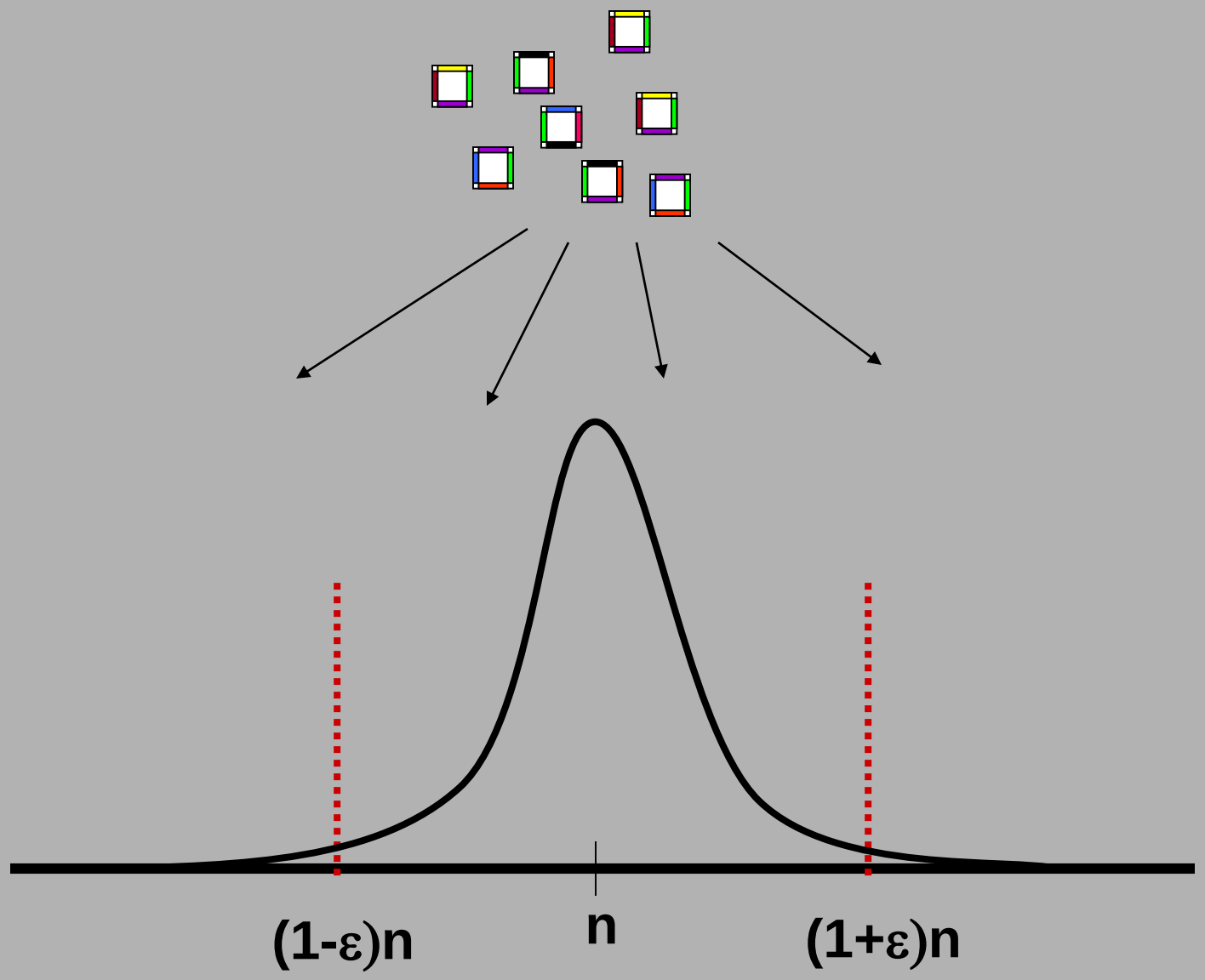
t = 2



Generic Tileset for all Squares



Generic Tileset for Approximate Squares



Generic Tileset for Approximate Squares

(ε , δ) – Approximate Square Assembly

Given:

- ε , $\delta < 0$
- integer n

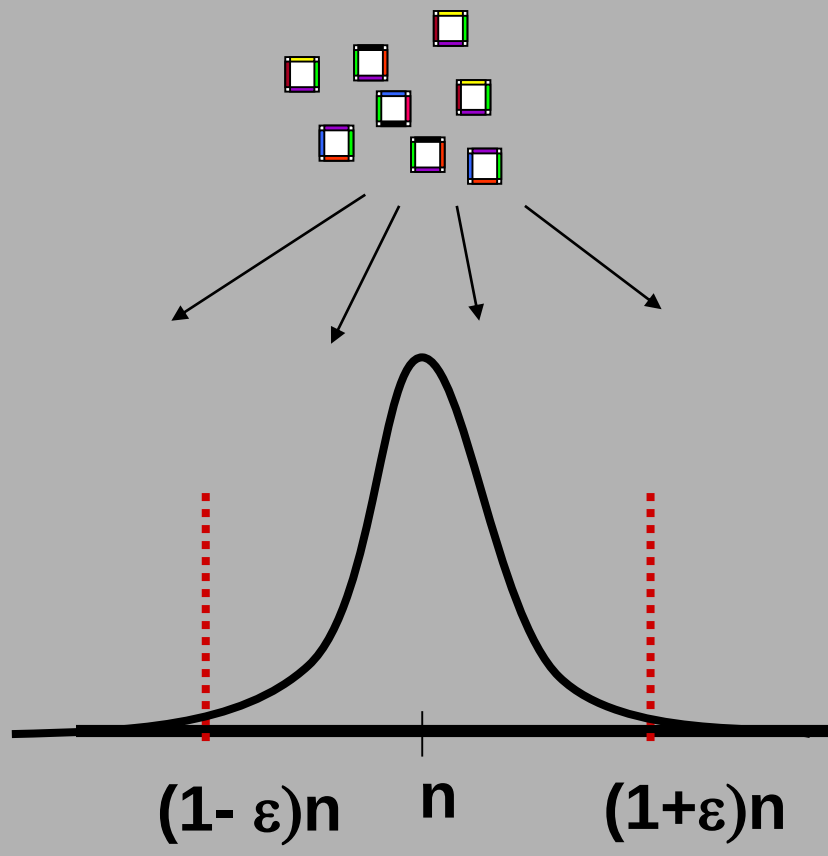
Design:

A probabilistic tile system that will assemble an n' x n' square with:

$$(1-\varepsilon)n < n' < (1+\varepsilon)n$$

With probability at least:

$$1 - \delta$$

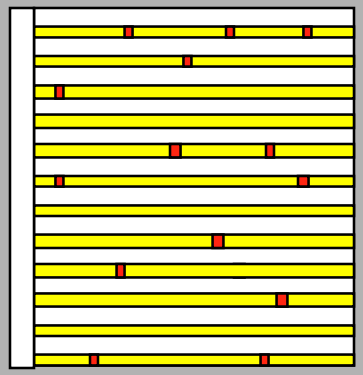


High Level Idea

-Build random structure:

-Dimensions are random

-Internal pattern is random



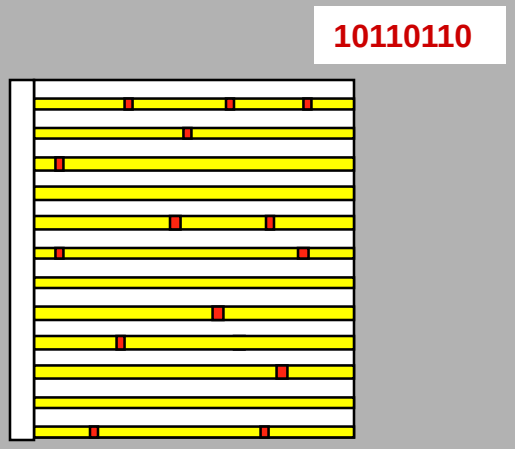
High Level Idea

-Build random structure:

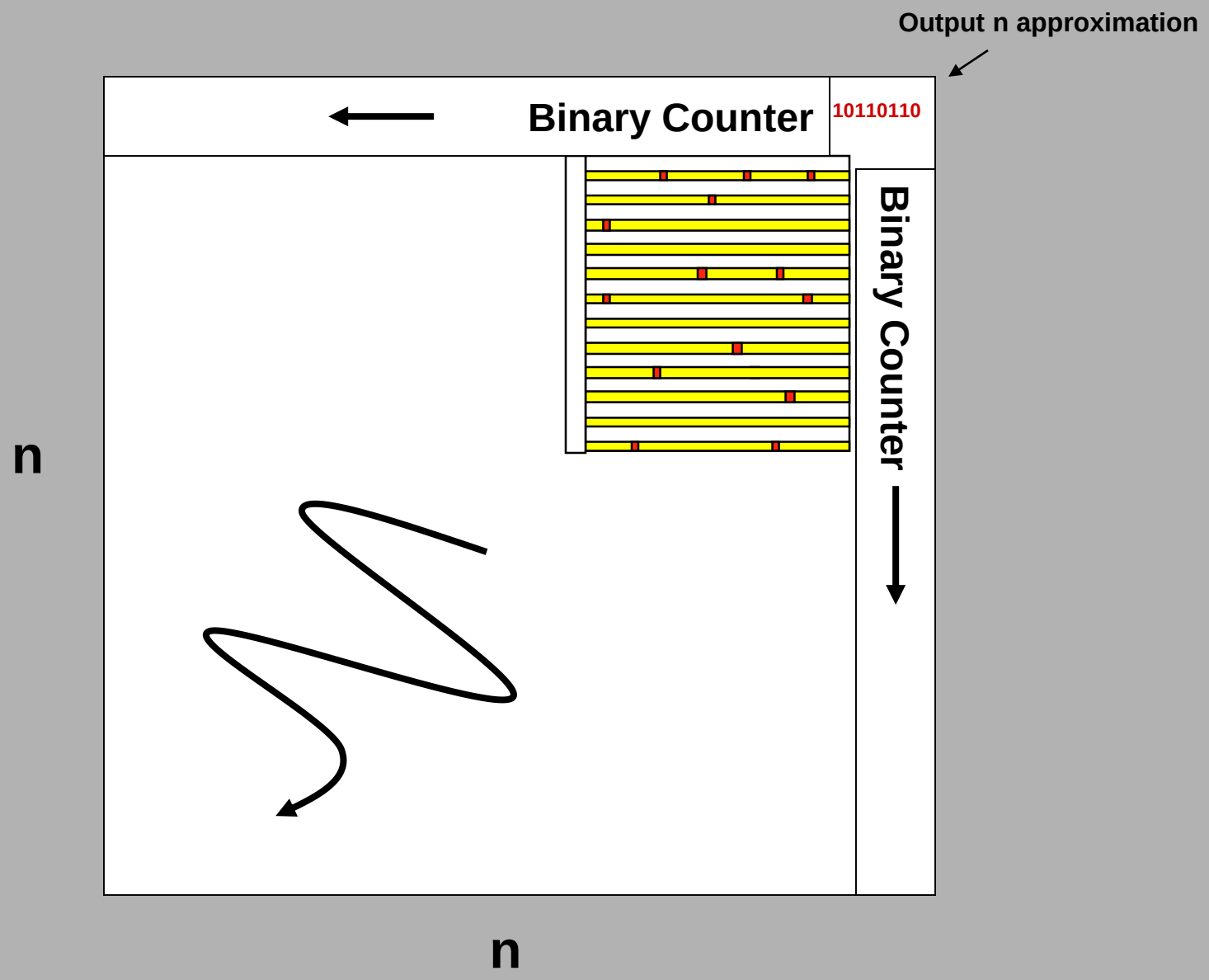
-Dimensions are random

-Internal pattern is random

-Incorporate arithmetic tiles to extract a binary number from the random pattern

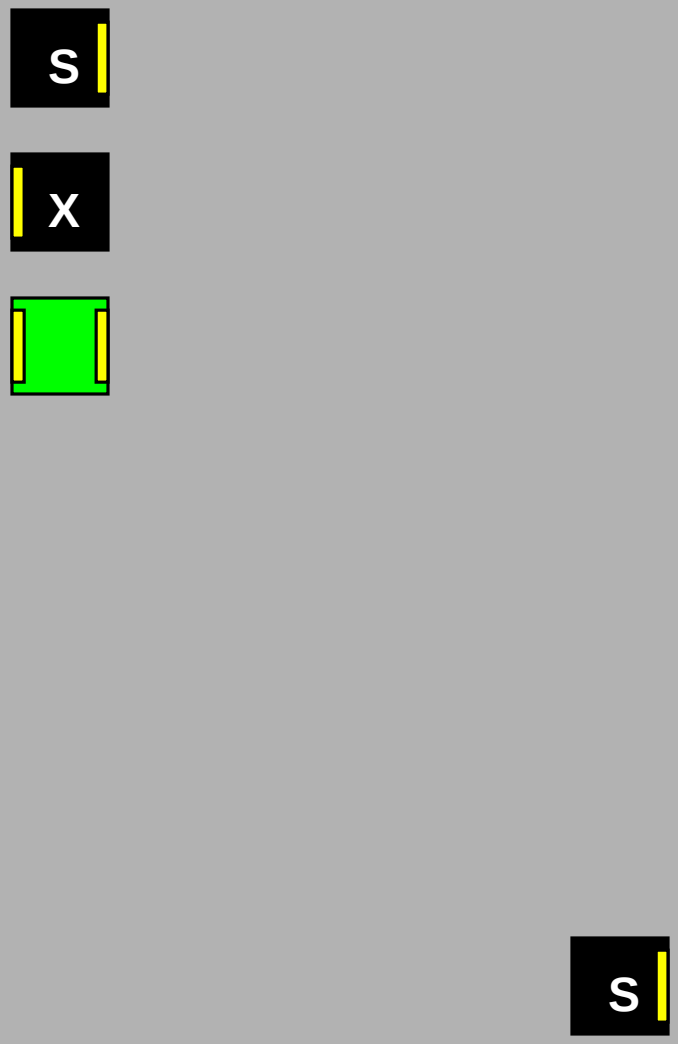


Finish off Square



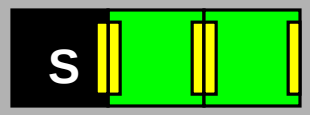
Line Estimation of n

Line Estimation of n

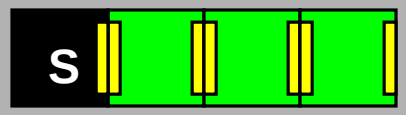


Line Estimation of n

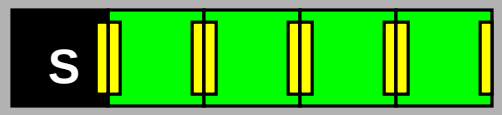
Line Estimation of n



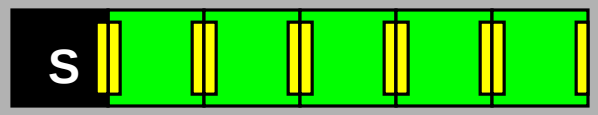
Line Estimation of n



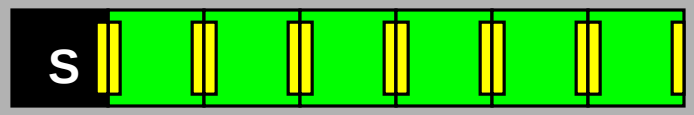
Line Estimation of n



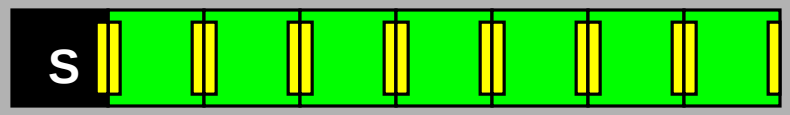
Line Estimation of n



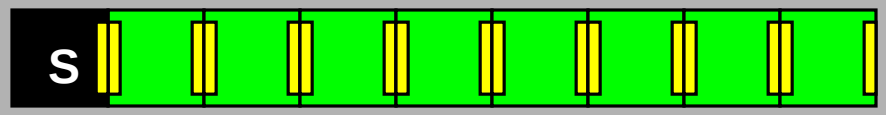
Line Estimation of n



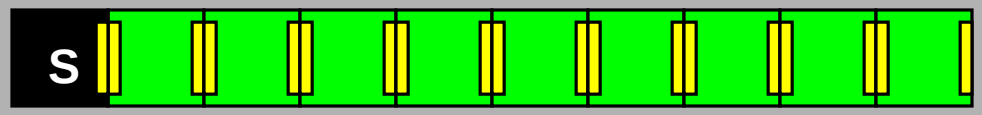
Line Estimation of n



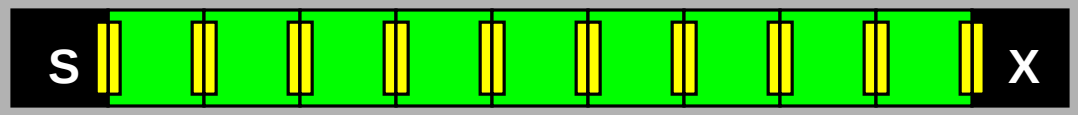
Line Estimation of n



Line Estimation of n

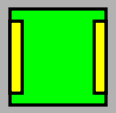


Line Estimation of n

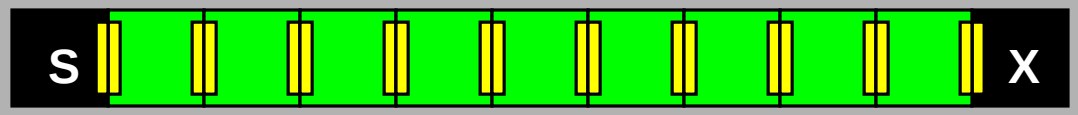


Line Estimation of n

% 1/n

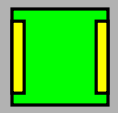


% (n-1)/n



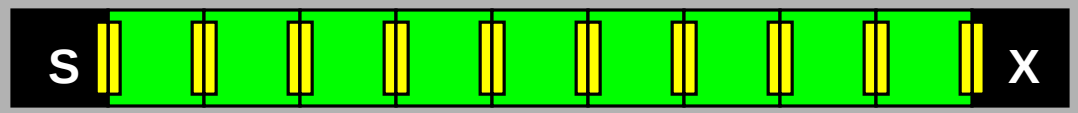
Line Estimation of n

% 1/n



% (n-1)/n

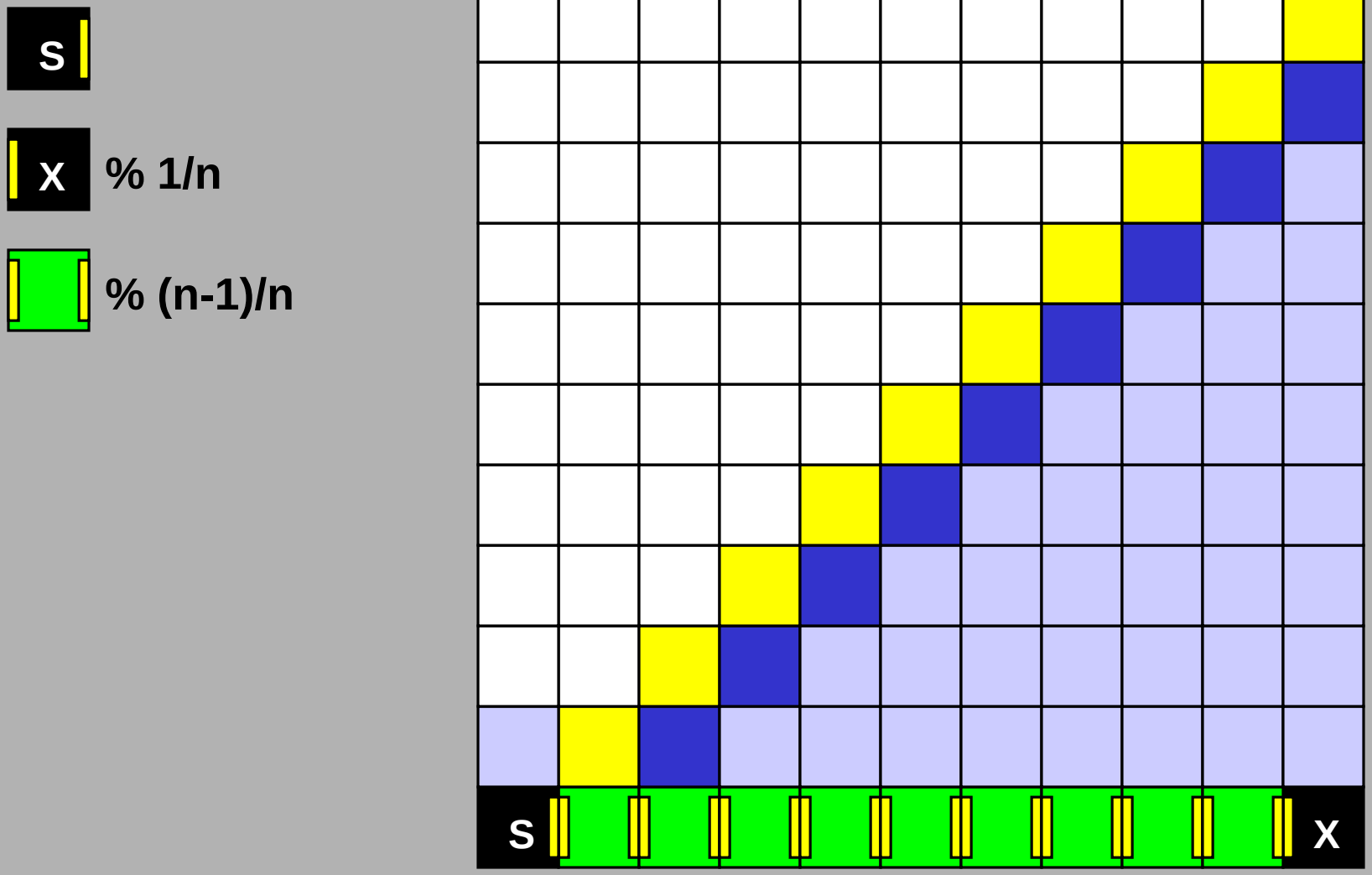
Length has Geometric distribution with p = 1/n



E[Length] = n

Line Estimation of n

[Becker, Rapaport Remila, 2006]



E[Length] = n

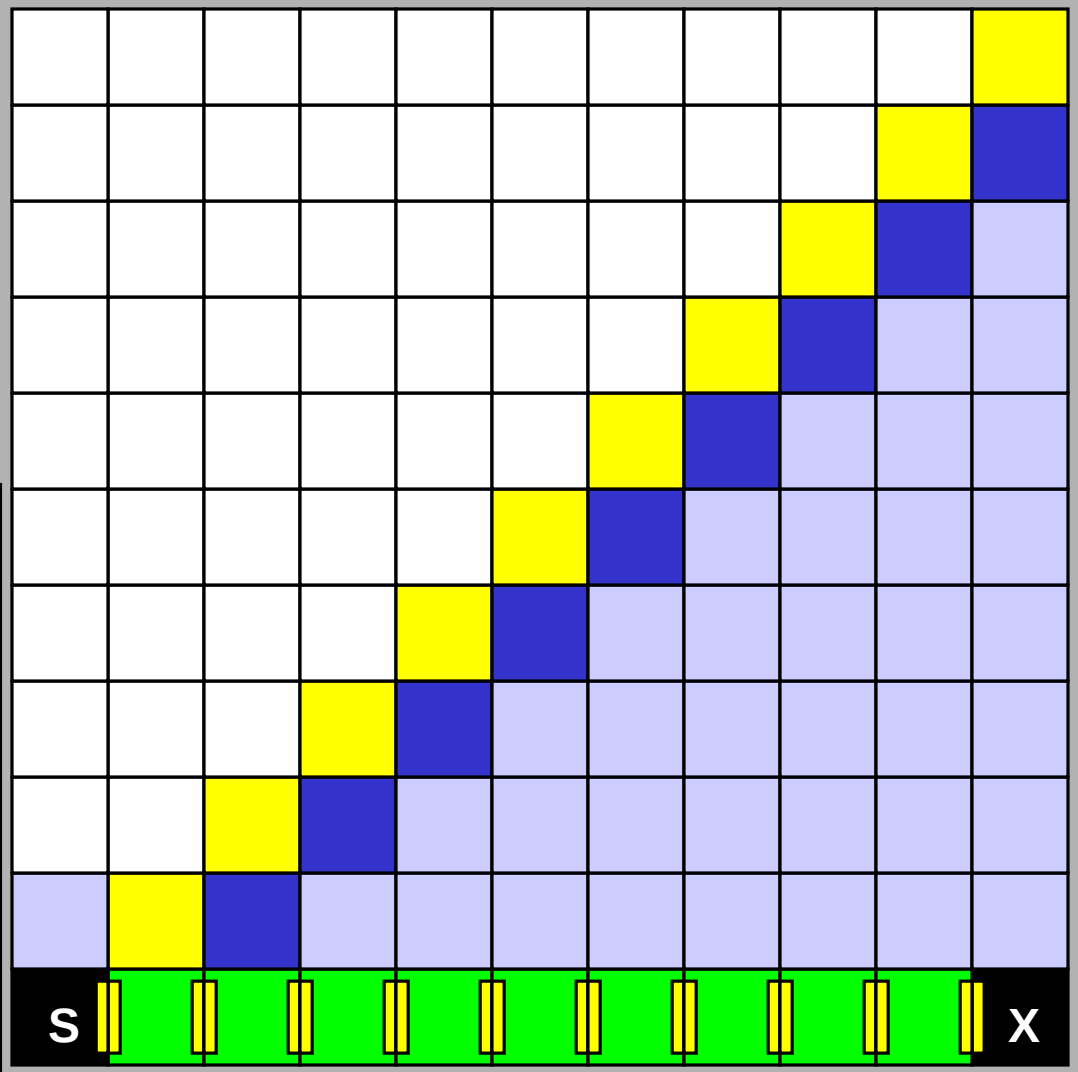
Line Estimation of n

[Becker, Rapaport Remila, 2006]

s

x % 1/n

% (n-1)/n



- Assembles all n x n squares.

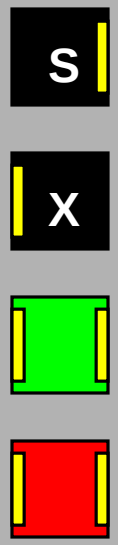
-Expected dimension specified by percentages.

-Geometric distribution:

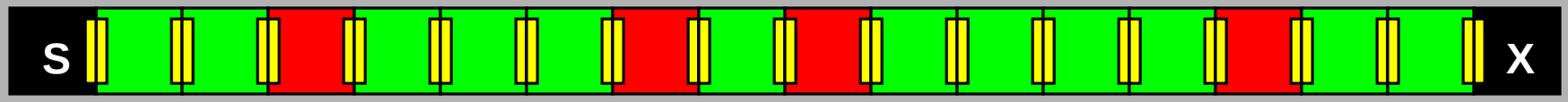
Large variance

E[Length] = n

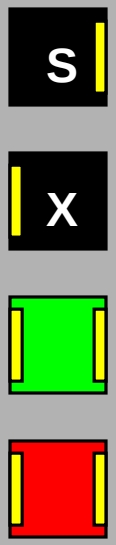
Improved Estimation of n: Binomial Distribution



Key Idea



Improved Estimation of n: Binomial Distribution



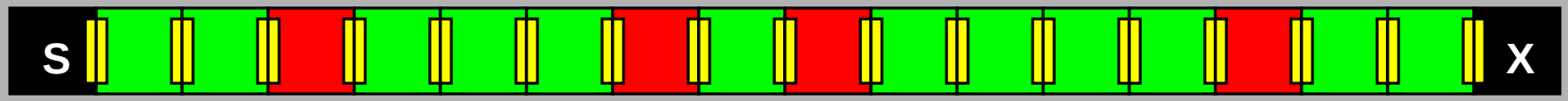
% 1/L

% (L+1)(n-1)/Ln

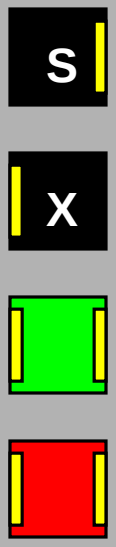
% (L+1)/Ln

Probability of placing a red tile given either a red or green tile is placed:

1/n



Improved Estimation of n: Binomial Distribution



% c

% c/L

% c(L+1)(n-1)/Ln

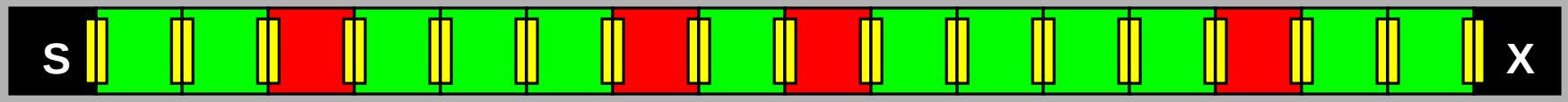
% c(L+1)/Ln

Probability of placing a red tile given either a red or green tile is placed:

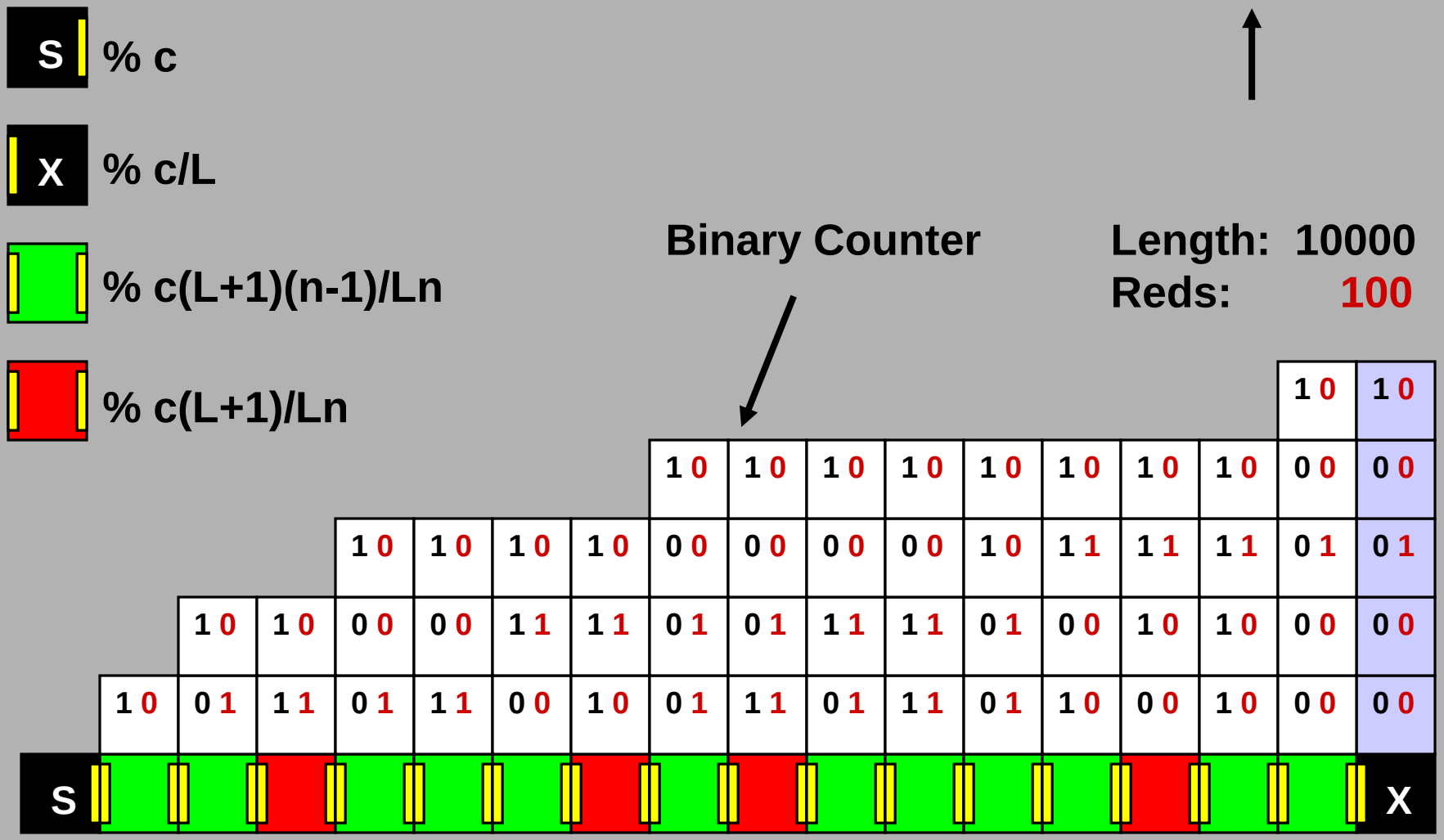
1/n

To compute estimation of n:

Compute LENGTH / REDS



Improved Estimation of n: Binomial Distribution



Improved Estimation of n: Binomial Distribution

s % c

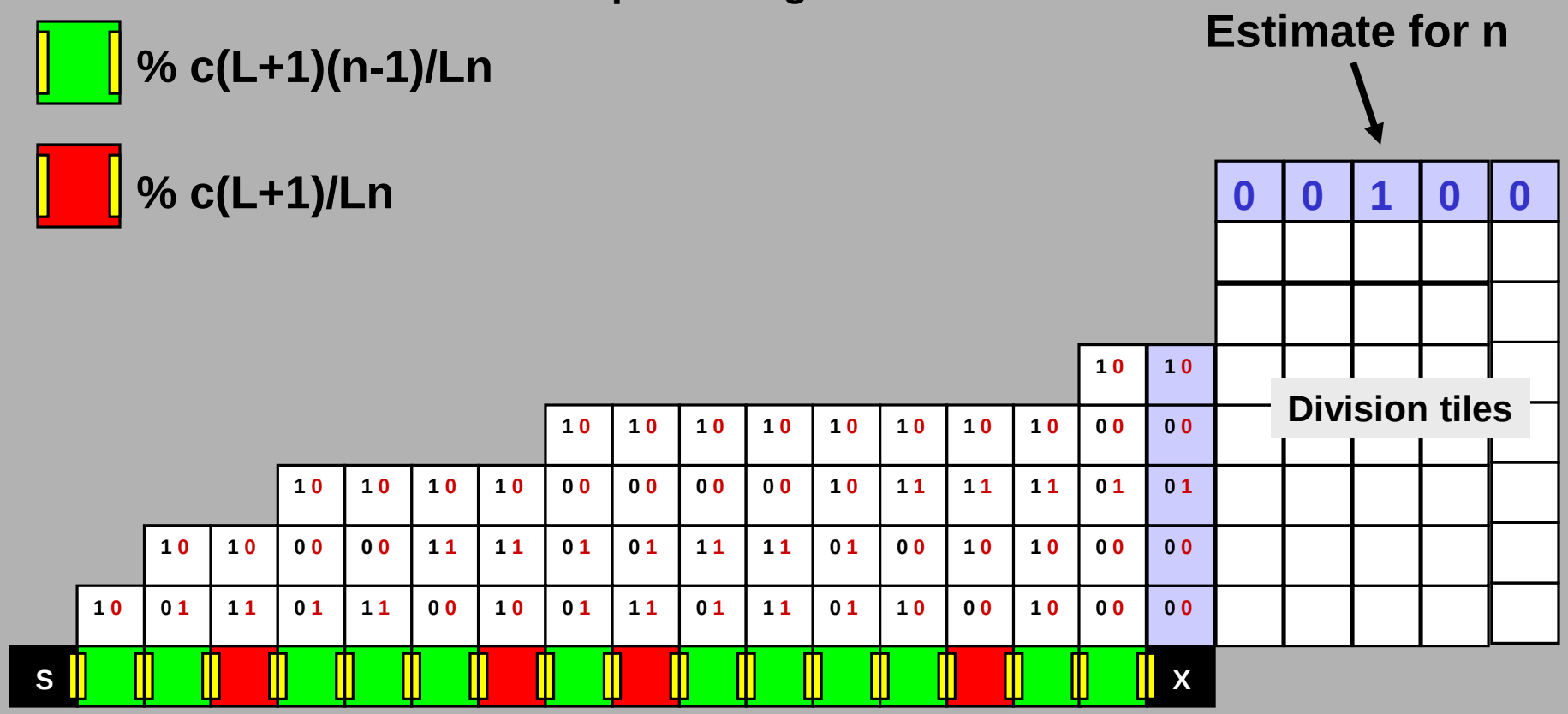
x % c/L

% c(L+1)(n-1)/Ln

% c(L+1)/Ln

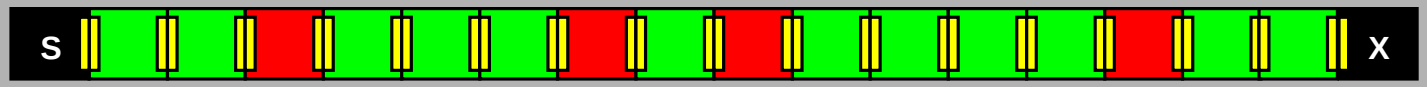
Length: 10000
Reds: 100

Compute Length / Reds: 100



Problem: Estimation Line too Long

Chernoff Bounds only yield high accuracy for Length >> n

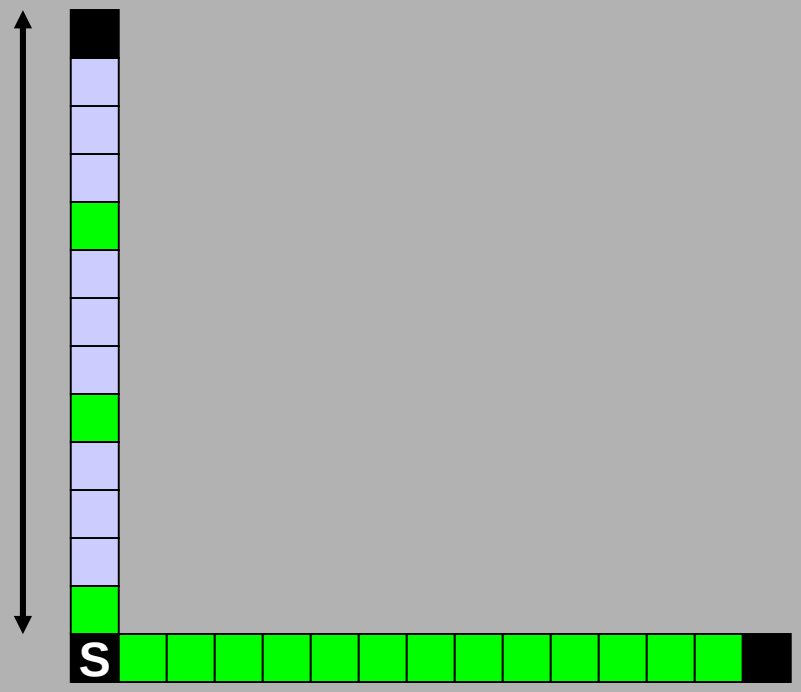


Length >> n :
Too long for an n x n square…

Solution: Estimation Frame

Phase 1:
Build dimensions of frame.

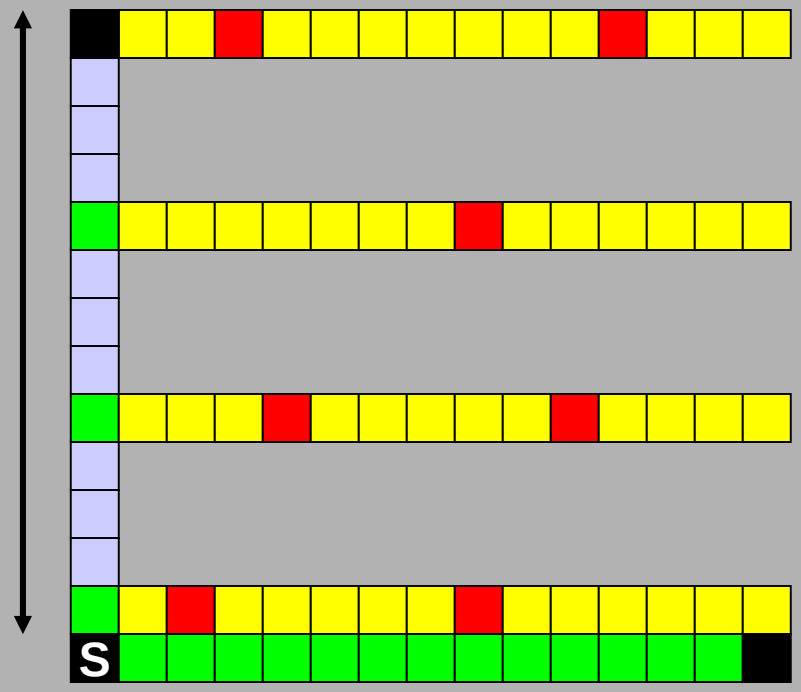
Multiple lines = HEIGHT: Determined by Geometric Distribution



WIDTH: Determined by Geometric Distribution

Solution: Estimation Frame

Multiple lines
= HEIGHT:
Determined by
Geometric
Distribution

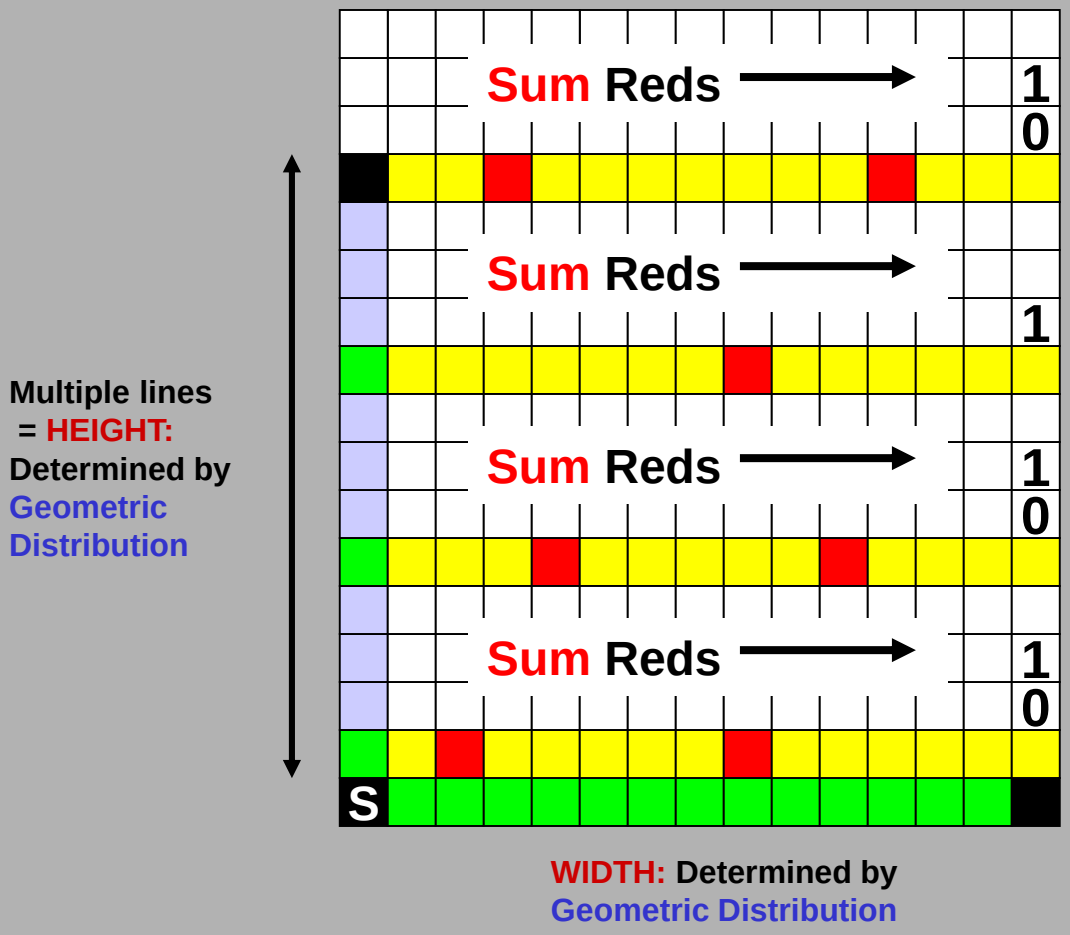


WIDTH: Determined by
Geometric Distribution

Phase 1:
Build dimensions of frame.

Phase 2:
Build Sampling Lines

Solution: Estimation Frame

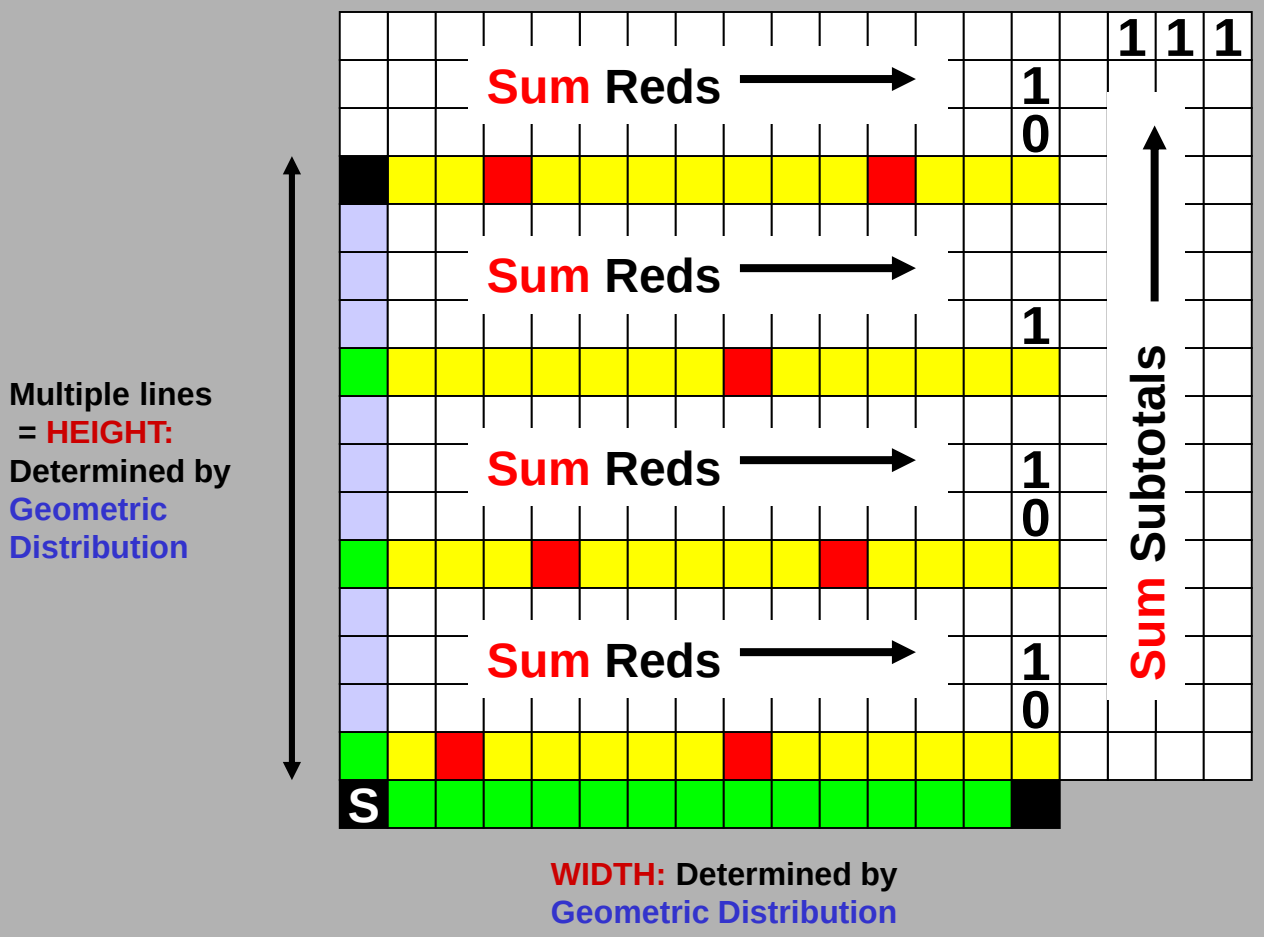


Phase 1:
Build dimensions of frame.

Phase 2:
Build Sampling Lines

Phase 3: Sum Reds and Length for each Line

Solution: Estimation Frame



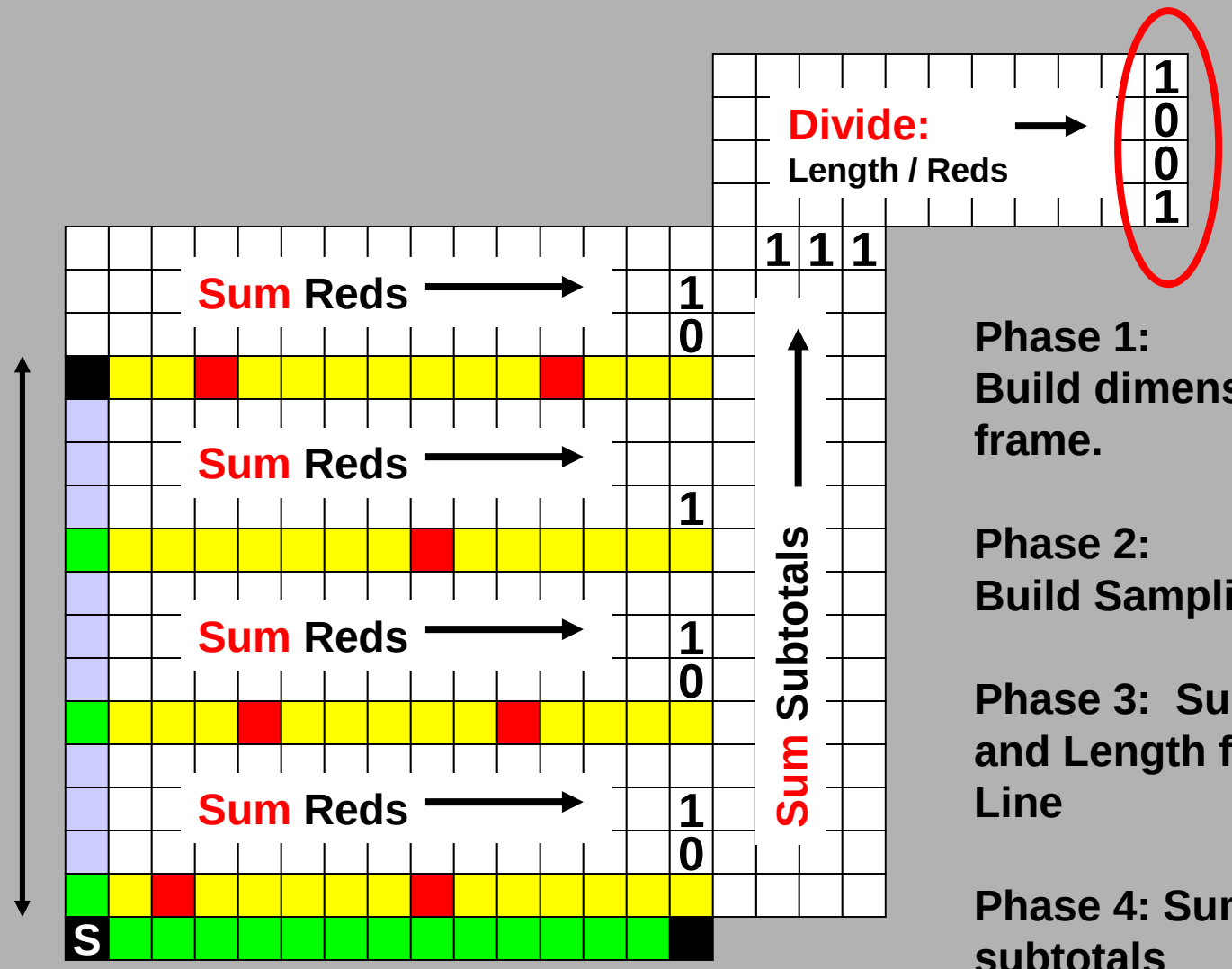
Phase 1:
Build dimensions of frame.

Phase 2:
Build Sampling Lines

Phase 3: Sum Reds and Length for each Line

Phase 4: Sum subtotals

Solution: Estimation Frame



Multiple lines = HEIGHT: Determined by Geometric Distribution

WIDTH: Determined by Geometric Distribution

Phase 1:
Build dimensions of frame.

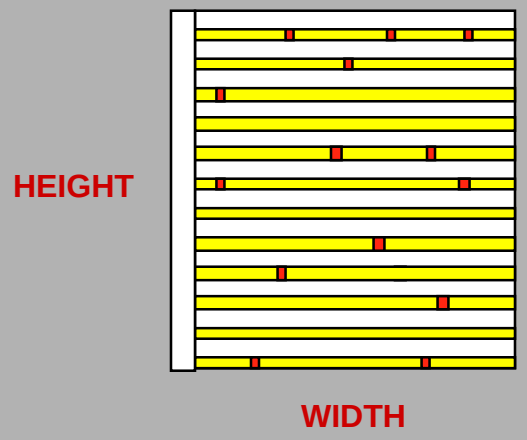
Phase 2:
Build Sampling Lines

Phase 3: Sum Reds and Length for each Line

Phase 4: Sum subtotals

Phase 5: Compute Length to Reds ratio

Solution: Estimation Frame



With high probability:

HEIGHT < n

WIDTH < n

→ **Frame fits within n x n square**

With high probability:

HEIGHT x WIDTH >> n

Chernoff Bounds imply:

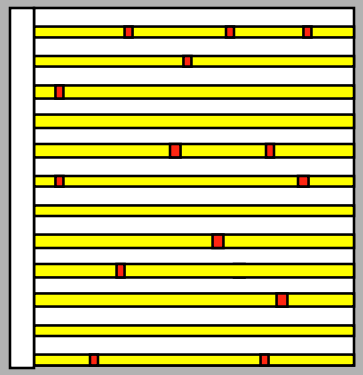
→ **Estimation is accurate with high probability:**

$(1 - \varepsilon)n' < n < (1 + \varepsilon)\, n'$

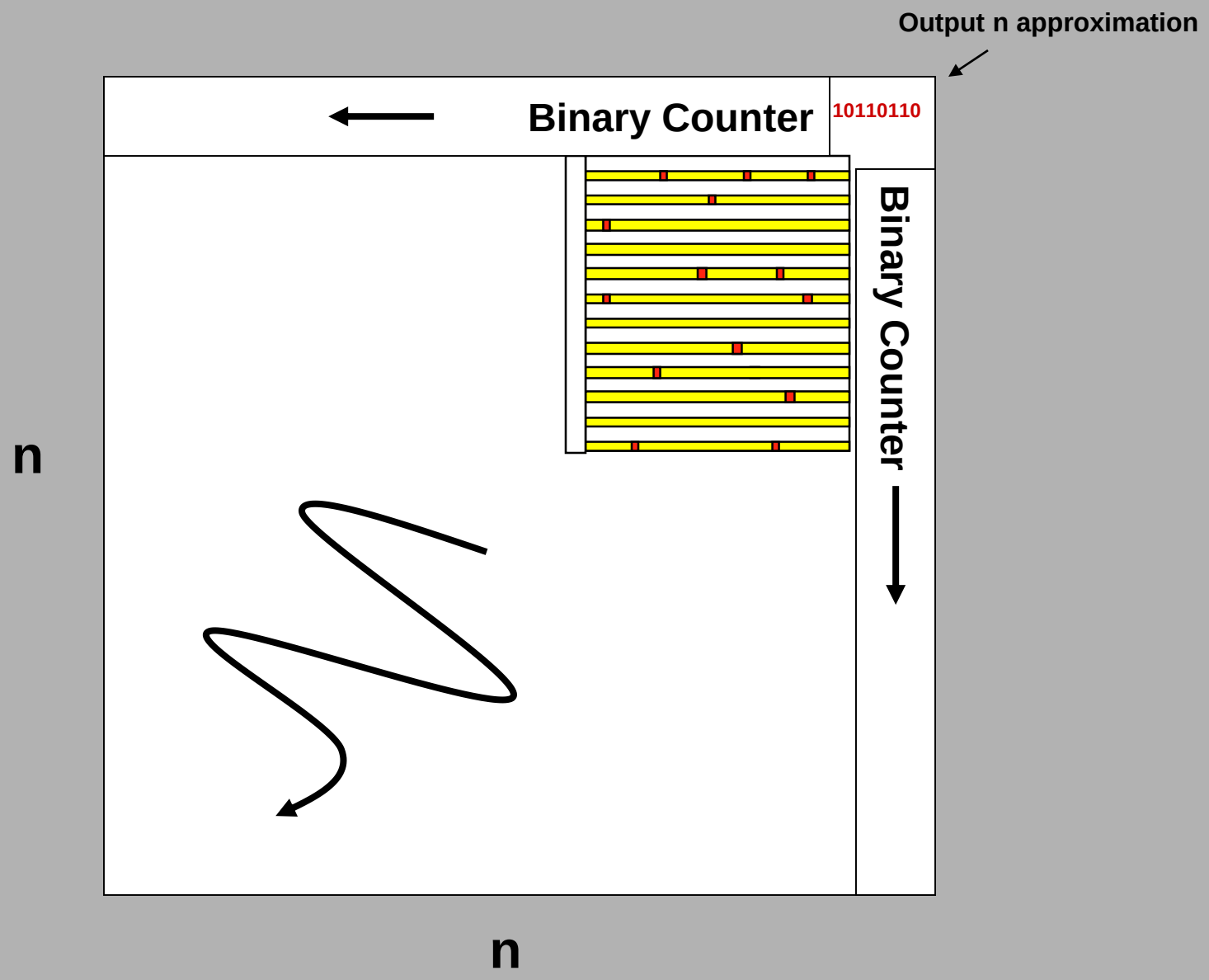
Finish off Square

Output n approximation

10110110



Finish off Square



Result:

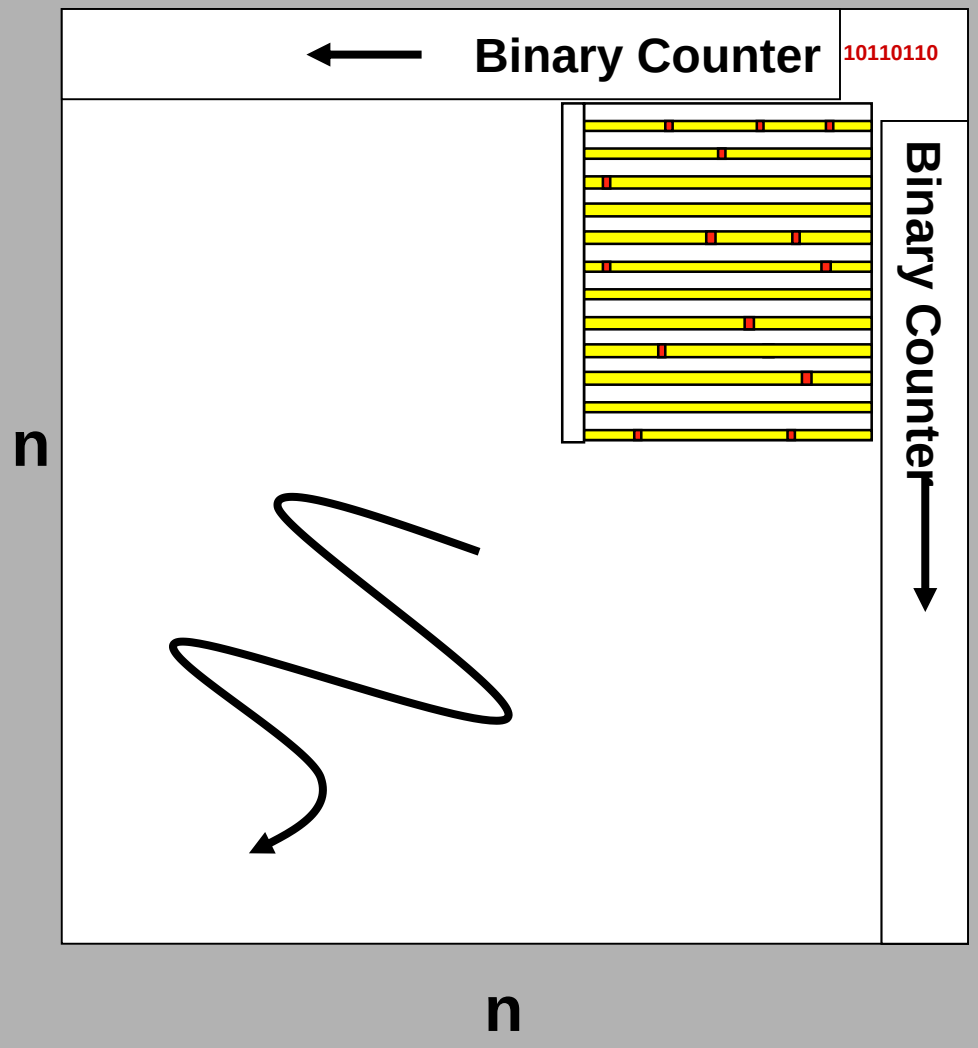
We have a fixed size O(1) tileset that:

-For any given ε, δ
- $n > C(\varepsilon, \delta)$

We can assign percentages such that:

With probability at least $1-\delta$, a size $n' \times n'$ square is assembled with

$$(1-\varepsilon)n < n' < (1+\varepsilon)n$$



Result:

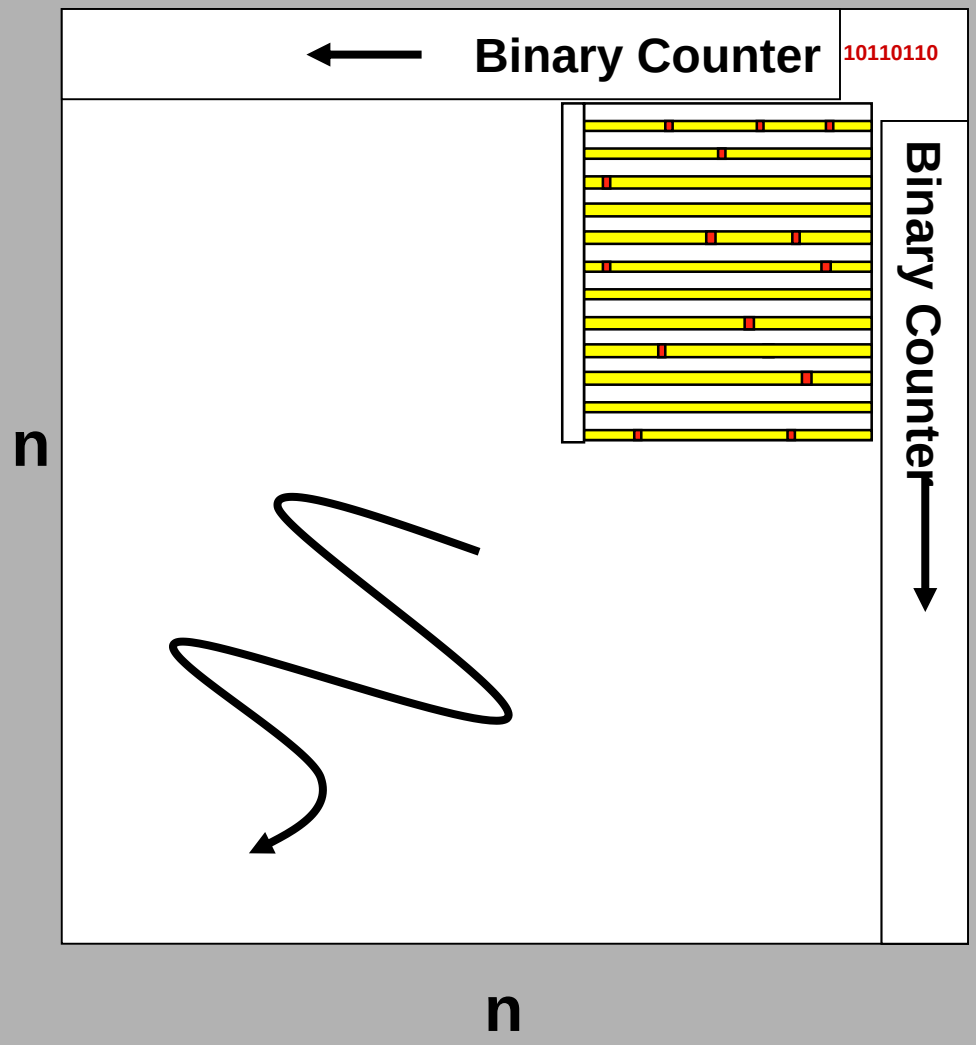
We have a fixed size O(1) tileset that:

-For any given ε, δ
- $n > C(\varepsilon, \delta)$

We can assign percentages such that:

With probability at least $1-\delta$, a size $n' \times n'$ square is assembled with

$$(1-\varepsilon)n < n' < (1+\varepsilon)n$$



Question: Exact squares with high probability?
(yes, see the paper in your reading)