ORDER

ORGANIZATION

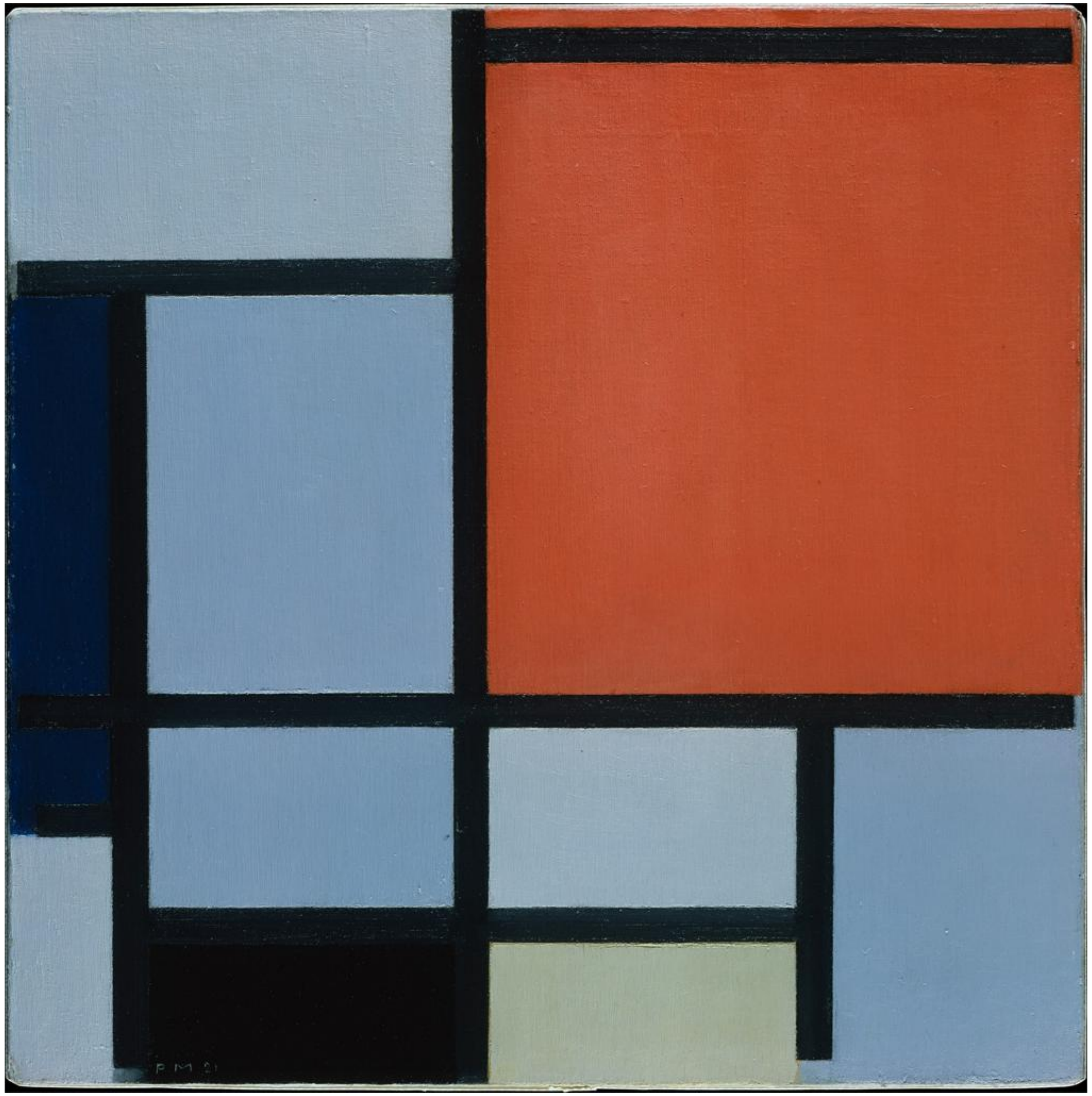
INFORMATION

Dov Levine

I SOME RANDOM THOUGHTS WHICH MAY HAVE SOME CONNECTION.

WHAT IS ORDER?

PIET MONDRIAN



BARNETT
NEWMAN

MARK
ROTHKO

MoMA MoMA

JACKSON POLLOCK

MoMA MoMA

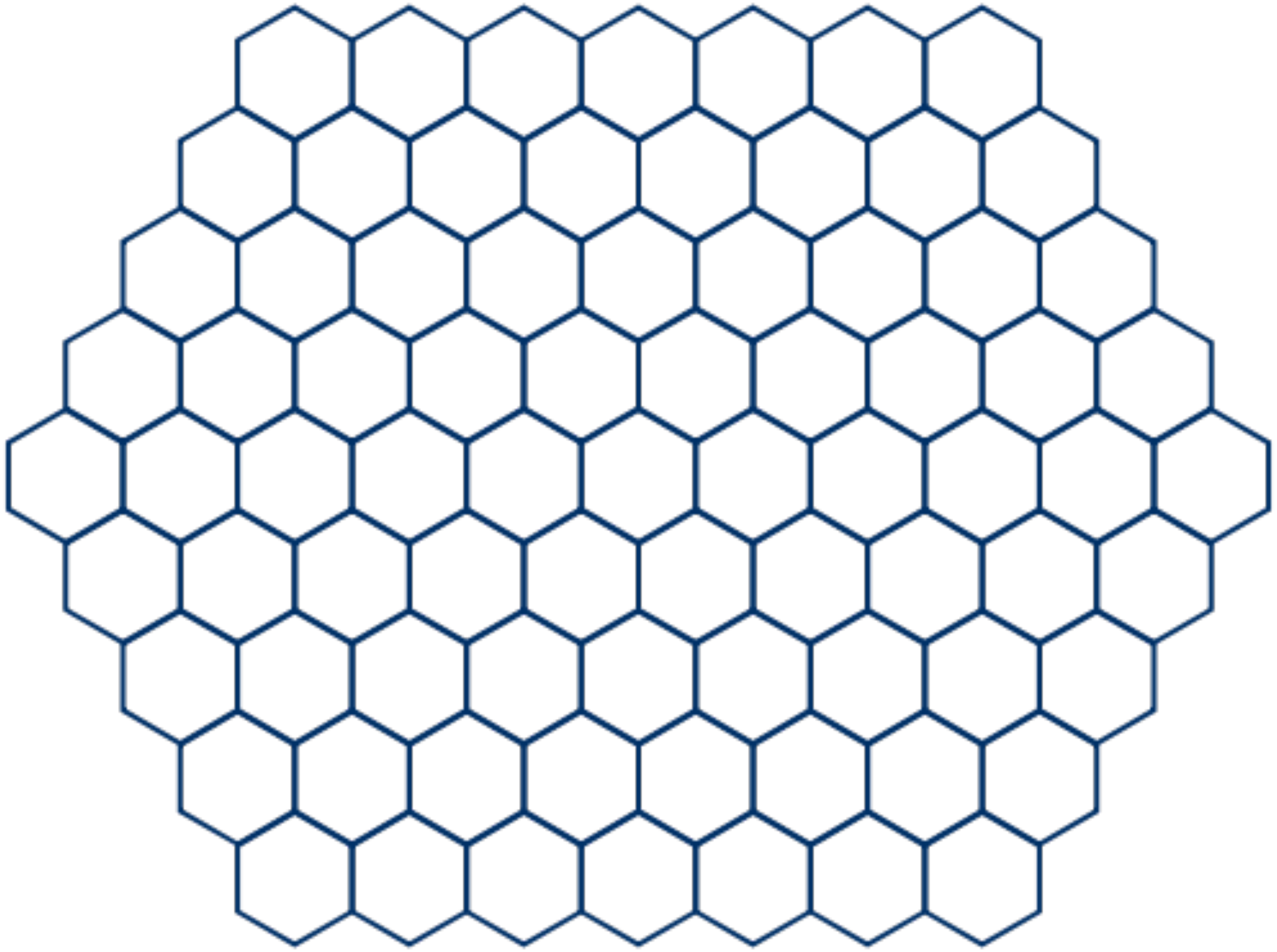
JACKSON POLLOCK

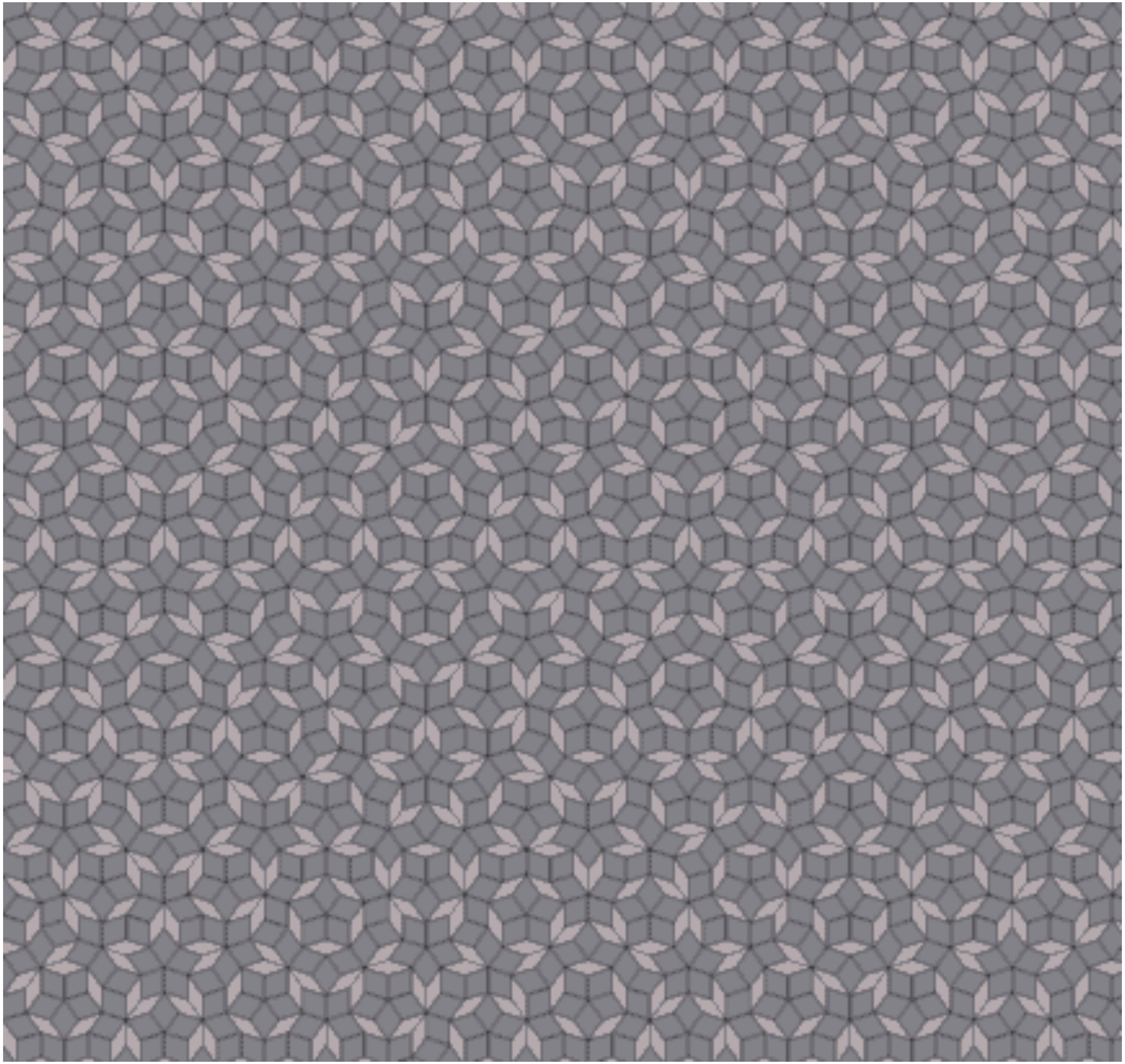
MoMA MoMA

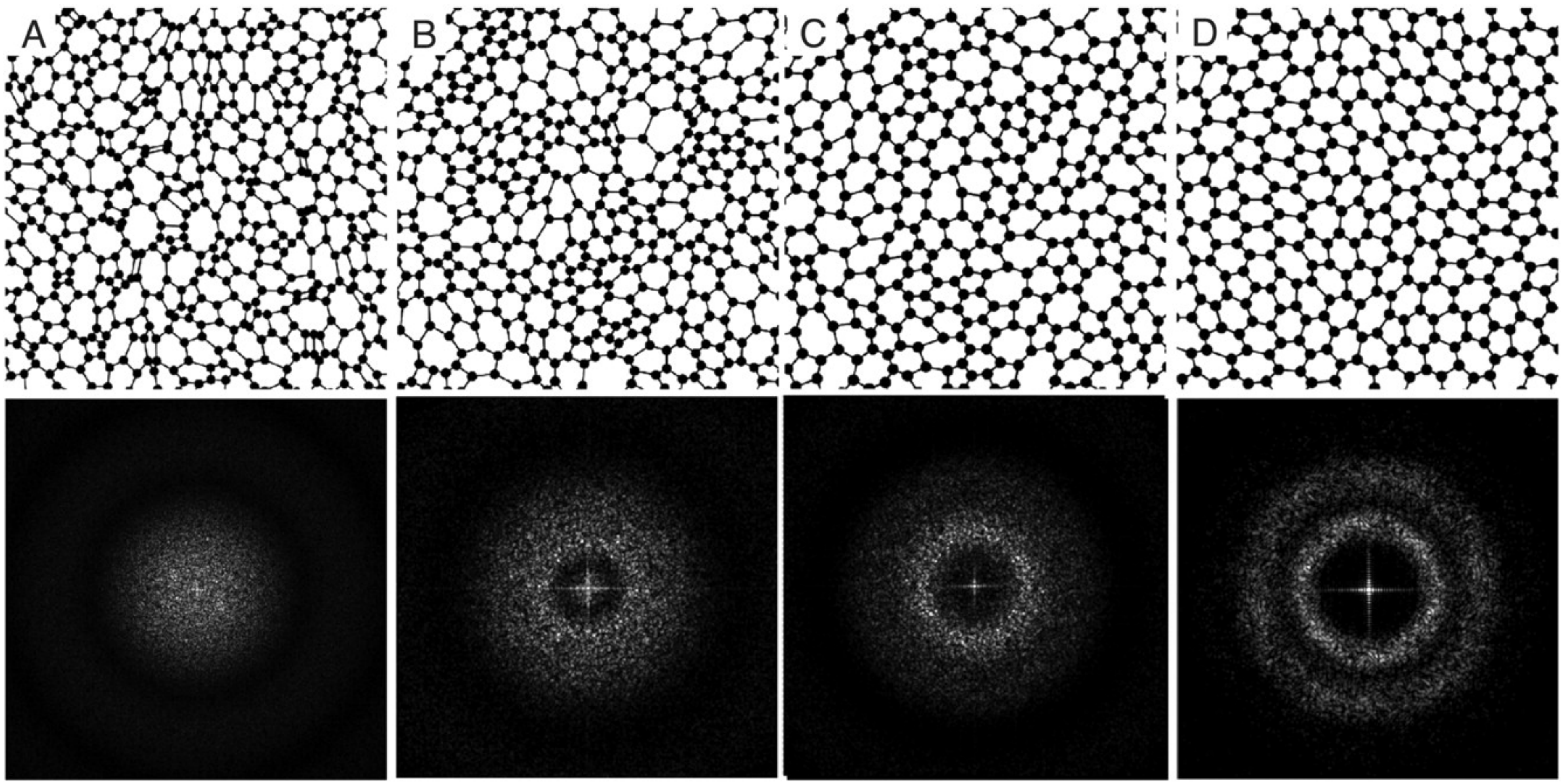
CAN WE IDENTIFY

ORDER?

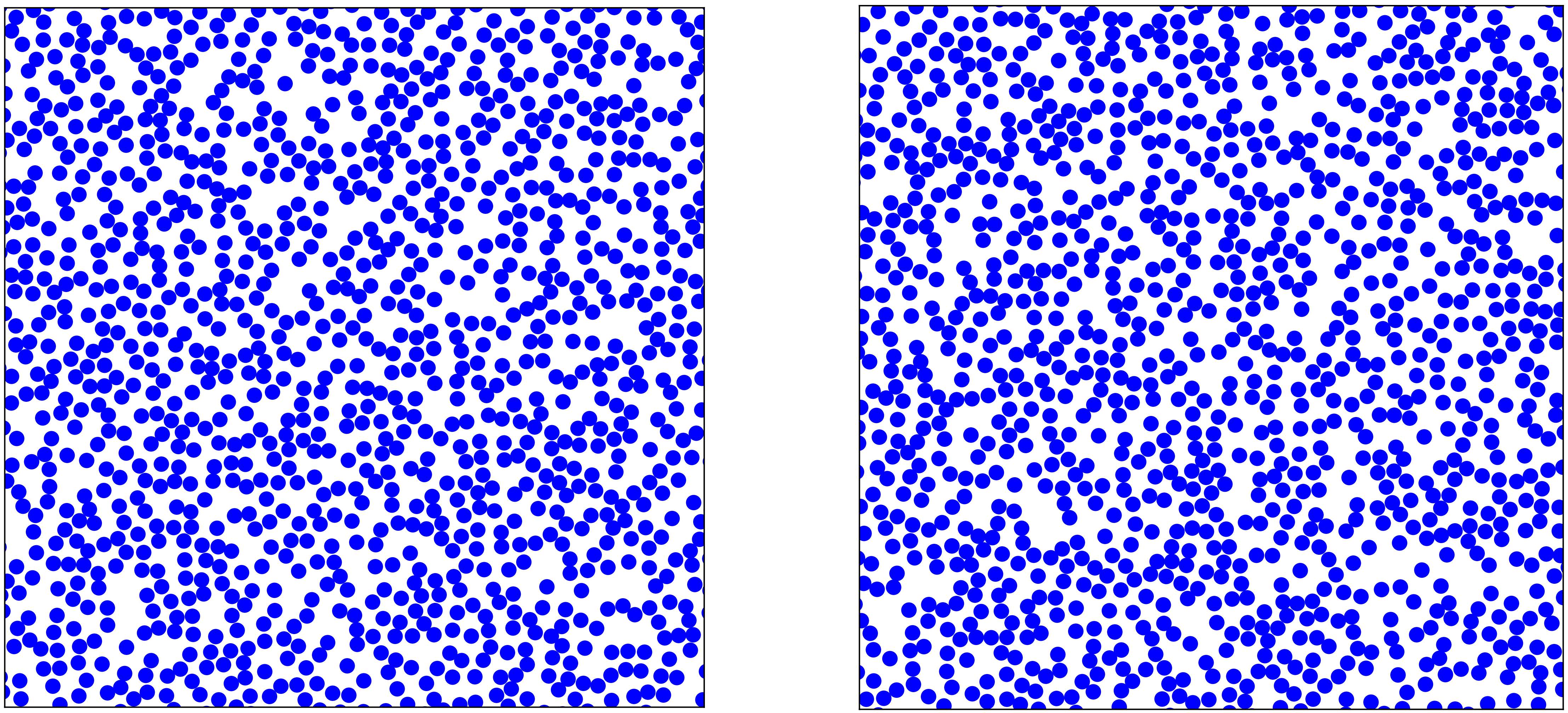
SOMETIMES IT'S EASY

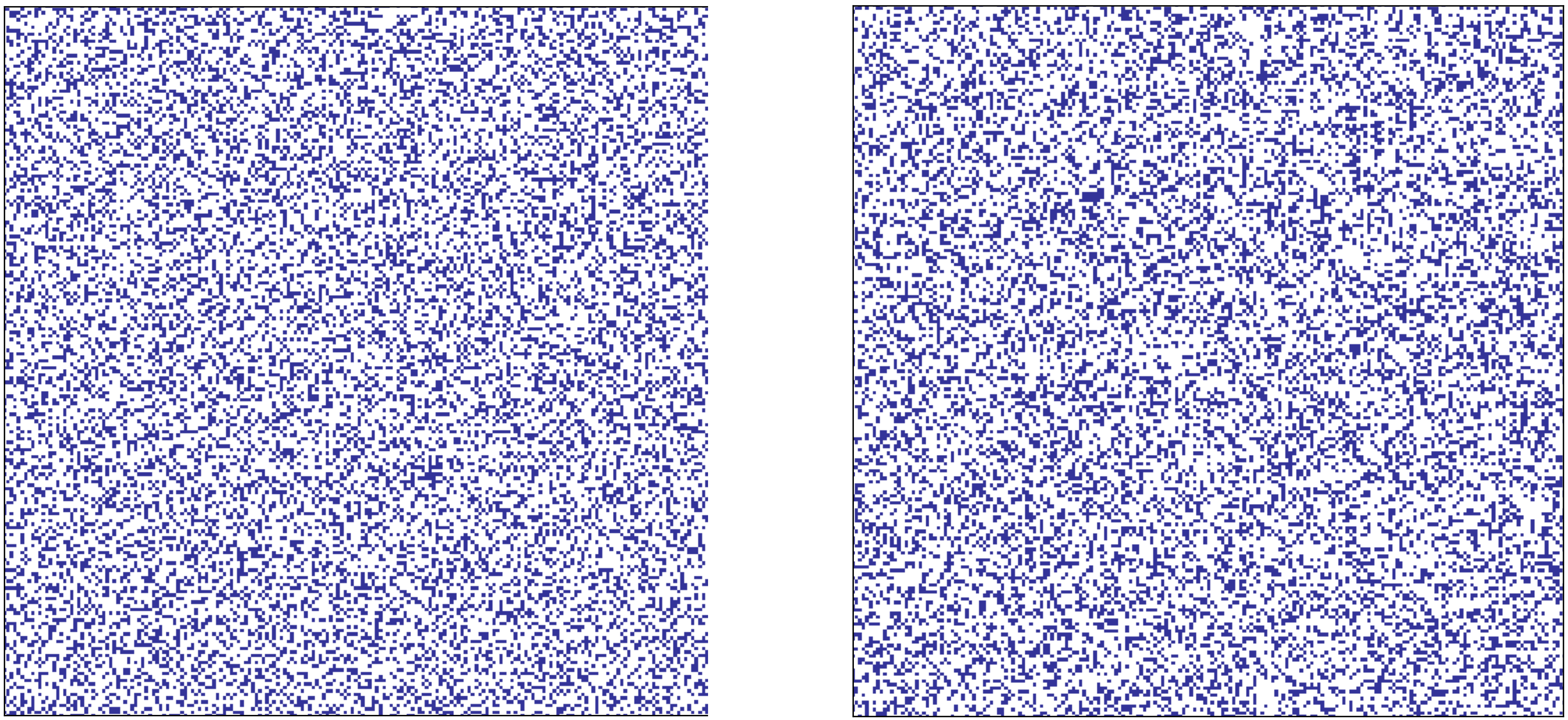


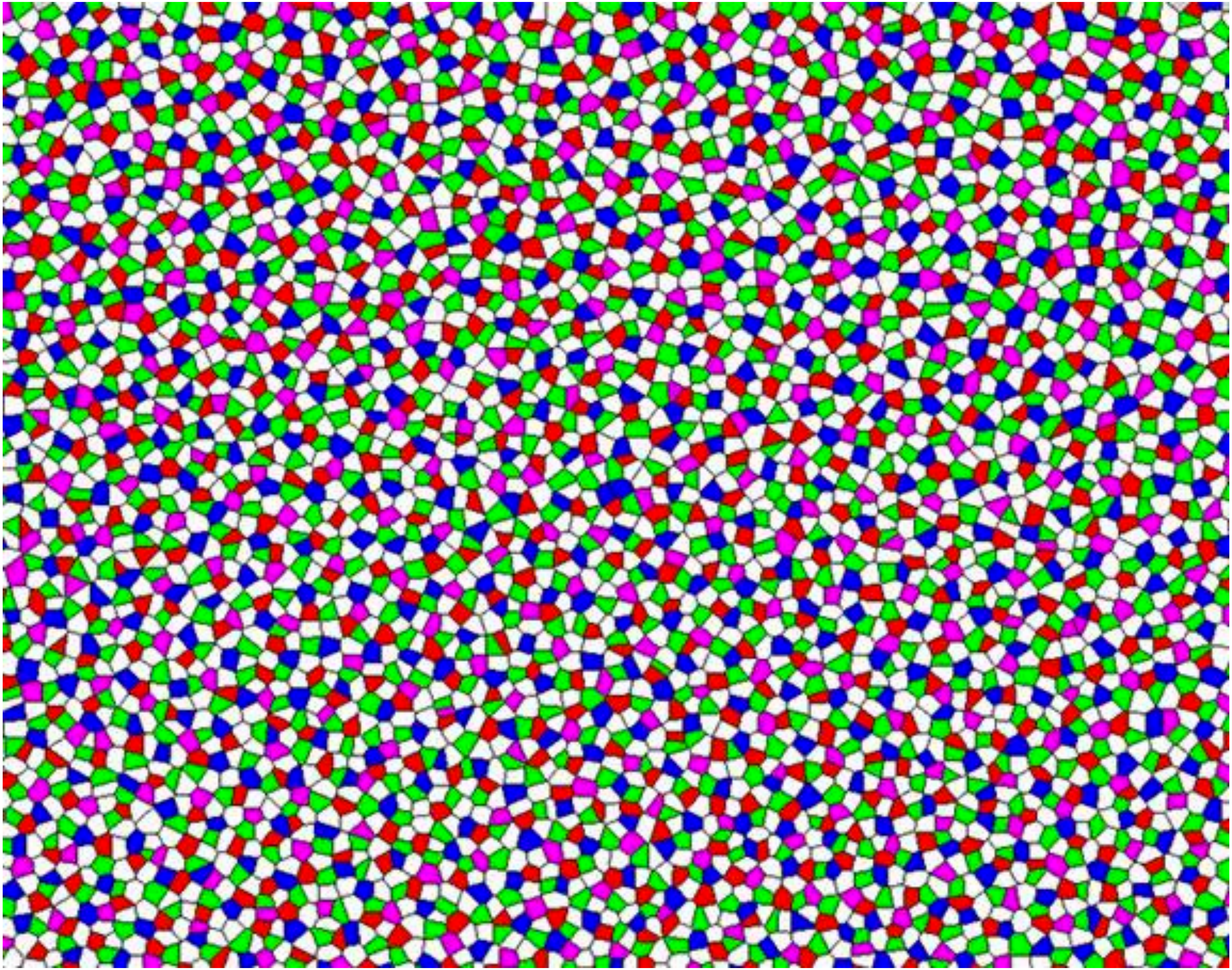



A
B
C
D

SOMETIMES IT'S
NOT QUITE
SO TRIVIAL







QUESTION: HOW CAN WE ASSESS THE EXTENT OF THE ORDERING?

MORE SPECIFICALLY, IF WE SUCCEED IN QUANTIFYING ORDER, WHAT DO WE HOPE TO MEASURE?

- TIME SCALES FOR ORGANIZATION
- POSITION & NATURE OF PHASE TRANSITIONS
- CORRELATION LENGTHS
- DIVERGENCES OF VARIOUS QUANTITIES & CRITICAL EXPONENTS
- NATURE & EXTENT OF SELF ORGANIZATION

THE QUESTION IS HOW TO DO THIS IN PRINCIPLE, AND WHETHER IT WORKS IN PRACTICE.

OK, LET'S CHANGE GEARS FOR A SECOND

Quora

Search for questions, people, and topics

Number Sequence Puzzles | Logic Puzzles | Puzzles and Trick Questions

What is the next number in the sequence 7, 2, 0, 7, 8, 4, 3?

Plaban Biswas, Game Developer, Rendered Ideas
Updated Mar 2 2017 · Author has **58** answers and **285.6k** answer views

Answer = **11**

Since,

The difference between a term and the *fourth* term from its position decreasing by **1.**

i.e.,

Sequence => 7 , 2 , 0 , 7 , 8 , 4 , 3 , ?

For Terms,

1st and 5th = 7 - 8 = **-1**

2nd and 6th = 2 - 4 = **-2**

3rd and 7th = 0 - 3 = **-3**

4th and 8th = 7 - ? = **-4** (By observation)

Therefore,

Missing no. (?) = 7 - (-4)

? = **11**

Tor Djärv, PhD student in Theoretical Nuclear Physics

Answered Mar 26 2017

The sequence is clearly approximated by the polynomial

$$p(n) = -\frac{2012572143228197}{74246277105000} + \frac{1792456010419157}{37123138552500}n + \frac{579311770487903}{37123138552500}n^2 - \frac{1647041896494713}{29698510842000}n^3 + \frac{842629883877863}{24748759035000}n^4 - \frac{107036193068797}{10999448460000}n^5 + \frac{268735884488563}{182760066720000}n^6 - \frac{21260170804811}{186343597440000}n^7 + \frac{2914278946885751}{810967336058880000}n^8$$

from which it follows that the next element is 42

Don Engelmeyer

Answered Sep 2, 2018

Originally Answered: What is the next number of 7, 2, 0, 7, 8, 4, 3, __?

The answer is irrelevant,useless and a waste of time.

THE POINT IS THAT
THE QUESTION IS
ILL-DEFINED

TO MAKE THIS POINT CLEAR, LET'S ASK A TRIVIAL QUESTION:

WHAT IS THE NEXT TERM IN THE SEQUENCE

1, 3, 5, ?

CLEARLY THE RIGHT ANSWER IS 7.

RIGHT?

ODD NUMBERS AND ALL THAT...

$$a(n) = 2n - 1 \qquad n = 1, 2, 3, \ldots$$

BUT WHAT'S WRONG WITH 8 ?

$$a(n) = \sum_{k=1}^{n} \left\lfloor \frac{n}{k} \right\rfloor$$ FLOOR FUNCTION

$$a(1) = 1 \qquad a(2) = \left\lfloor \frac{2}{1} \right\rfloor + \left\lfloor \frac{2}{2} \right\rfloor = 3$$

$$a(3) = \left\lfloor \frac{3}{1} \right\rfloor + \left\lfloor \frac{3}{2} \right\rfloor + \left\lfloor \frac{3}{3} \right\rfloor = 5$$

$$a(4) = \left\lfloor 4/1 \right\rfloor + \left\lfloor 4/2 \right\rfloor + \left\lfloor 4/3 \right\rfloor + \left\lfloor 4/4 \right\rfloor = 8$$

WHY IS 7 BETTER THAN 8?

THIS LEADS TO THE NOTION OF SIMPLICITY. IS

$a(n) = 2n-1$ SIMPLER THAN

$a(n) = \sum_{k=1}^{n} \lfloor n/k \rfloor$?

CAN "SIMPLICITY" BE QUANTIFIED?

KOLMOGOROV COMPLEXITY K

THE KOLMOGOROV COMPLEXITY OF A OF AN OBJECT (LINE OF TEXT, STRING OF NUMBERS, PICTURE, ...) IS THE LENGTH OF THE SHORTEST COMPUTER PROGRAM WHICH PRODUCES THE OBJECT AS ITS OUTPUT.

[KOLMOGOROV, CHAITIN] (1965)

ESSENTIALLY, THIS IS ONE QUANTIFICATION OF OCCAM'S RAZOR.

SO, LET'S LOOK AT SOME SIMPLE OBJECTS.

FOR SIMPLICITY'S SAKE, LOOK AT 1D OBJECTS: SEQUENCES OF 2 LETTERS.

I . . . . . ABABABAB . . . .

OBTAINED BY SUBSTITUTION :

$A \to AB$

$B \to AB$

PERIODIC

II . . . . ABABB ABABBABB ABABB . . . .

$A \to AB$

$B \to ABB$

QUASIPERIODIC

III ... ABBABAABBAABABBA ...

A → AB
B → BA

SINGULAR CONTINUOUS

IV ... |||T||T||||TTT|T|||T||T| ...

|| → |||T
|T → ||T|
T| → TT|T
TT → TTT|

ABSOLUTELY CONTINUOUS

INTERPRET AS SPIN $\frac{1}{2}$ (± 1).

$$g(j,k) = \langle \sigma_j \sigma_k \rangle = \delta_{jk}.$$

THESE ALL HAVE RATHER SMALL KOLMOGOROV COMPLEXITY.

II RANDOM COIN TOSS
BERNOULLI SEQUENCE (± 1)

FOR A GIVEN SUCH SEQUENCE, SAY

1 T 1 1 T T T 1 1 T 1 T 1 1 1 T T 1

SHORTEST PROGRAM IS

"PRINT 1 T 1 1 T T T 1 1 T 1 T 1 1 1 T T 1"

$K = N$ BITS + OVERHEAD

THE PROBLEM WITH KOLMOGOROV COMPLEXITY IS THAT IS NOT TYPICALLY COMPUTABLE. IF WE DO NOT KNOW HOW A SEQUENCE WAS GENERATED, WE CANNOT COMPUTE K.

SO FOR PHYSICS PROBLEMS, K IS NOT A USEFUL QUANTITY.

SHANNON ENTROPY $\mathbf{H}$

THE SHANNON ENTROPY OF A RANDOM VARIABLE X WITH PROBABILITY MASS FUNCTION $p(x)$ IS

$$H(X) = -\sum_{x} p(x) \log_2 p(x)$$

H IS A MEASURE OF THE AVERAGE UNCERTAINTY OF X.
H IS ALSO THE NUMBER OF BITS ON AVERAGE REQUIRED TO DESCRIBE X.

REMARKABLY, THE KOLMOGOROV COMPLEXITY & THE SHANNON ENTROPY ARE CLOSELY RELATED. CONSIDER A RANDOM VARIABLE X, AND DRAW IT N TIMES. NOW MAKE A STRING BY CONCATINATING THESE VALUES:

$$x_1 x_2 \cdots x_N$$

THEN

$$\lim_{N \to \infty} \frac{1}{N} K(x_1 x_2 \cdots x_N) = H(X)$$

WHAT DOES THIS HAVE TO DO WITH PHYSICS?

A PROCESS OR MODEL GENERATES AN ENSEMBLE OF MICROSTATES x_k WITH PROBABILITIES $p(x_k)$. THE RANDOM VARIABLE IS THE MICROSTATE.

NOTE THAT IN EQUILIBRIUM SM,

$$S = -k \sum_j p_j \ln p_j , \quad \text{WITH}$$

$$p_k = \frac{e^{-\beta E_k}}{Z}$$

(CANONICAL ENSEMBLE)

SO THE SHANNON ENTROPY REDUCES TO THE BOLTZMAN ENTROPY IN EQUILIBRIUM.

IN EQUILIBRIUM, THE ENTROPY IS A CENTRAL, FUNDAMENTAL QUANTITY.

IS THE SHANNON ENTROPY USEFUL OUT OF EQUILIBRIUM?

POSSIBILITIES

1. NO, IT'S NOT USEFUL
2. YES, IT IS USEFUL

LET'S BE OPTIMISTIC —
ASSUME POSSIBILITY 2.

BUT $H = -\sum p_k \log_2 p_k$,

AND OUT OF EQUILIBRIUM

WE DON'T KNOW p_k.

CAN WE APPROXIMATE H?

SO NOW WE NEED TO TALK ABOUT CODING.

FIRST, EXAMPLES:

① X = INTEGER BETWEEN 1 & 64, WITH EQUAL PROBABILITY $p = 1/64$.

THEN $H = -\sum p \log_2 p = -64 \cdot \frac{1}{64} \cdot \log(1/64)$
$= 6$ BITS

THIS IS TRIVIAL. WE'LL RETURN TO IT LATER.

② UNFAIR ROULETTE WHEEL WITH 8 POSSIBLE OUTCOMES; WITH PROBABILITIES:

OUTCOME	PROBABILITY
x_1	$1/2$
x_2	$1/4$
x_3	$1/8$
x_4	$1/16$
x_5	$1/64$
x_6	$1/64$
x_7	$1/64$
x_8	$1/64$

THE ENTROPY OF THIS PROCESS IS

$$H = \frac{1}{2}\log 2 + \frac{1}{4}\log 4 + \frac{1}{8}\log 8 + \frac{1}{16}\log 16 + 4 \times \frac{1}{64}\log 64$$

$$= 2 \text{ BITS}$$

QUESTION

SUPPOSE I AM THINKING OF AN INTEGER BETWEEN 1 & 8, WITH NO BIAS. YOU CAN ASK ME YES/NO QUESTIONS. WHAT IS YOUR BEST STRATEGY FOR GUESSING THE NUMBER?

ANSWER IS OBVIOUS: DIVIDE GROUP INTO TWO EQUAL PARTS:
"IS IT >4?", ETC...

QUESTION 2

SUPPOSE I AM THINKING OF A NUMBER BETWEEN 1 & 8, BUT I TELL YOU THE PROBABILITIES FOR MY CHOICE ARE

NUM.	PROB.
1	$1/2$
2	$1/4$
3	$1/8$
4	$1/16$
5	$1/64$
6	$1/64$
7	$1/64$
8	$1/64$

WHAT IS THE BEST STRATEGY NOW? HOW MANY QUESTIONS DO YOU NEED TO ASK ON AVERAGE?

GUESS! DIVIDE INTO GROUPS OF EQUAL TOTAL PROBABILITY.

SO, THE FIRST QUESTION WOULD BE:

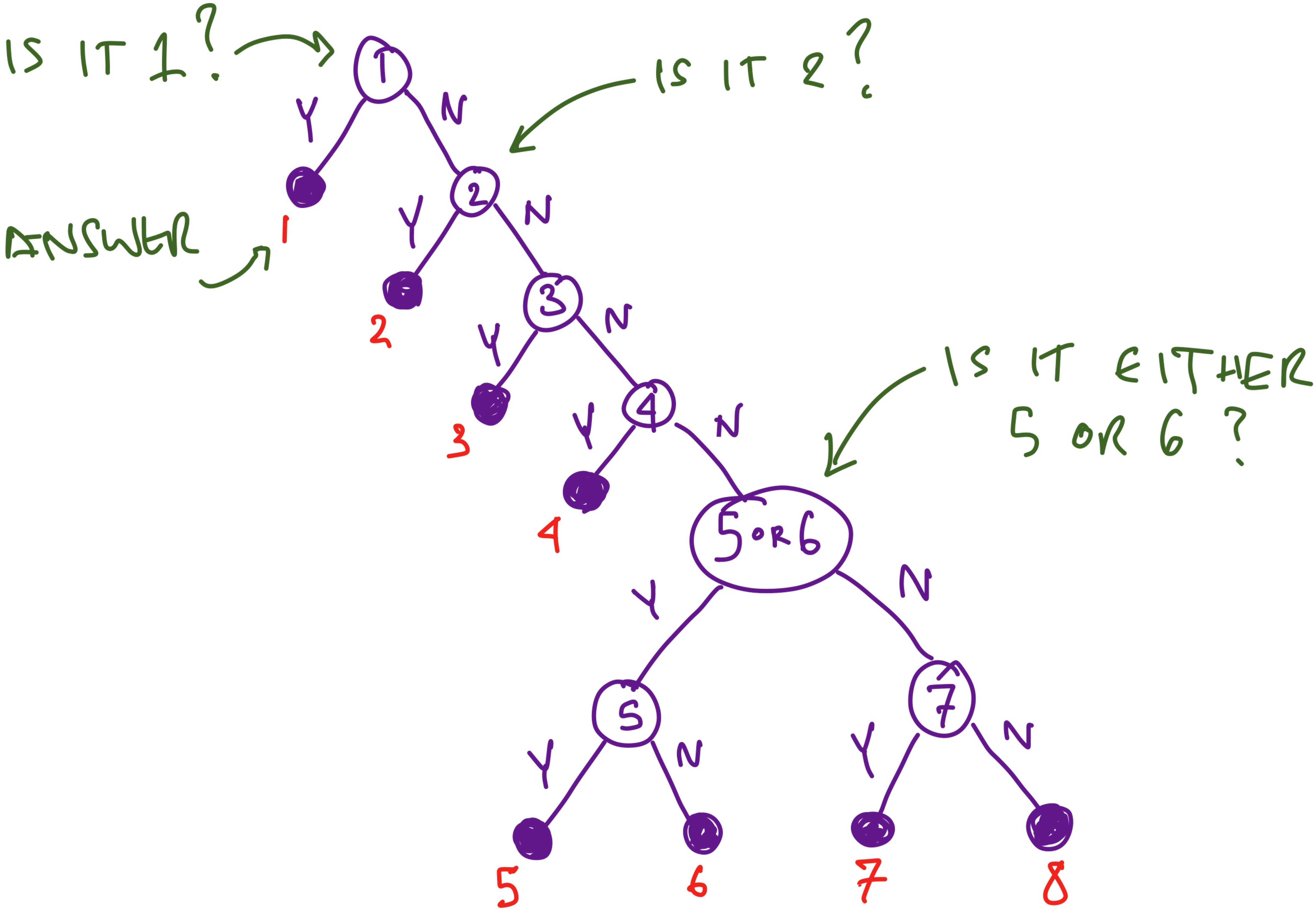
"IS IT 1?"

IF NO, THEN ASK

"IS IT 2"

etc.

GUESS TREE



HUFFMAN CODE

WE WANT TO SPIN OUR ROULETTE WHEEL MANY TIMES, AND TRANSMIT THE SEQUENCE OF OUTCOMES TO SOMEONE. IS THERE A CODE WE CAN DEVISE TO DO THIS EFFICIENTLY? LET'S USE BINARY CODES, FOR CONCRETENESS.

THERE ARE 8 POSSIBLE OUTCOMES, SO WE NEED 8 SYMBOLS. WE WANT OUR CODESTRING TO BE SHORT.

ALSO, WE'D LIKE TO BE ABLE TO DECODE THE INCOMING STRING AS IT ARRIVES.

SO A CODE LIKE

1	2	3	4	5	6	7	8
0	1	00	01	10	11	000	001

WOULD NOT WORK, SINCE THE OUTCOME STRINGS

342 AND 136

BOTH HAVE THE SAME CODE:

00011

FIRST GUESS: BINARY REP. OF (NUM − 1):

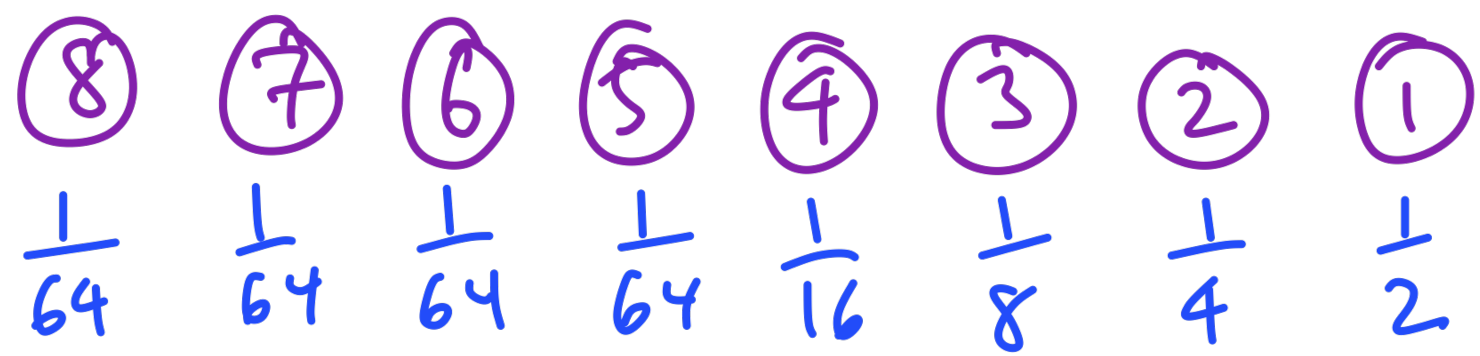
NUM.	CODE
1	000
2	001
3	010
4	011
5	100
6	101
7	110
8	111

$$\frac{\text{AVERAGE CODE LENGTH}}{\text{OUTCOME}} = 3 \text{ BITS}$$

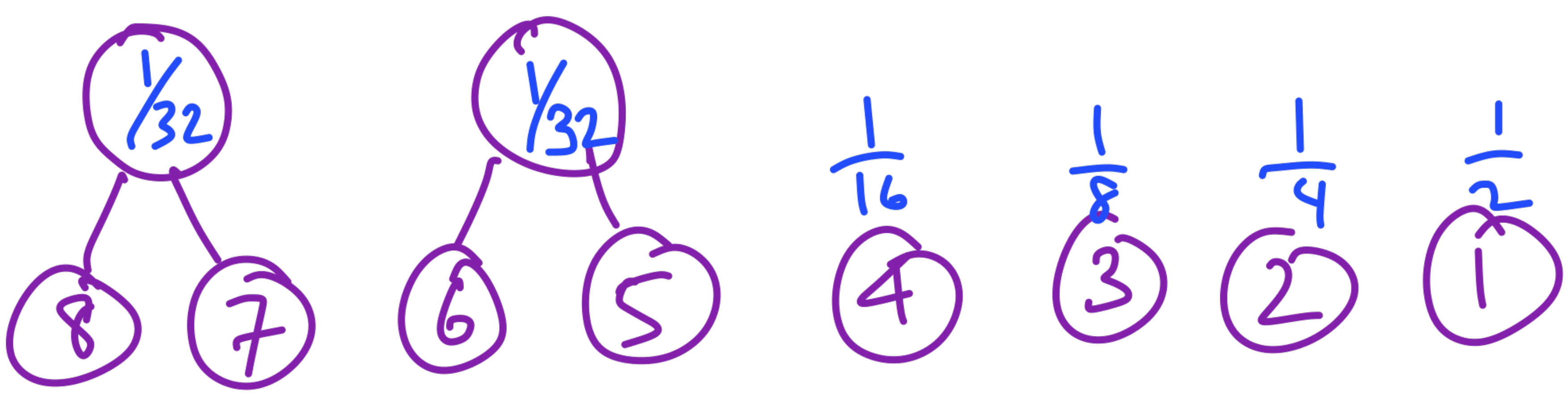
IS THERE A BETTER WAY ?

HUFFMAN (1952):

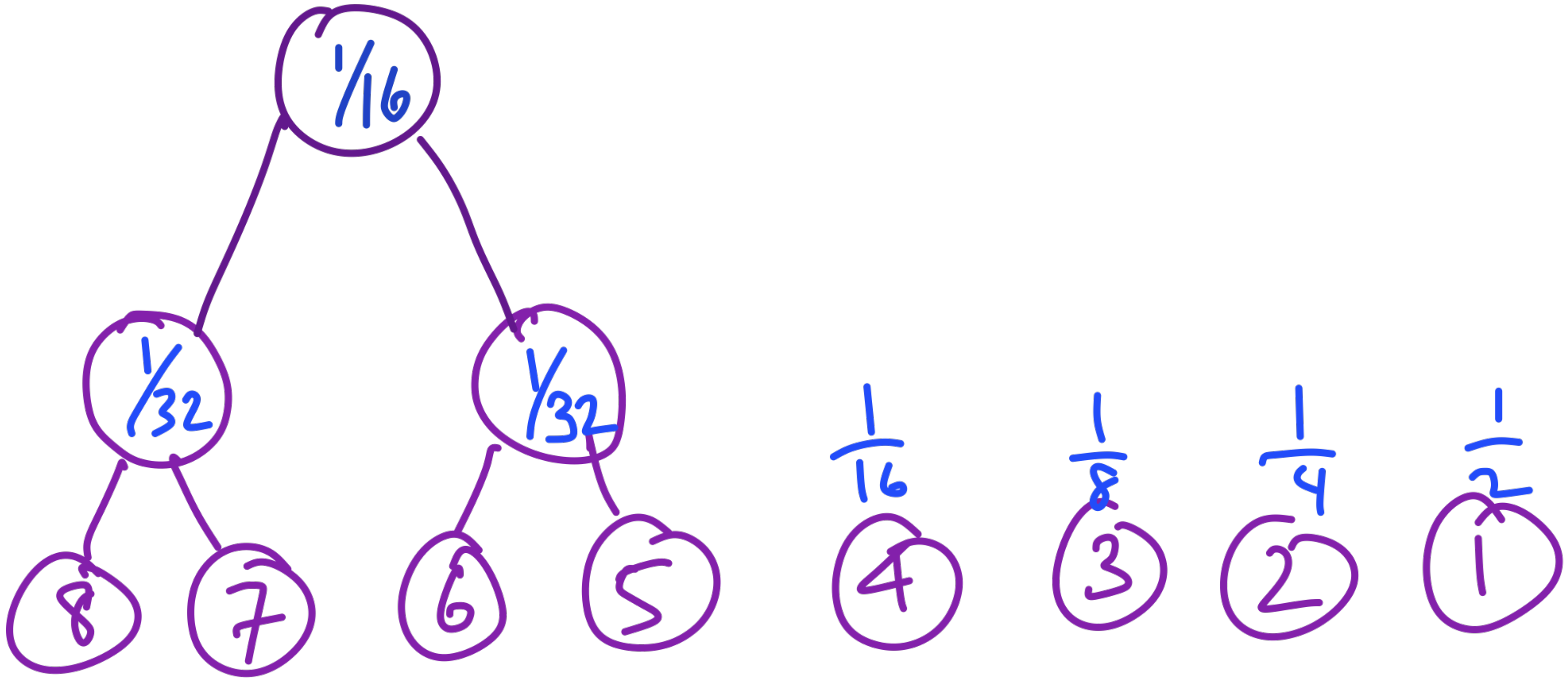
SORT OUTCOMES IN INCREASING p:



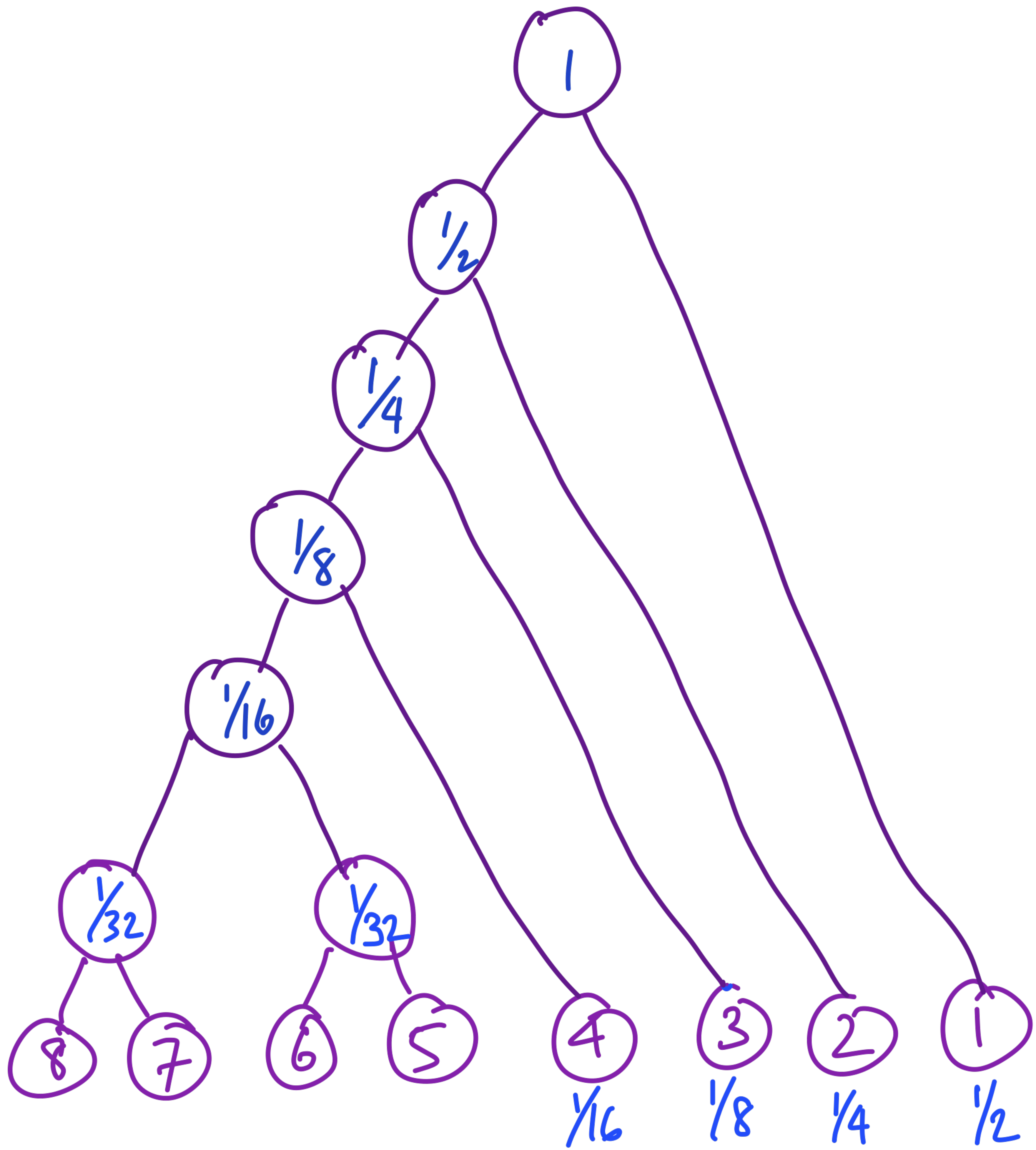
NOW, GROUP THE 2 LOWEST p OUTCOMES:
(HERE THERE ARE 2 SUCH PAIRS)



REPEAT:



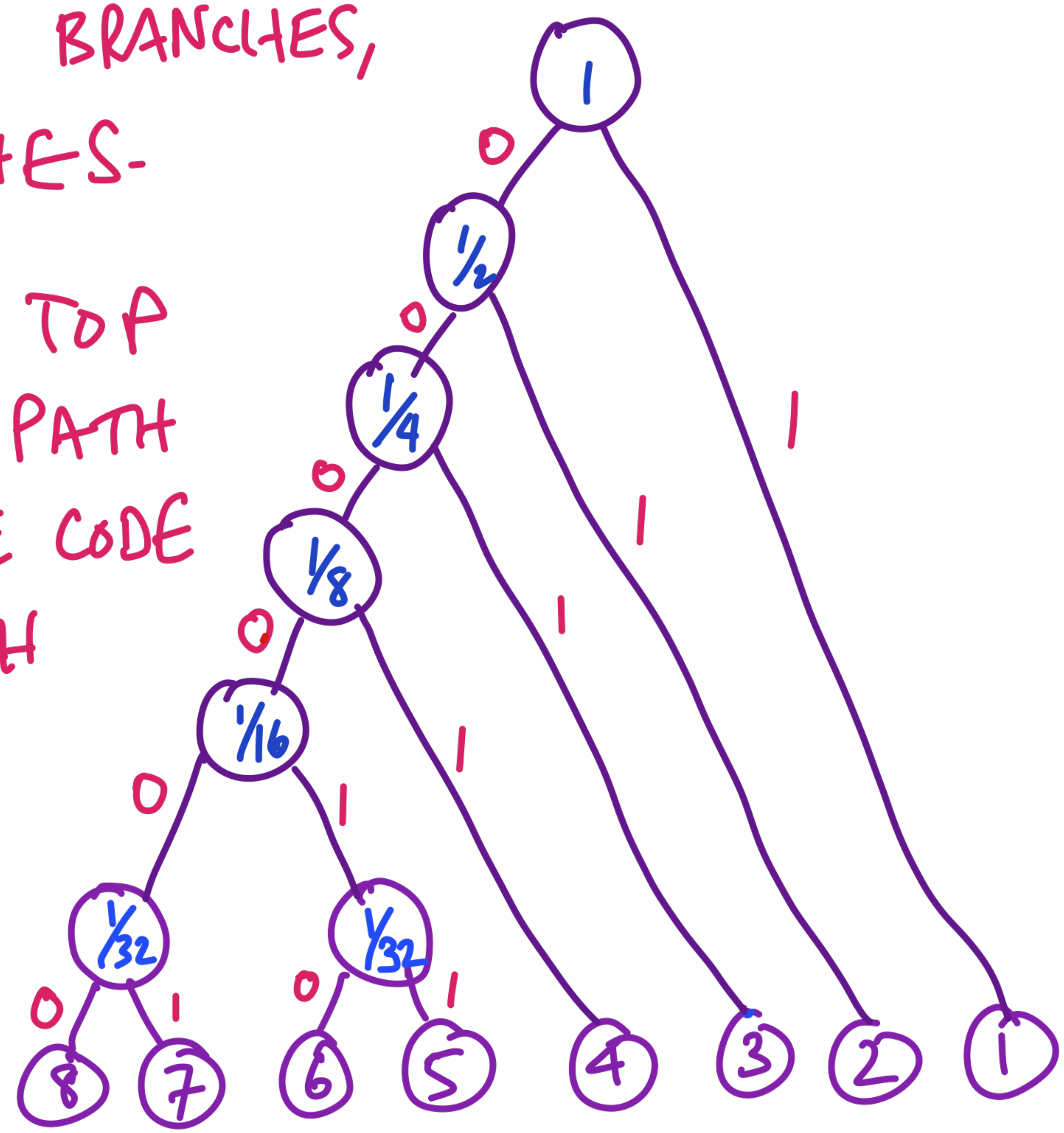
REPEAT PROCEDURE UNTIL ONLY ONE NODE REMAINS.



1
1/2
1/4
1/8
1/16
1/32
1/32
8
7
6
5
4
3
2
1
1/16
1/8
1/4
1/2

ASSIGN 0 TO LEFT BRANCHES,
1 TO RIGHT BRANCHES.

STARTING FROM THE TOP (ROOT), FOLLOW THE PATH TO THE BOTTOM. THE CODE WORDS ARE THE PATH SYMBOLS:



1	1	5	000011
2	01	6	000010
3	001	7	000001
4	0001	8	000000

SEEMS WORSE, SINCE THERE ARE 6 BIT CODEWORDS.

1	1	5	000011
2	01	6	000010
3	001	7	000001
4	0001	8	000000

BUT WHAT IS THE AVERAGE (EXPECTED) CODE WORD LENGTH?

$$L = \sum p_k \ell_k$$

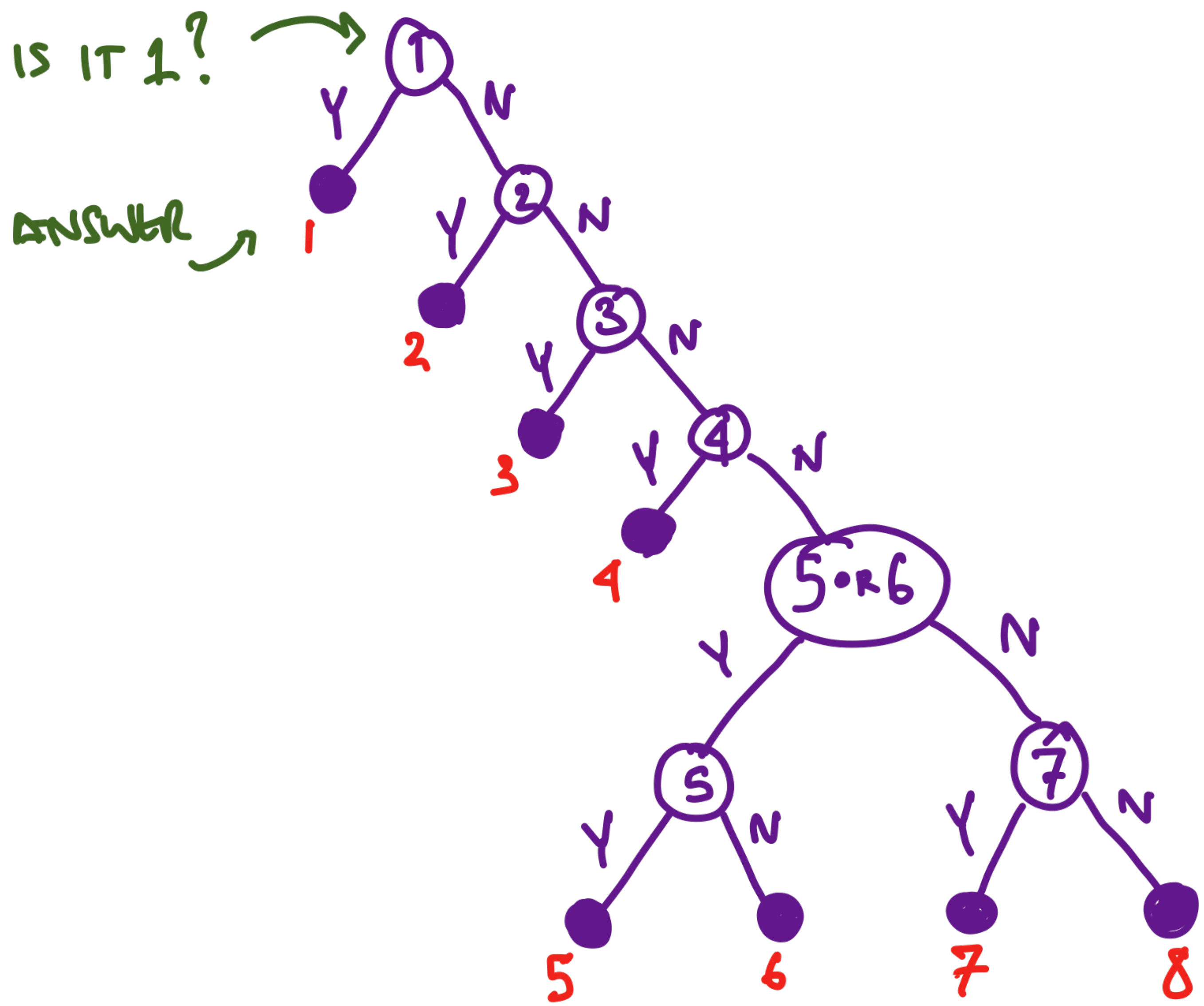
ℓ_k = LENGTH OF CODE WORD FOR k.

$$L = \frac{1}{2}\cdot 1 + \frac{1}{4}\cdot 2 + \frac{1}{8}\cdot 3 + \frac{1}{16}\cdot 4 + 4\cdot\frac{1}{64}\cdot 6$$

$$= \frac{1}{2} + \frac{1}{2} + \frac{3}{8} + \frac{1}{4} + \frac{3}{8} = 2 \text{ BITS.}$$

SO THIS IS BETTER THAN 1ST GUESS.

RECALL OUR GUESS TREE ...



ASSOCIATE Y WITH 1, N WITH 0:
SO SEQUENCE OF GUESSES LEADING TO, SAY
4 IS NNNY → 0001
7 NNNNNY → 000001

} SAME AS THE HUFFMAN CODE

NATURAL QUESTION : CAN WE DO BETTER ?

SHANNON'S SOURCE CODING THEOREM (1948) SAYS THAT THERE IS A LOWER LIMIT ON ANY ENCODING : THE AVERAGE CODEWORD LENGTH CANNOT BE SHORTER THAN THE SHANNON ENTROPY.

LET'S PROVE THIS, BUT FIRST, SOME DEFINITIONS AND AN INEQUALITY.

DEFINITION: A **SOURCE CODE** FOR A RANDOM VARIABLE X IS A MAPPING FROM $\mathcal{X}$, THE RANGE OF X, TO $\mathcal{D}^*$, THE SET OF FINITE LENGTH STRINGS OF SYMBOLS OF D LETTERS.

[WE WILL TAKE $D = 2$, 0 & 1, TYPICALLY]

$C(x)$ DENOTES THE CODEWORD CORRESPONDING TO x, AND $\ell(x)$ DENOTES THE LENGTH OF $C(x)$.

EXAMPLE: RANDOM VARIABLE IS COLOR OF PIXEL (RGB)

WITH $C(\text{RED}) = 00 \quad C(\text{GREEN}) = 10 \quad C(\text{BLUE}) = 11$

HERE $\mathcal{X} = (\text{RED}, \text{GREEN}, \text{BLUE})$ & $\mathcal{D} = \{0, 1\}$

DEFINITION: THE EXPECTED LENGTH $L(C)$ OF A SOURCE CODE $C(X)$ FOR A RANDOM VARIABLE X WITH PROBABILITY DISTRIBUTION $p(x)$ IS

$$L(C) = \sum_{x \in \mathcal{X}} p(x)\, l(x)$$

DEFINITION: A CODE IS NONSINGULAR IF DIFFERENT ELEMENTS OF $\mathcal{X}$ MAP INTO DIFFERENT STRINGS IN $\mathcal{D}^*$:

$$x \neq x' \implies C(x) \neq C(x')$$

DEFINITION: THE EXTENSION C^* OF A CODE C IS THE MAPPING OF FINITE LENGTH STRINGS OF $\mathcal{X}$ TO FINITE LENGTH STRINGS OF $\mathcal{D}$, DEFINED BY

$$C(x_1 x_2 \cdots x_m) = C(x_1)C(x_2) \cdots C(x_m)$$

(CONCATENATION)

EXAMPLE: IF $C(x_1) = 00$ $\quad C(x_2) = 11$

THEN $C(x_1 x_2 x_1 x_1) = C(x_1)C(x_2)C(x_1)C(x_1)$

$$= 00110000$$

DEFINITION: A CODE IS CALLED UNIQUELY DECODABLE IF ITS EXTENSION IS NON SINGULAR.

IN OTHER WORDS, ANY ENCODED STRING IN A UNIQUELY DECODABLE CODE HAS ONLY ONE POSSIBLE SOURCE STRING PRODUCING IT.

DEFINITION: A CODE IS CALLED A PREFIX CODE OR AN INSTANTANEOUS CODE IF NO CODEWORD IS A PREFIX OF ANY OTHER CODEWORD.

AN INSTANTANEOUS CODE CAN BE DECODED WITHOUT REFERENCE TO WHAT COMES LATER.

EXAMPLES

X	SINGULAR	NONSINGULAR BUT NOT UNIQUELY DECODABLE	UNIQUELY DECODABLE BUT NOT INSTANTANEOUS	INSTANTANEOUS
1	0	0	10	0
2	0	010	00	10
3	0	01	11	110
4	0	10	110	111

STRING 010 HAS 3 POSSIBLE SOURCE WORDS: 2, 14, 31

CAN ULTIMATELY DECODE 1100, BUT UNTIL FINAL 0, DO NOT KNOW WHETHER STRING BEGINS WITH 3 OR 4

HUFFMAN CODES ARE INSTANTANEOUS (PREFIX) CODES.

KRAFT INEQUALITY (THEOREM)

FOR ANY INSTANTANEOUS CODE OVER AN ALPHABET OF SIZE D, THE CODEWORD LENGTHS $l_1\ l_2 \ldots l_m$ MUST SATISFY THE INEQUALITY

$$\boxed{\sum_j D^{-l_j} \leq 1}$$

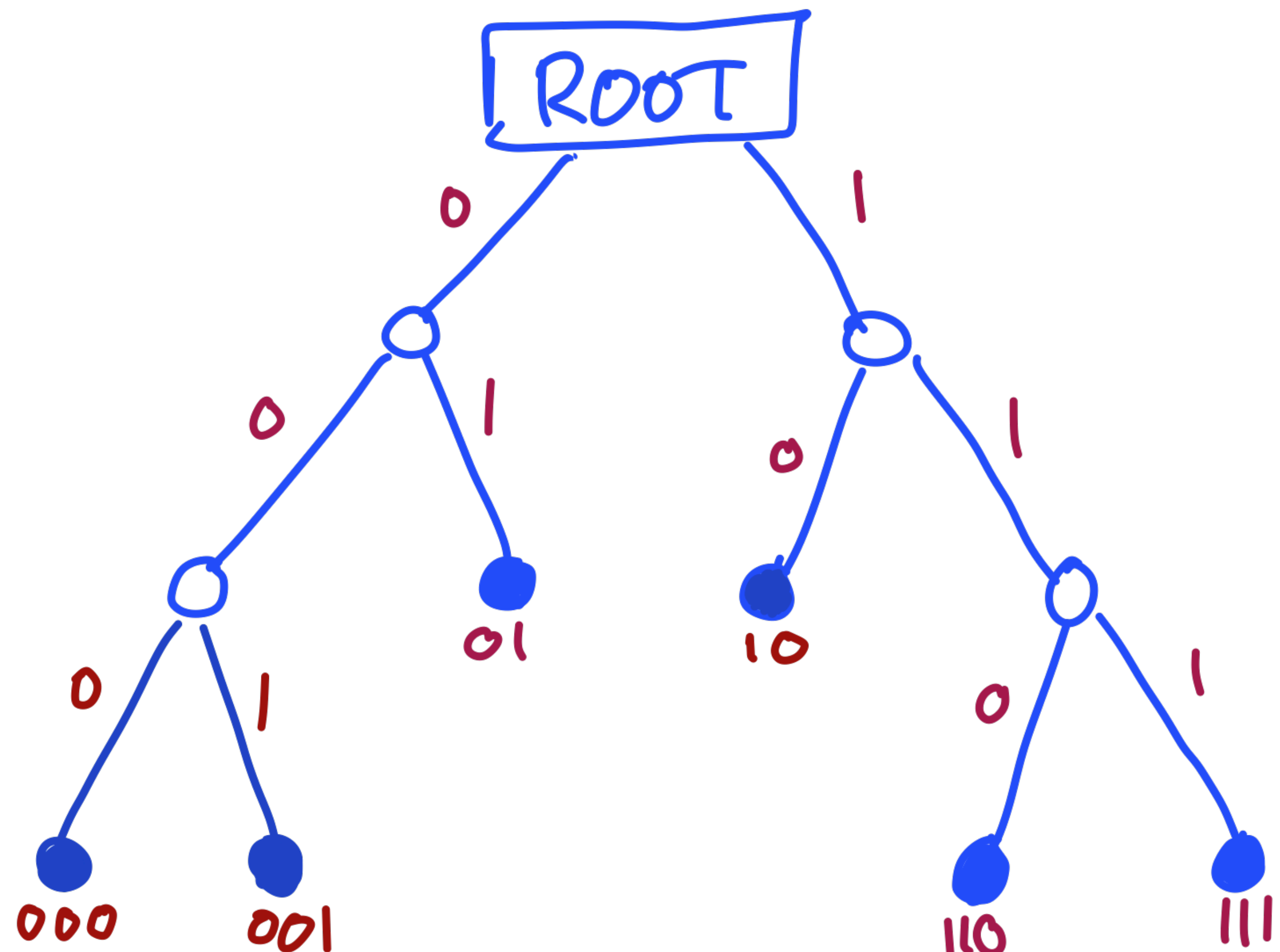
IN PARTICULAR, IF $D=2$ (BINARY CODE OF 0 & 1),

$$\boxed{\sum_j 2^{-l_j} \leq 1}$$

MOREOVER, IF A GIVEN SET OF CODEWORD LENGTHS SATISFIES THE INEQUALITY, THEN THERE EXISTS AN INSTANTANEOUS CODE WITH THESE WORD LENGTHS.

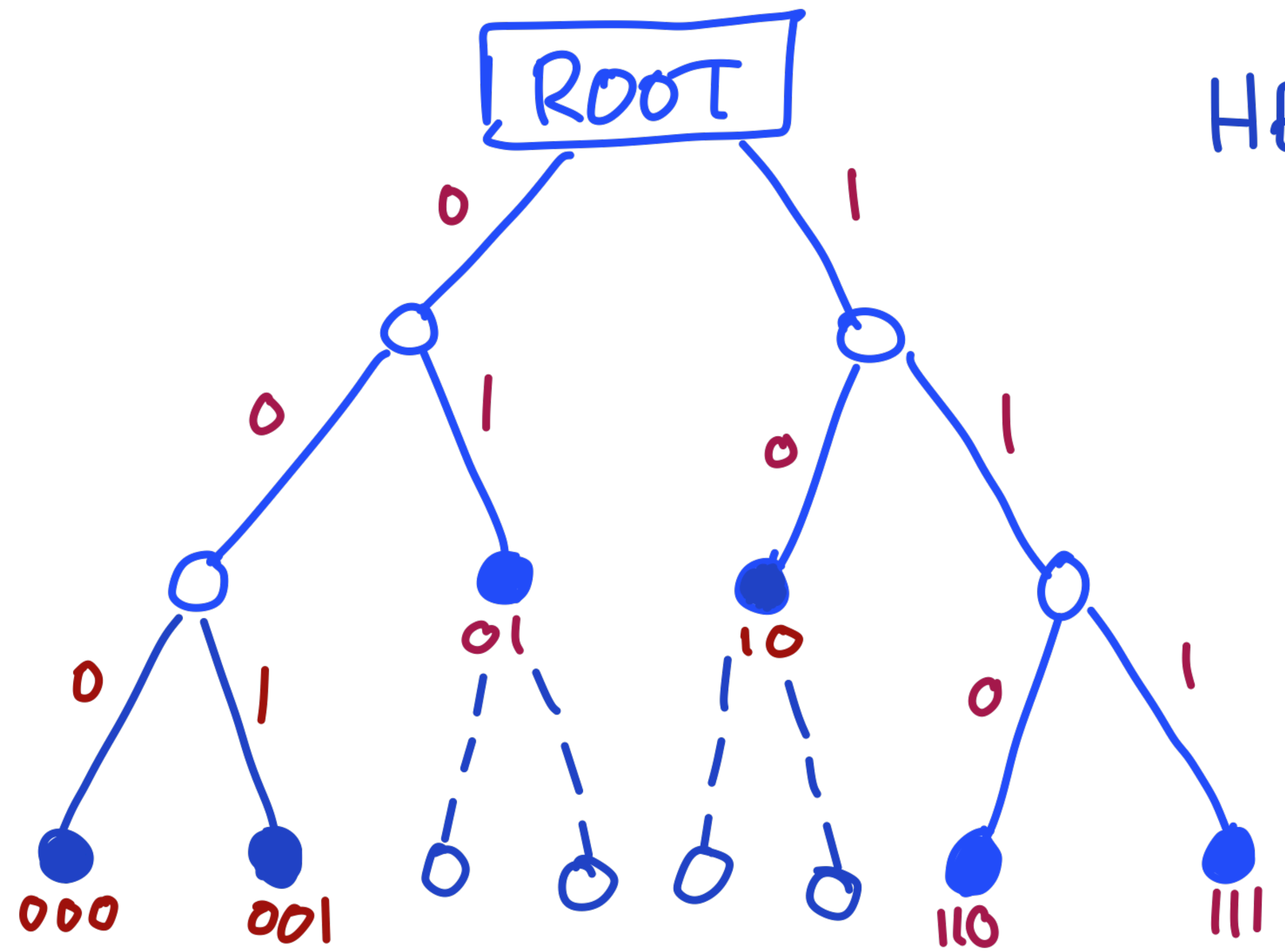
PROOF (D=2 ; GENERALIZES TRIVIALLY)

CONSIDER A BINARY TREE. EACH NODE HAS 2 CHILDREN. LET THE BRANCHES OF THE TREE REPRESENT THE SYMBOLS (0,1) OF THE CODEWORD.



EACH CODEWORD IS REPRESENTED BY A LEAF ●, WHERE A BRANCH TERMINATES. BECAUSE OF THE PREFIX CONDITION, A LEAF HAS NO DESCENDENTS.

LET l_{MAX} = LENGTH OF THE LONGEST CODEWORD.
CONSIDER ALL NODES AT THIS LEVEL. SOME ARE CODEWORDS, SOME ARE NOT.



HERE, $l_{MAX} = 3$

A CODEWORD AT LEVEL l_k HAS $2^{l_{MAX} - l_k}$ DESCENDENT NODES AT LEVEL l_{MAX}. ALL THESE DESCENDENT NODES FORM DISJOINT SETS. MOREOVER, THE NUMBER OF NODES IN THESE SETS MUST BE LESS THAN OR EQUAL TO $2^{l_{MAX}}$.

So: $\sum_1 2^{l_{max} - l_K} \leq 2^{l_{max}}$

OR

$$\sum_1 2^{-l_K} \leq 1$$

THIS IS THE KRAFT INEQUALITY.

NOW THE CONVERSE: GIVEN ANY SET OF CODEWORD LENGTHS $l_1\, l_2 \cdots l_m$ THAT SATISFY THE KRAFT INEQUALITY, WE CAN ALWAYS CONSTRUCT A TREE LIKE THE ONE WE JUST STUDIED. LABEL THE FIRST ONE OF DEPTH l_1 AS CODEWORD 1, AND REMOVE ALL OF ITS DESCENDENTS.

THEN LABEL THE FIRST REMAINING NODE OF LENGTH l_2 AS CODEWORD 2, AND REMOVE ITS DESCENDENTS. CONTINUE THIS PROCESS UNTIL THERE ARE NO CODEWORDS LEFT.

IN THIS WAY WE WILL GENERATE A PREFIX CODE WITH THE SPECIFIED $l_1, l_2 \cdots l_m$.

WE SHALL SOON SEE WHY THIS RESULT IS IMPORTANT.

OPTIMAL CODES

LET'S NOW CONCERN OURSELVES WITH FINDING THE PREFIX CODE WITH THE MINIMUM EXPECTED LENGTH.

THIS IS EQUIVALENT TO FINDING THE LENGTHS $l_1, l_2 \cdots l_m$ WHICH SATISFY THE KRAFT INEQUALITY AND WHOSE EXPECTED LENGTH

$$L = \sum_i p_k l_k$$

IS LESS THAN FOR ANY OTHER PREFIX CODE.

THIS IS JUST A MINIMIZATION PROBLEM:

MINIMIZE $L = \sum p_k l_k$ OVER ALL INTEGERS l_k SATISFYING

$$\sum 2^{-l_k} \leq 1$$

TO DO THIS, FIRST RELAX THE INTEGER CONDITION & ASSUME EQUALITY FOR THE CONSTRAINT:

$$\sum 2^{-l_k} = 1$$

THIS IS JUST A LAGRANGE MULTIPLIER PROBLEM

$$J = \sum p_k l_k + \lambda \left(\sum 2^{-l_k} \right)$$

SETTING $\frac{\partial J}{\partial l_k} = 0$ GIVES

$$\frac{\partial J}{\partial l_k} = p_k - \lambda 2^{-l_k} \ln 2 = 0$$

$$\Rightarrow 2^{-l_k} = \frac{p_k}{\lambda \ln 2}$$

SUBSTITUTE THIS INTO THE CONSTRAINT $\sum 2^{-l_k} = 1$ TO GET

$$\frac{1}{\lambda \ln 2} \sum p_k = 1$$

So $\lambda = 1/\ln 2$

And

$$p_k = 2^{-l_k}.$$

Thus, the optimal code lengths are

$$l_k^* = -\log_2 p_k$$

The expected codeword length for these (non-integer) l_k is

$$L^* = \sum p_k l_k^* = -\sum_i p_k \log_2 p_k = \underline{\underline{H(X)}}$$

The Shannon Entropy!

THIS BRINGS US TO ONE OF THE MOST IMPORTANT RESULTS OF INFORMATION THEORY, SHANNON'S SOURCE CODING THEOREM:

THEOREM: THE EXPECTED LENGTH L OF ANY BINARY PREFIX CODE FOR A RANDOM VARIABLE X IS GREATER THAN OR EQUAL TO THE SHANNON ENTROPY:

$$\boxed{L \geq H(X)}$$

EQUALITY HOLDS ONLY IF $p_k = 2^{-l_k}$.

THIS MEANS THAT THE SHANNON ENTROPY IS A LOWER BOUND ON THE SHORTEST ENCODING.

PROOF: THE DIFFERENCE BETWEEN THE EXPECTED LENGTH AND THE ENTROPY IS

$$L - H(X) = \sum p_k l_k + \sum p_k \log_2 p_k$$

$$= -\sum p_k \log_2 2^{-l_k} + \sum p_k \log_2 p_k$$

DEFINE $r_k \equiv \dfrac{2^{-l_k}}{\sum 2^{-l_k}}$, AND $C \equiv \sum 2^{-l_k}$

$\left[\sum r_k = 1 \text{, SO } r_k \text{ IS A PROBABILITY MEASURE} \right]$

THEN $\quad L-H = -\sum p_k \log 2^{-l_k} + \sum p_k \log p_k$

$$= -\sum_k p_k \left[\log r_k + \log c\right] + \sum_k p_k \log p_k$$

$$= \sum_k p_k \log\left(\frac{p_k}{r_k}\right) - \log c$$

BUT $\quad c = \sum 2^{-l_k} \leq 1$, SO $\log c \leq 0$

AND

$$\sum p_k \log\left(\frac{p_k}{r_k}\right) \geq 0\,; \quad *$$

SO

$$L-H \geq 0$$

* $$\sum p_j \log\left(\frac{p_j}{r_j}\right) \geq 0 \Rightarrow \sum p_j \log\left(\frac{r_j}{p_j}\right) \leq 0$$

THE PROOF THAT $\sum p_j \log\left(\frac{r_j}{p_j}\right) \leq 0$ USES JENSEN'S INEQUALITY :

$$\langle f(x)\rangle \geq f(\langle x\rangle) \quad \text{FOR } f(x) \text{ CONVEX}$$

$$\sum p_j \log\left(\frac{r_j}{p_j}\right) \leq \log \sum p_j \frac{r_j}{p_j} = \log \overbrace{\sum r_j}^{=1} = 0$$

HOW WELL CAN WE DO? CAN WE EVER GET CLOSE TO THE BOUND? IS THERE ANY CODE WHICH COMES CLOSE?

IN FACT, THERE EXISTS A CODE FOR WHICH $H \leq L \leq H+1$

TO SEE THIS, RECALL DISCUSSION ABOUT OPTIMAL CODES, WHERE WE SOUGHT TO MINIMIZE $L = \sum p_j l_j$ SUBJECT TO THE CONDITIONS OF INTEGER l_j, & $\sum 2^{-l_j} \leq 1$.

WE FOUND THAT THE OPTIMAL CODEWORD LENGTHS WERE

$$l_j = \log_2 \left(\frac{1}{p_j} \right)$$

WHICH GIVES $L = H$.

BUT $\log_2\left(\frac{1}{p_j}\right)$ MAY NOT ALL BE INTEGERS (IF ANY ARE AT ALL). LET'S FIX THIS BY ROUNDING UP TO THE NEAREST INTEGER AND CONSIDER

$$l_j = \left\lceil \log_2\left(\frac{1}{p_j}\right) \right\rceil$$

THESE l_j SATISFY THE KRAFT INEQUALITY, SINCE

$$\sum 2^{-\left\lceil \log\left(\frac{1}{p_j}\right) \right\rceil} \leq \sum 2^{-\log\left(\frac{1}{p_j}\right)} = \sum p_j = 1$$

CLEARLY, THESE l_j SATISFY

$$\log_2\left(\frac{1}{p_j}\right) \leq l_j \leq \log_2\left(\frac{1}{p_j}\right) + 1$$

$$\log_2\left(\frac{1}{p_j}\right) \leq l_j \leq \log_2\left(\frac{1}{p_j}\right) + 1$$

MULTIPLY BY p_j & SUM OVER j:

$$-\sum p_j \log_2 p_j \leq \sum p_j l_j \leq -\sum p_j \log p_j + \underbrace{\sum p_j}^{1}$$

OR

$$H \leq L \leq H+1$$

SINCE AN OPTIMAL CODE CAN ONLY BE BETTER THAN THIS CODE, ANY OPTIMAL CODE SATISFIES

$$H \leq L^* \leq H+1 .$$

SO, COMPUTING THE AVERAGE WORD LENGTH OF AN OPTIMAL CODE WILL GIVE AN EXCELLENT ESTIMATE OF THE SHANNON ENTROPY.

SUPPOSE WE KNOW THE CODEWORD FOR A SINGLE MICROSTATE IN AN OPTIMAL CODE. CAN WE LEARN ANYTHING FROM THIS?

PERHAPS SURPRISINGLY, THE ANSWER IS YES. TO SEE THIS, LET'S CONSIDER A MICROSTATE IN THE CANONICAL ENSEMBLE, WHOSE PROBABILITY OF OCCURENCE IS

$$P_j = \frac{1}{Z} e^{-\beta E_j}$$

THE CODEWORD FOR THIS MICROSTATE WILL BE OF LENGTH l_j, WHERE

$$l_j = -\log_2 p_j$$ FOR AN OPTIMAL BINARY CODE.

THAT IS, $$l_j = -\log_2\left(e^{-\beta E_j}\right) + \log_2 Z$$

AND, SINCE $\log_2 x = \ln x / \ln 2$,

$$l_j = \frac{1}{\ln 2}\left[-\ln e^{-\beta E_j} + \ln Z\right] = \frac{1}{\ln 2}\left[\beta E_j + \ln Z\right]$$

NOW THE FREE ENERGY IS $F = -kT \ln Z$, SO

$$l_j = \frac{\beta}{\ln 2}\left[E_j - F\right]$$

Since $F = \bar{E} - TS$,

$$l_j = \frac{\beta}{\ln 2}\left[TS + (E_j - \bar{E})\right] = \frac{1}{\ln 2}\left[\frac{S}{k} + \beta(E_j - \bar{E})\right]$$

NOW FOR A TYPICAL MICROSTATE, $E_j - \bar{E}$ SHOULD BE SMALL, WITH

$$\frac{\bar{E} - E_j}{kT} \sim \frac{\sqrt{\langle E^2\rangle - \langle E\rangle^2}}{kT} \sim \sqrt{\frac{C_v}{k}} \sim \sqrt{N}$$

So $\quad l_j \sim \left[\frac{S}{k} + O(\sqrt{N})\right] \qquad (N = \#\ \text{DOF's})$

SINCE $S \sim N$,

$$\frac{l_j}{N} \sim \frac{A}{k} + O\left(\frac{1}{\sqrt{N}}\right) \qquad \left[A \equiv S/N\right]$$

THUS, KNOWLEDGE OF A CODEWORD OF A TYPICAL MICROSTATE GIVES A GOOD ESTIMATE OF S, IF THE CODE IS OPTIMAL.

THE QUESTION THEN BECOMES — CAN WE FIND AN OPTIMAL CODE?

AS IT HAPPENS, THE ANSWER IS YES, USING DATA COMPRESSION ALGORITHMS.

DATA COMPRESSION ALGORITHMS TAKE A STRING OF DATA AND RETURN A CODEWORD. FOR US, THE DATA STRING IS A MICROSTATE OF A SYSTEM, EITHER IN EQUILIBRIUM, OR, MORE INTERESTINGLY, OUT OF EQUILIBRIUM.

WE HAVE JUST SEEN THAT THE CODEWORD OF A TYPICAL MICROSTATE GIVES US AN EXCELLENT ESTIMATE OF THE SHANNON ENTROPY. WE SHOWED THIS FOR EQUILIBRIUM SYSTEMS. WE MAY HOPE THAT IT IS TRUE FOR NON-EQUILIBRIUM MANY-BODY SYSTEMS AS WELL.

FOR THIS TO HAVE A PRAYER,

MICROSTATES OF THE NON-EQUILIBRIUM SYSTEM MUST HAVE A SENSE OF TYPICALITY.

AT LEAST FOR THE SYSTEMS WE HAVE STUDIED, THIS SEEMS TO BE TRUE.

WE WILL GET TO COMPRESSION ALGORITHMS PRESENTLY, BUT BEFORE WE DO, LET'S ASK WHAT WE CAN HOPE FOR, AND WHERE THINGS CAN GO WRONG.

I. SYSTEMS HAVE TYPICAL STATES

II. THE LENGTHS OF COMPRESSED MICROSTATES GIVES A GOOD ESTIMATE OF THE SHANNON ENTROPY H OF THE ENSEMBLE.

III. H IS A RELEVANT, INTERESTING MEASURE FOR, AND GIVES USEFUL INFORMATION ABOUT AT LEAST SOME NON-EQUILIBRIUM SYSTEMS. (WE KNOW IT DOES FOR EQUILIBRIUM.)

IV. AS A CONTROL PARAMETER (TEMPERATURE, DENSITY, ...) IS VARIED, THE APPROACH IS SENSITIVE ENOUGH TO SEE CHANGES: THE SIGNAL IS NOT WIPED OUT BY THE NOISE.

NONE OF THESE POINTS ARE OBVIOUSLY TRUE.

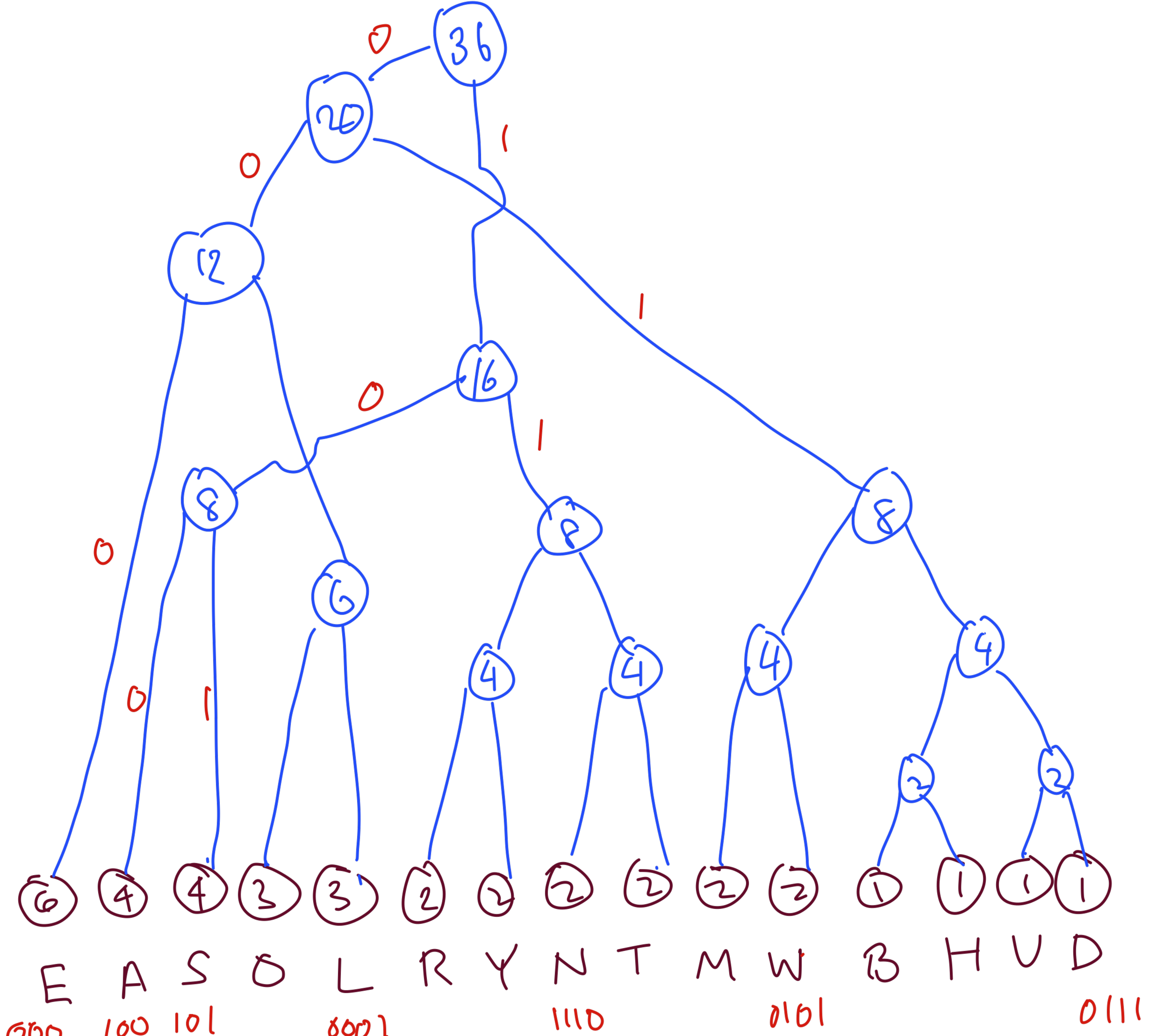
SPOILER ALERT: IT ALL SEEMS TO WORK OUT.

WE HAVE ALREADY DISCUSSED THE HUFFMAN CODES. BUT TO CONSTRUCT A HUFFMAN CODE, WE MUST HAVE A-PRIORI KNOWLEGE OF THE OCCURENCE FREQUENCIES OF THE OUTCOMES.

FOR EXAMPLE, TO CONSTRUCT A HUFFMAN CODE FOR THE LETTERS OF THE PHRASE: "A ROSE BY ANY OTHER NAME WOULD SMELL AS SWEET", WE USE THE FREQUENCY OF OCCURENCE OF THE LETTERS:

A: 4
R: 2
O: 3
S: 4
E: 6
B: 1
Y: 2
N: 2

T: 2
H: 1
M: 2
W: 2
U: 1
L: 3
D: 1



36
20
12
16
8
8
8
6
4
4
4
4
2
2
6 4 4 3 3 2 2 2 2 2 2 1 1 1 1
E A S O L R Y N T M W B H U D
000
100
101
0001
1110
0101
01111

WHAT CAN WE DO IF WE DO NOT KNOW THE PROBABILITIES? HOW CAN WE CONSTRUCT CODEWORDS? IS THE RESULTANT CODE OPTIMAL?

LEMPEL-ZIV ALGORITHMS

"CODE AS YOU GO..." MAKES A DICTIONARY AS IT READS A DATA STRING.

L-Z CODES ARE ASYMPTOTICALLY OPTIMAL (IN THE THERMODYNAMIC LIMIT)

LZ78

A, B, A B, B A, B A B, B A B B, A B A, B B, A B A B, B A B B A, B A B B

WORD	WORD NUMBER	CODE FOR WORD #
A	1	0A
B	2	0B
AB	3	1A
BA	4	2A
BAB	5	4B
BABB	6	5B
ABA	7	3A
BB	8	2B
ABAB	9	7B
BABBA	10	6A
BABB	11	6

THE CODEWORD FOR THIS STRING IS, THEN, SOMETHING LIKE

0A0B1A2A4B5B3A2B7B6A6

THE CHARACTERS A & B CAN BE CODED IN 1 BIT (0 OR 1). THE NUMBERS REFERRING TO THE DICTIONARY REFERRALS CAN BE LABELED FROM 1 TO C, WHERE C = # WORDS IN THE DICTIONARY. IT TAKES $\log_2 C$ BITS TO WRITE THESE NUMBERS, AT MOST. SO THE LENGTH OF THE CODE WORD IS

$$\ell \sim c \log_2 C + C$$

WE CANNOT COMPRESS A RANDOM STRING OF 0'S AND 1'S.