CS124 **How to get 1 fair flip ?** (efficiently) Lecture 1

Biased coin, get unbiased

symmetry: heads p independent
tails $1-p$

flip until HT or TH → NOPE, no symmetry
$p(1-p)$ $(1-p)p$

flip 2 coins, if HT I win, TH you win
otherwise

efficiency

# of times to flip $\quad 2 \cdot \frac{1}{2pq} = \boxed{\frac{1}{pq}}$

$q = 1-p$

X = expected # of flips

$X = 2pq\,(2) + (1-2pq)(2+X)$ ← recursive based on independence of trials

$(2pq)X = 2 \quad \boxed{X = \frac{1}{pq}}$

pick $p = 2/3$, make algorithm more efficient?

2x efficiency of flipping

HH → 4/9
HT → 2/9
TH → 2/9 — group together
TT

But depends on specific p value

generalize?

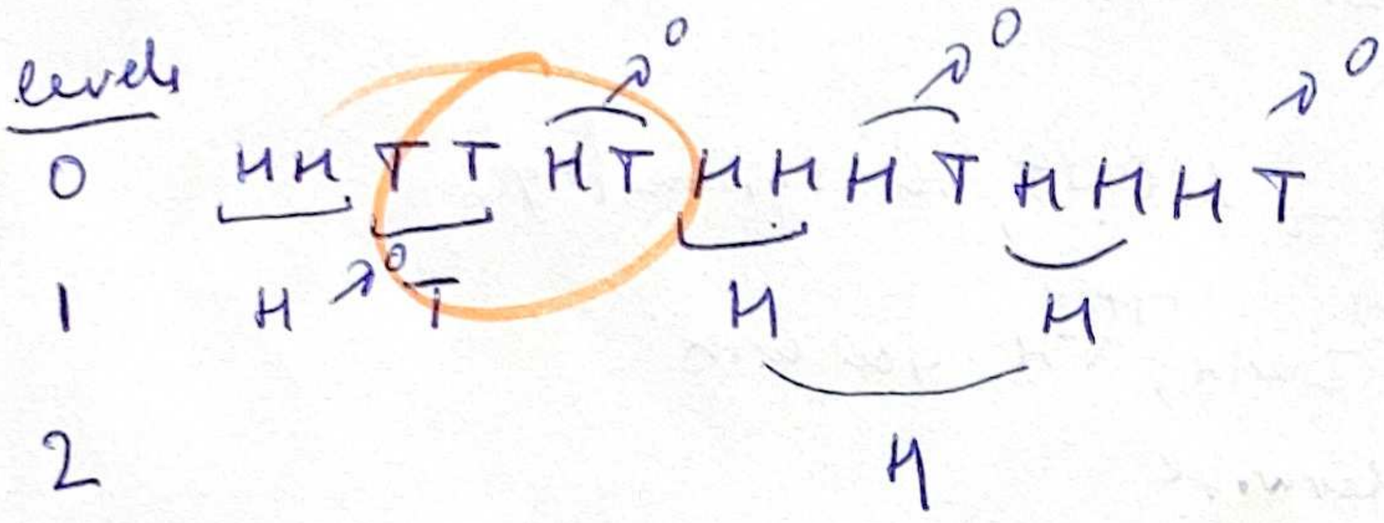
HT — HHTT — HHHHTTTT — ... — by induction
TH — TTHH — TTTTHHHH — ... — only fail if 2^k all H or all T

(only 1 failure flips) (only 1 failure)

still consider HT independently
TH

next question: how to get many coin flips ?... efficiently.

extract max # of flips from a string of flips

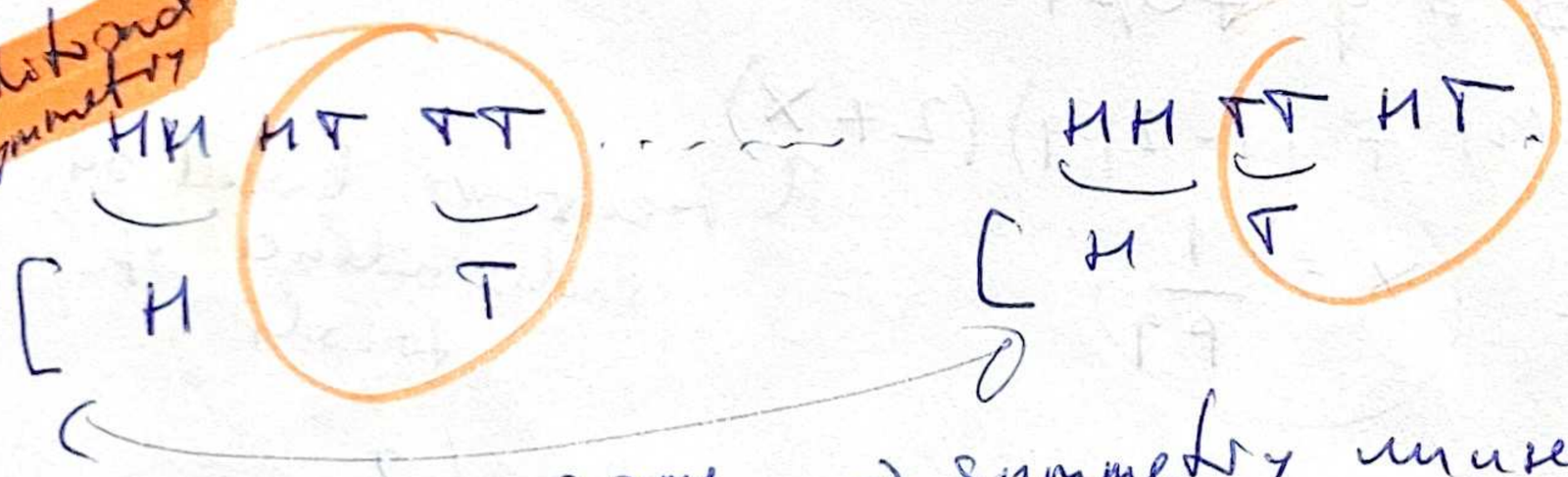


is it all? indep. flips are we getting all randomness out?

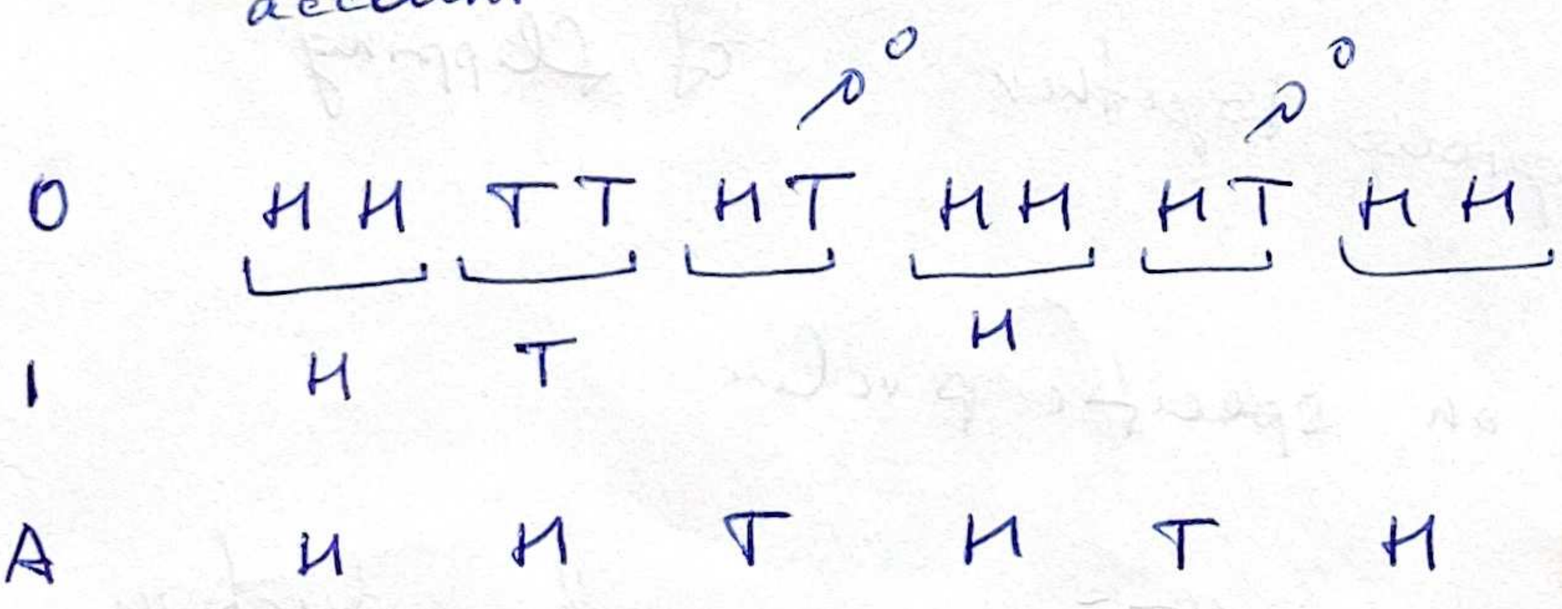
cannot shift by one → not indep!

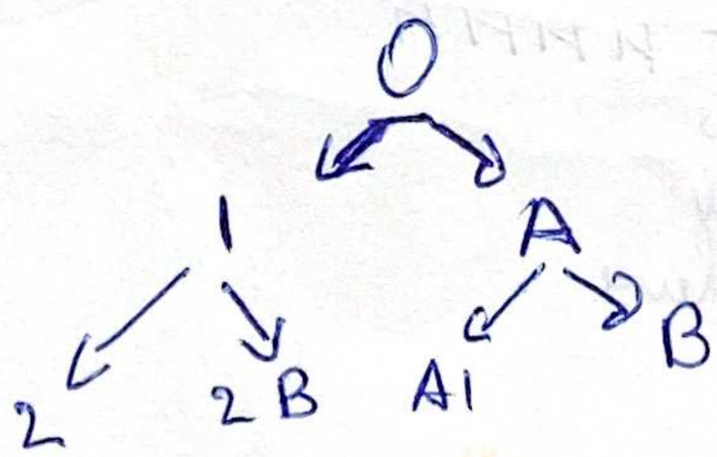
additional symmetry

switch



generate another sequence to take this symmetry into account





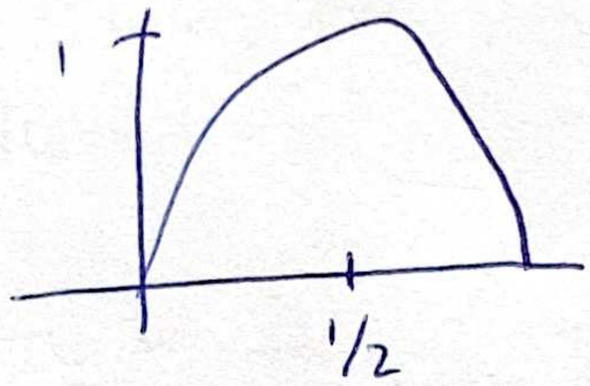
all unbiased indep coin flips pulled out of a string of biased

can reconstruct sequence from flips

entropy = measure randomness

Biased coin p

$H(p)$ = entropy → average # of bits available per coin flip



$H(1/2) = 1$ ↑ 1 bit per flip

$H(p) = 0.72$ ↑ 0.72 bits per flip

$$H(p) = -p\log_2 p - (1-p)\log_2(1-p)$$ for coin flipping process

Proof for A construction

$A(p)$ = avg # both pulled out per flip, when bias is p asymptotically in expect ~ top layer ~ flip from HH or TT

$$= \frac{2pq}{2} + \frac{p^2+q^2}{2} A\left(\frac{p^2}{p^2+q^2}\right) + \frac{1}{2} A(p^2+q^2)$$

avg length ↖ one char for every 2 in original

↖ av. # of bits when got a flip from HH or TT

can show $A(p) = H(p)$

thus $A(p)$ pulls out as many bits as $H(p)$