

- 0. MOTIVATION
- 1. PROBLEM SOLVING
- 2. INFORMATION
- 3. COUNTING
- 4. BITS
- 5. CODES
- 6. ALGORITHMS
- 7. COMPUTERS
- 8. ARITHMETIC
- 9. MEMORY
- 10. ANALOG VS. DIGITAL

The slides are meant as visual support for the lecture.  
They are neither a documentation nor a script.

Please do not print the slides.

Comments and feedback at [n.meseth@hs-osnabrueck.de](mailto:n.meseth@hs-osnabrueck.de)

# MOTIVATION

a few  
experts

digitally uneducated  
society



digitally illiterate society with a few experts

# collective understanding

you?



society with a distributed and high degree of digital education

representation

representation

processing

representation

processing

programming

representation

processing

programming

digital fundamentals

data analysis

representation

processing

programming

digital fundamentals

data analysis

artificial  
intelligence

representation

processing

programming

digital fundamentals

digital applications

data analysis

artificial  
intelligence

representation

processing

programming

digital fundamentals

digital applications

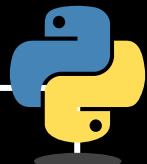
data analysis

artificial  
intelligence

representation

processing

programming

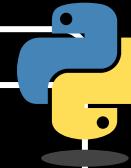


digital fundamentals

digital applications

data analysis

artificial  
intelligence



representation

processing

programming



digital fundamentals



# PROBLEM SOLVING

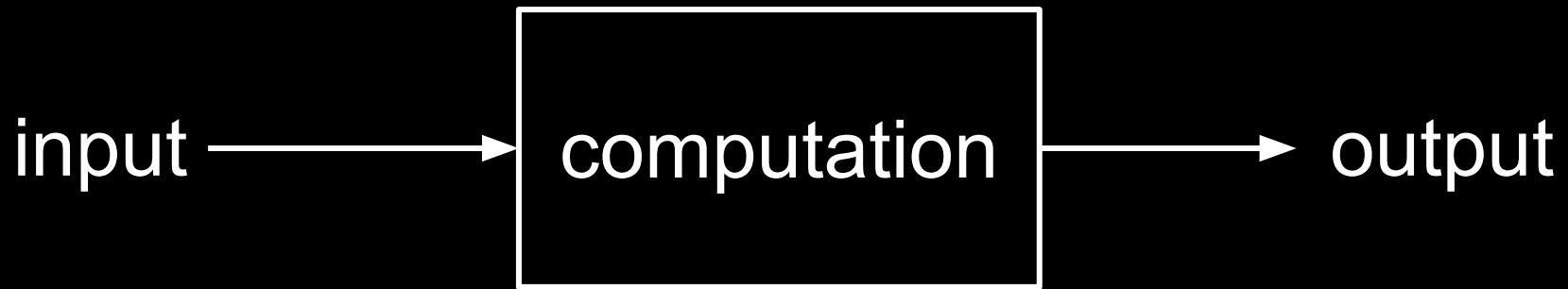
# a model for solving problems



# a model for solving problems

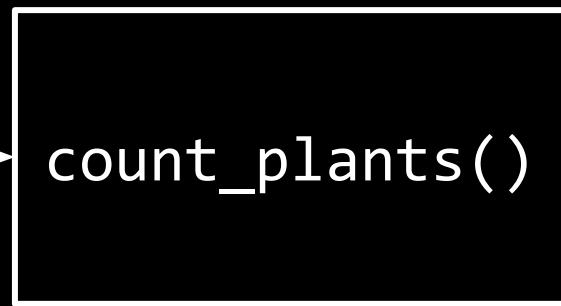


# a model for solving problems









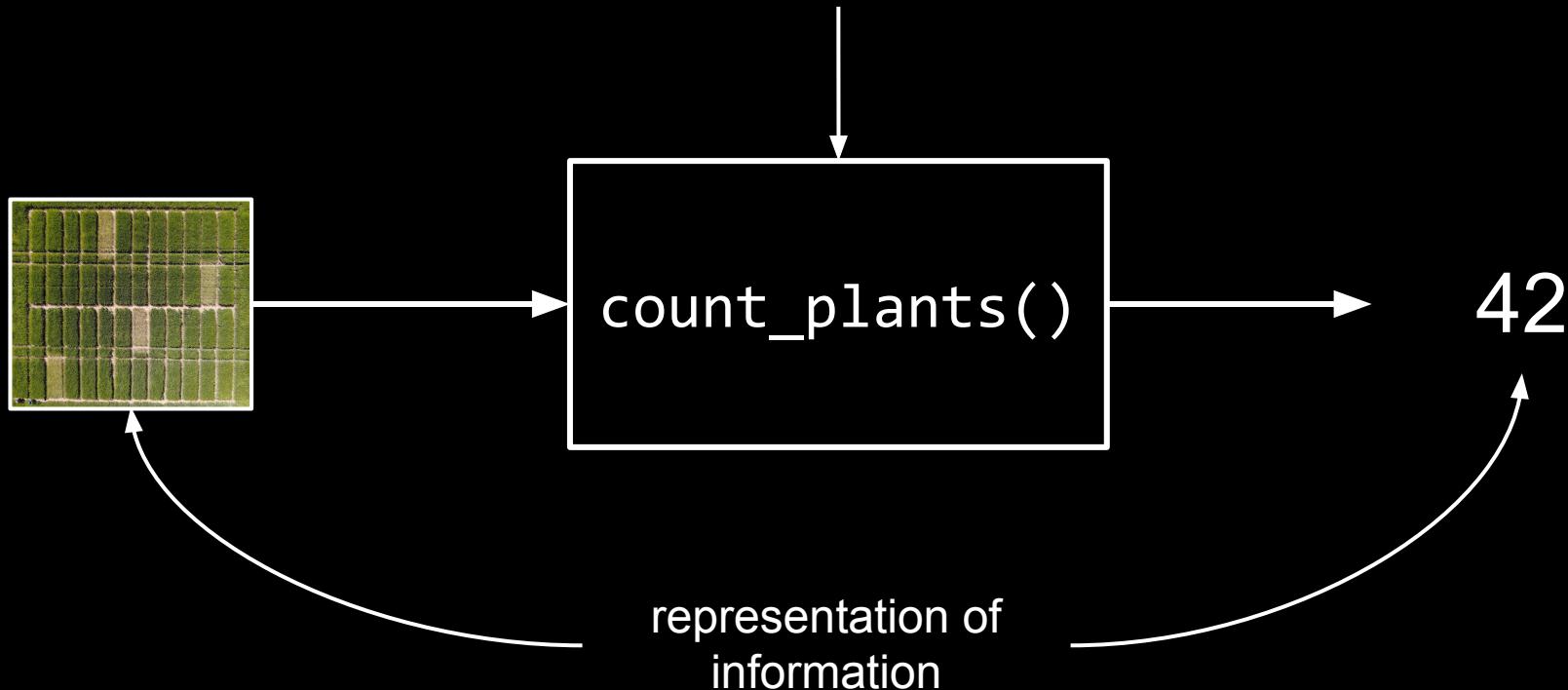
output



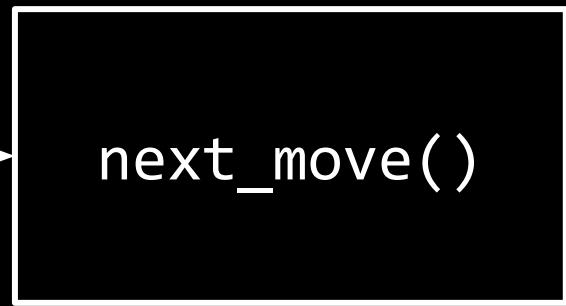
count\_plants()

42

processing of  
information



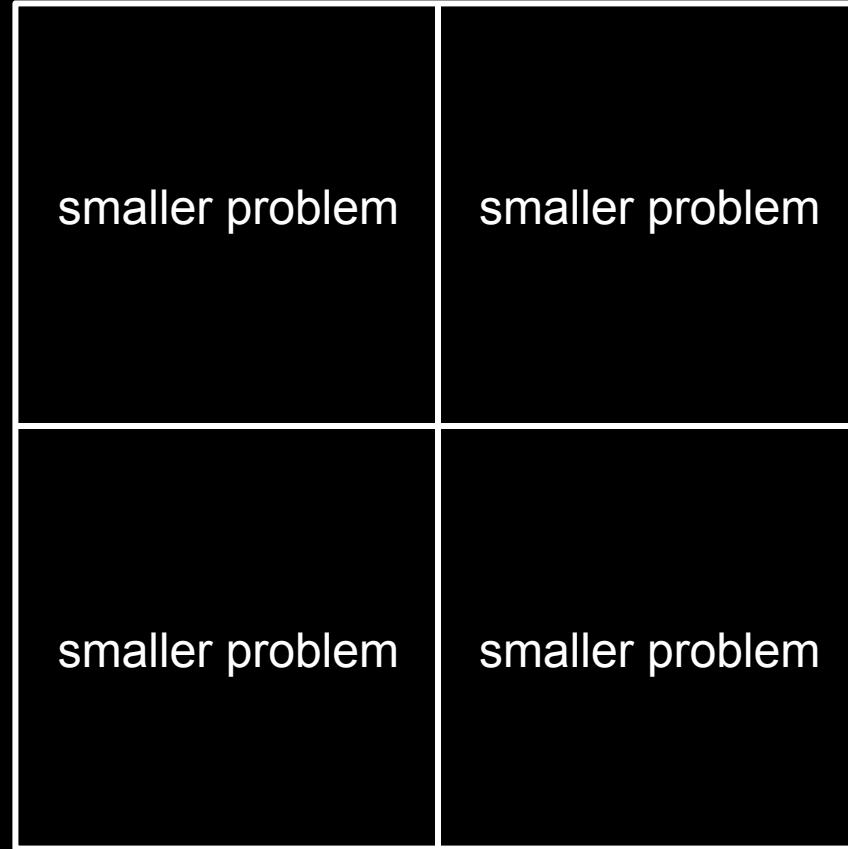




# problem solving strategies

divide and conquer

large and complex problem





sorted list +  
element



is 67 a prime number?

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

## linear search



2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

## linear search



2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

## linear search



2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

## linear search



2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

## linear search

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97



19 steps... can't we do better?

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97



large and complex  
problem

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

## large and complex problem

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

## smaller problem

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41

## smaller problem

43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

## binary search

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

binary search

67 != 41



2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

binary search

$67 > 41$



2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, **41**,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

binary search

$67 > 41$



2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

## binary search

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97



$$67 \neq 71$$

## binary search

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97



$$67 \neq 71$$

## binary search

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97



$$67 < 71$$

## binary search

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97



67 != 59

## binary search

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97



$$67 > 59$$

## binary search

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97



$$67 = 67$$

## binary search

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97



$$67 = 67$$

3 splits → much better

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,  
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97



$$67 = 67$$



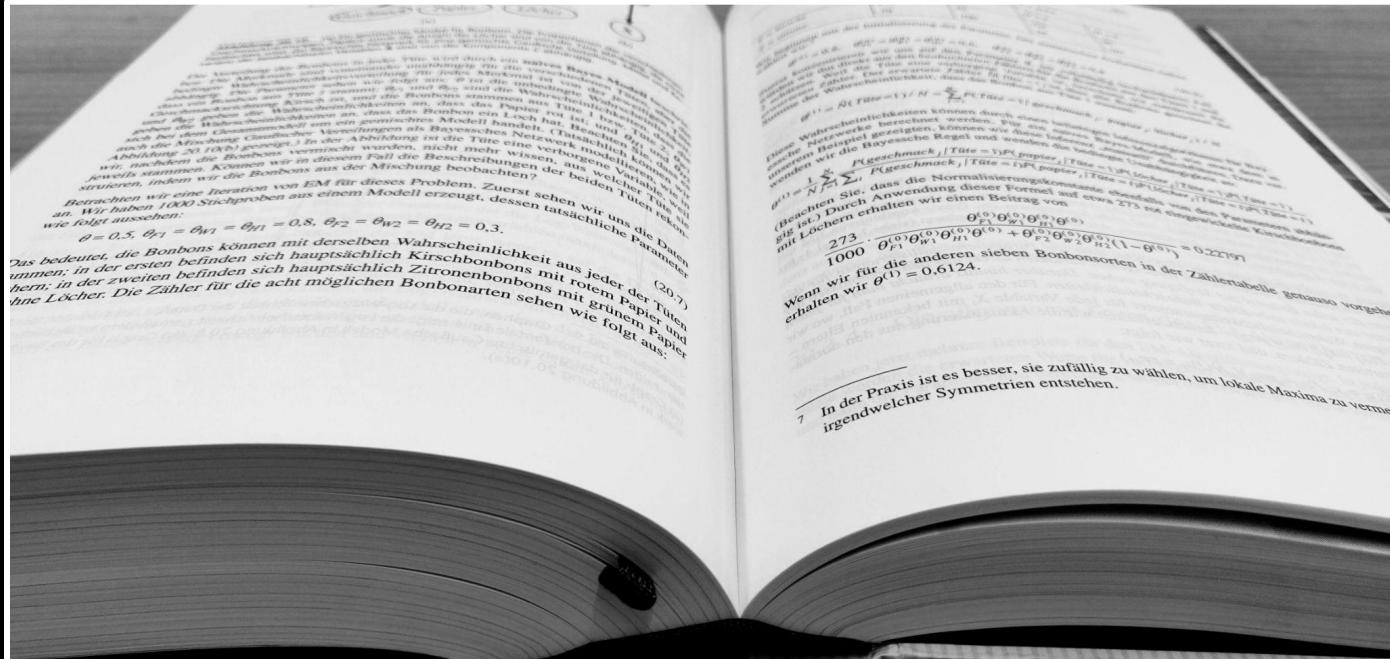
how efficient are linear and  
binary search in general?



count\_words()

word count

**how many words are in the book?**



strategies, anyone?

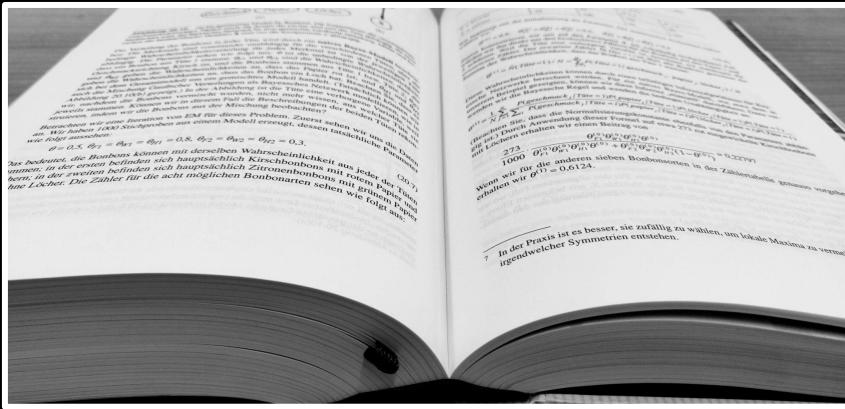


7 In der Praxis ist es besser, sie zufällig zu wählen, um lokale Maxima zu vermeiden.



page 1









187



page 1

212



page 2

55



page 3



187



page 1

212



page 2

55



page 3

142



page n



7. In der Praxis ist es besser, sie zufällig zu wählen, um lokale Maxima zu vermeiden.



$n = 1327$  pages

$\varnothing 2:23$  minutes per page

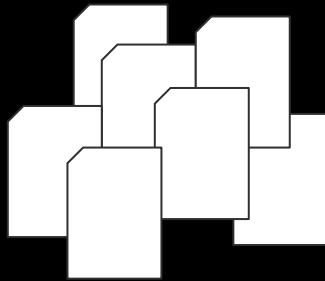
$\sim 52.34$  hours

divide and conquer

+

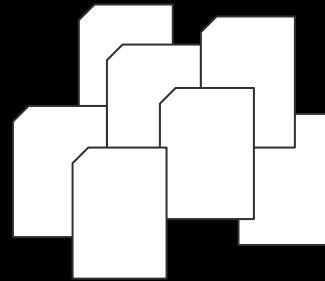
?

pages 1 - 700



student 1

pages 701 - 1327



student 2

pages 1 - 350

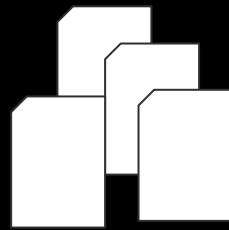
pages 351 - 700

pages 701 - 1050

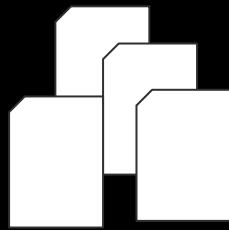
pages 1051- 1327



student 1



student 2



student 3



student 4

divide and conquer

+

distribution and parallelization



# INFORMATION

“Information is that which allows you to make a correct prediction with accuracy better than chance.”

Adami, Christoph. “What Is Information?” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2063, Mar. 2016, p. 20150230, <https://doi.org/10.1098/rsta.2015.0230>.

“Information is that which allows you to make a correct prediction with accuracy better than chance.”

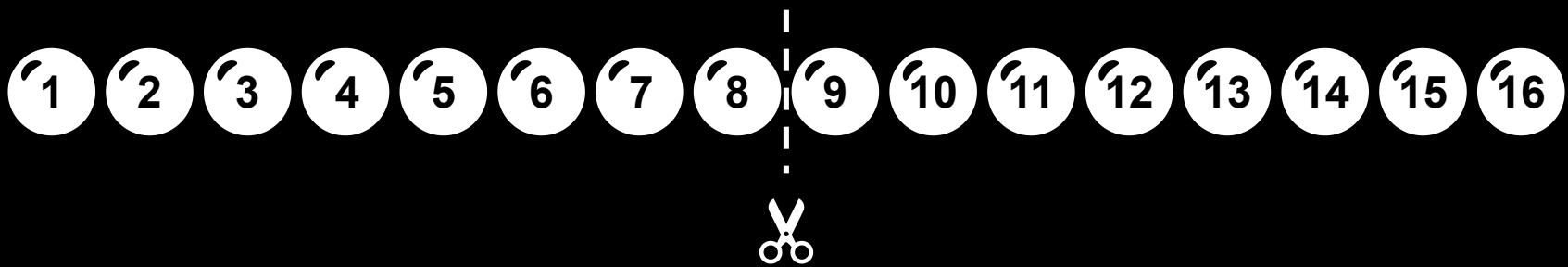
Adami, Christoph. “What Is Information?” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2063, Mar. 2016, p. 20150230, <https://doi.org/10.1098/rsta.2015.0230>.

what number am I thinking of?

what is the most efficient approach?



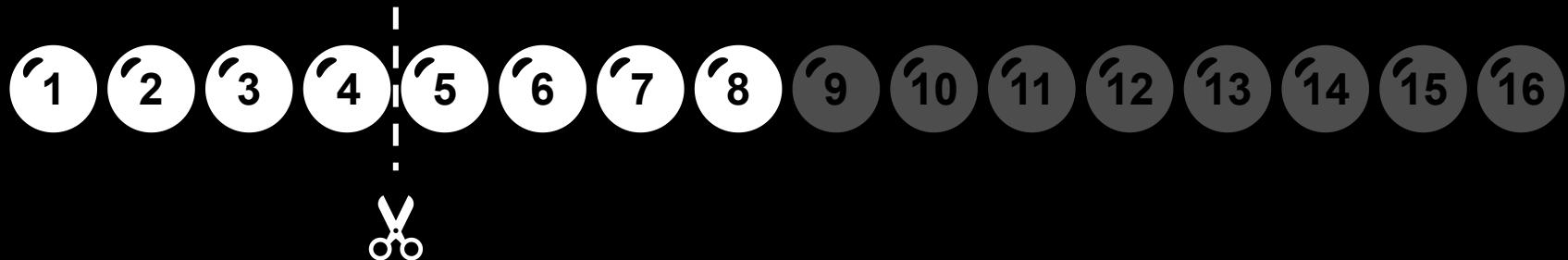
is it  $> 8$ ?



is it  $> 8?$  ✗



is it  $> 8?$  X  
is it  $> 4?$



is it  $> 8?$  ✗

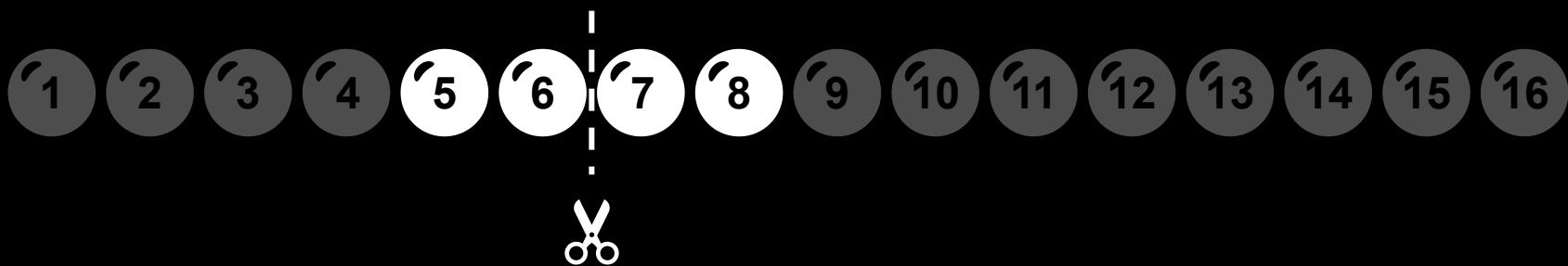
is it  $> 4?$  ✓



is it > 8? ✗

is it > 4? ✓

is it > 6?



is it > 8? ✗

is it > 4? ✓

is it > 6? ✓

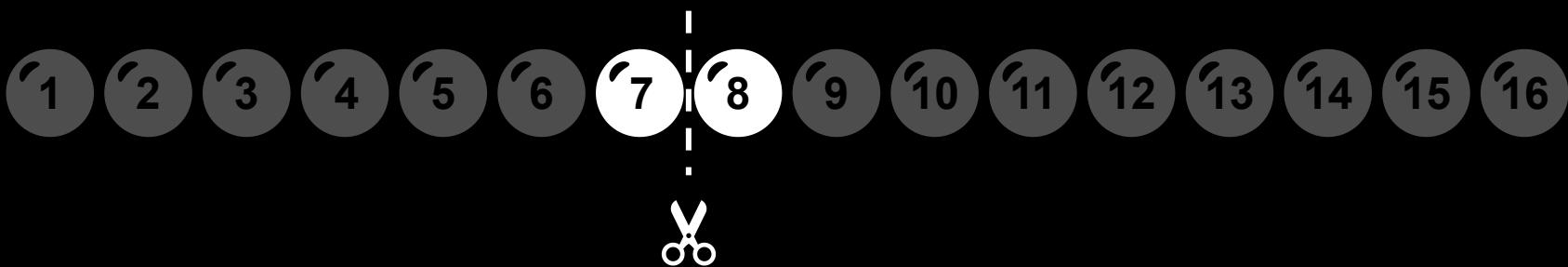


is it > 8? ✗

is it > 4? ✓

is it > 6? ✓

is it > 7?



is it > 8? ✗

is it > 4? ✓

is it > 6? ✓

is it > 7? ✗



is it > 8? ✗

is it > 4? ✓

is it > 6? ✓

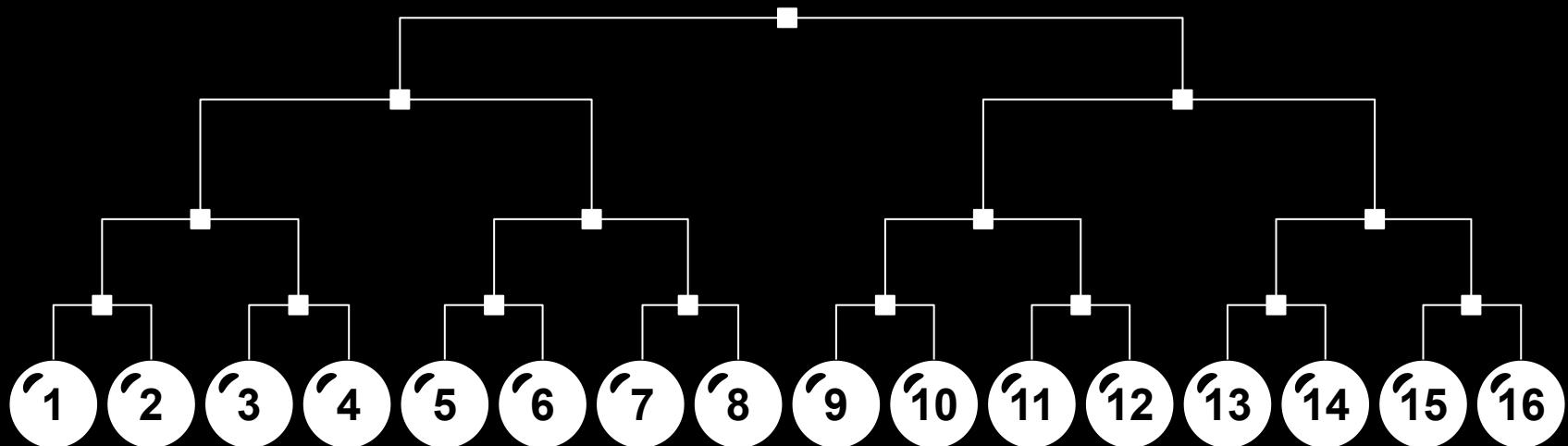
is it > 7? ✗



with 4 questions from 16 to 1

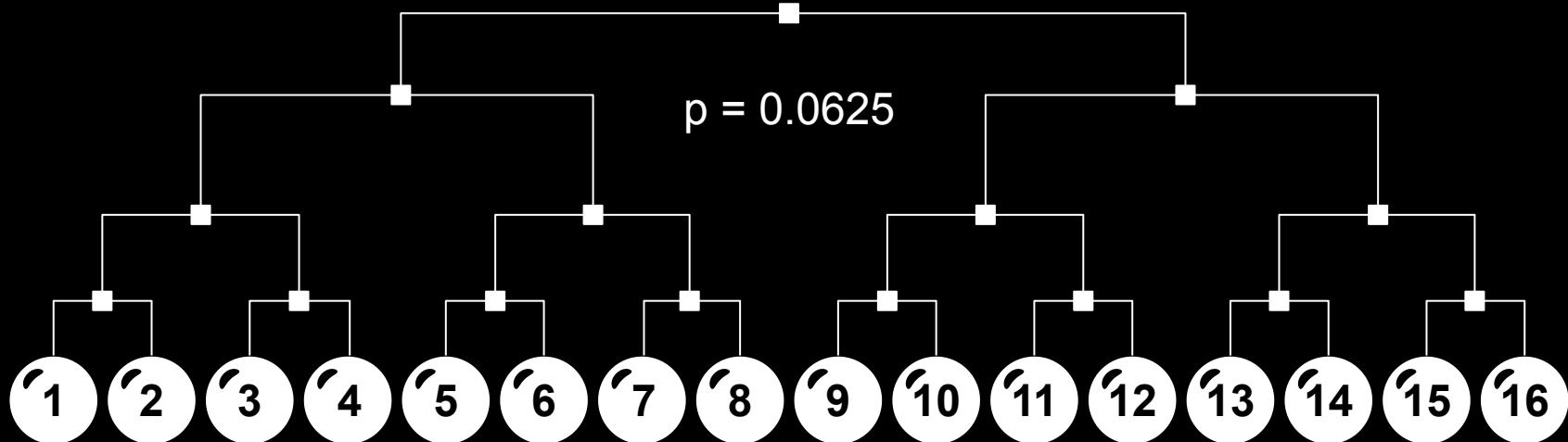
where is the information?

# where is the information?



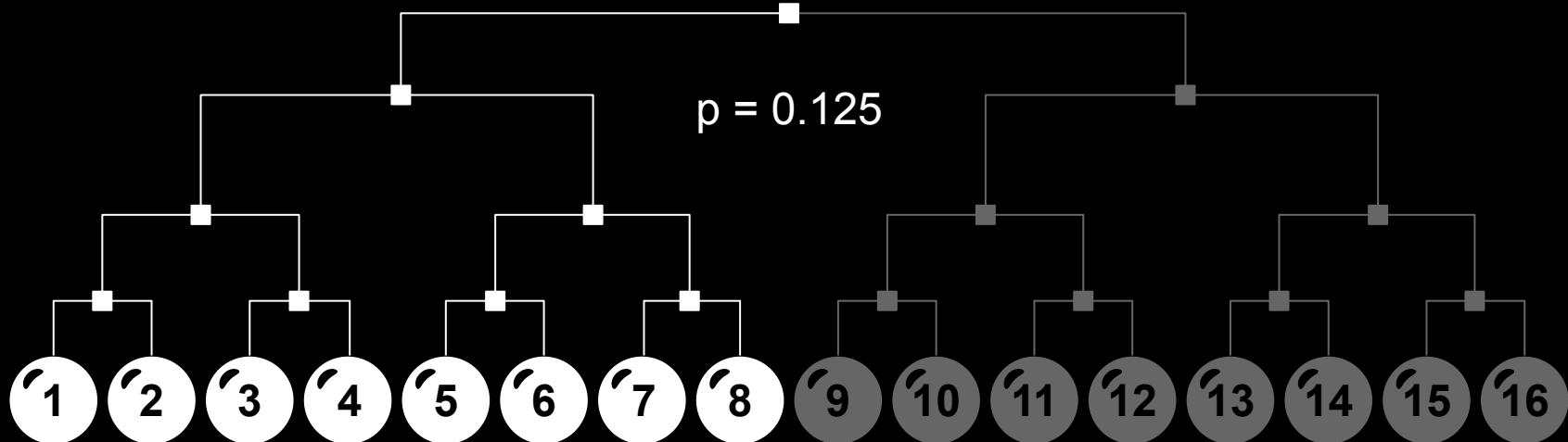
# where is the information?

$N = 16$



# where is the information?

$$\begin{matrix} N = 16 \\ N = 8 \end{matrix}$$

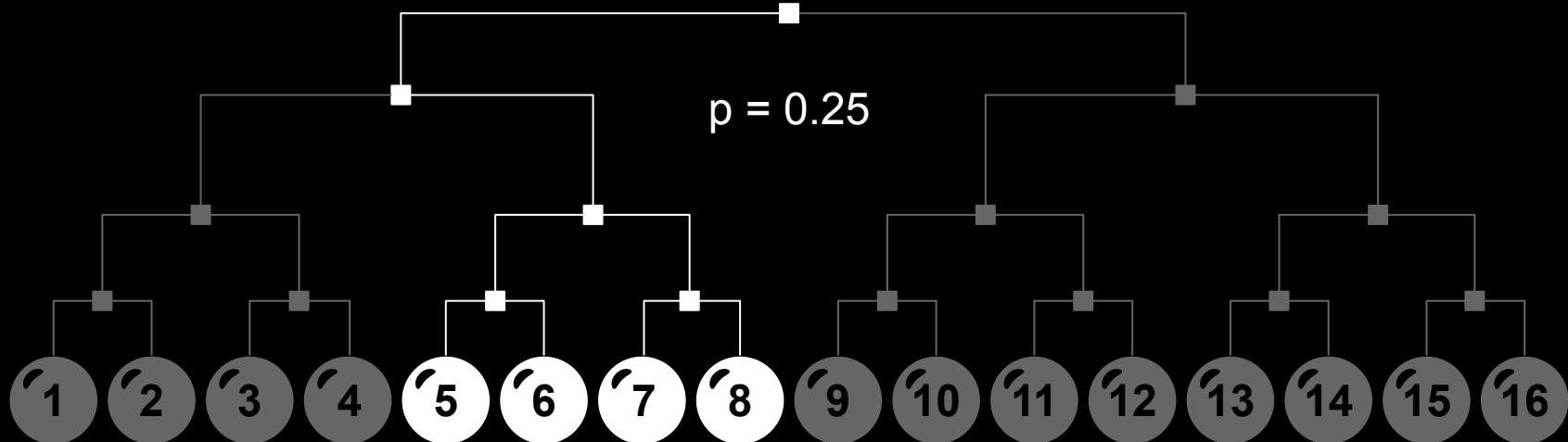


# where is the information?

$\overline{N} = 16$

$\overline{N} = 8$

$N = 4$



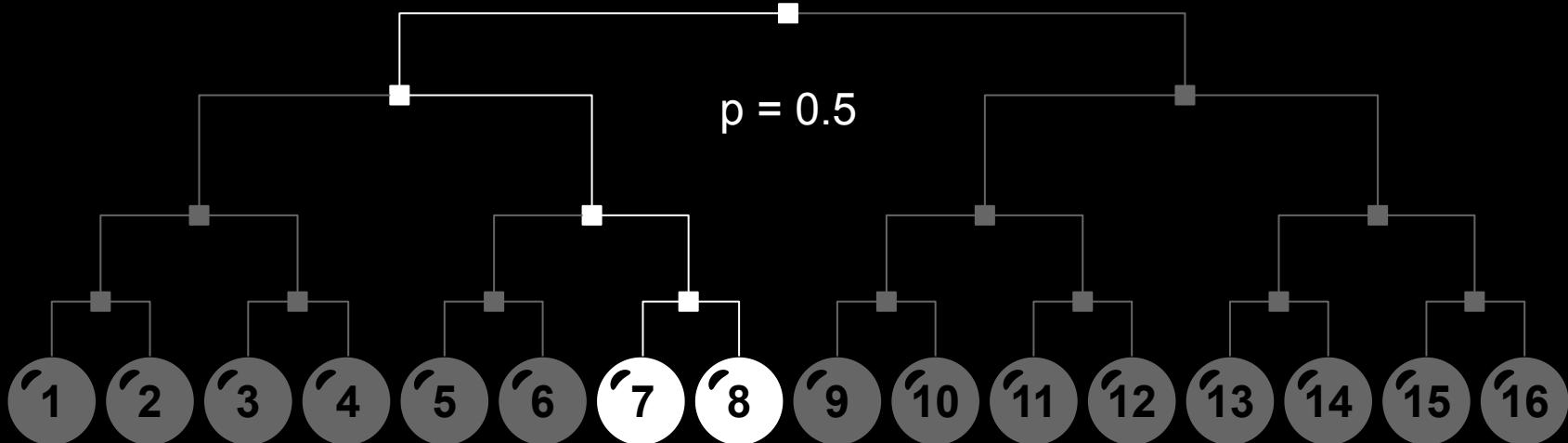
# where is the information?

$N = 16$

$N = 8$

$N = 4$

$N = 2$



# where is the information?

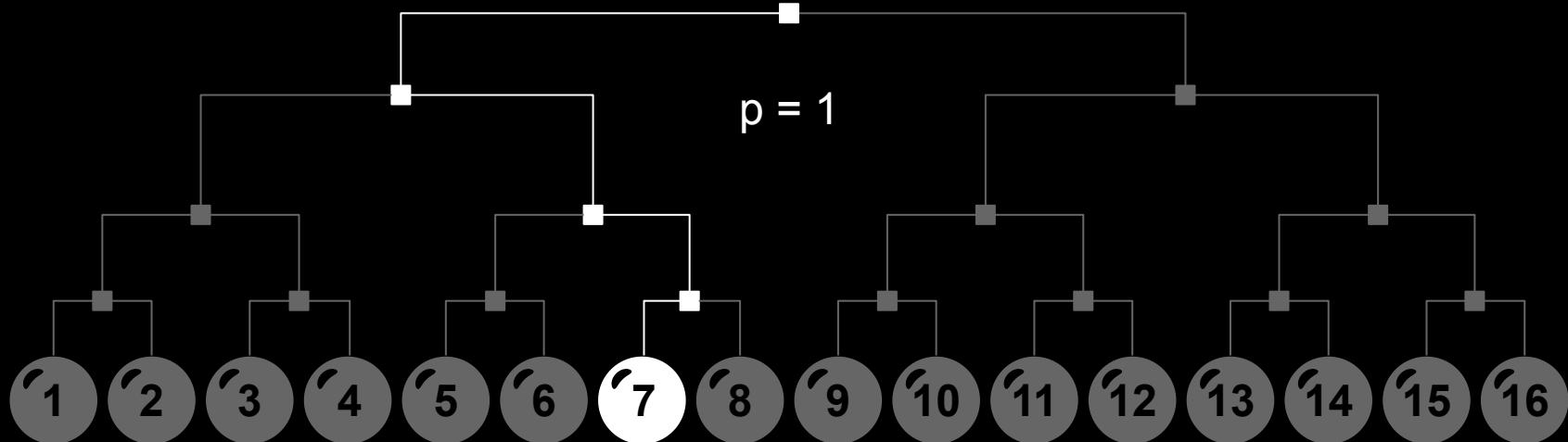
$N = 16$

$N = 8$

$N = 4$

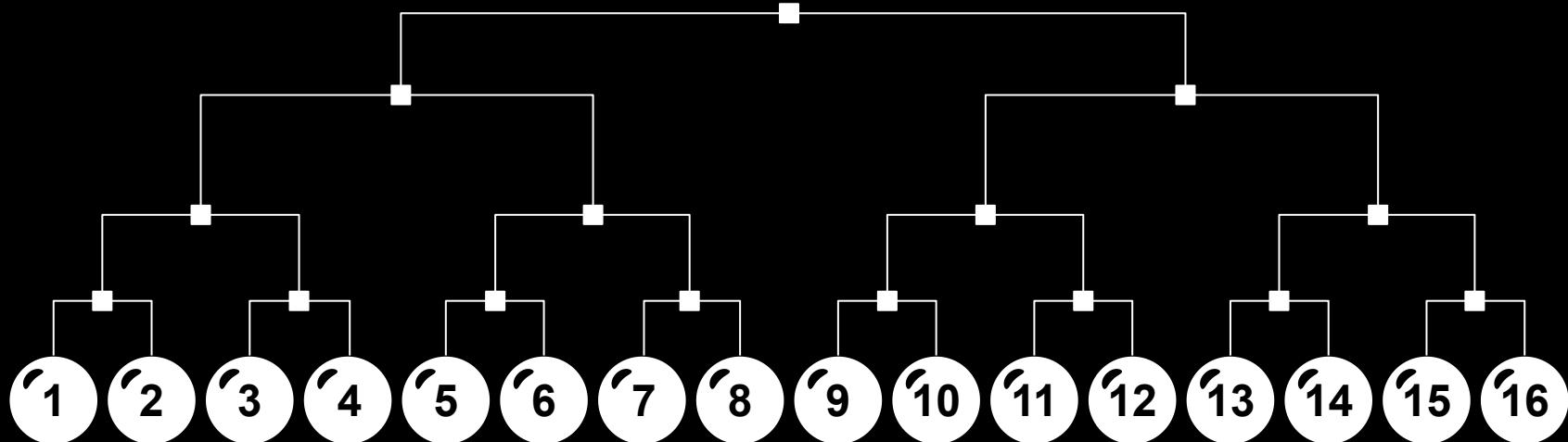
$N = 2$

$N = 1$



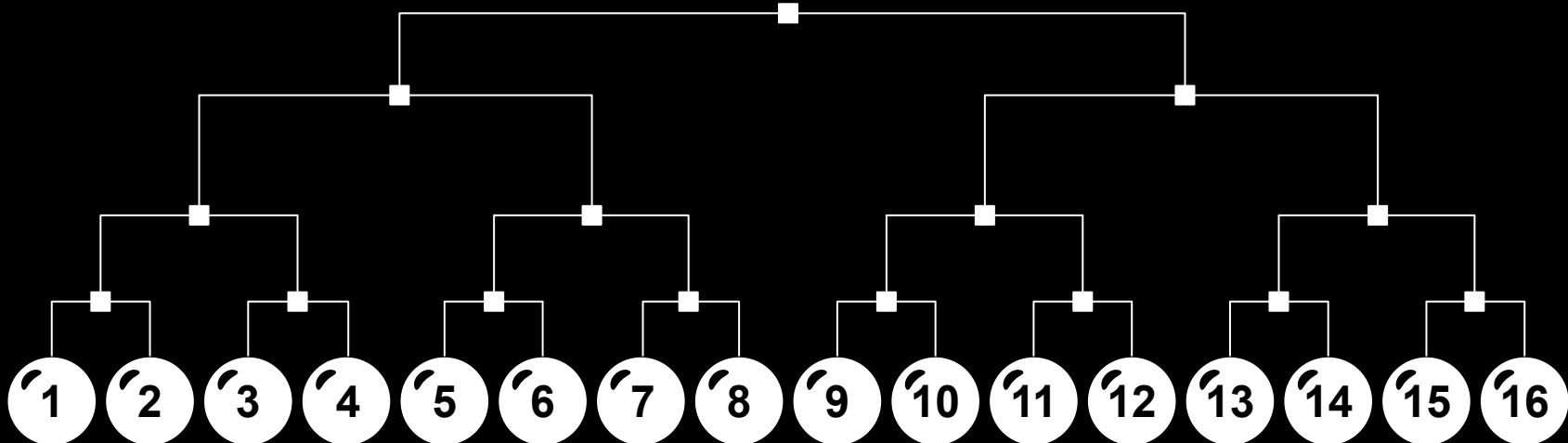
information = reduced uncertainty  
uncertainty is measured with the logarithm of N

$$H = \log_2(N)$$

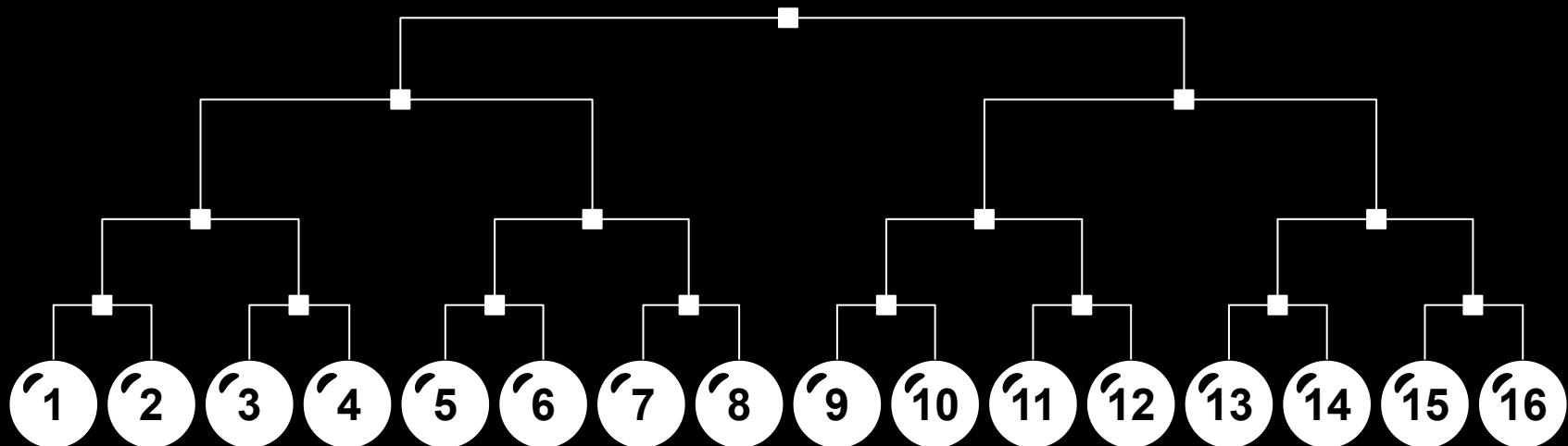


or: how often can we cut the remaining possibilities in half?

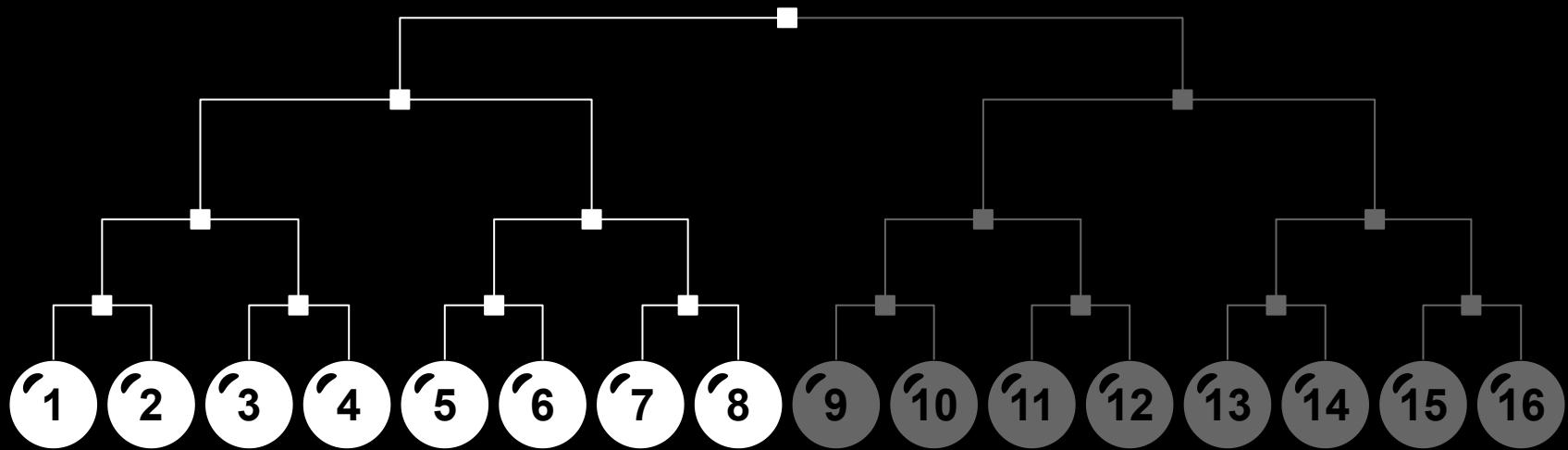
$$H = \log_2(N)$$



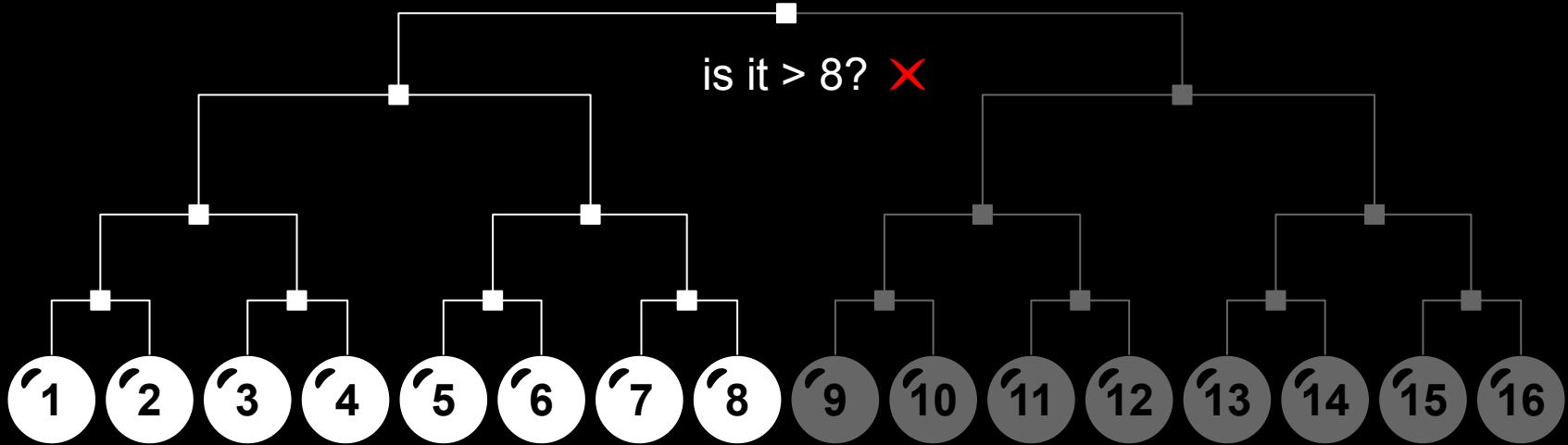
$$H_0 = \log_2(16) = 4$$



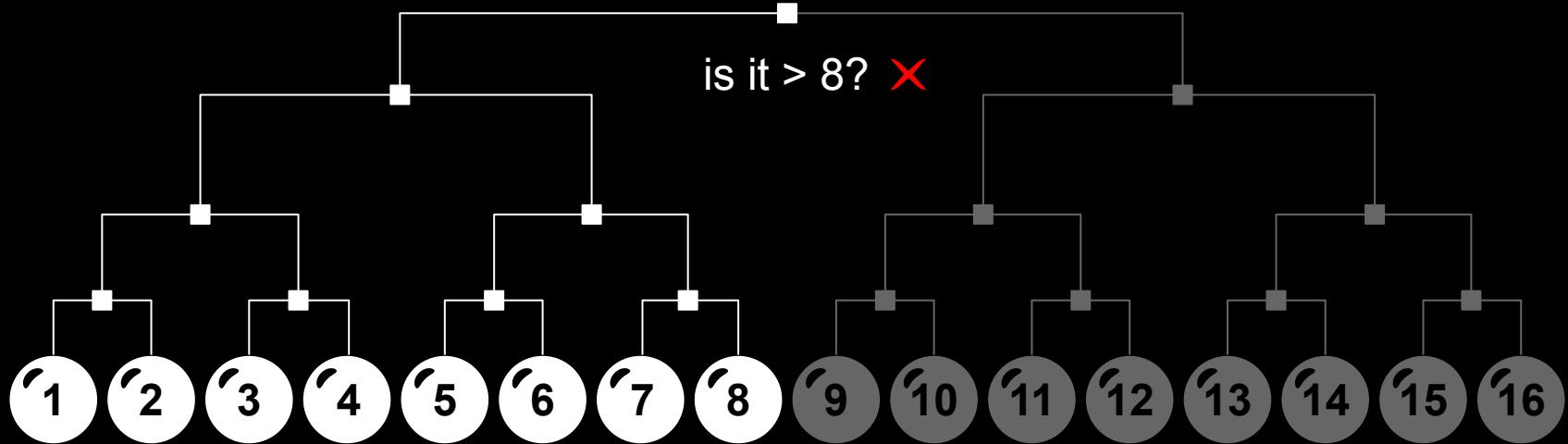
$$H_1 = \log_2(8) = 3$$



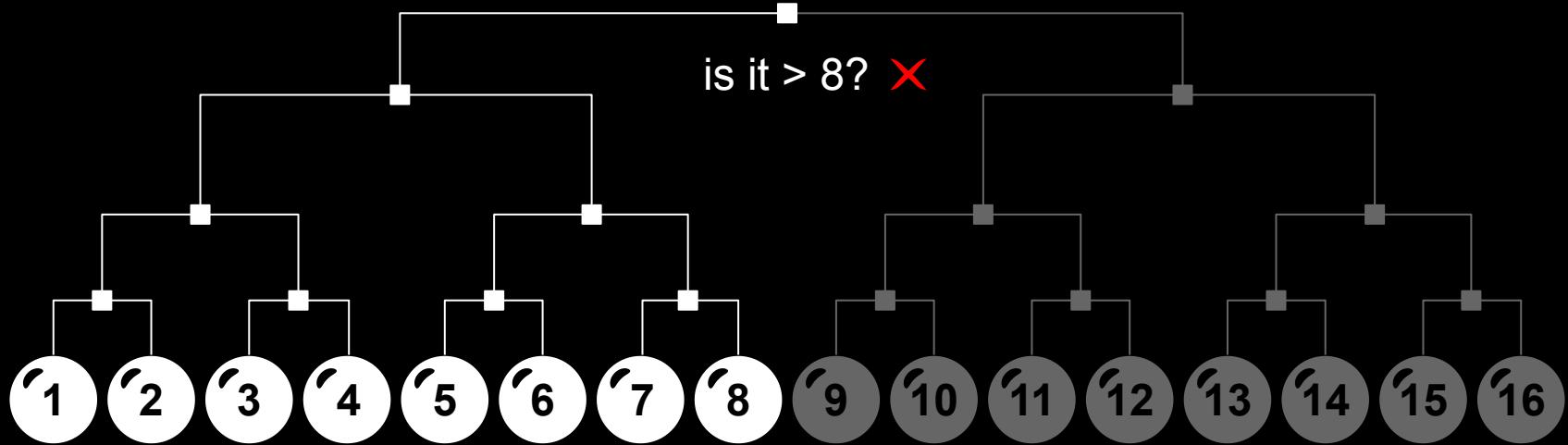
$$I = H_0 - H_1$$



$$I = \log_2(16) - \log_2(8)$$



$$I = 4 - 3 = 1$$

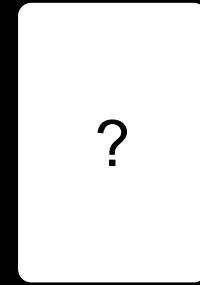
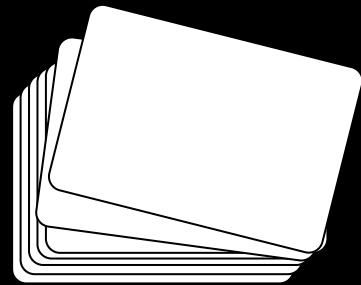


uncertainty and information are  
measured in bits

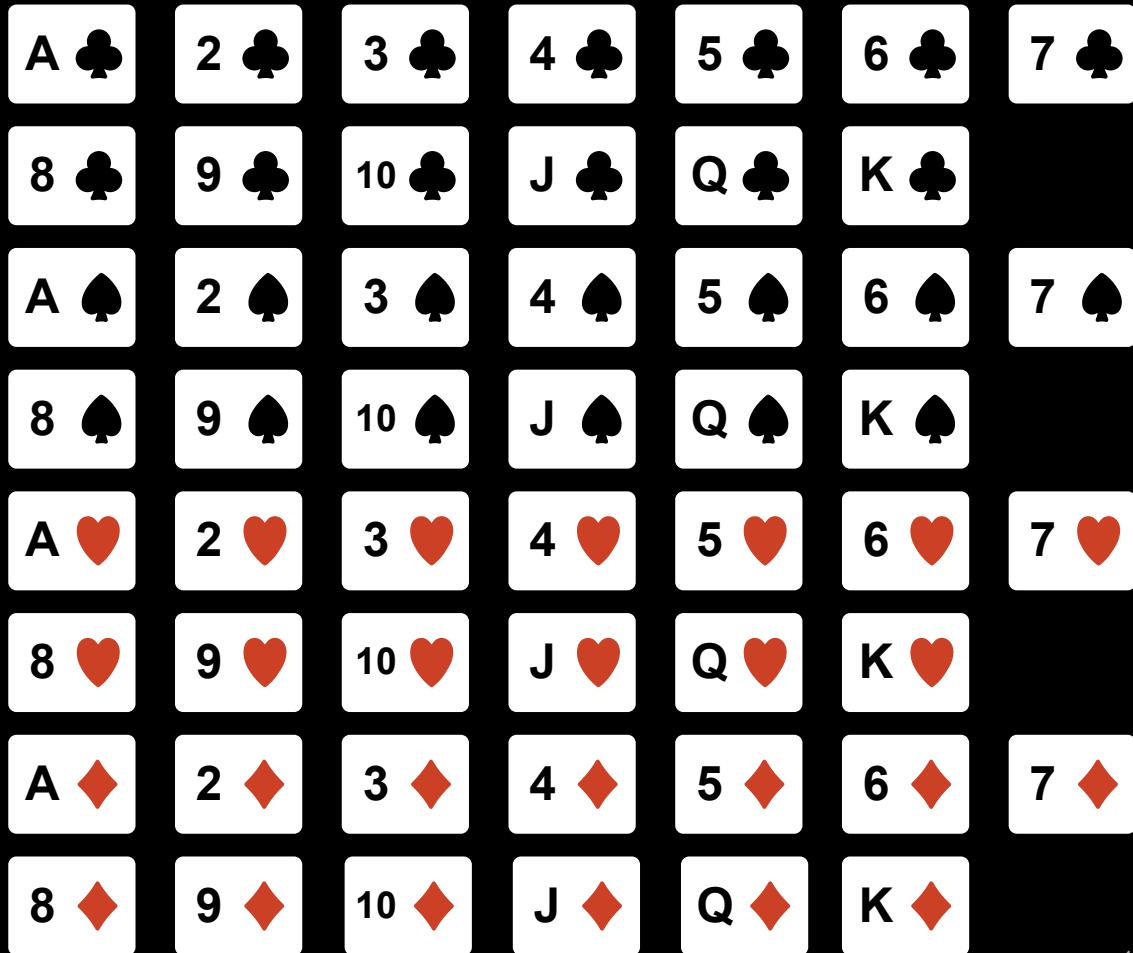
how many yes/no questions to reduce uncertainty to zero?

$$H = 0 = \log_2(1)$$

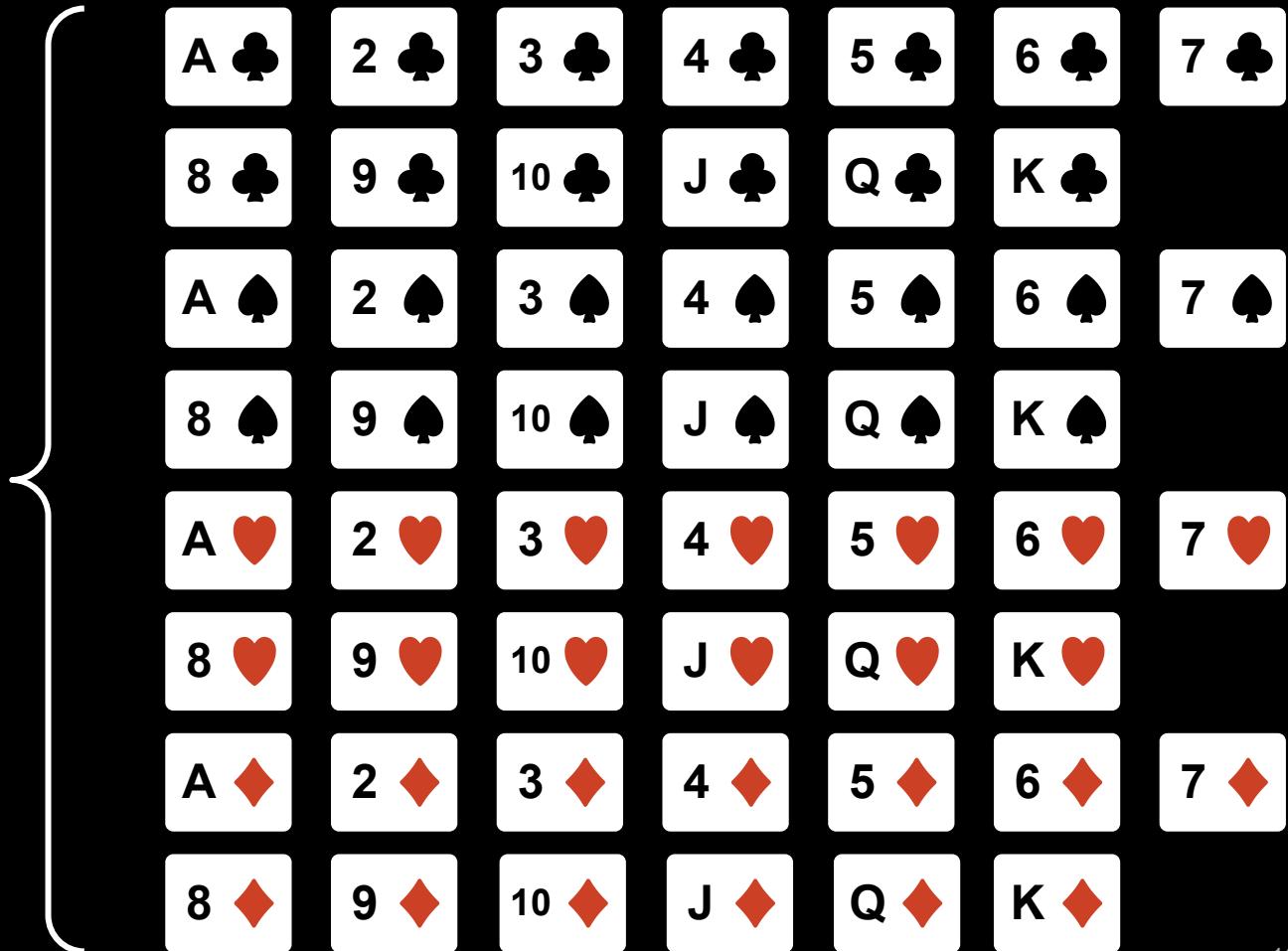
which card am I holding?



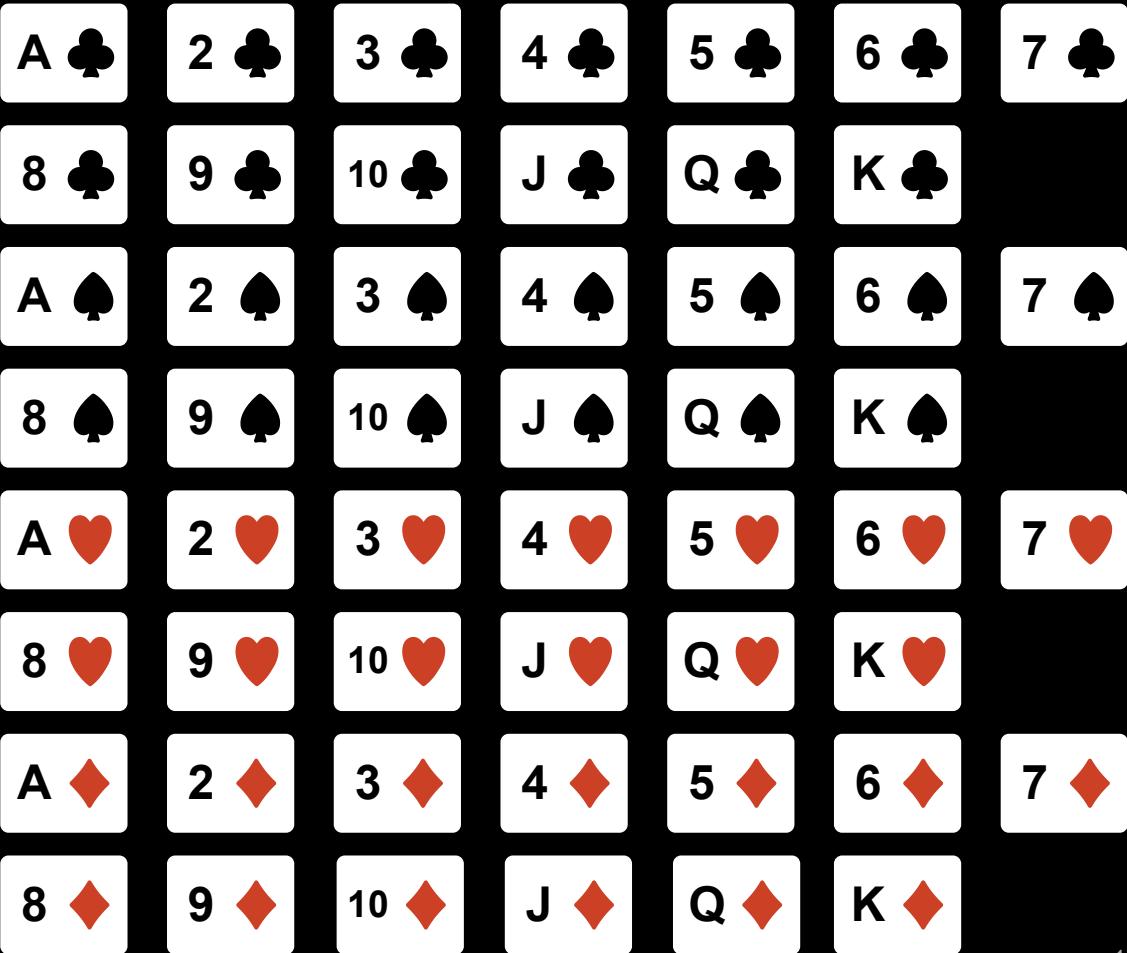
52 card poker deck



52 possible cards

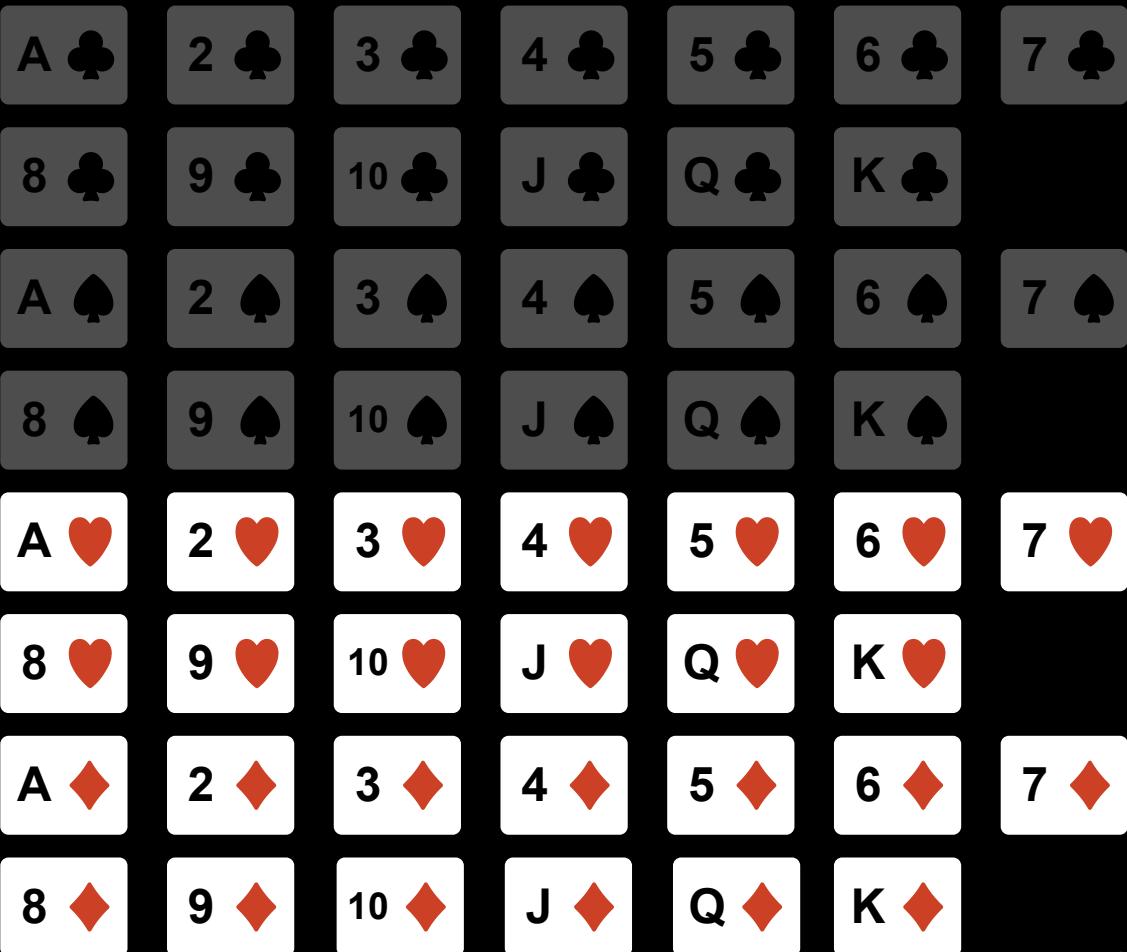


is the card black?



is the card black?

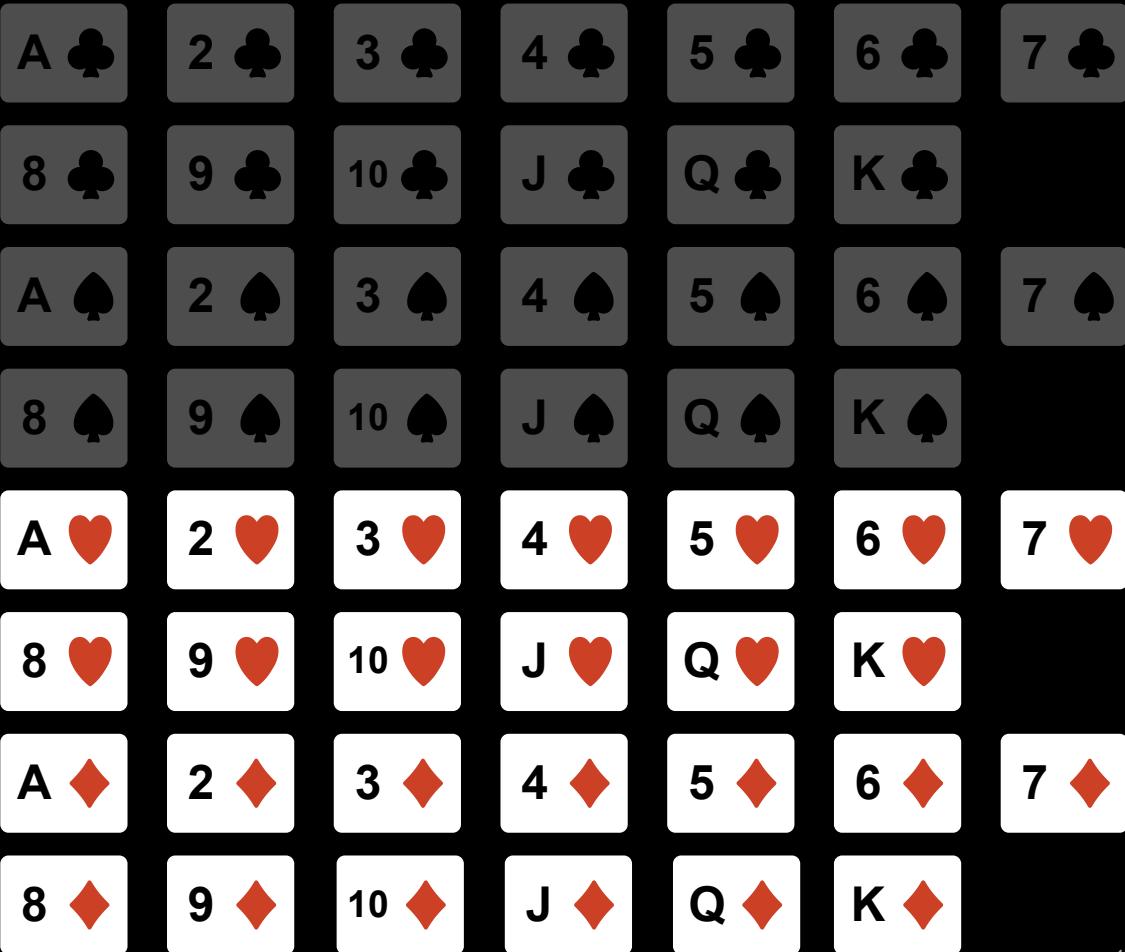
no



is the card black?

no

is it hearts?

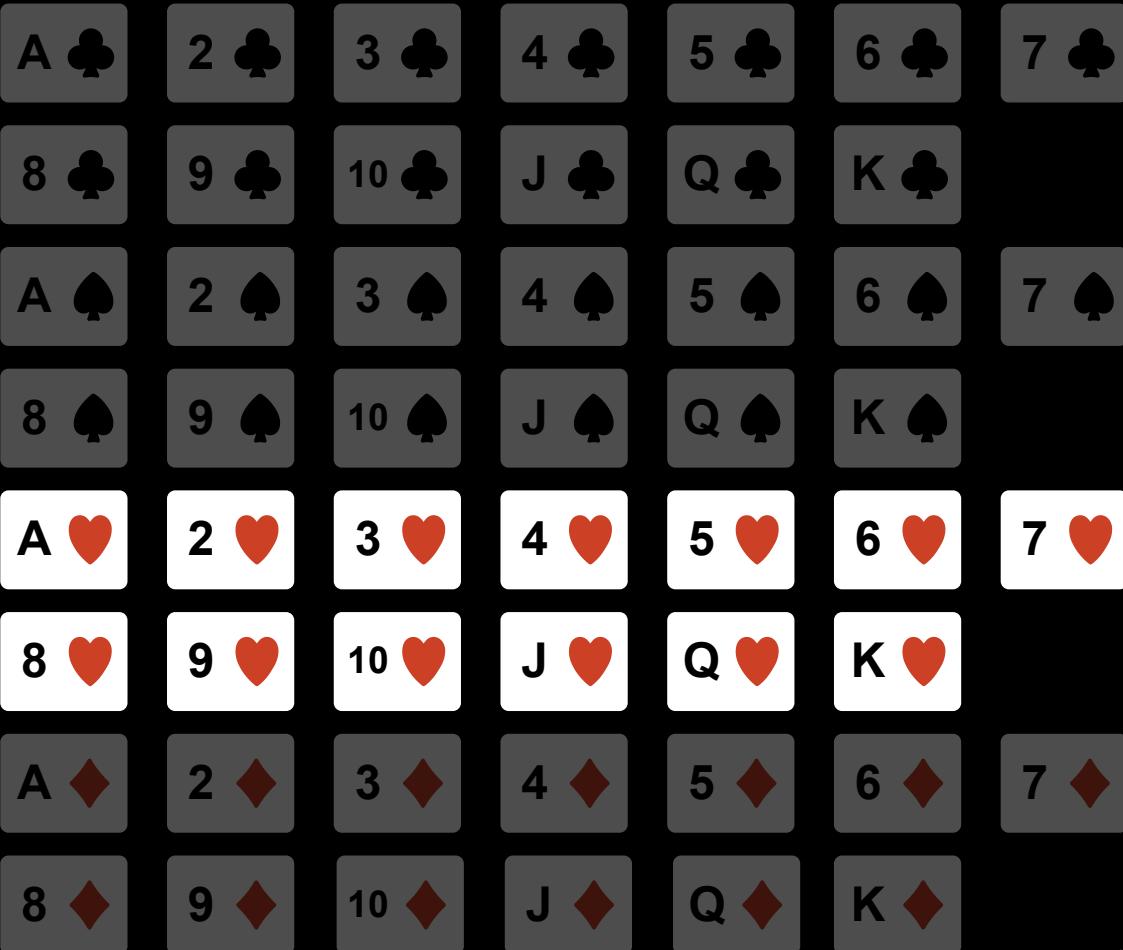


is the card black?

no

is it hearts?

yes



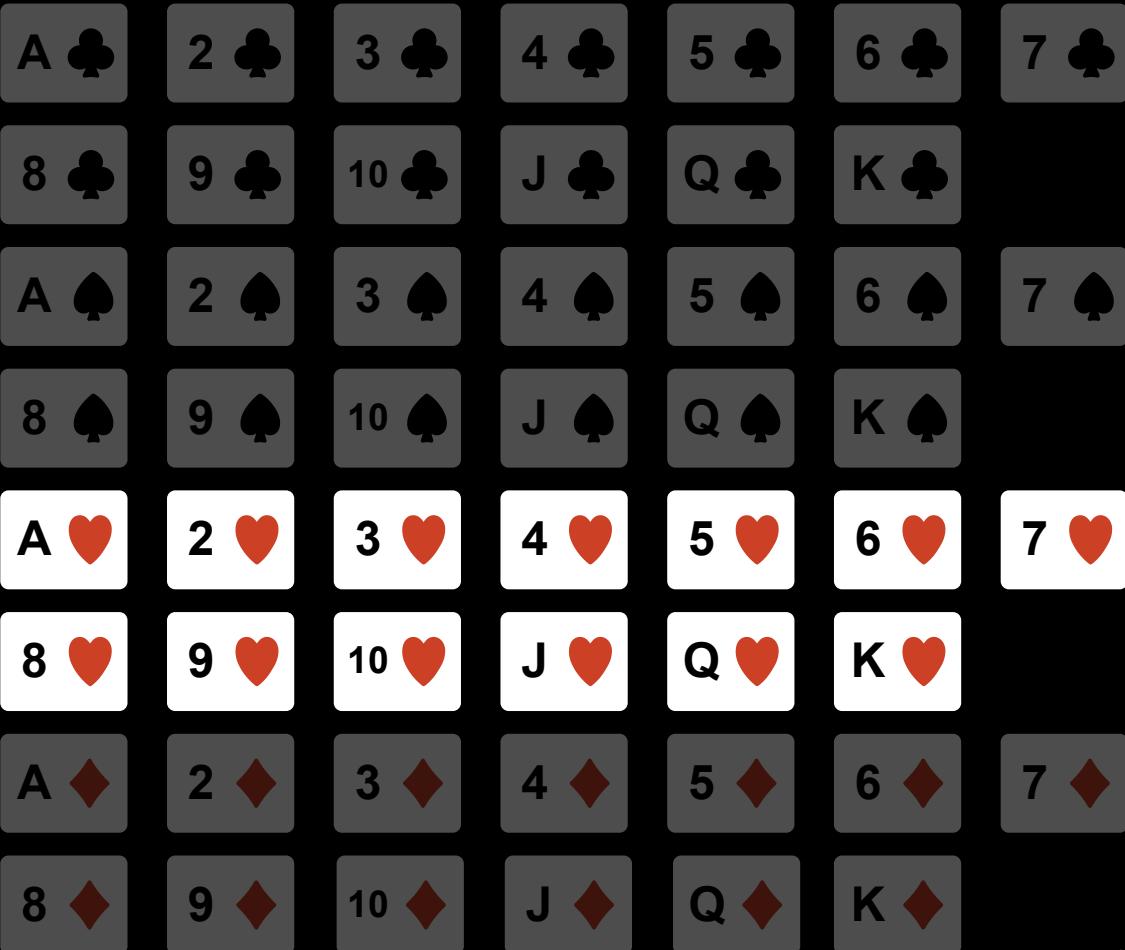
is the card black?

no

is it hearts?

yes

is it 8 or above?



is the card black?

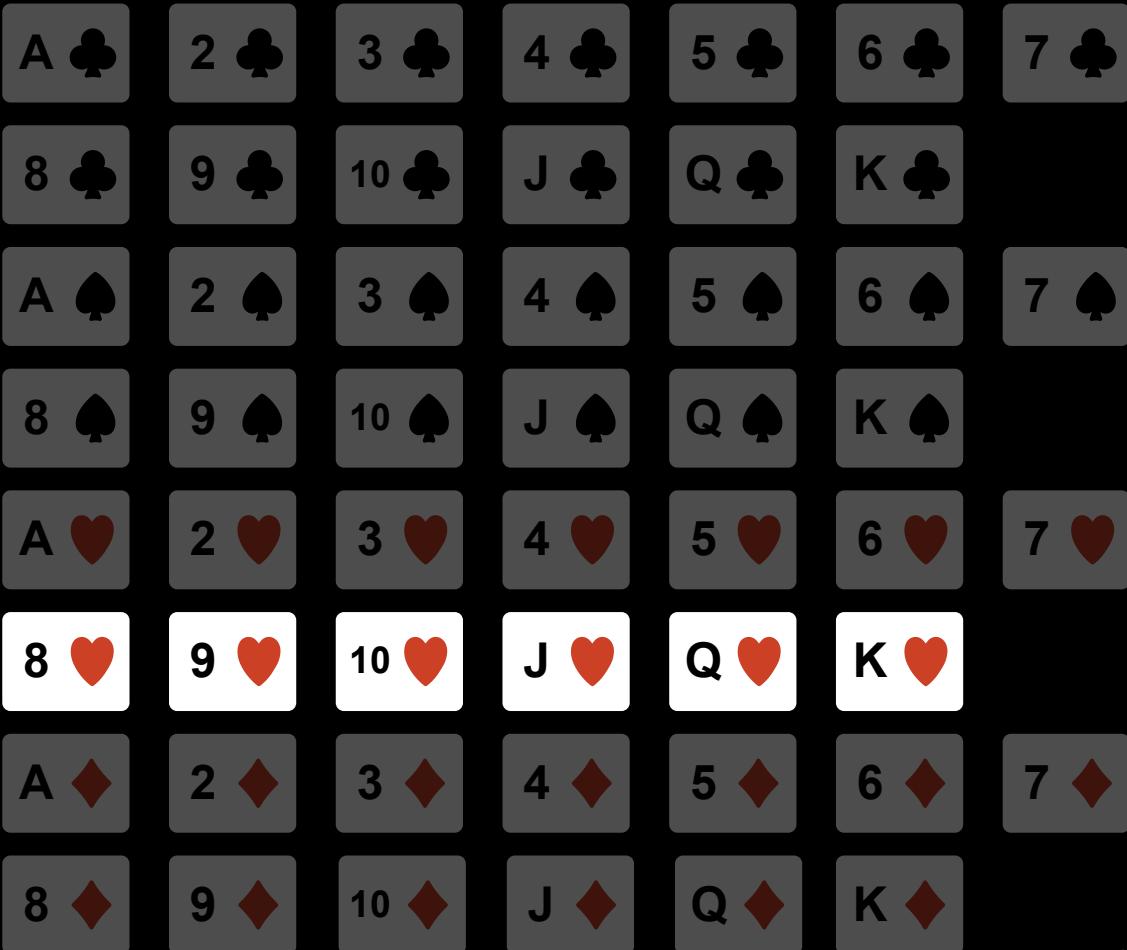
no

is it hearts?

yes

is it 8 or above?

yes



is the card black?

no

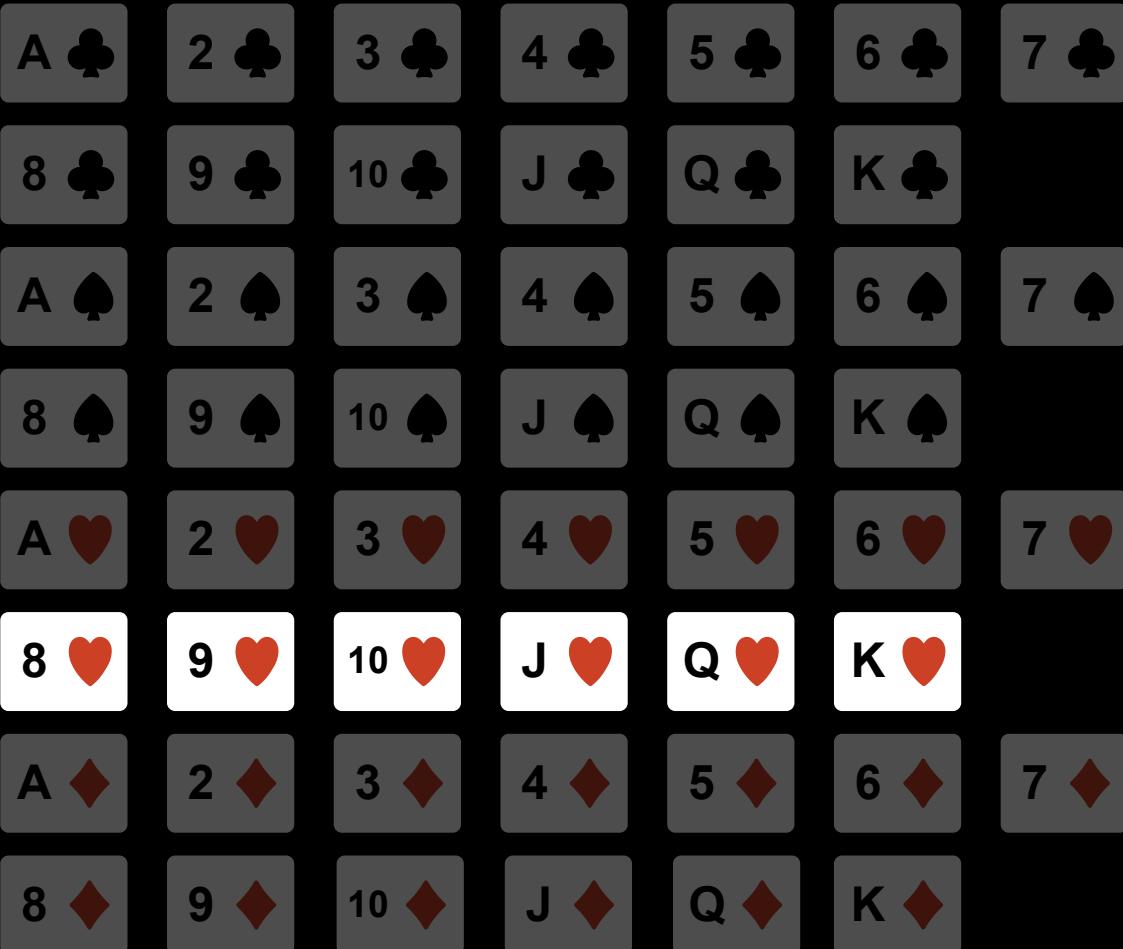
is it hearts?

yes

is it 8 or above?

yes

is it jack or above?



is the card black?

no

is it hearts?

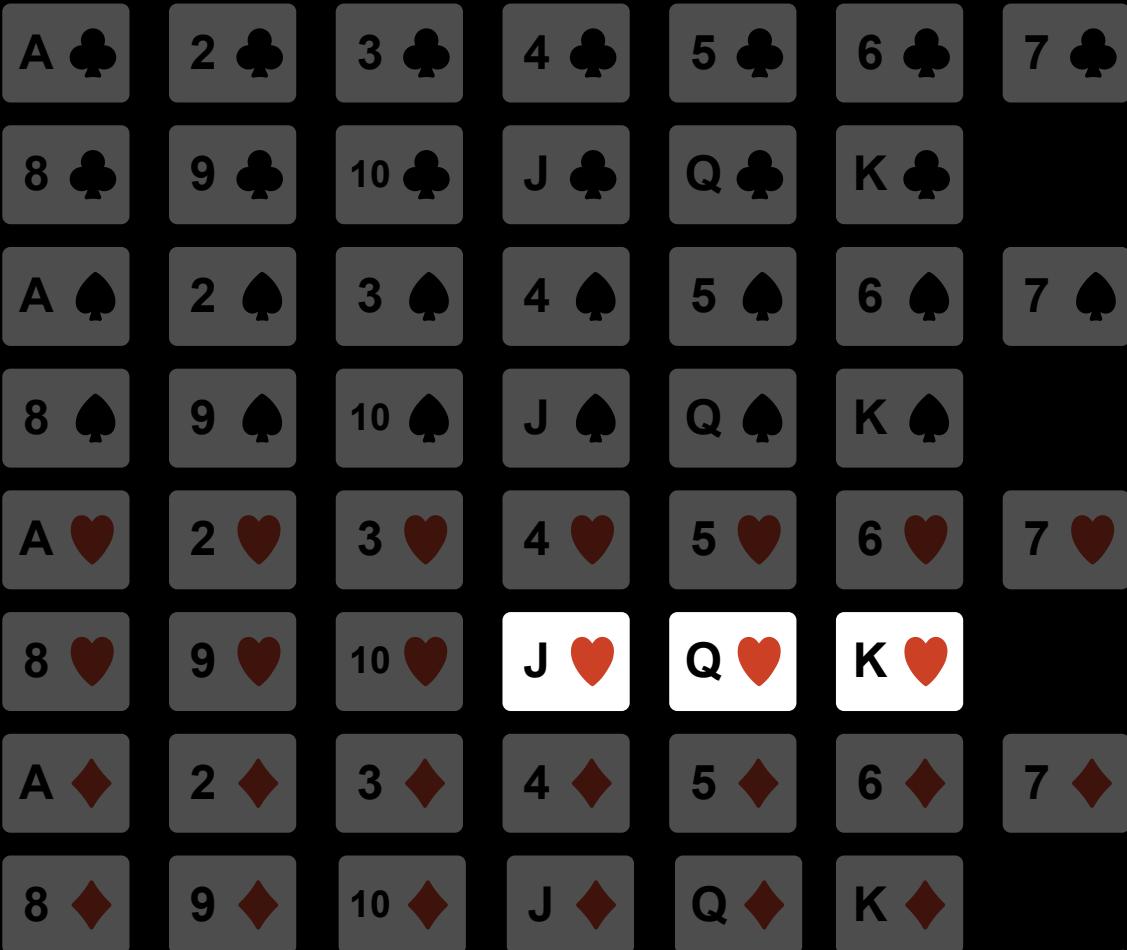
yes

is it 8 or above?

yes

is it jack or above?

yes



is the card black?

no

is it hearts?

yes

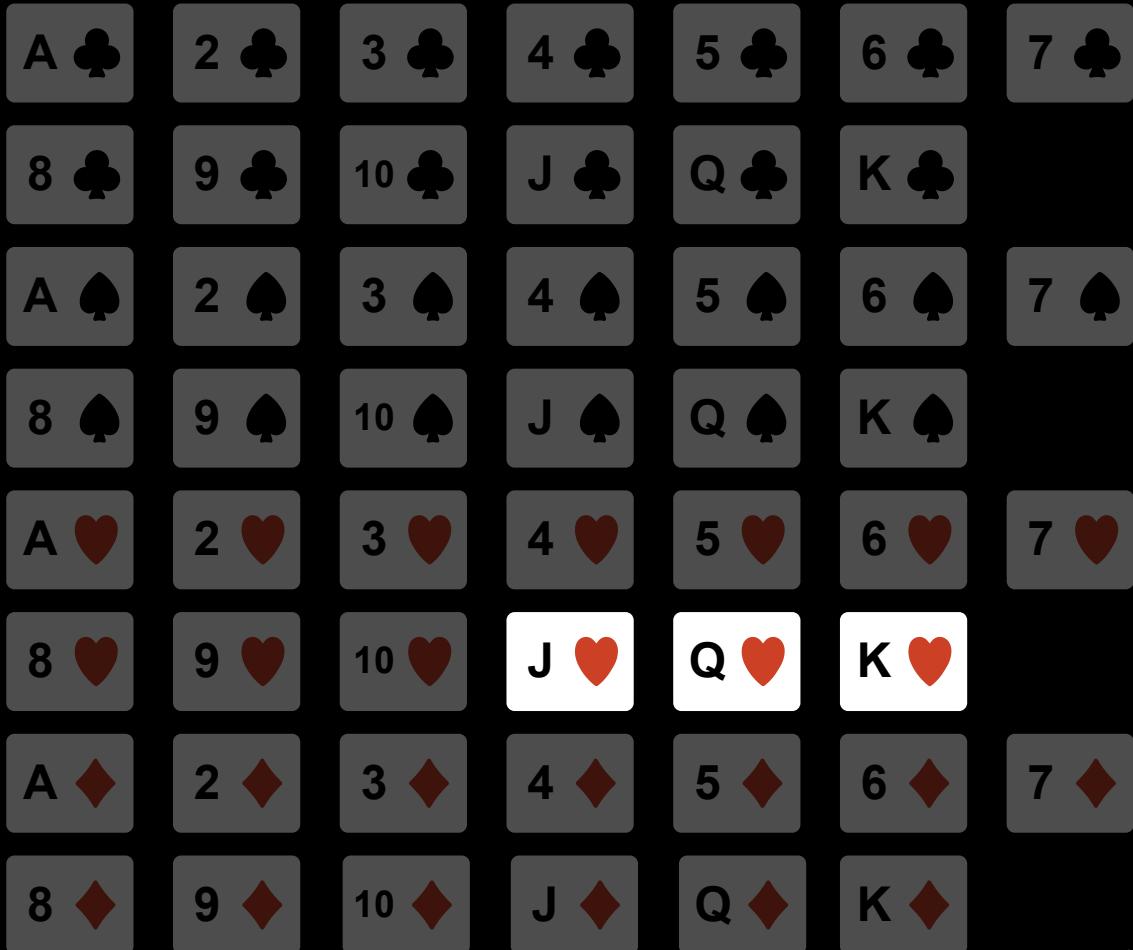
is it 8 or above?

yes

is it jack or above?

yes

is it queen or above?



is the card black?

no

is it hearts?

yes

is it 8 or above?

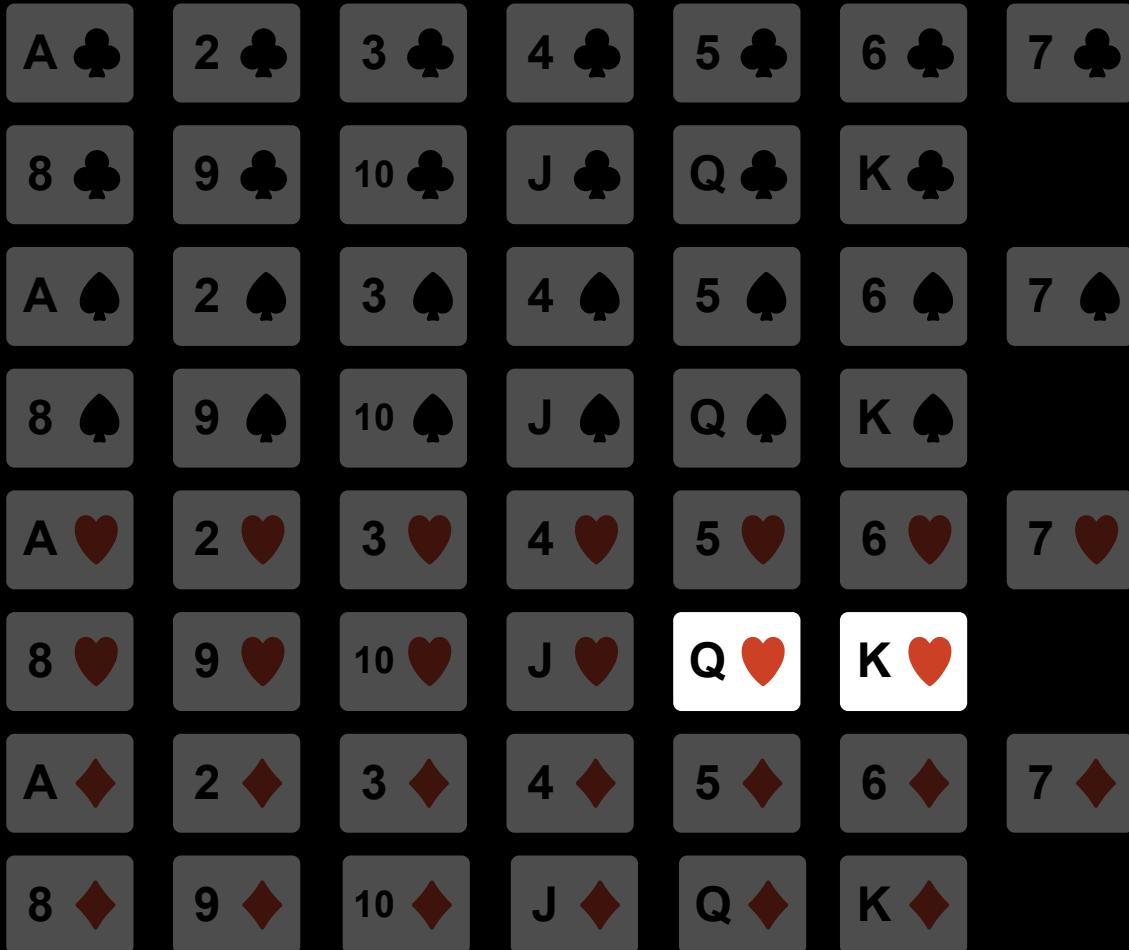
yes

is it jack or above?

yes

is it queen or above?

yes



is the card black?

no

is it hearts?

yes

is it 8 or above?

yes

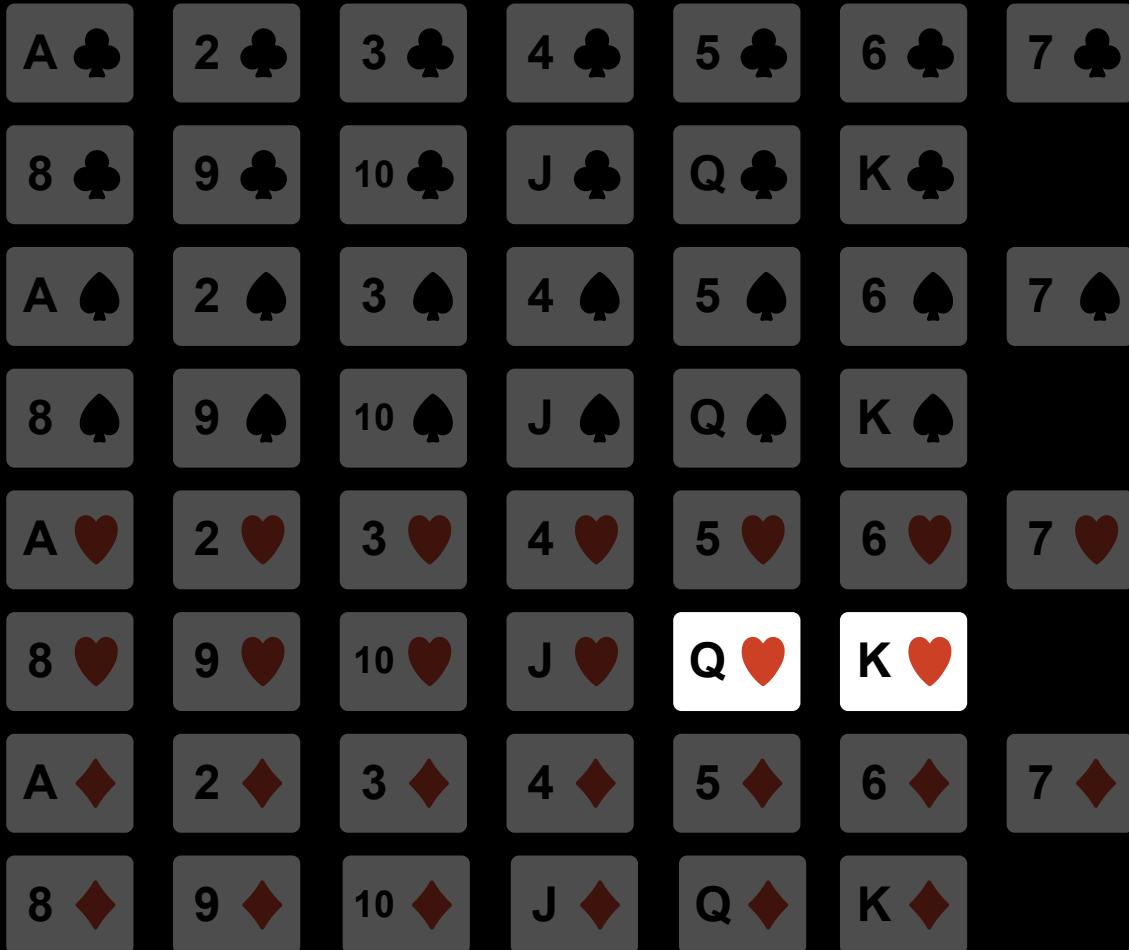
is it jack or above?

yes

is it queen or above?

yes

is it king?



is the card black?

no

is it hearts?

yes

is it 8 or above?

yes

is it jack or above?

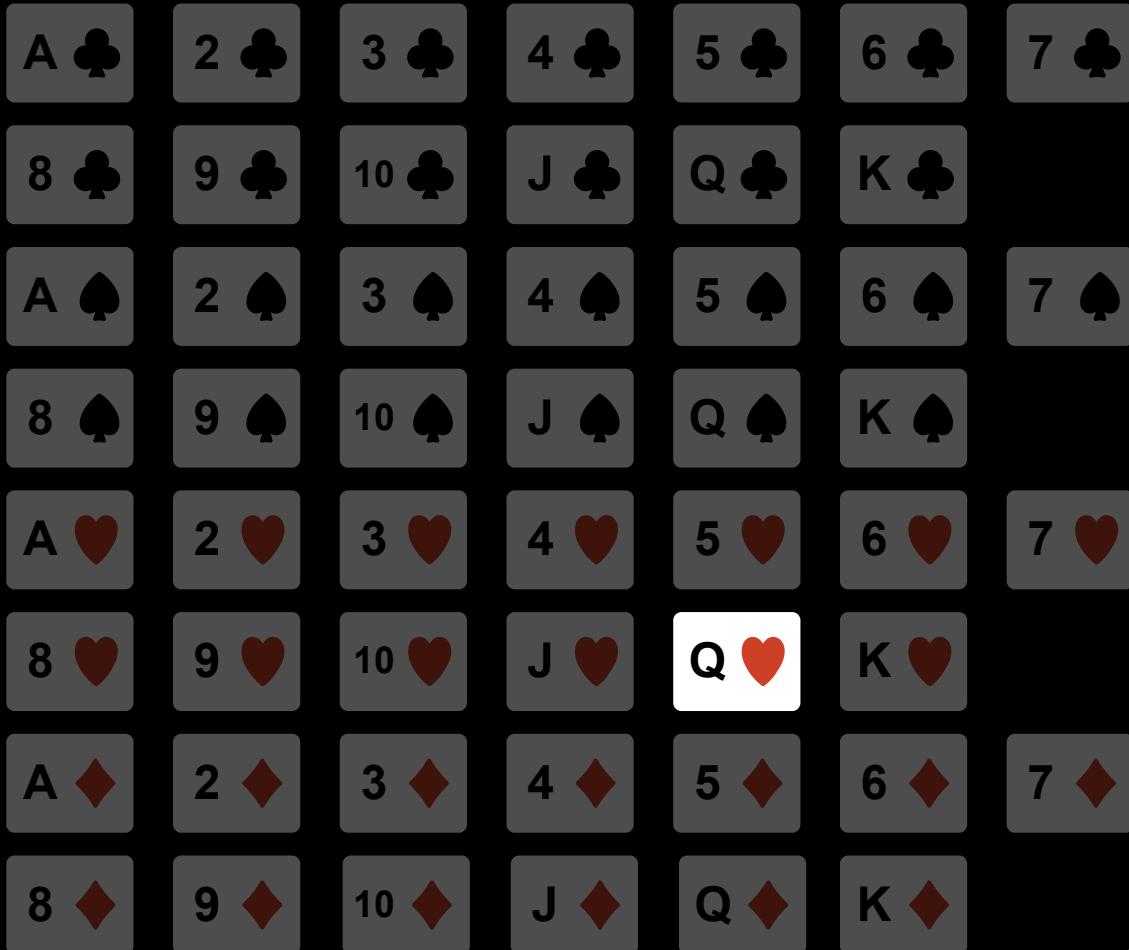
yes

is it queen or above?

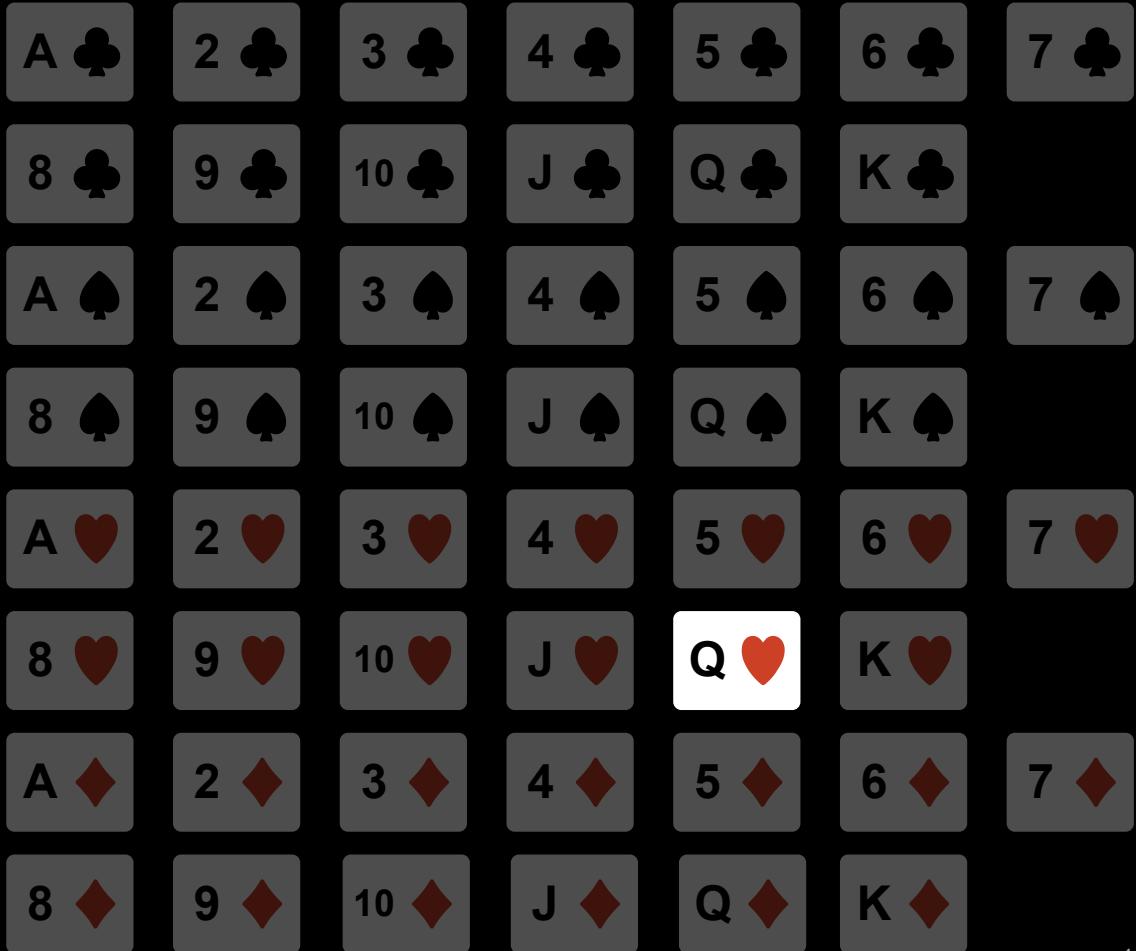
yes

is it king?

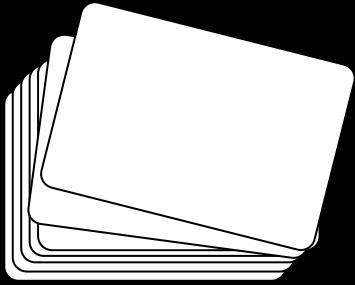
no



with 6 questions  
from 52 to 1

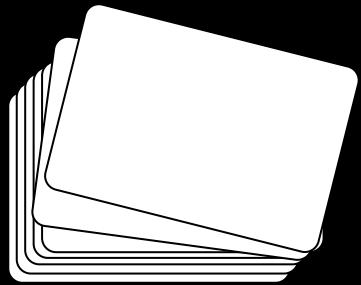


uncertainty with  $N = 52$  possibilities?

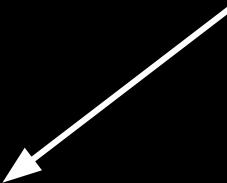


$$H = \log_2(52) \approx 5.7$$

uncertainty with  $N = 52$  possibilities?



average # of yes/no  
questions

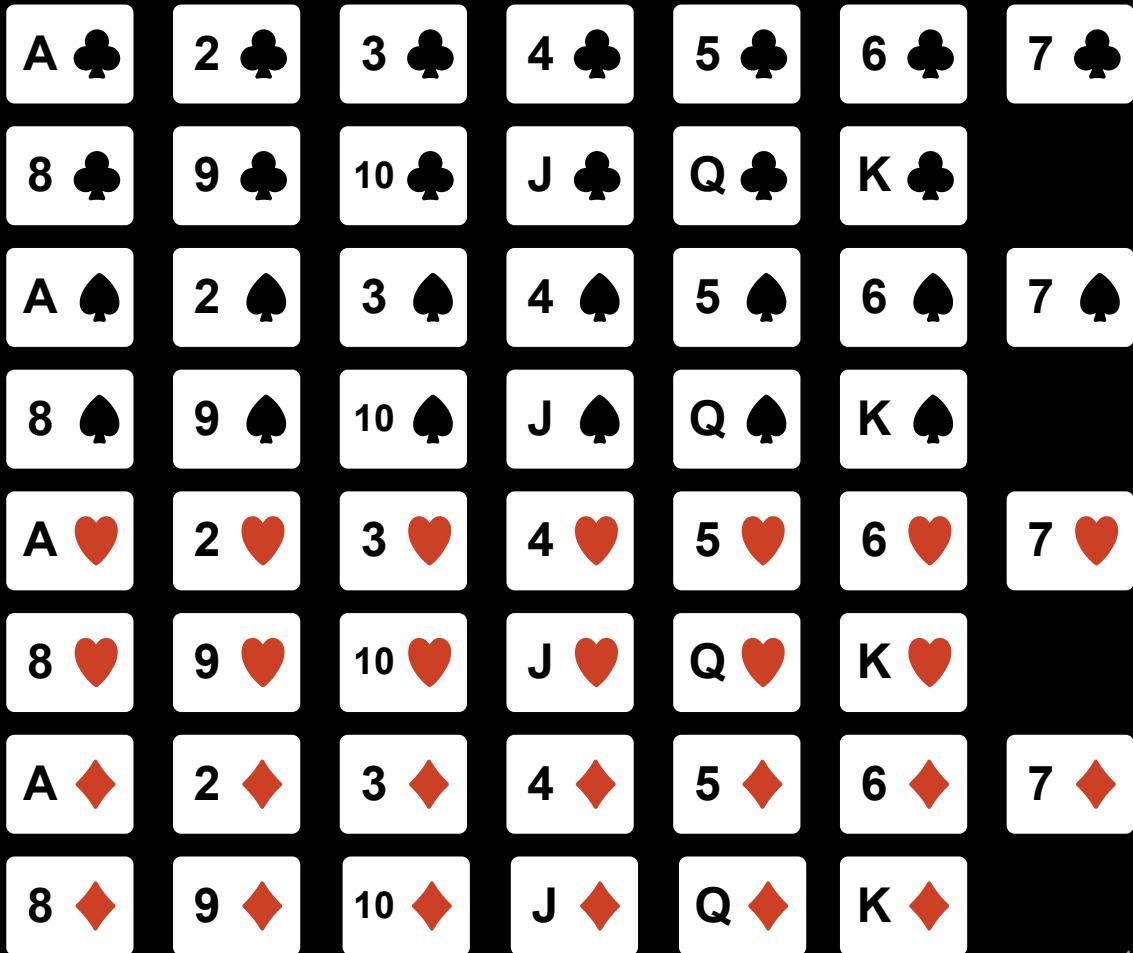


$$H = \log_2(52) \approx 5.7$$

one bit of information with each question

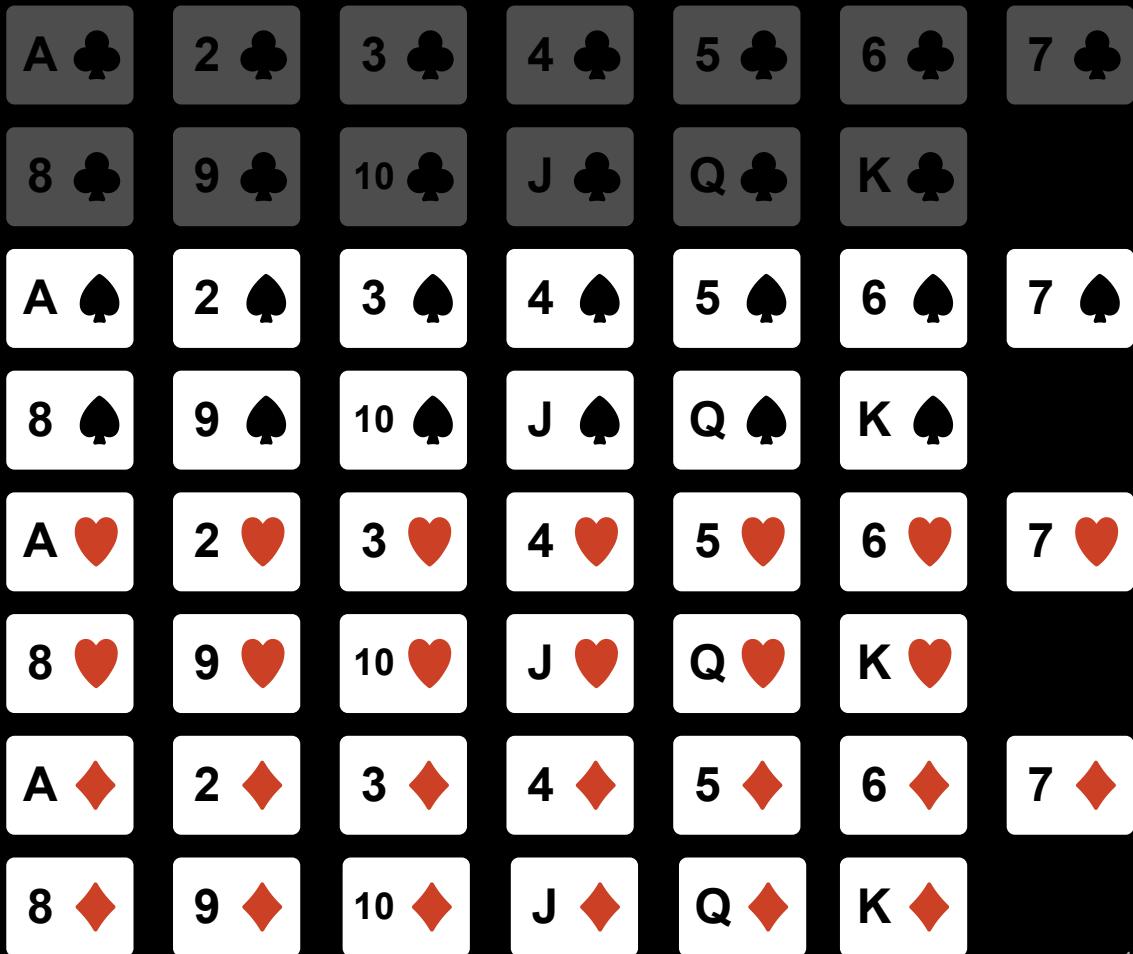
$$\log_2(52) - \log_2(26) = 1$$

is it a spades card?



is it a spades card?

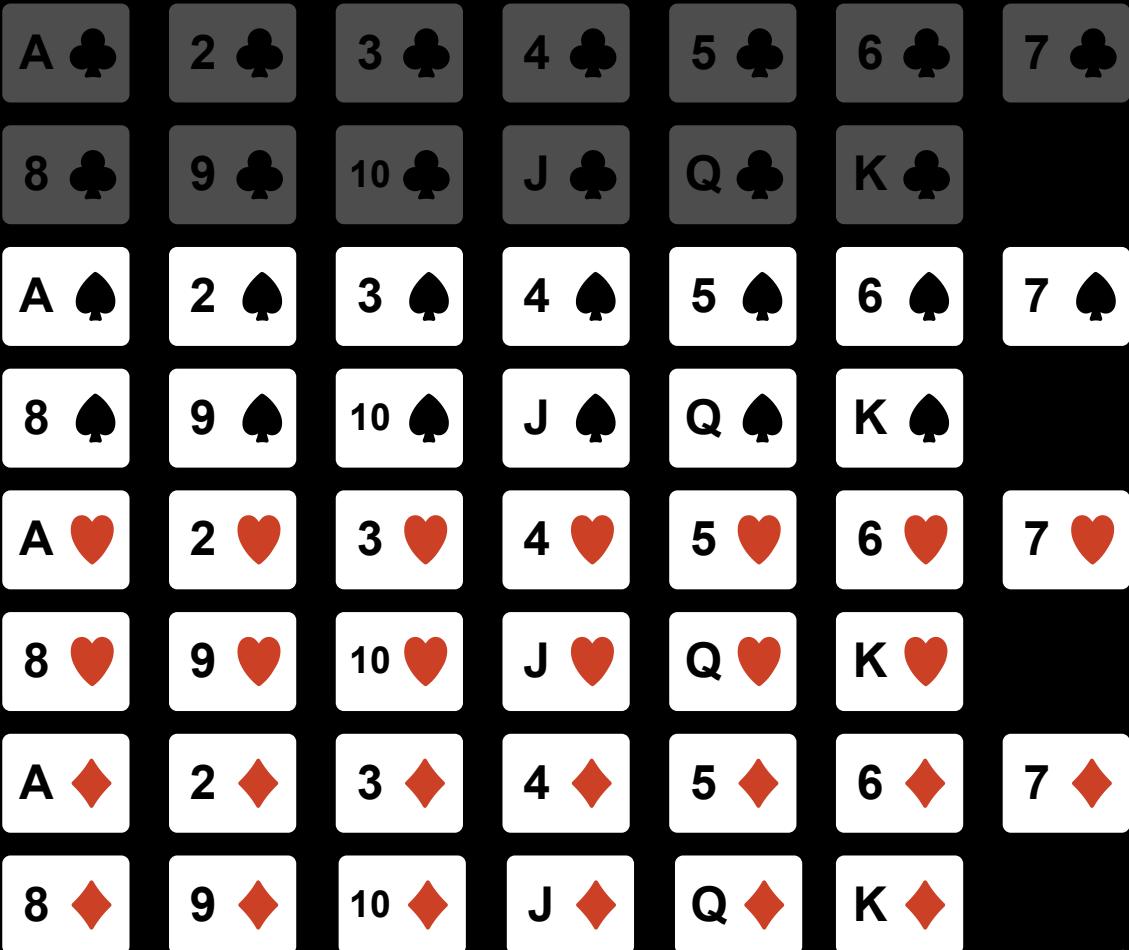
no



is it a spades card?

no

how much information?



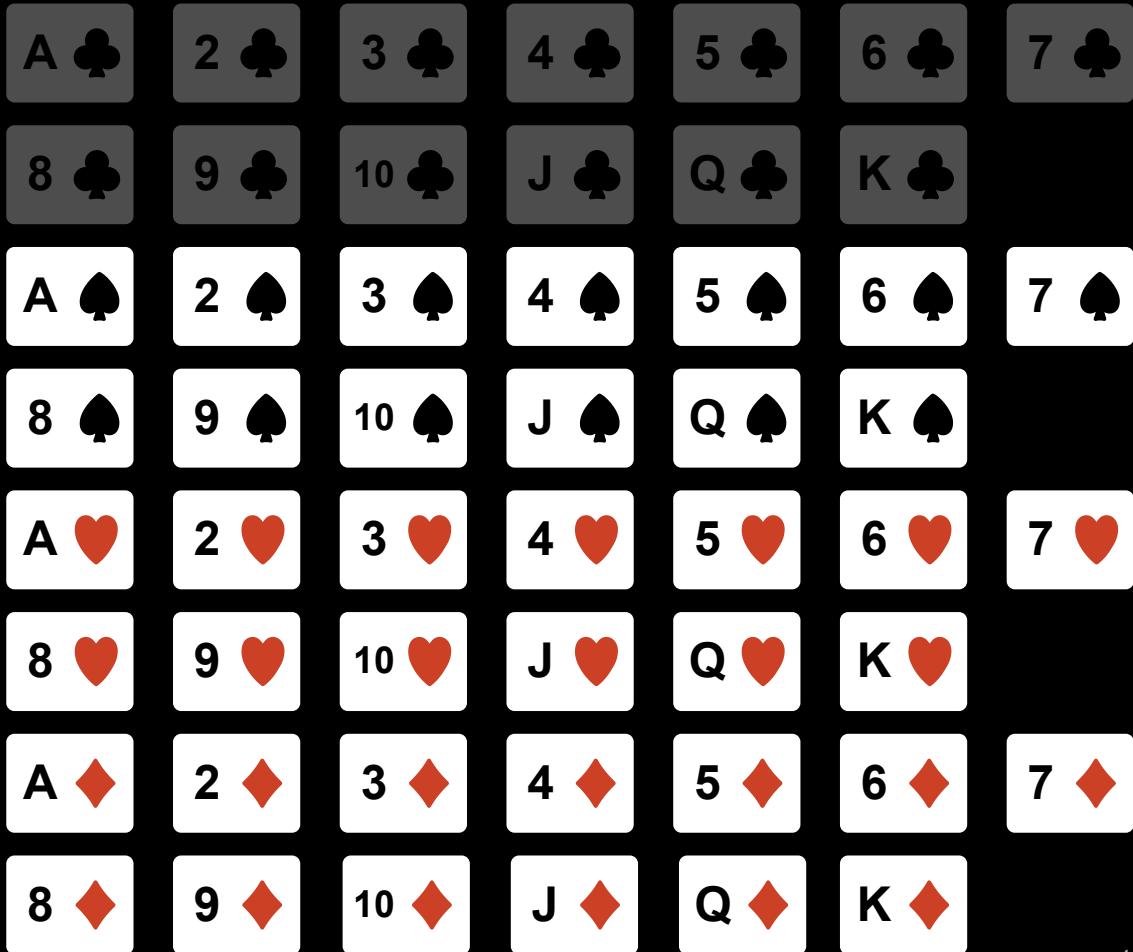
is it a spades card?

no

how much information?

$$H_0 = \log_2(52) \approx 5.7$$

$$H_1 = \log_2(39) \approx 5.29$$



is it a spades card?

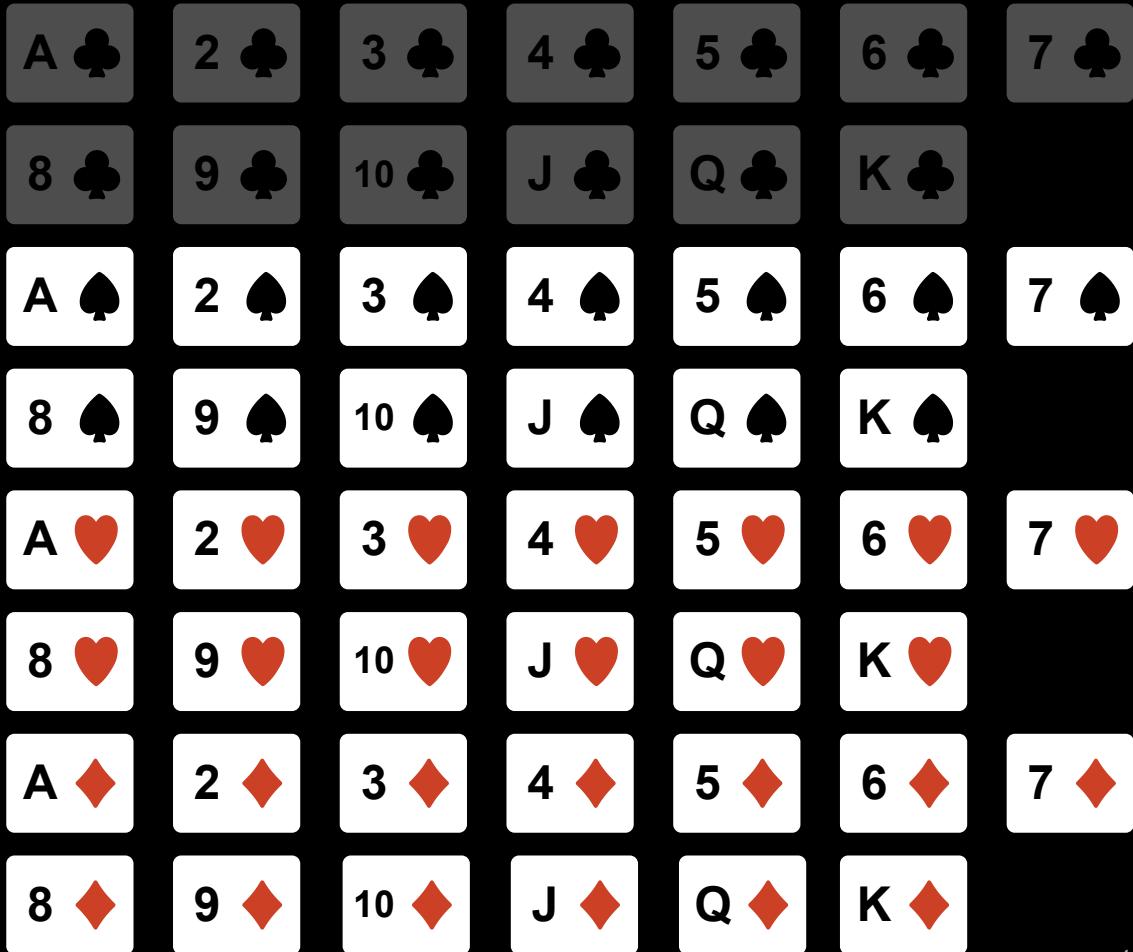
no

how much information?

$$H_0 = \log_2(52) \approx 5.7$$

$$H_1 = \log_2(39) \approx 5.29$$

$$H_0 - H_1 \approx 0.41$$



is it a spades card?

no

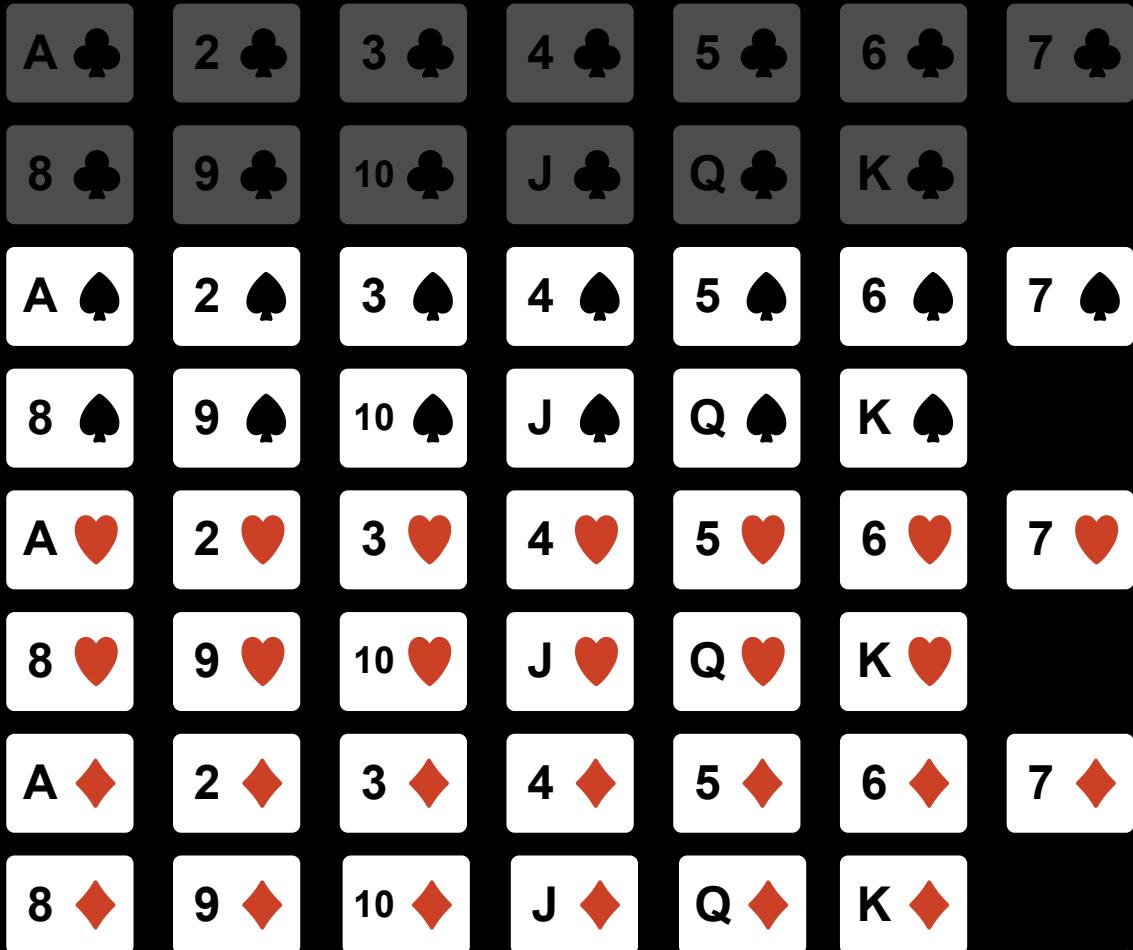
how much information?

$$H_0 = \log_2(52) \approx 5.7$$

$$H_1 = \log_2(39) \approx 5.29$$

$$H_0 - H_1 \approx 0.41$$

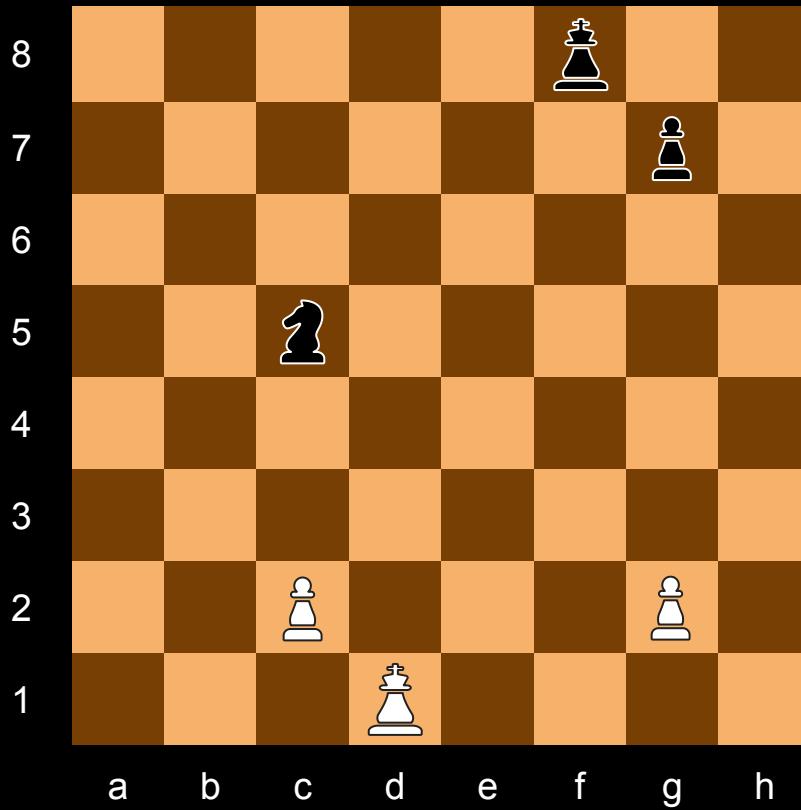
that's less than 1 bit

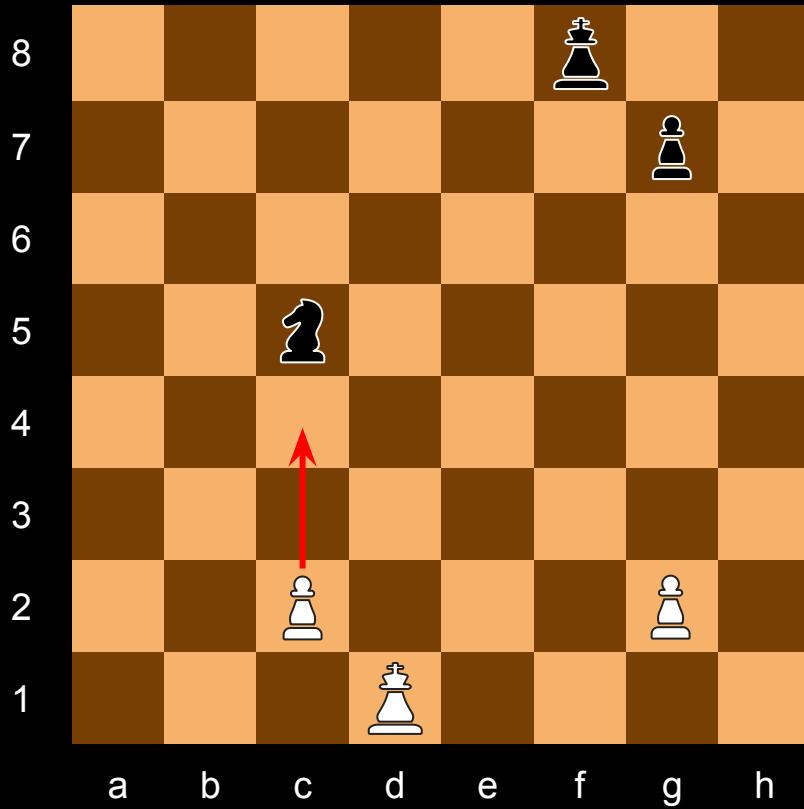


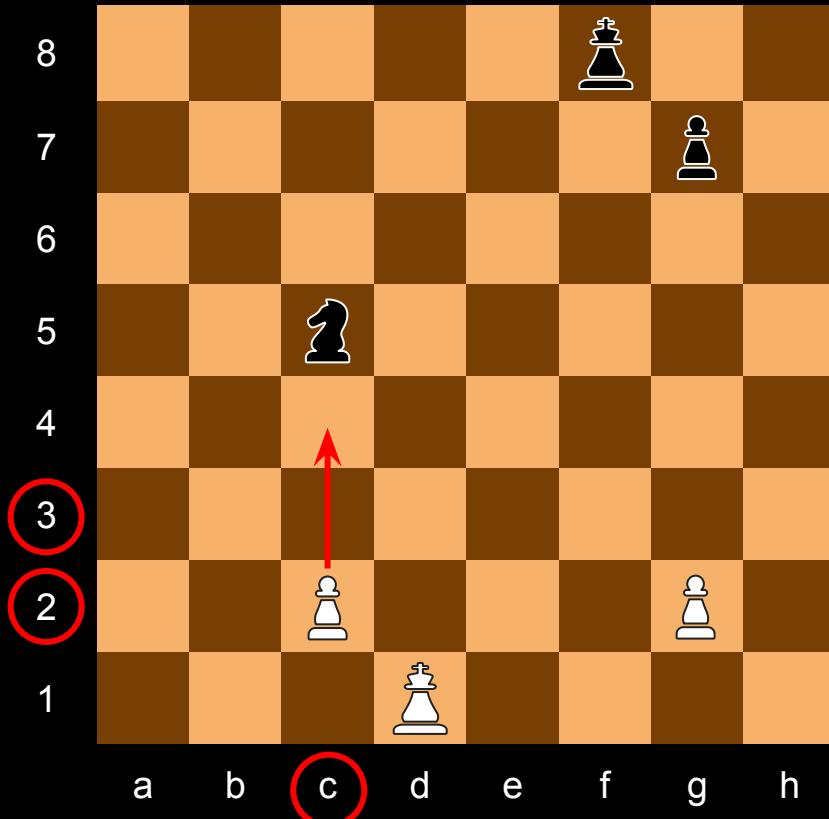
chess



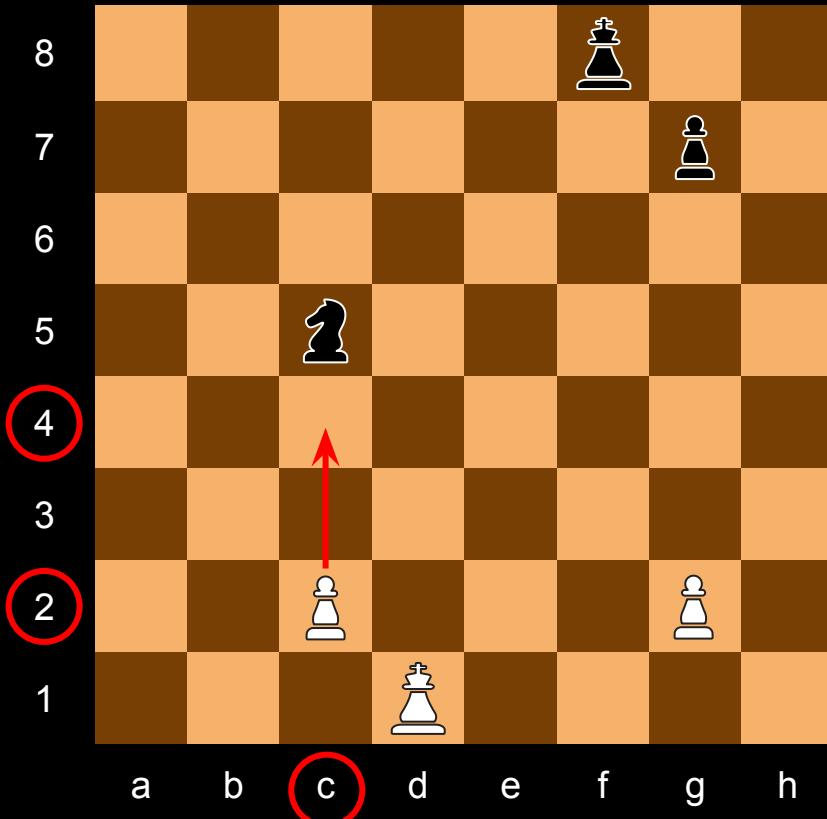
Image created with DALLE-3



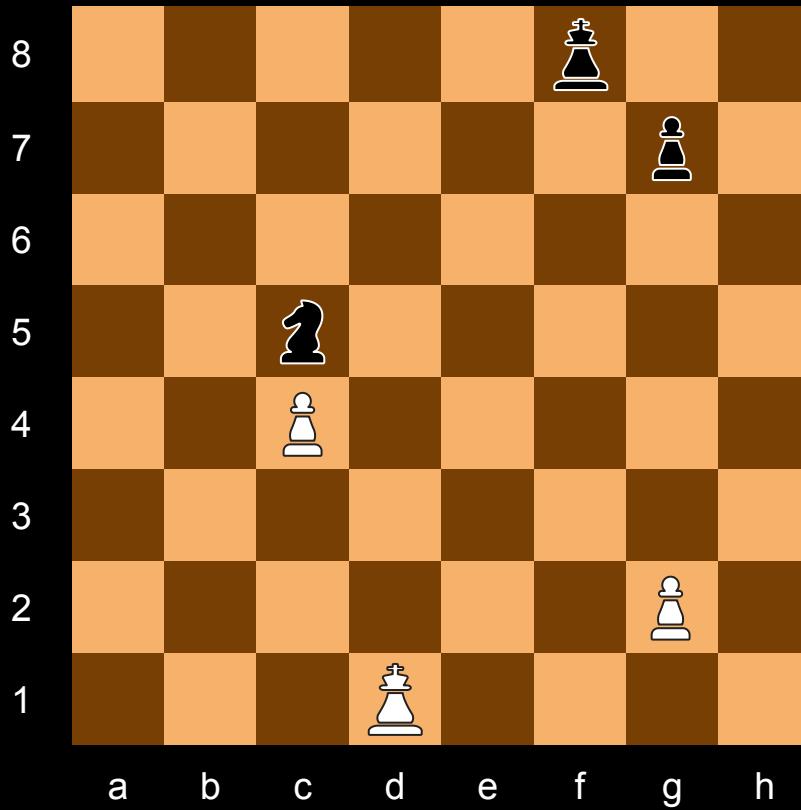




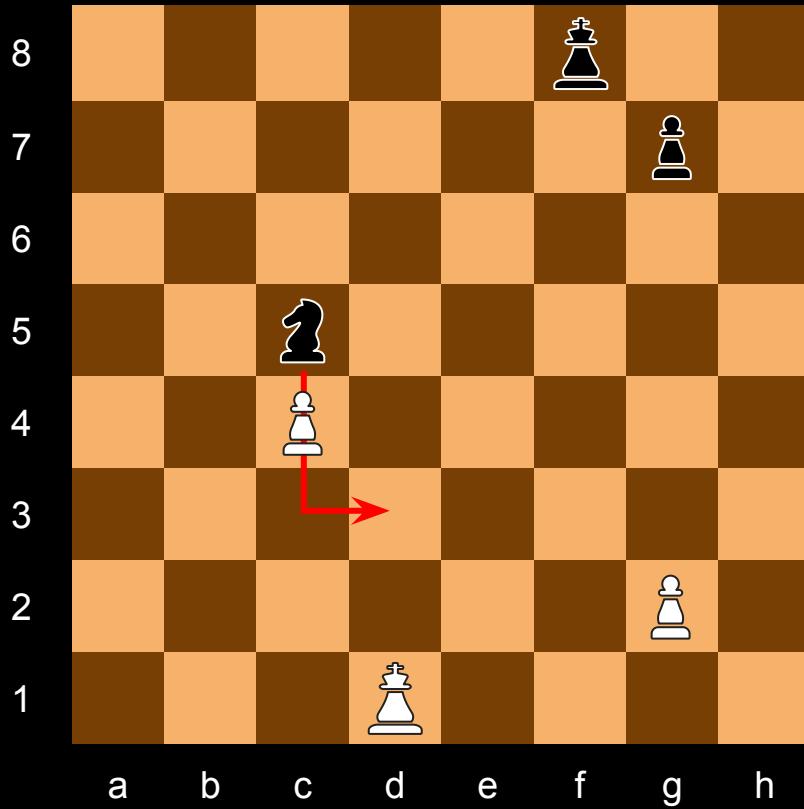
c2 → c4



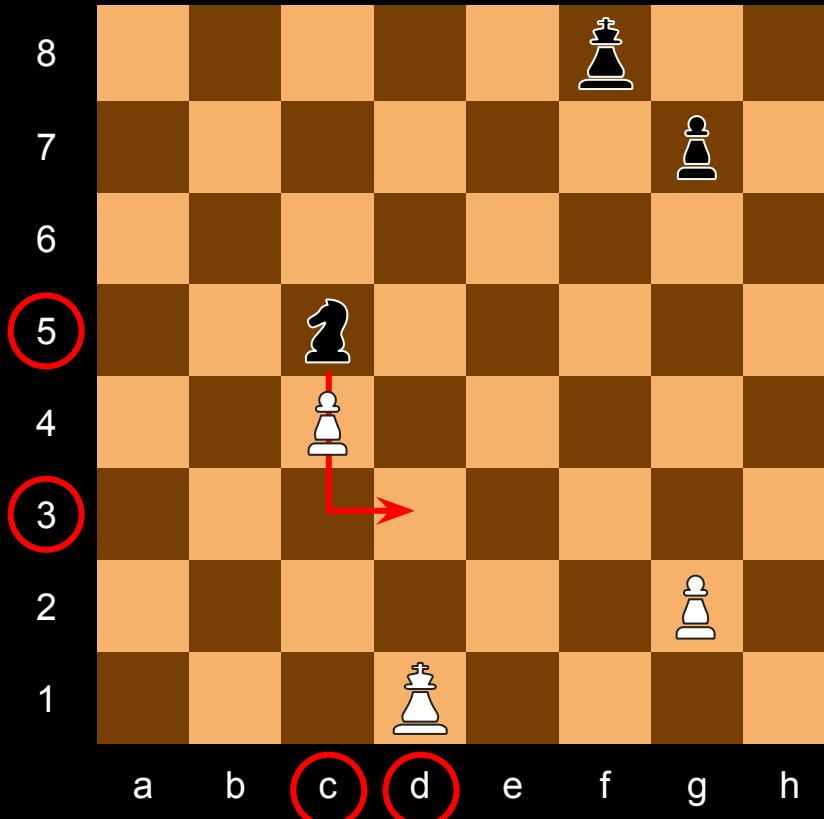
c2 → c4



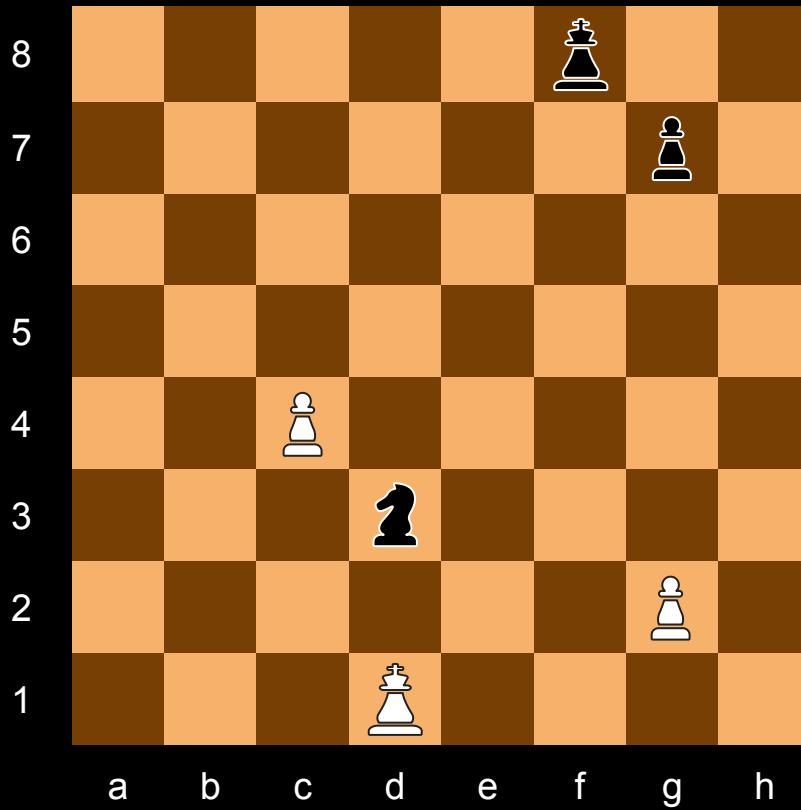
c2 → c4



$c2 \rightarrow c4$   
 $c5 \rightarrow d3$



$c2 \rightarrow c4$   
 $c5 \rightarrow d3$   
...



{A}

— —

{A}

A A

{A, B}

— —

{A, B}

A A

{A, B}

A B

{A, B}

B A

{A, B}

B B

{A, B}

AA, AB, BA, BB

{A, B, C}

— —

{A, B, C}

AA, AB, BA, BB,  
AC, BC, CA, CB, CC

{A, B, C, D}

— —

{A, B, C, D}

AA, AB, BA, BB, AC, BC, CA, CB,  
CC, AD, DA, BD, DB, CD, DC, DD

{A, B, C, D, E}

— —

{A, B, C, D, E}

AA, AB, BA, BB, AC, BC, CA, CB, CC,  
AD, DA, BD, DB, CD, DC, DD, AE, EA,  
BE, EB, CE, EC, DE, ED, EE

with length  $n = 2$

# symbols	# messages
1	1
2	4
3	9
4	16
5	25

with length  $n = 2$

# symbols	$f(x)$	# messages
1		1
2	$f(x)$	4
3	→	9
4		16
5		25

{A, B}

— — —

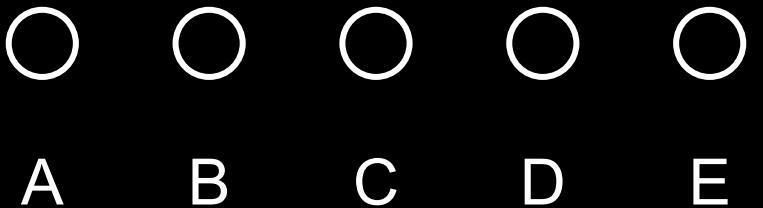
# mastermind



eight colors



five slots



# possible permutations

$$N = 5^8 = 390625$$

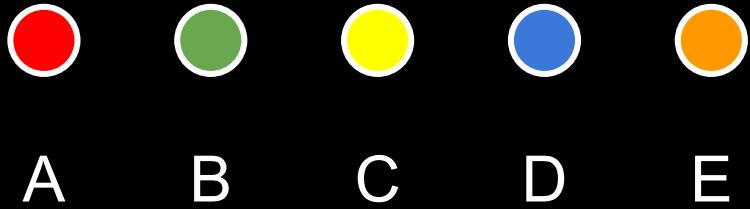
# maximum entropy in bits

$$H_{max} = \log_2 5^8 = 18.575$$

hints reduce uncertainty (entropy)



first guess



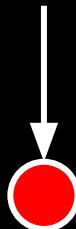
opponent's hints



first guess

opponent's hints

correct



A

B

C

D

E

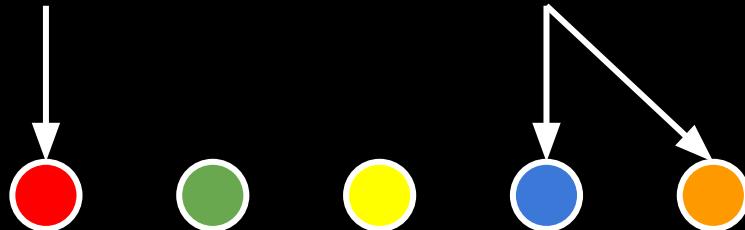


first guess

opponent's hints

correct

wrong position



A

B

C

D

E



how much information did we get?



A



B



C



D



E



colors left



slots left



A

B

C

D

E

# new possible permutations?

$$N = 4^8 = 65536$$

# reduced entropy in bits?

$$H = \log_2 4^8 = 16$$

# COUNTING

1

2

3

1

2

3

---

$10^2$

$10^1$

$10^0$

**1**

**2**

**3**

---

**$10^2$**

**$10^1$**

**$10^0$**

$$= 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0$$

$$= 1 \times 100 + 2 \times 10 + 3 \times 1$$

$$= 123$$

4

1

2

3

---

?

$10^2$

$10^1$

$10^0$

$$\begin{array}{r} 4 & 1 & 2 & 3 \\ \hline ? & 10^2 & 10^1 & 10^0 \end{array}$$

$$= 4 \times 10^3 + 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0$$

$$\begin{array}{r}
 4 \quad 1 \quad 2 \quad 3 \\
 \hline
 ? \quad 10^2 \quad 10^1 \quad 10^0
 \end{array}$$

$$= 4 \times 10^3 + 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0$$

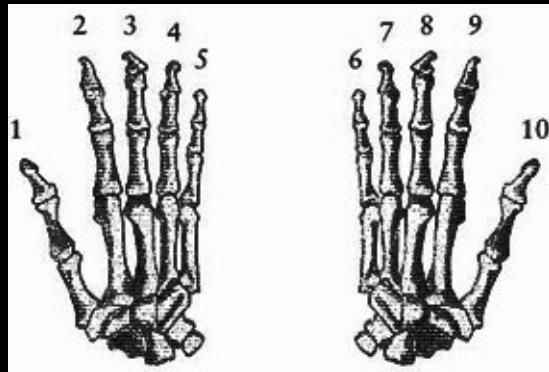
$$= 4 \times 1000 + 1 \times 100 + 2 \times 10 + 3 \times 1$$



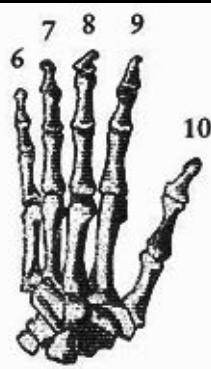
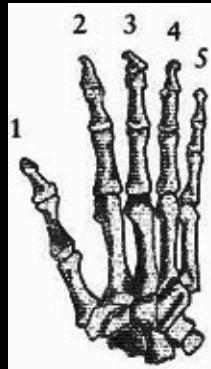
$$= 4 \times 10^3 + 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0$$

$$= 4 \times 1000 + 1 \times 100 + 2 \times 10 + 3 \times 1$$

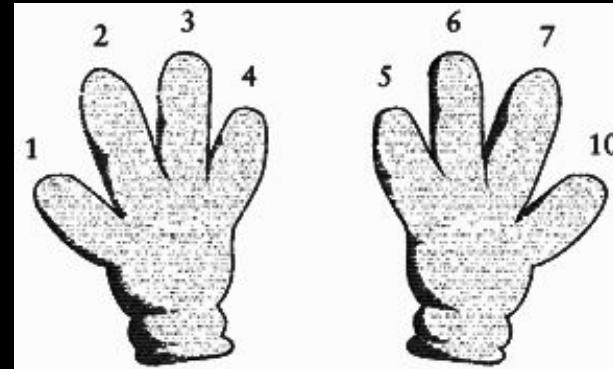
$$= 4123$$



human hand



human hand



cartoon character's hand

1

2

3

(octal)

**1**

**2**

**3**

(octal)

---

**8<sup>2</sup>**

**8<sup>1</sup>**

**8<sup>0</sup>**

**1**

**2**

**3**

(octal)

---

**8<sup>2</sup>**

**8<sup>1</sup>**

**8<sup>0</sup>**

$$= 1 \times 8^2 + 2 \times 8^1 + 3 \times 8^0$$

**1**

**2**

**3**

(octal)

---

**$8^2$**

**$8^1$**

**$8^0$**

$$= 1 \times 8^2 + 2 \times 8^1 + 3 \times 8^0$$

$$= 1 \times 64 + 2 \times 8 + 3 \times 1$$

**1**

**2**

**3**

(octal)

---

**$8^2$**

**$8^1$**

**$8^0$**

$$= 1 \times 8^2 + 2 \times 8^1 + 3 \times 8^0$$

$$= 1 \times 64 + 2 \times 8 + 3 \times 1$$

$$= 83 \text{ (decimal)}$$

decimal

8

octal

?

decimal

octal

?



7

decimal

16

octal

?



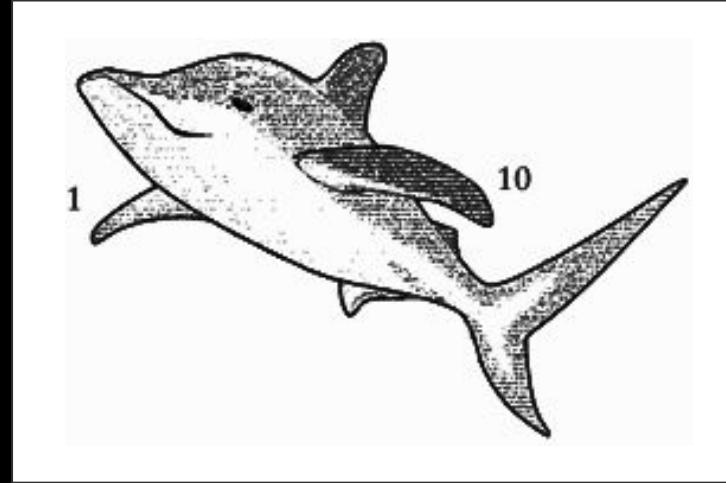
decimal

octal

?



100



what now?

0, 1, ...

0, 1, 10, ...

0, 1, 10, 11, ...

0, 1, 10, 11, 100, ...

0, 1, 10, 11, 100, 101, ...

0, 1, 10, 11, 100, 101, 110

1 1 0 (binary)

**1**

**1**

**0**

(binary)

---

**$2^2$**

**$2^1$**

**$2^0$**

$$\begin{array}{r} 1 & 1 & 0 \\ \hline 2^2 & 2^1 & 2^0 \end{array}$$

$$= 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$$

$$\begin{array}{r} 1 & 1 & 0 \\ \hline 2^2 & 2^1 & 2^0 \end{array}$$

$$= 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$$

$$= 1 \times 4 + 1 \times 2 + 0 \times 1$$

$$\begin{array}{r} 1 & 1 & 0 \\ \hline 2^2 & 2^1 & 2^0 \end{array}$$

$$= 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$$

$$= 1 \times 4 + 1 \times 2 + 0 \times 1$$

$$= 6 \text{ (decimal)}$$

2 3 4 5 6

0, 1, 10, 11, 100, 101, 110

# place value systems

$$N = d_n * R^{n-1} + \dots + d_1 * R^1 + d_0 *$$
$$R^0$$

$$d \in \{ 0, 1, \dots R-1 \}$$

$n$  = number of digits

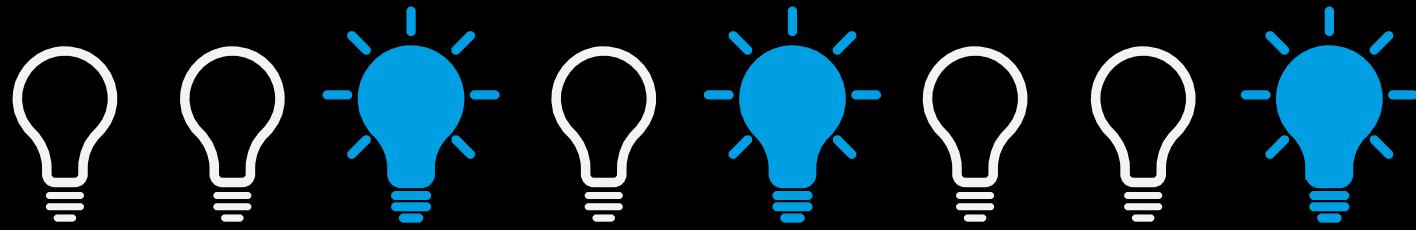
place value systems

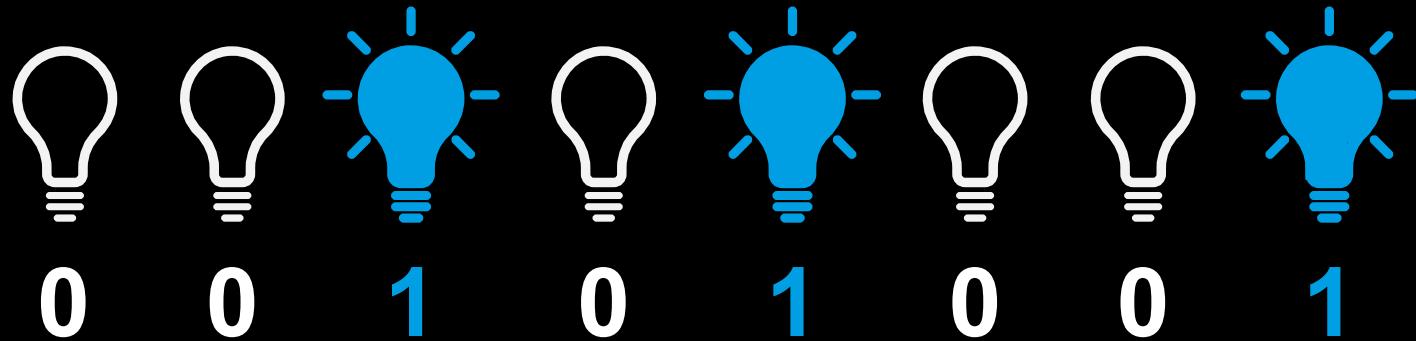
$$R \geq 2$$

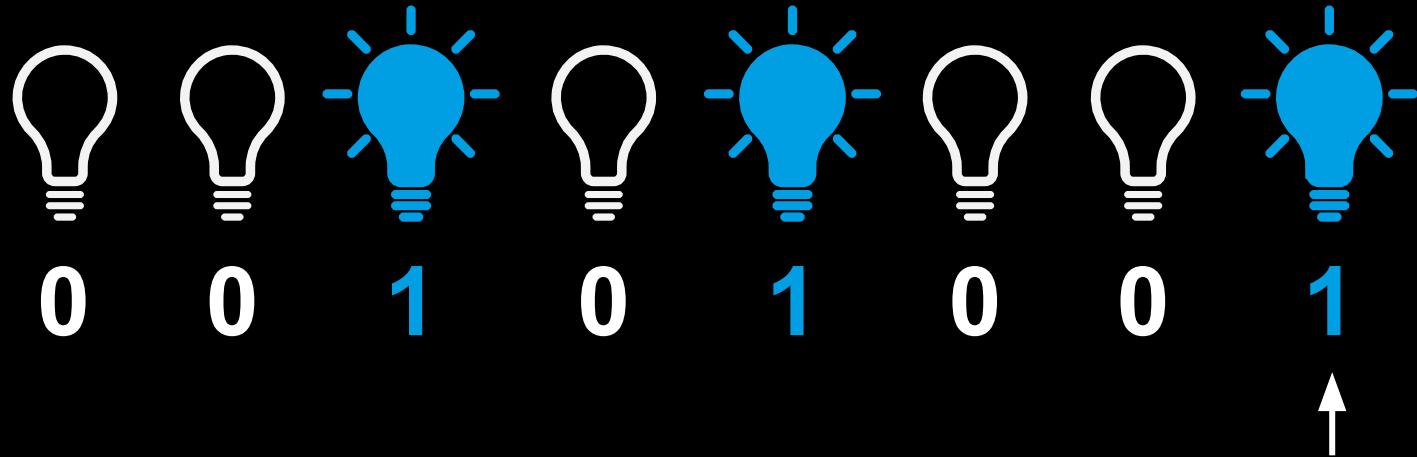
# BITS

why do computers think **binary**?

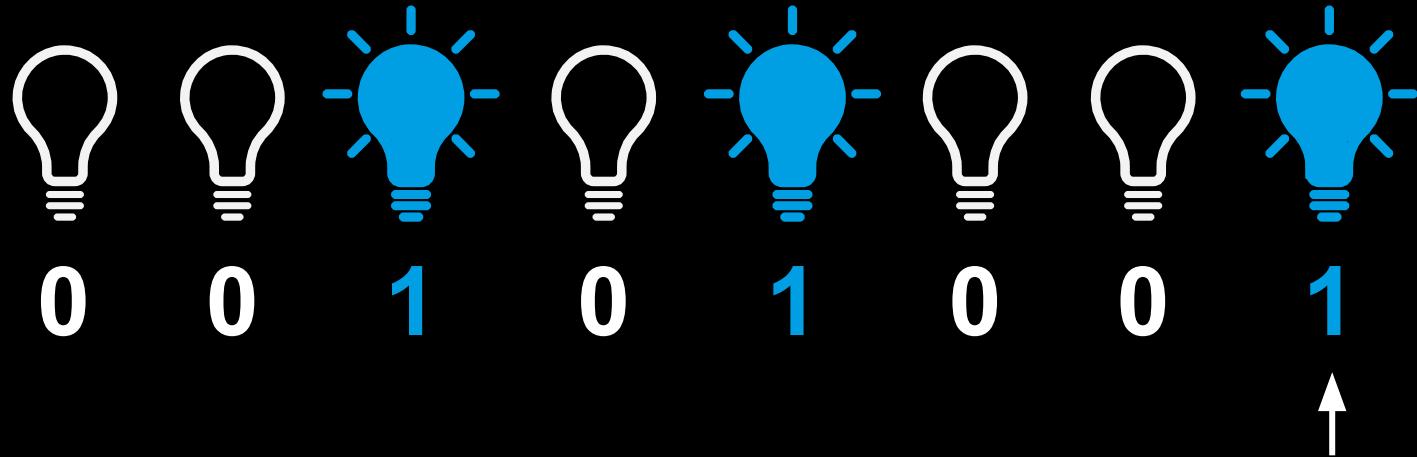








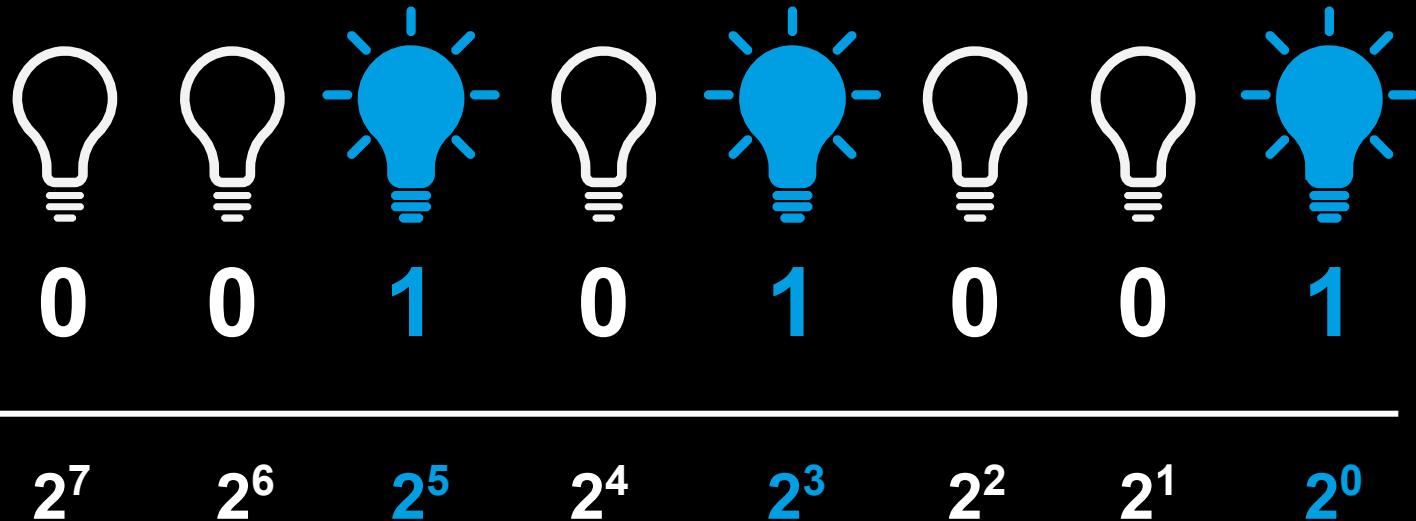
a **bit** (binary digit)

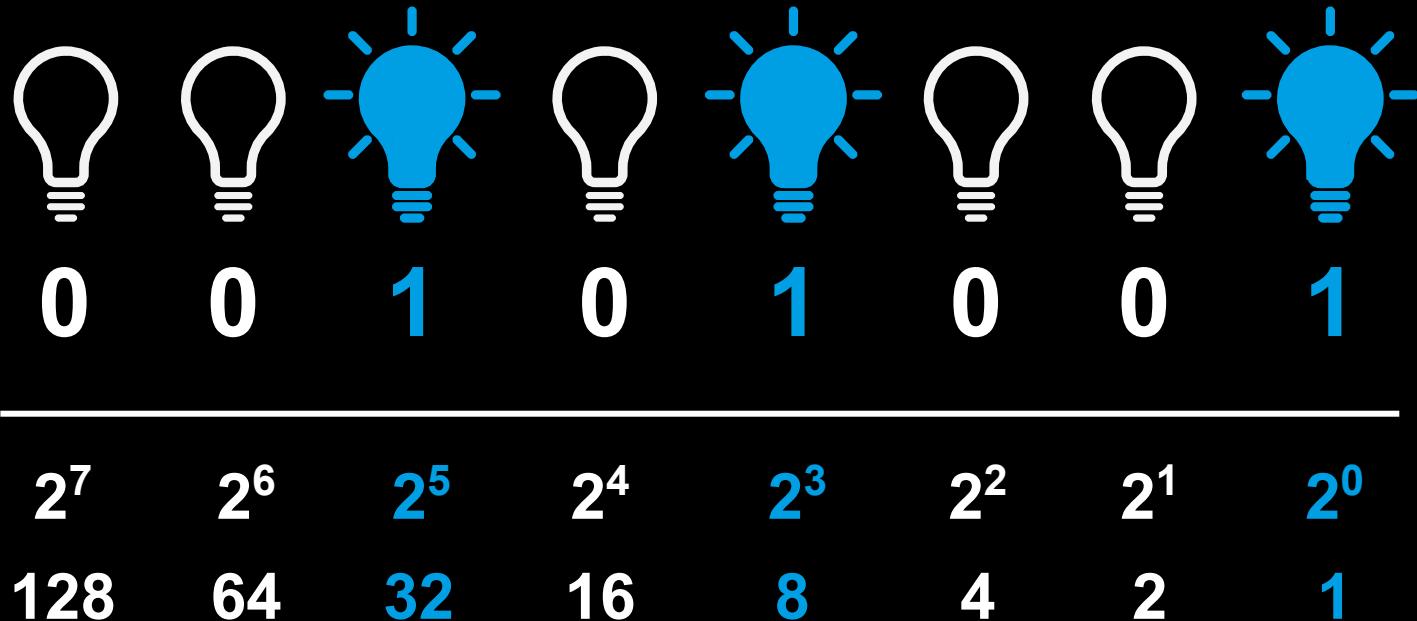


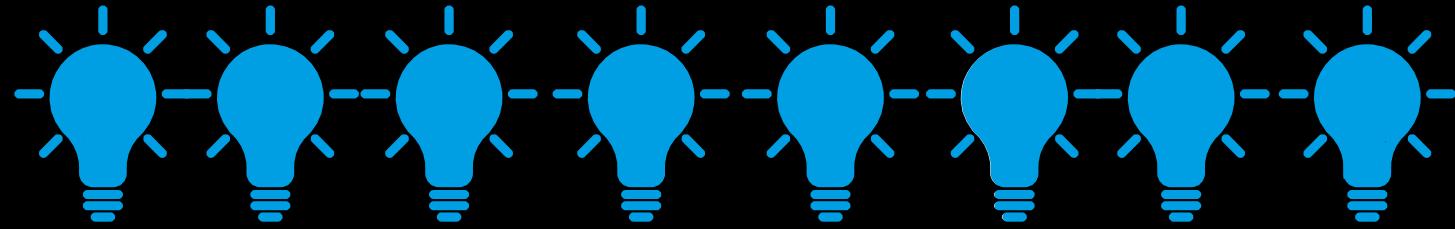
a **bit** (binary digit)

{

a **byte** (8 bits)

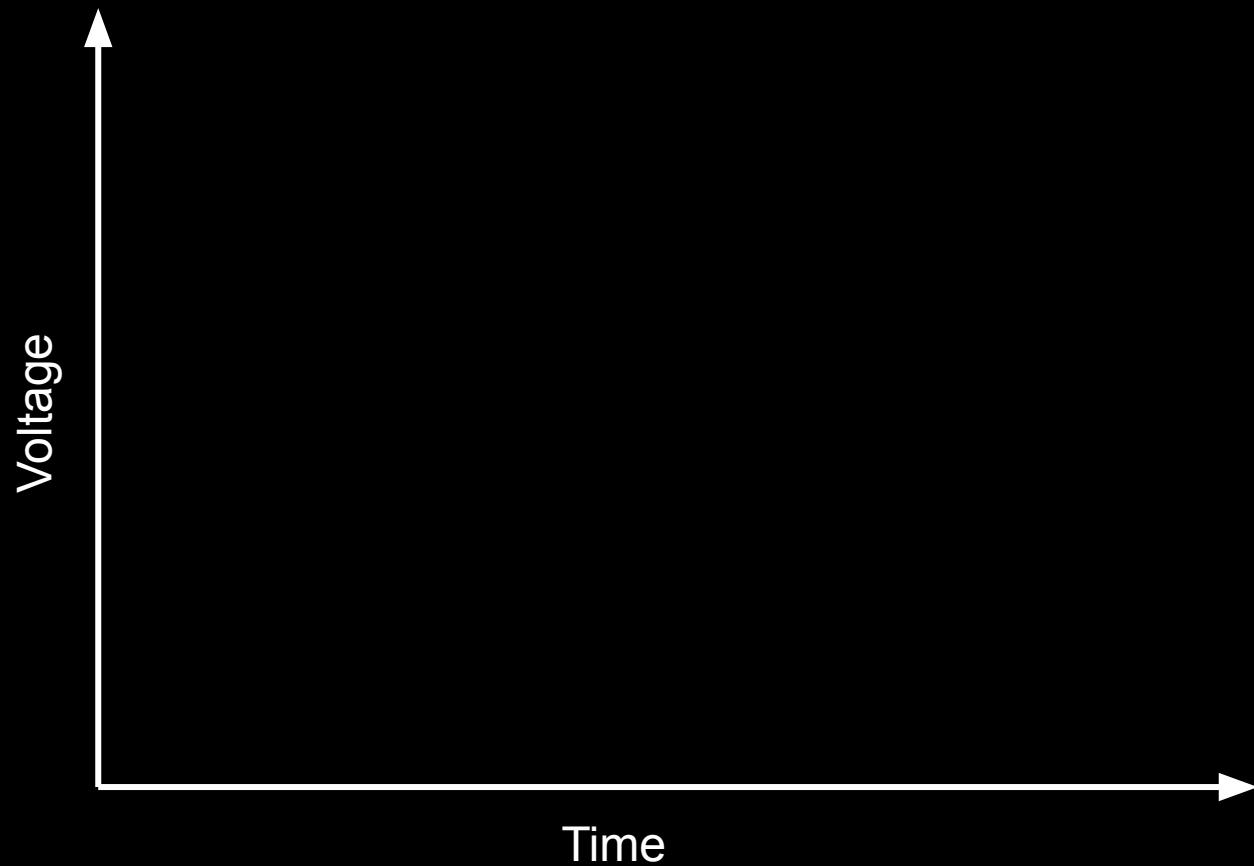


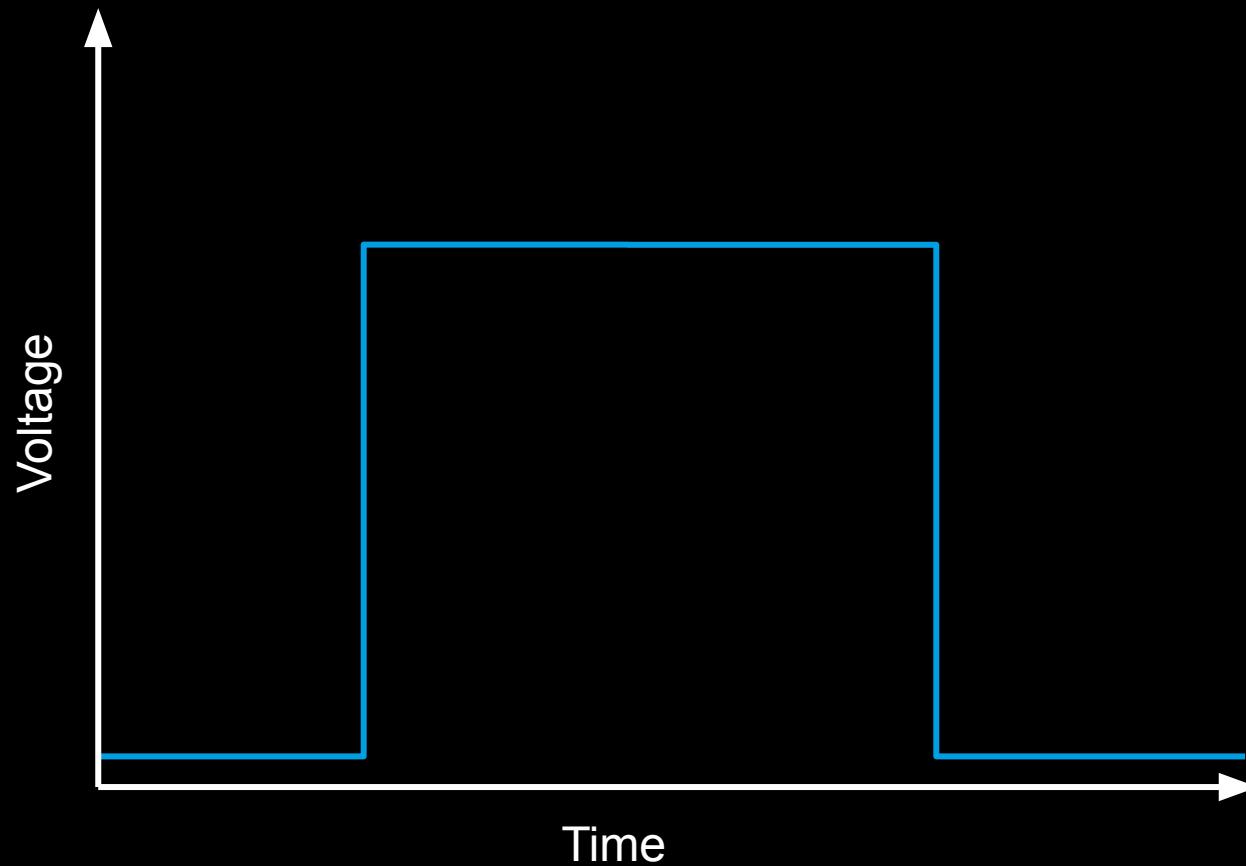


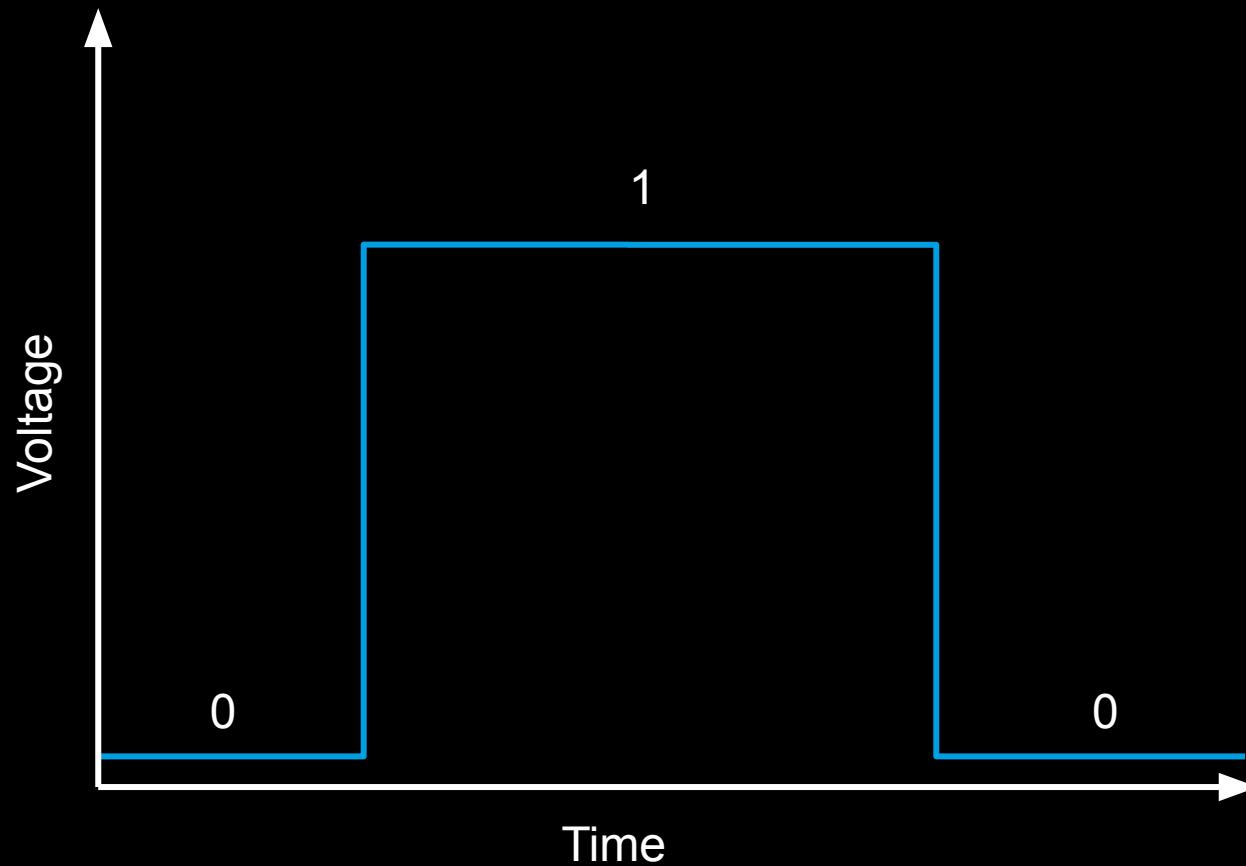


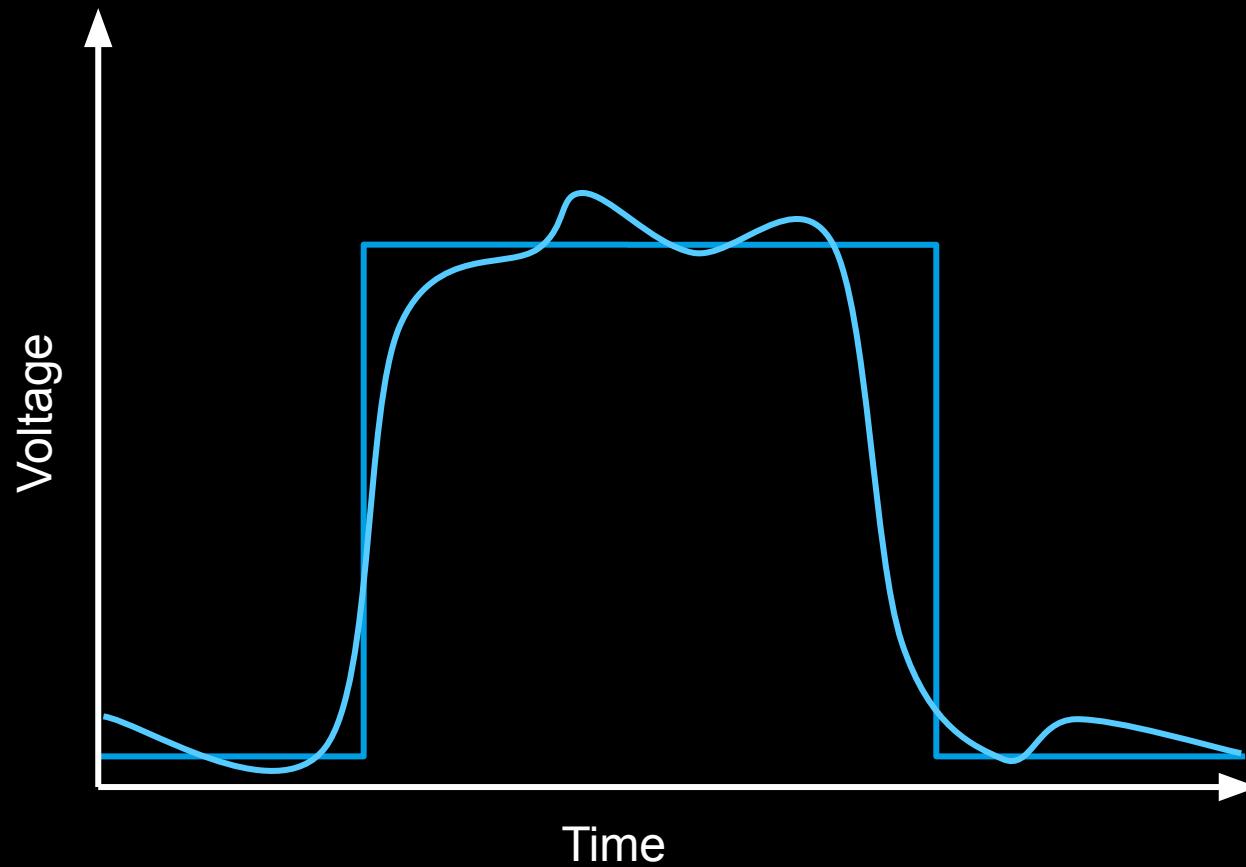
what can we store in one byte?

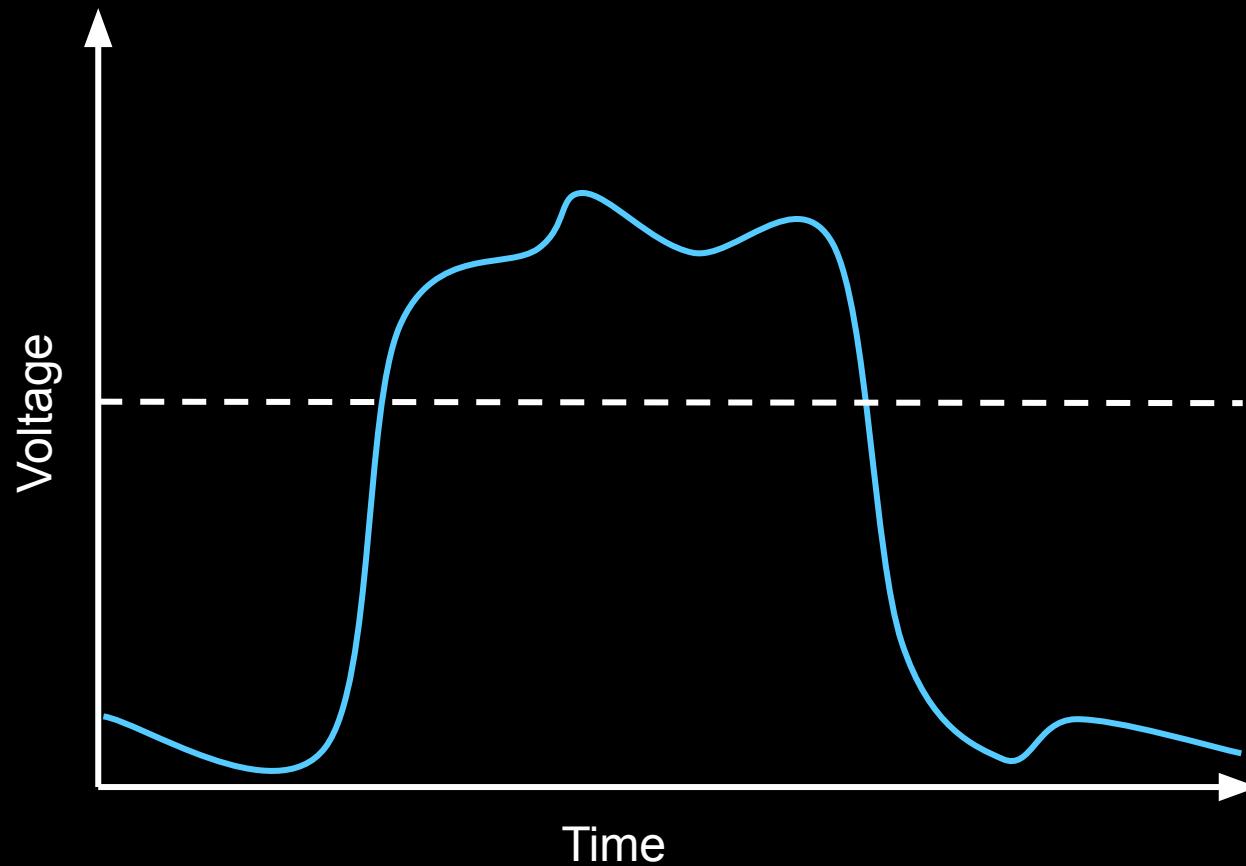
are we stuck with binary?

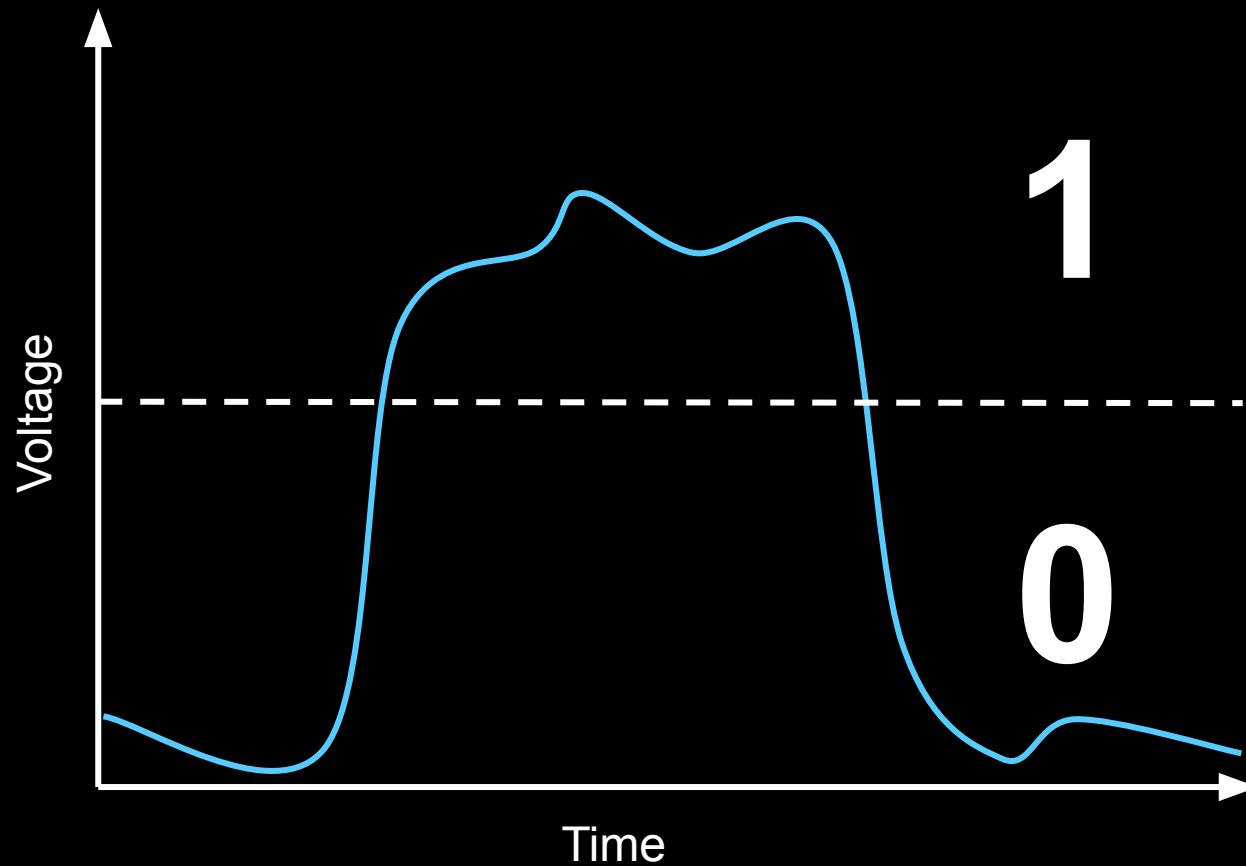


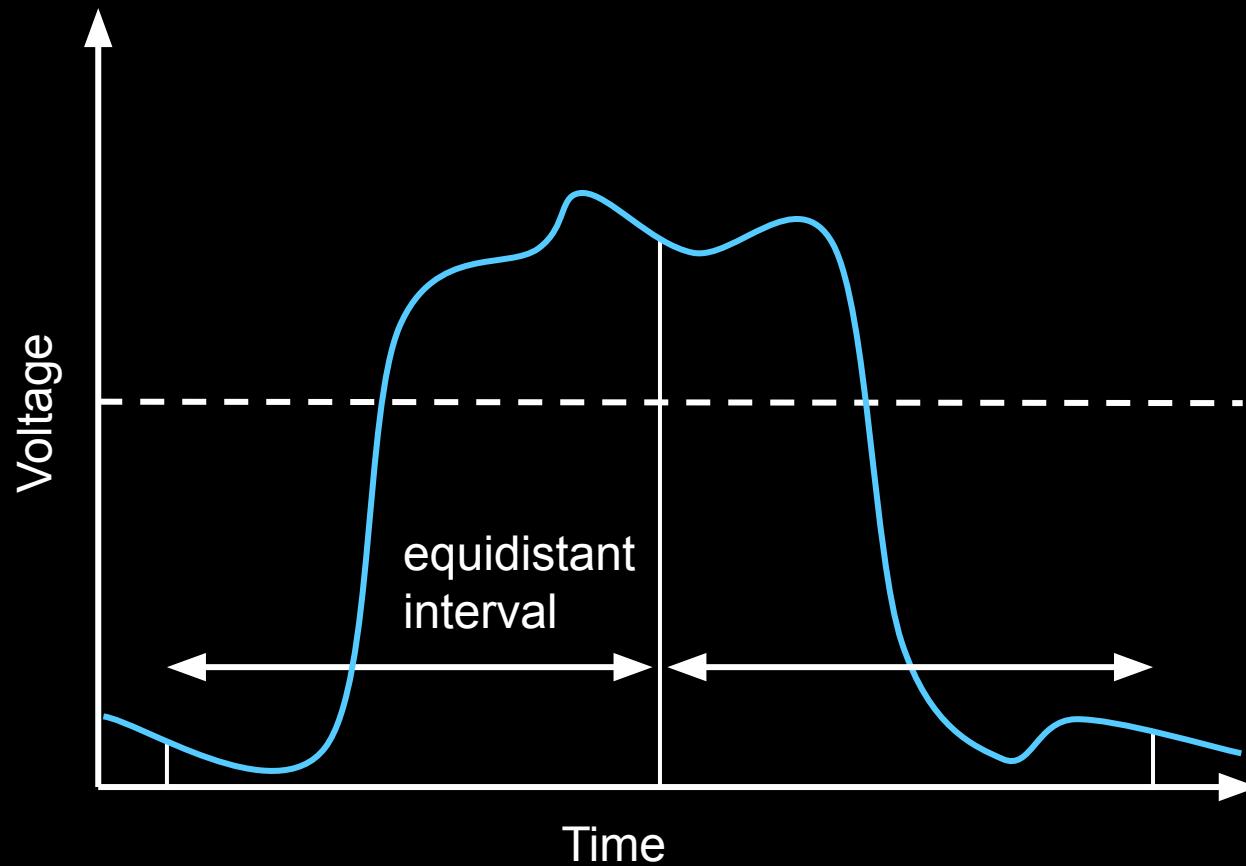


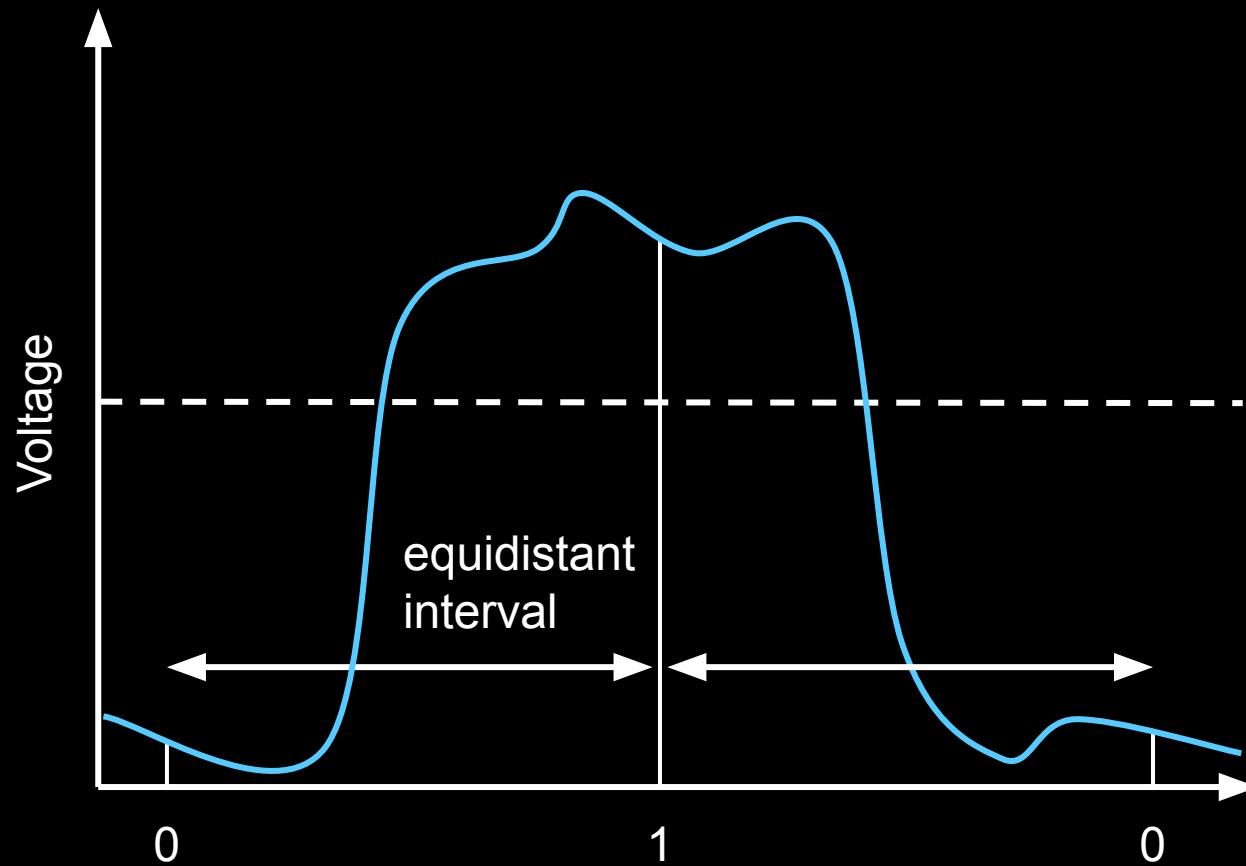


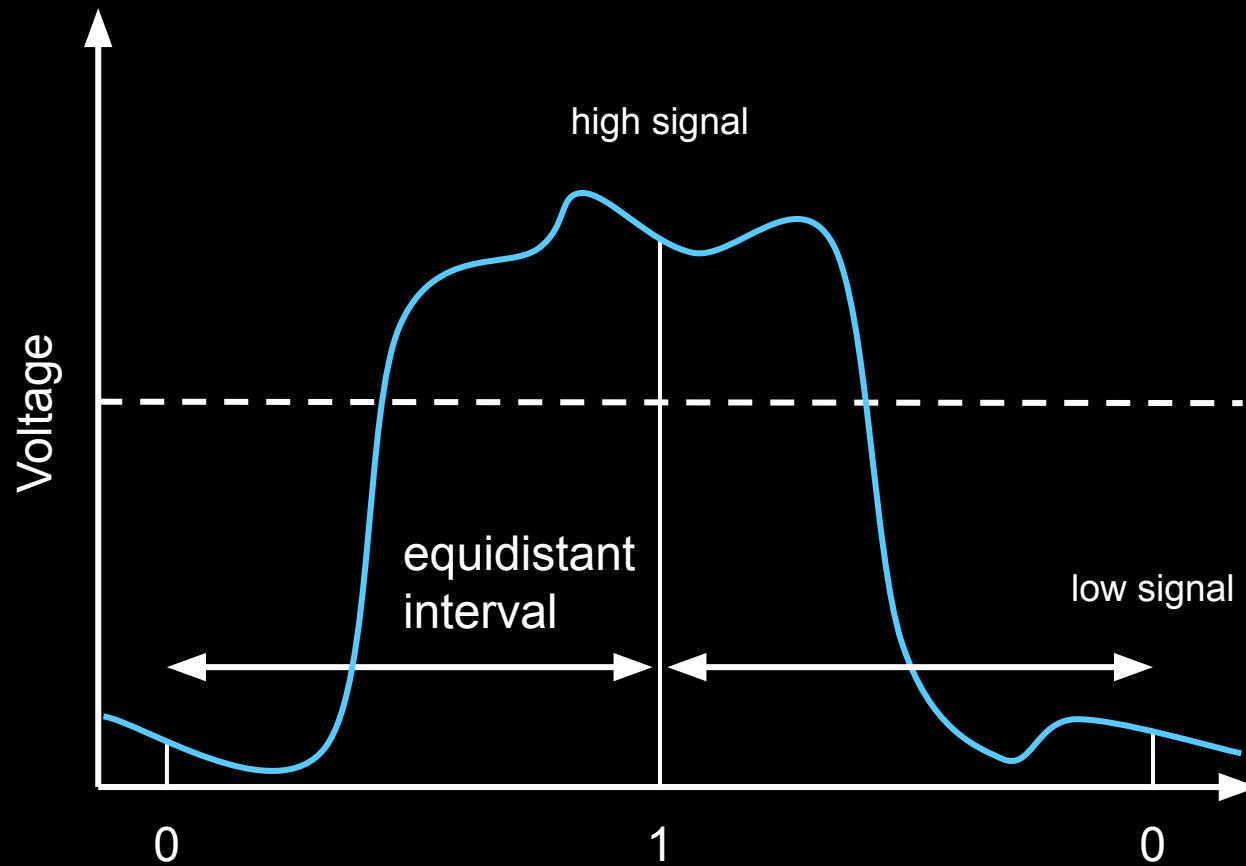




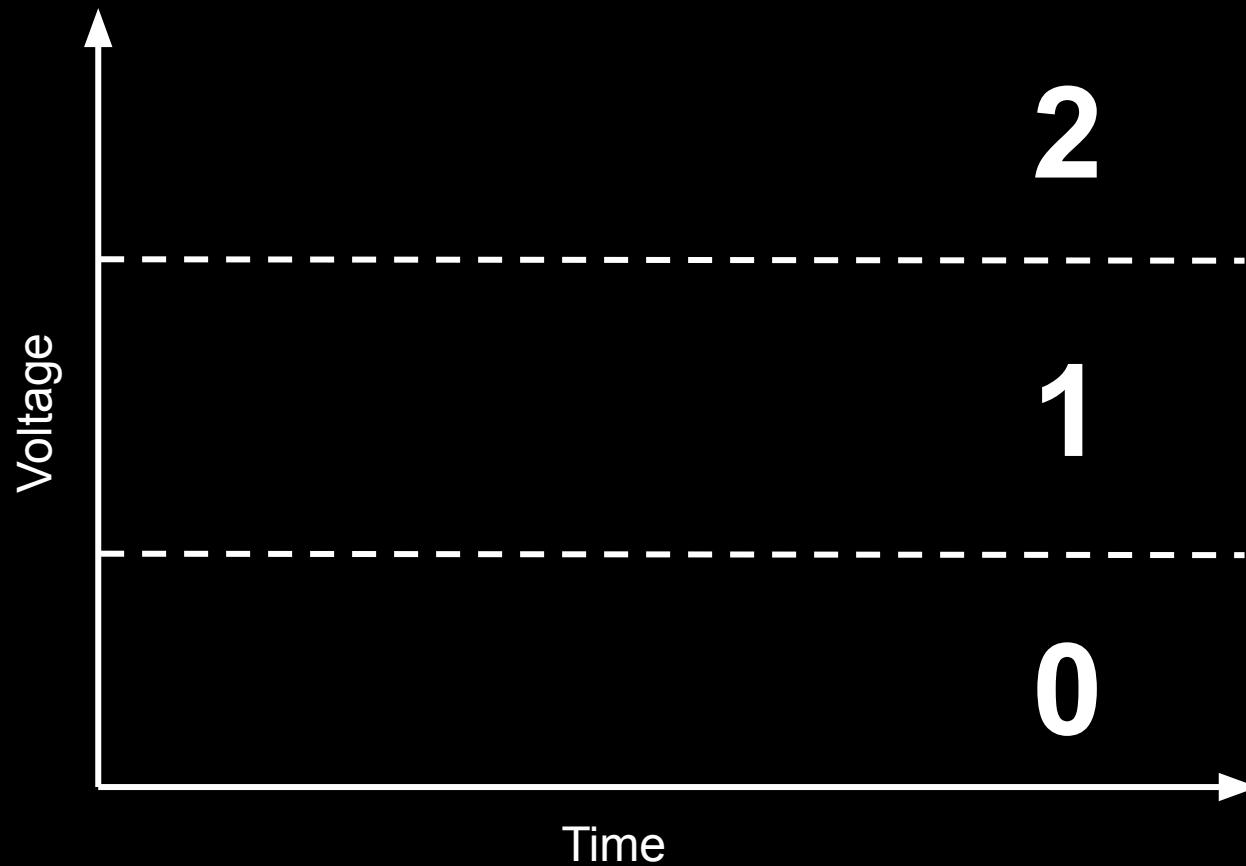


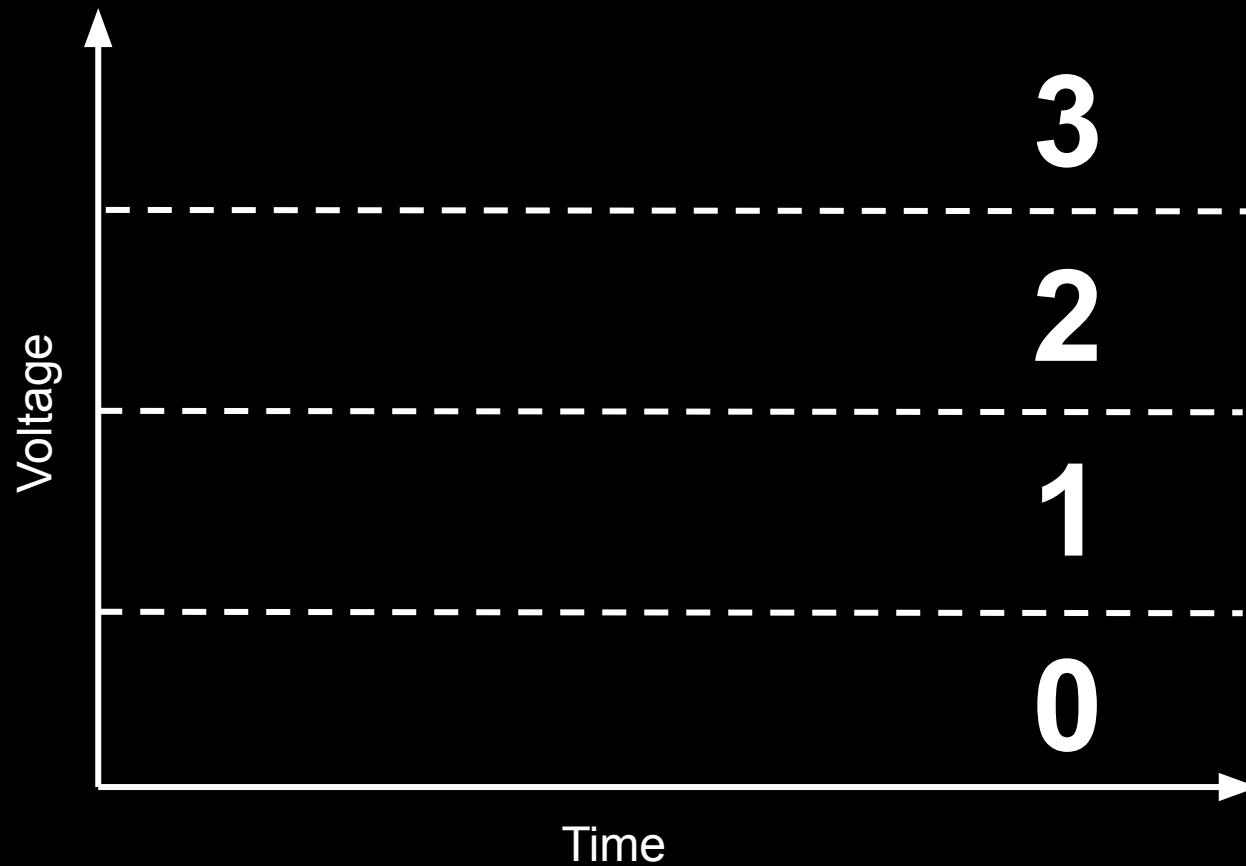


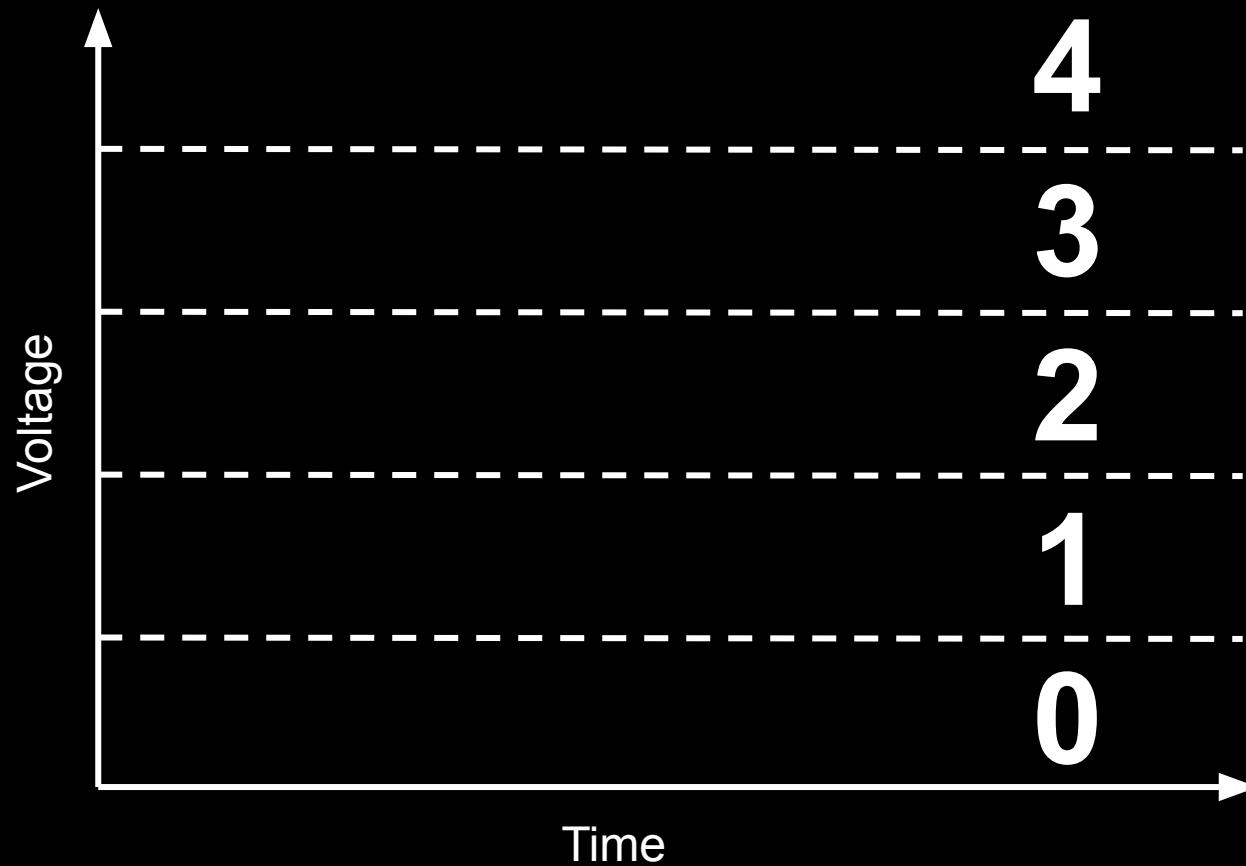


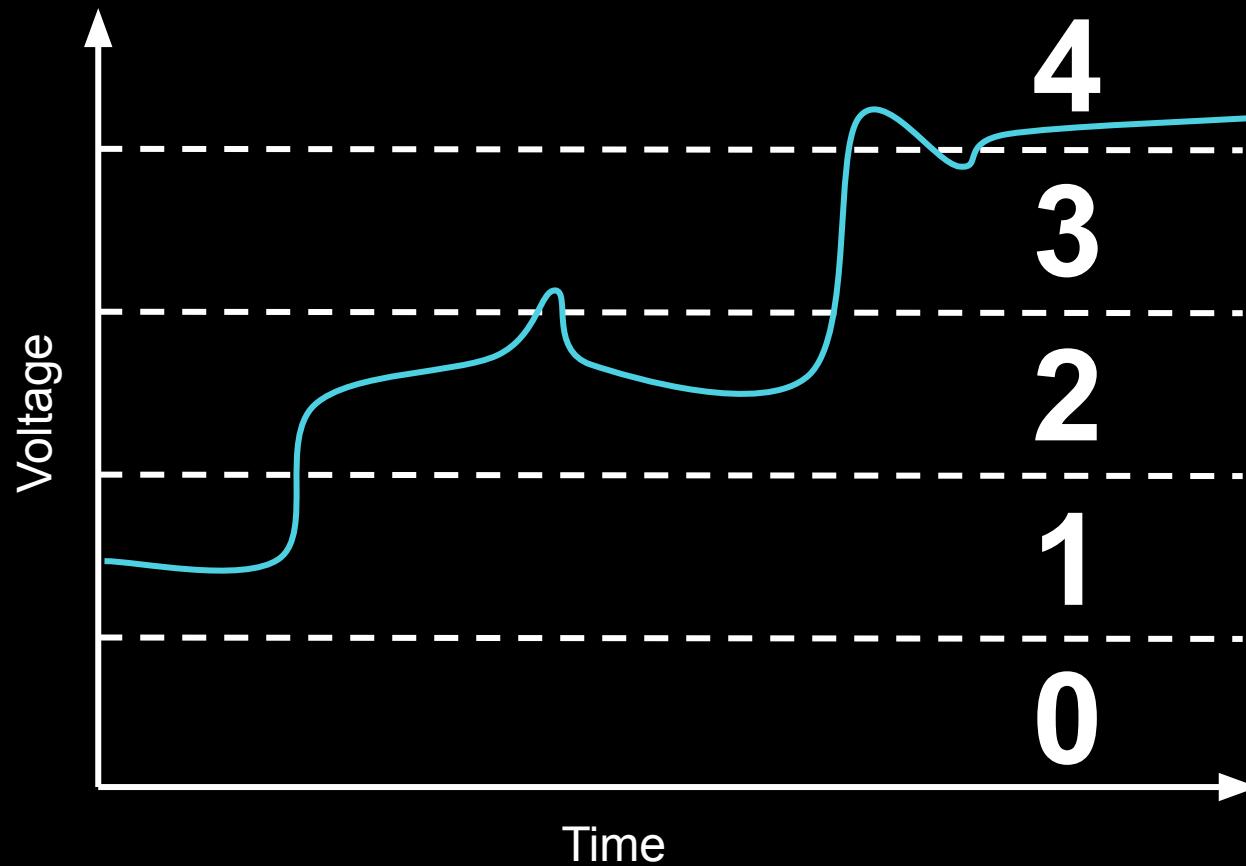


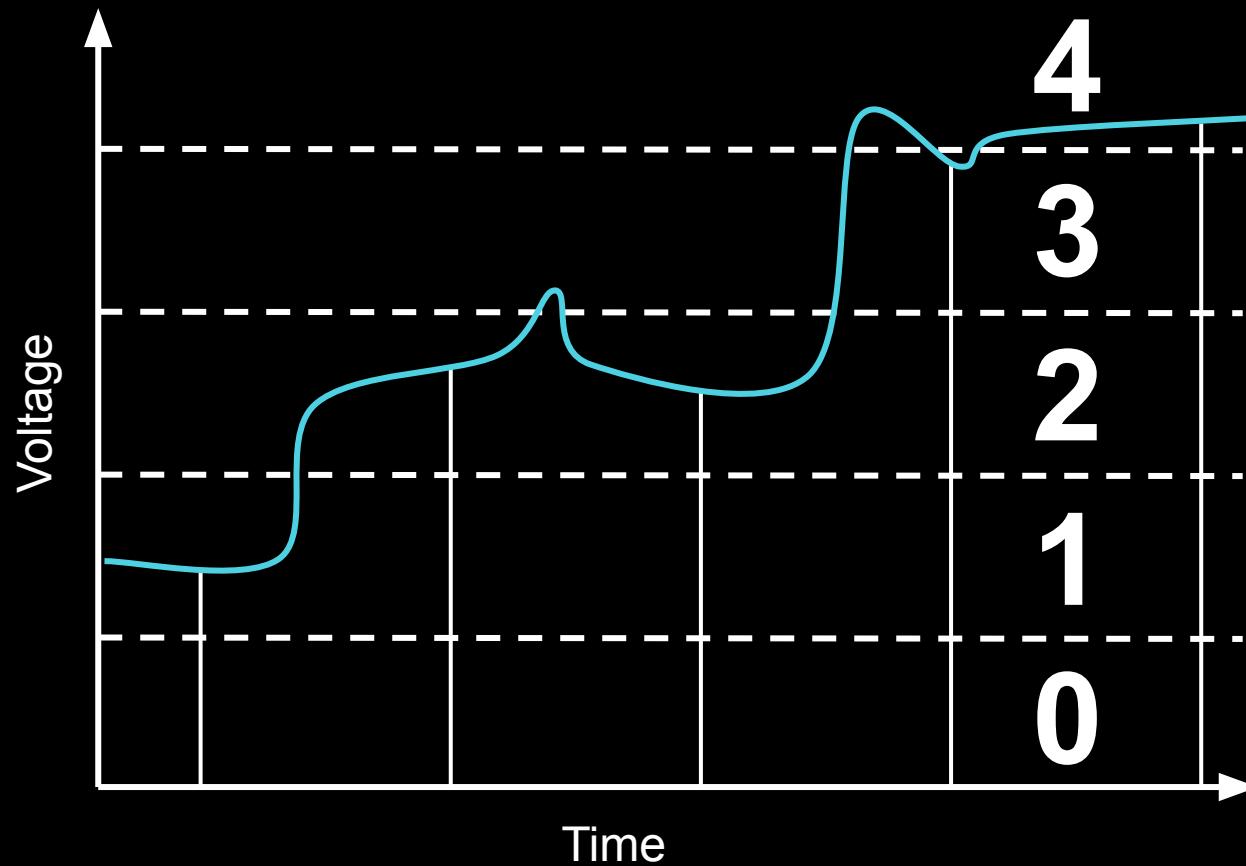
what about ternary?

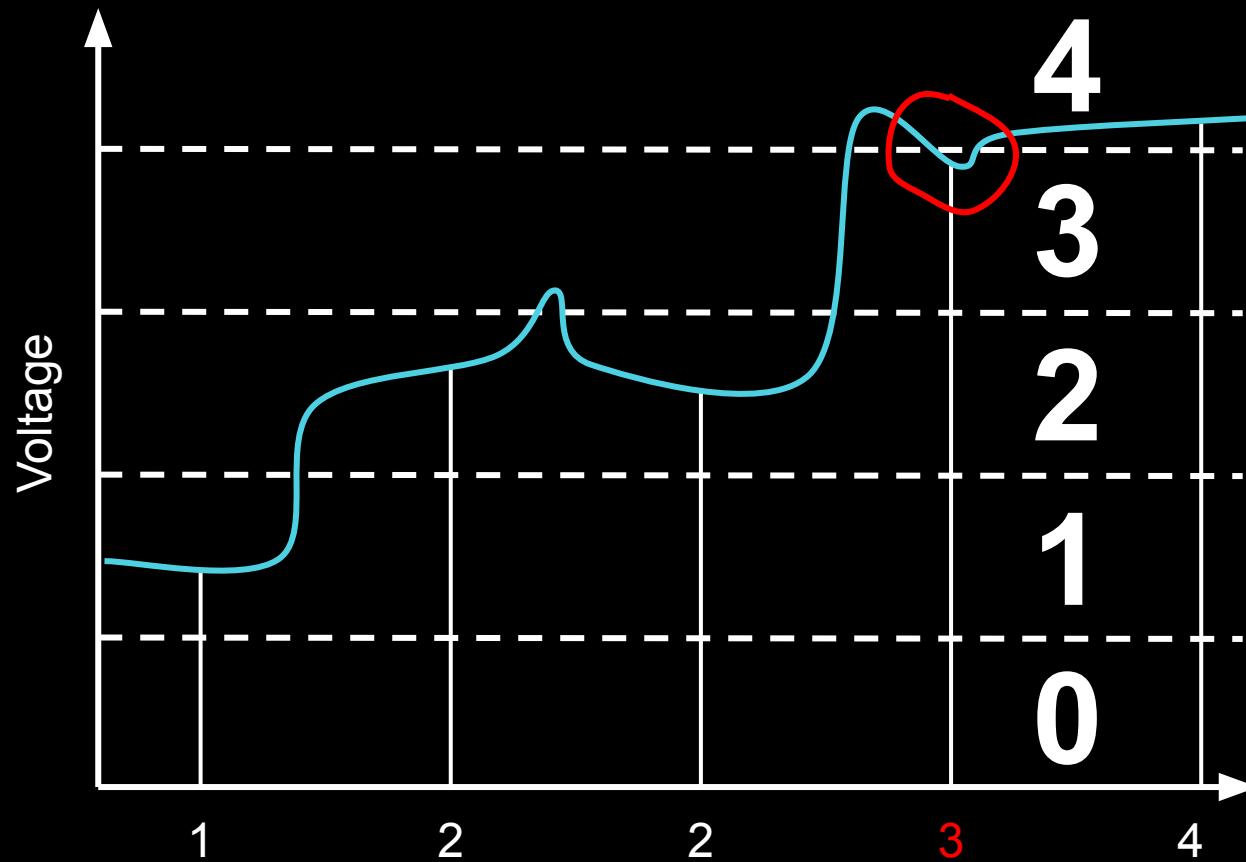












# CODES



# representing text

A	B	C	D	...	a	b	c	d
65	66	67	68		97	98	99	100

# ASCII Code

A	B	C	D	...	a	b	c	d
65	66	67	68		97	98	99	100



1F600



1F601



1F602



1F603

...



1F648



1F649



1F64A



1F64B

# Unicode



1F600



1F601



1F602



1F603

...



1F648



1F649



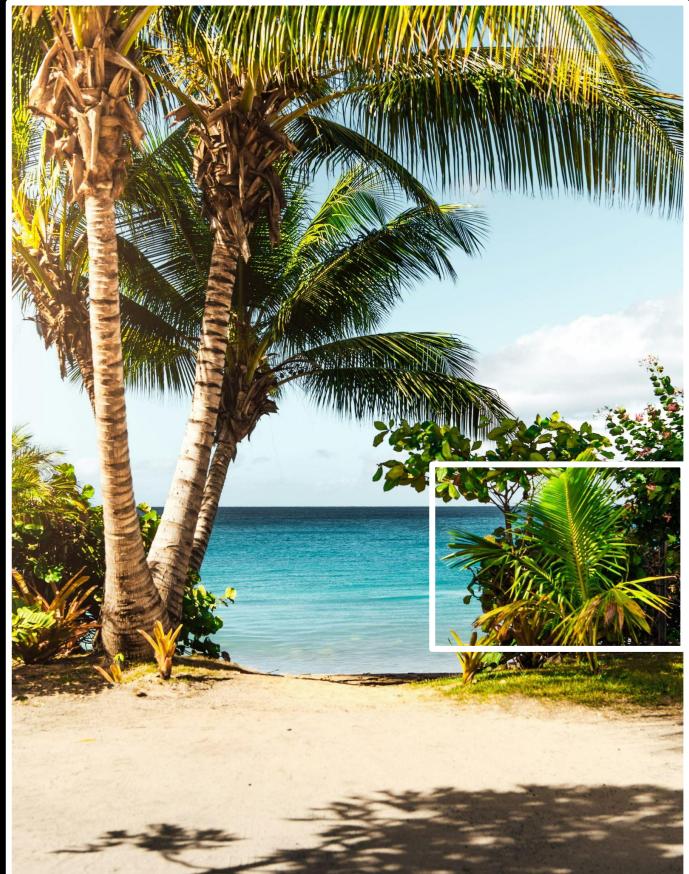
1F64A

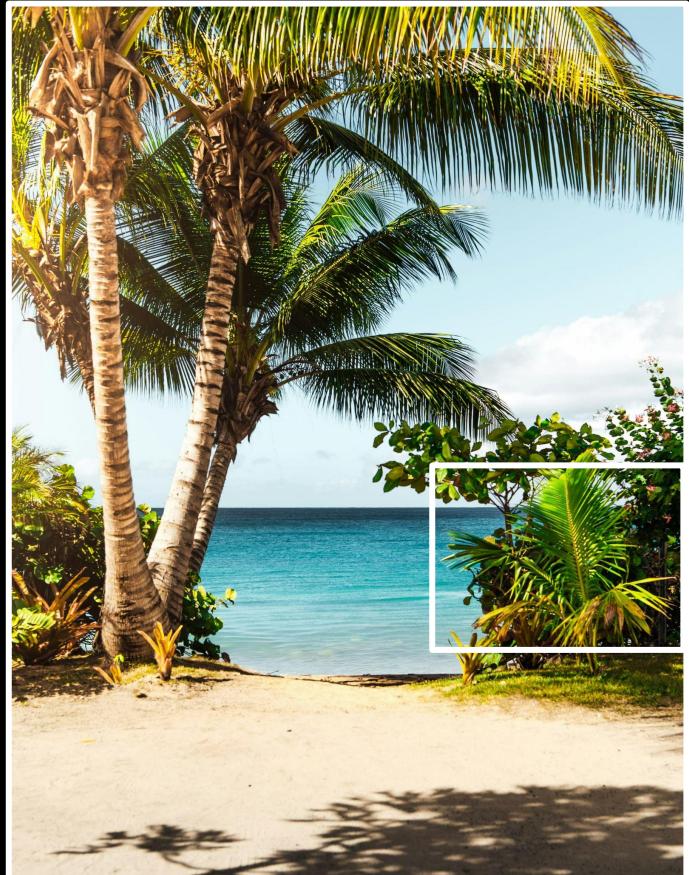


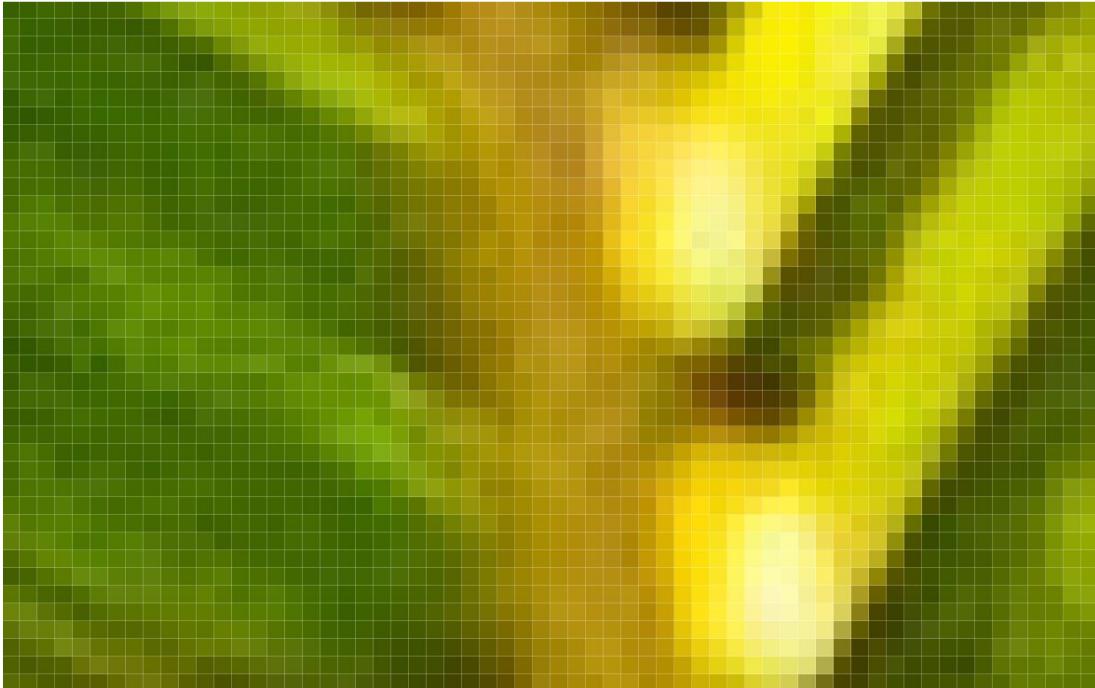
1F64B

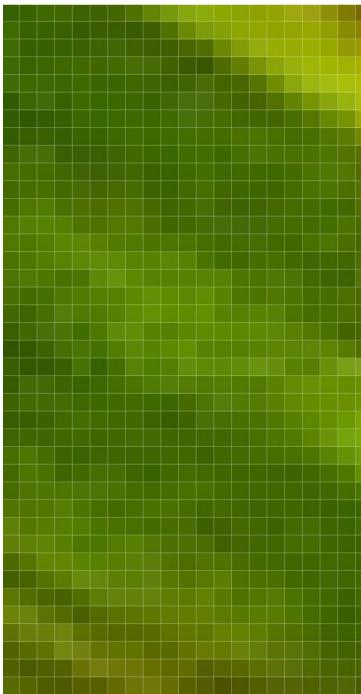
# representing images

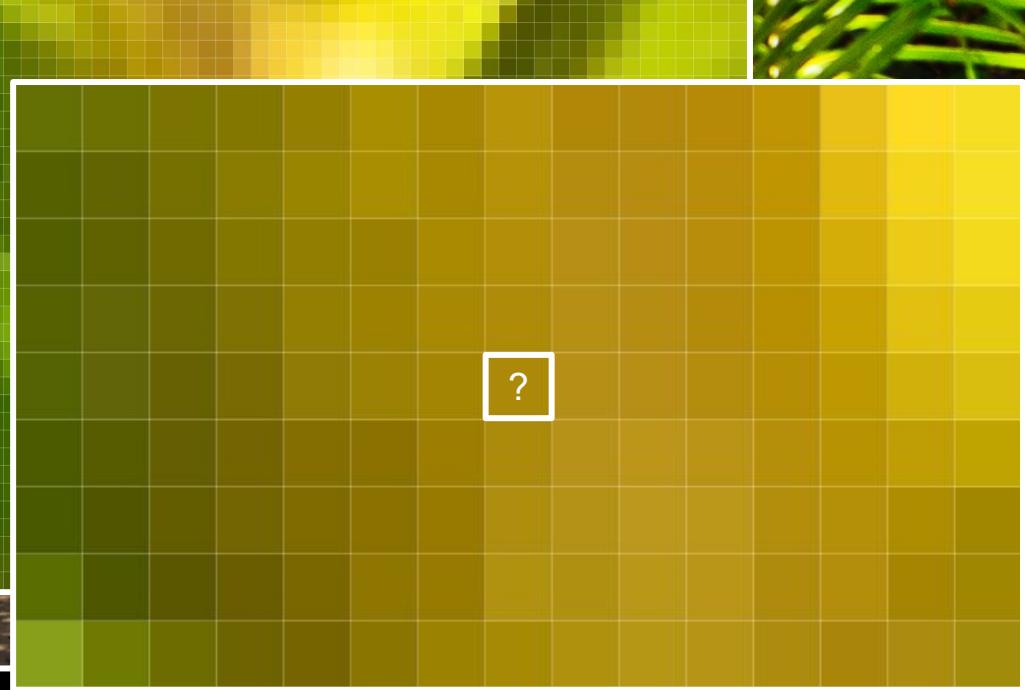




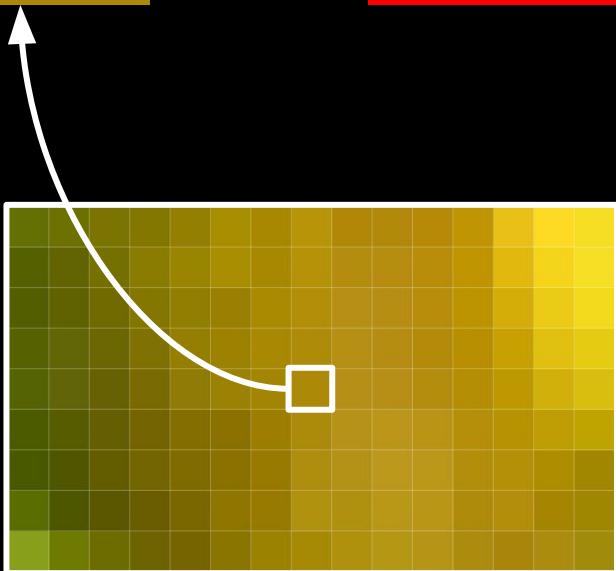




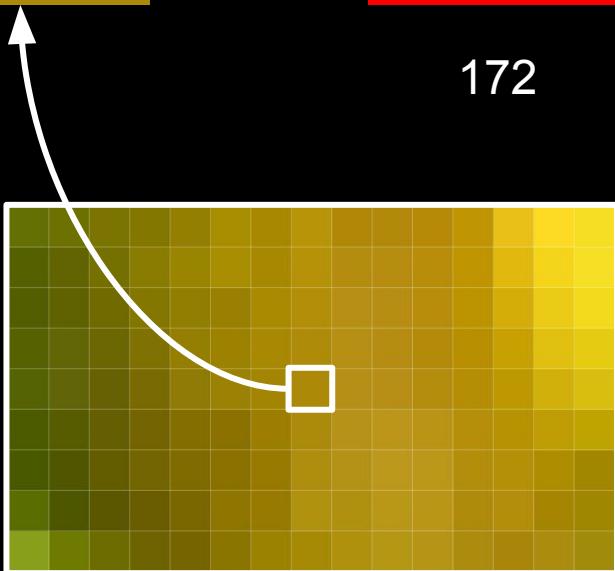




$$\text{Yellow} = \text{R} + \text{G} + \text{B}$$



$$\text{Yellow} = \text{R} + \text{G} + \text{B}$$



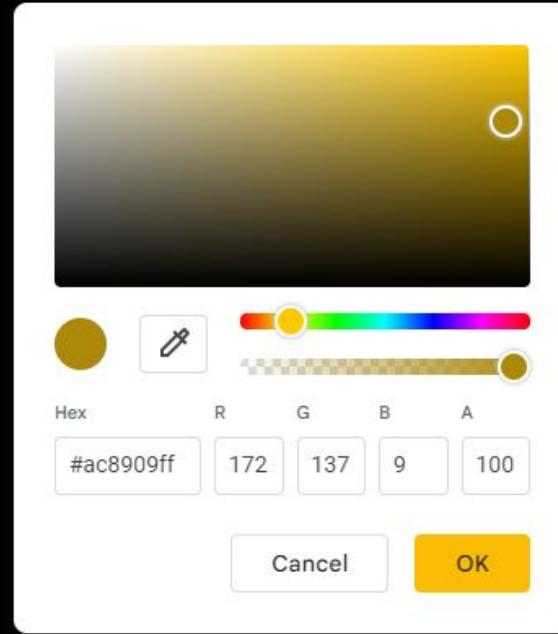
172

137

9

$$\begin{matrix} \text{Yellow} \\ = \end{matrix} \begin{matrix} \text{R} \\ 172 \end{matrix} + \begin{matrix} \text{G} \\ 137 \end{matrix} + \begin{matrix} \text{B} \\ 9 \end{matrix}$$

#AC8909



$$\text{#AC8909} = \text{R} + \text{G} + \text{B}$$

172

137

9

#AC8909

AC

89

09

$$\begin{array}{c} \text{Yellow} \\ = \end{array} \quad \begin{array}{c} \text{R} \\ 172 \\ \text{AC} \\ 10101100 \end{array} \quad + \quad \begin{array}{c} \text{G} \\ 137 \\ 89 \\ 01011001 \end{array} \quad + \quad \begin{array}{c} \text{B} \\ 9 \\ 09 \\ 00001001 \end{array}$$

# possible colors?

R

$2^7 \quad 2^6 \quad 2^5 \quad 2^4 \quad 2^3 \quad 2^2 \quad 2^1 \quad 2^0$

R

G

B

$2^{23} 2^{22} 2^{21} 2^{20} 2^{19} 2^{18} 2^{17} 2^{16}$

$2^{15} 2^{14} 2^{13} 2^{12} 2^{11} 2^{10} 2^9 2^8$

$2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0$

R

G

B

$2^{23} 2^{22} 2^{21} 2^{20} 2^{19} 2^{18} 2^{17} 2^{16}$      $2^{15} 2^{14} 2^{13} 2^{12} 2^{11} 2^{10} 2^9 2^8$      $2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0$

$$8.388.608 + 8.388.607 = 16.777.215$$

R

G

B

$2^{23} 2^{22} 2^{21} 2^{20} 2^{19} 2^{18} 2^{17} 2^{16}$

$2^{15} 2^{14} 2^{13} 2^{12} 2^{11} 2^{10} 2^9 2^8$

$2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0$

{

{

{

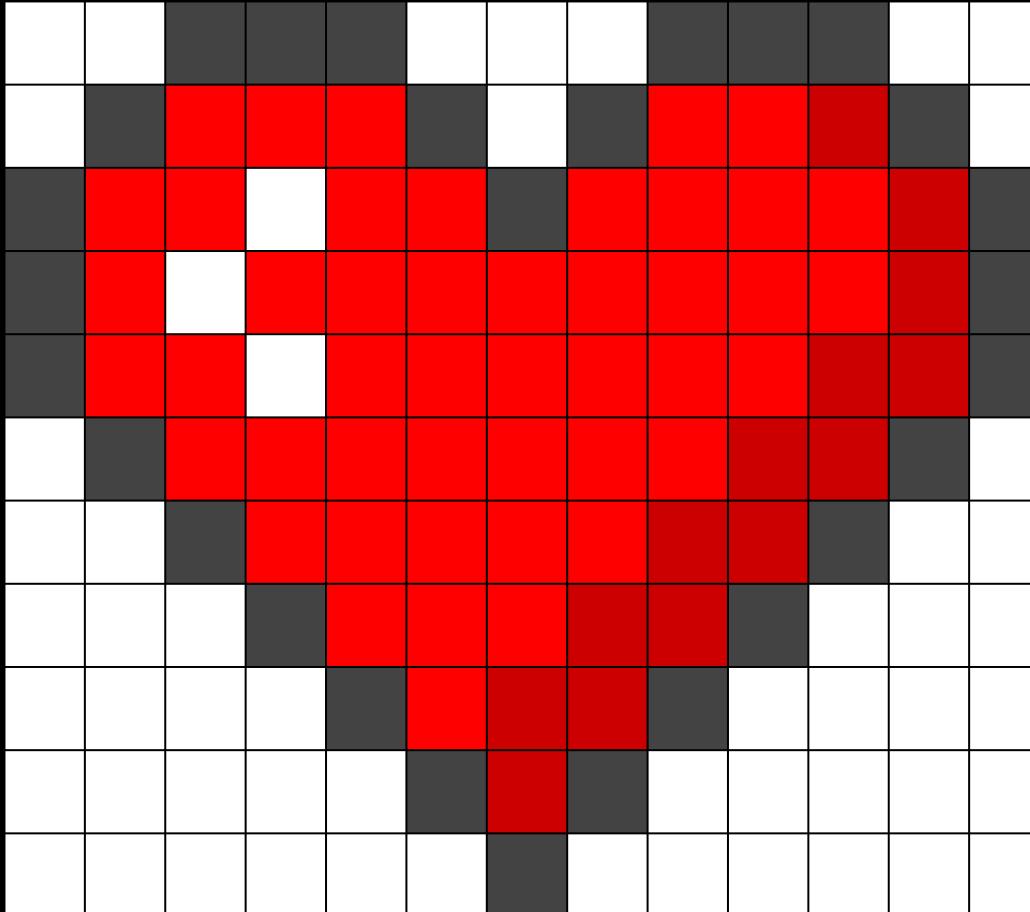
256

X

256

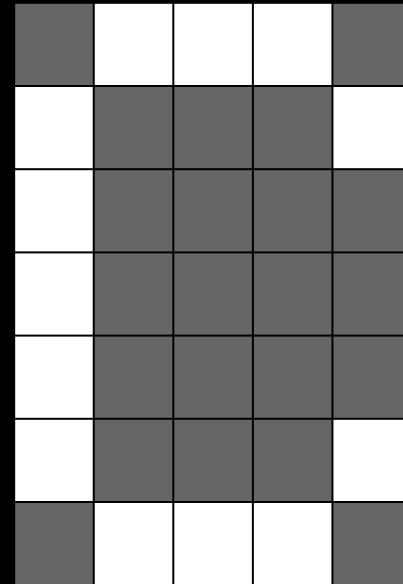
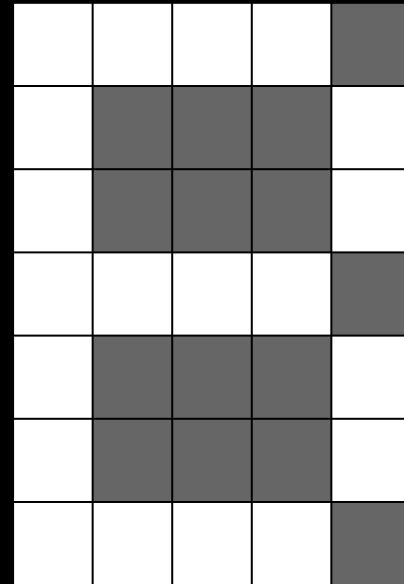
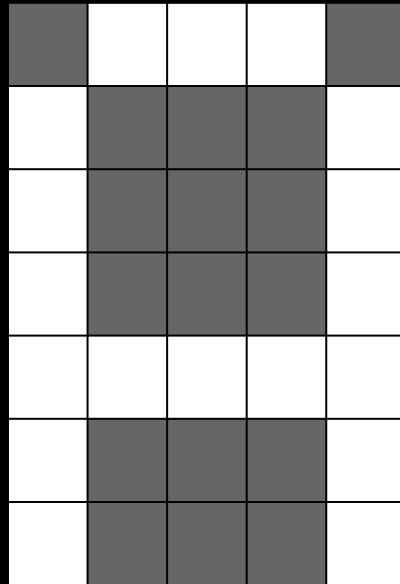
X

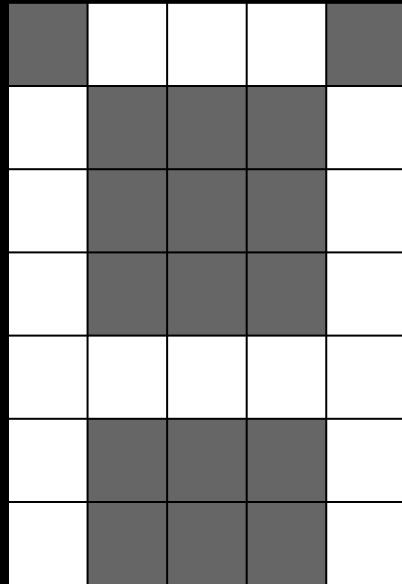
256





compression?





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

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



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

5

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

7



011101000110001100  
0111111000110001

bitmap

# vector graphics

grayscale

# representing sound

telephone

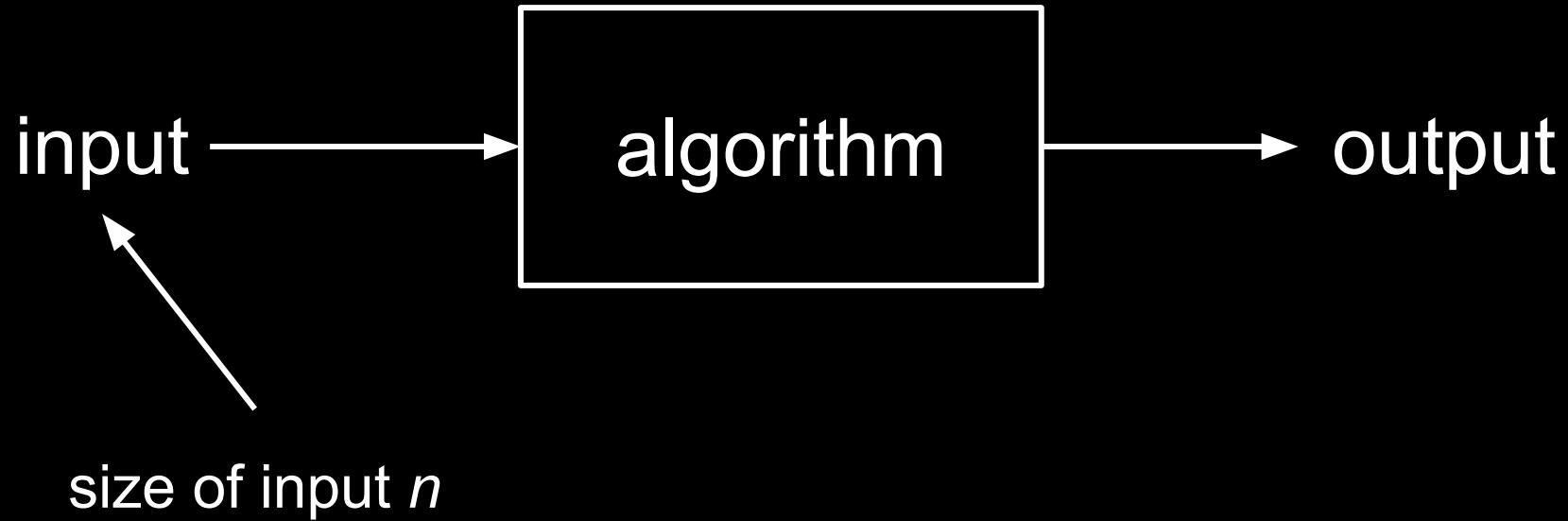
music

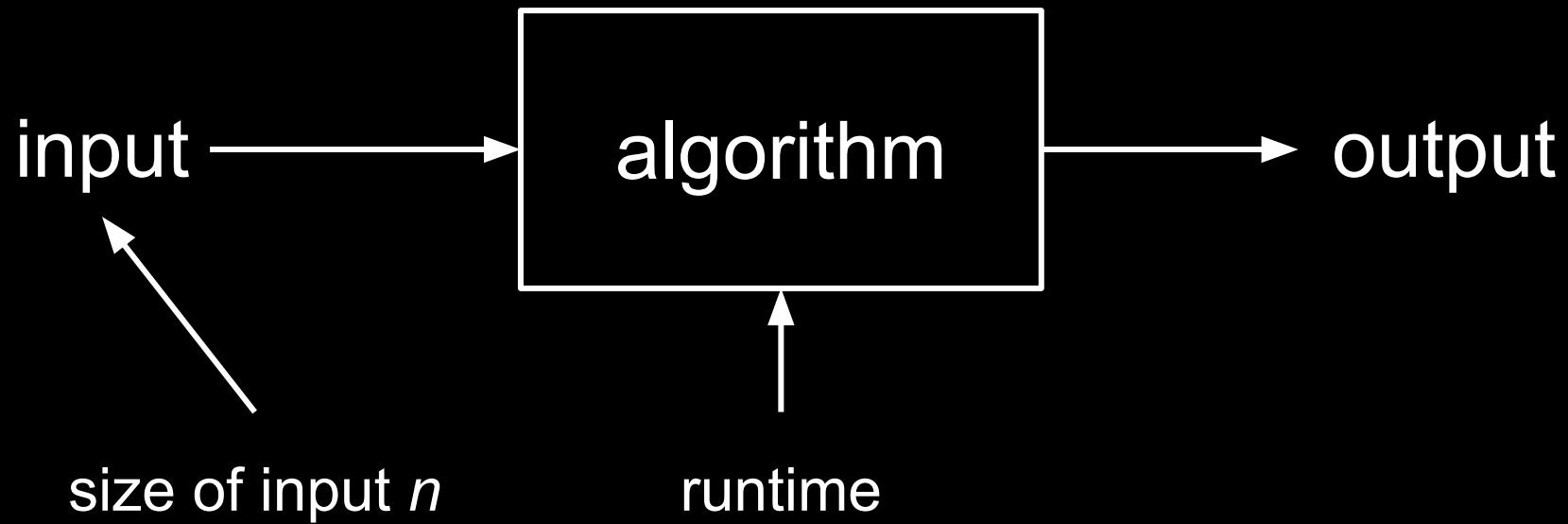
# musical instrument digital interface (MIDI)

# ALGORITHMS

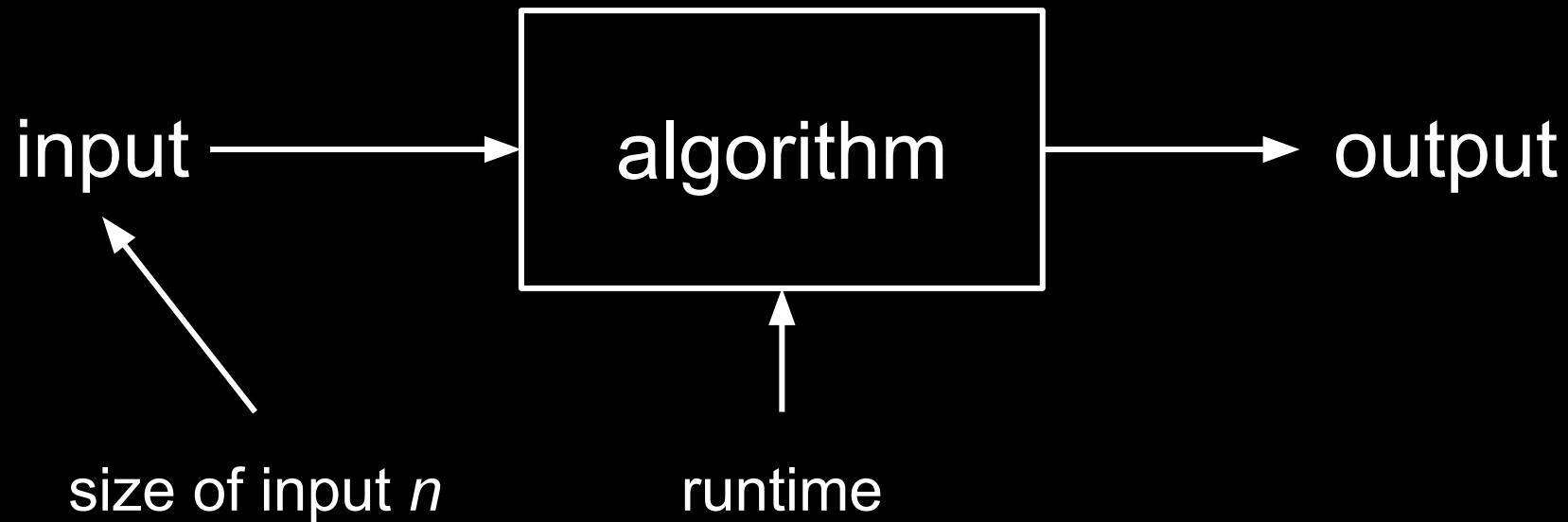
# complexity







$O(n)$



# COMPUTERS

# ARITHMETIC

# MEMORY

# ANALOG VS. DIGITAL

which is better?

