ILLC Project Course in Information Theory

Crash course

13 January – 17 January 2014
12:00 to 14:00

Student presentations

27 January – 31 January 2014
12:00 to 14:00

Location

ILLC, room F1.15,
Science Park 107, Amsterdam

Materials

informationtheory.weebly.com

Contact

Mathias Winther Madsen
mathias.winther@gmail.com

Monday

Probability theory
Uncertainty and coding

Tuesday

The weak law of large numbers
The source coding theorem

Wednesday

Random processes
Arithmetic coding

Thursday

Divergence
Kelly Gambling

Friday

Kolmogorov Complexity
The limits of statistics

PLAN

- Some combinatorical preliminaries
- Turing machines
- Kolmogorov complexity
- The universality of Kolmogorov complexity
- The equivalence of Kolmogorov complexity and coin flipping entropy
- Monkeys with typewriters

PLAN

- Some combinatorical preliminaries:
  - Factorials
  - Stirling's approximation
  - Binomial coefficients
  - The binary entropy approximation

There are $3 \cdot 2 \cdot 1$ ways to sort three letters:

`ABC, ACB, BAC, BCA, CAB, CBA`

Notation:

$$n! == 1 \cdot 2 \cdot 3 \cdot \cdots \cdot n$$

or "n factorial."

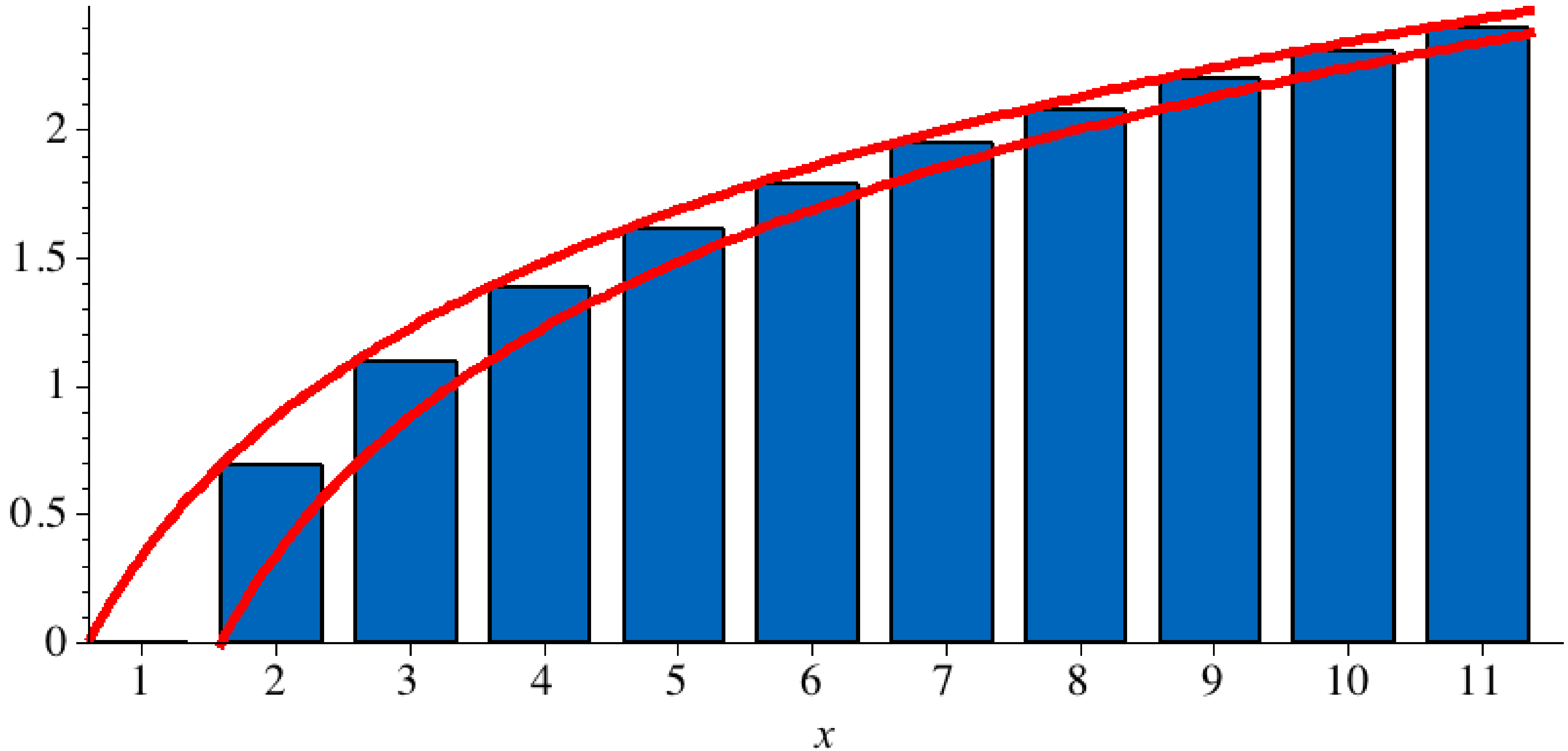
The **natural logarithm** of a factorial can be approximated by Stirling's approximation,

$$\ln(n!) == n \ln n - n$$

The error of this approximation grows slower than linearly.

n	10	20	30	40	50
$\ln(n!)$	15.1	42.3	74.6	110.3	148.5
$\mathrm{Stir}(n)$	13.0	40.0	72.0	107.6	145.6

Sproof:



The anti-derivative of ln(x) is x ln(x) – x.

William Feller: *An Introduction to Probability Theory and its Applications* (1950)

There are

$$4 \cdot 3 \;==\; \frac{4!}{2!} \;==\; \frac{24}{2} \;==\; 12$$

ways to put two objects into four boxes:

1	2	1	2	1	2						
2	1					1	2	1	2		
		2	1			2	1			1	2
				2	1			2	1	2	1

If the objects are identical, the number of options is a factor of 2! smaller:

$$\frac{4!}{2!\ 2!} == \frac{24}{4} == 6$$

*	*	*			
*			*	*	
	*		*		*
		*		*	*

In general, the number of ways to put k identical objects into n distinct boxes is

$$\binom{n}{k} == \frac{n!}{(n-k)!\,k!}$$

This is the **binomial coeff cient**, or "*n* choose *k*."

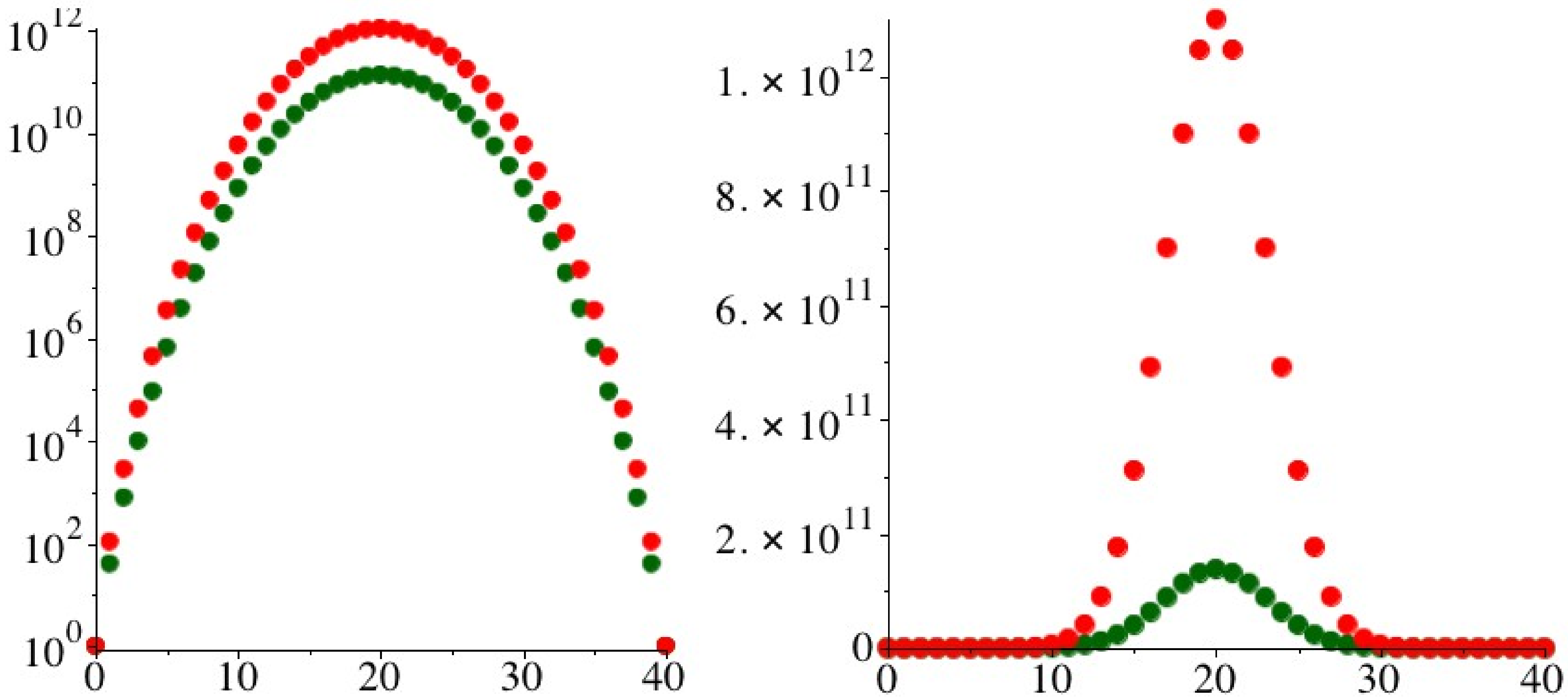
For a nice introduction, see the f rst chapter of Victor Bryant: *Aspects of Combinatorics* (1993)

When applied to a binomial coeff cient,
Stirling's approximation gives

$$\ln \binom{n}{k} == n\,H_2\left(\frac{k}{n}\right)$$

H_2 is here the binary entropy function **measured in nats** (natural units, 1.44 bits).

The error grows slower than a linearly.



Example: $n = 40$, $k = 20$

$$\log \frac{137846528820}{1099511627776} == 25.649 - 27.726.$$

PLAN

- Some combinatorical preliminaries
- Turing machines
- Kolmogorov complexity
- The universality of Kolmogorov complexity
- The equivalence of Kolmogorov complexity and coin flipping entropy
- Monkeys with typewriters

Shortest description?

001001001001.

1100111100111111001111111100.

01011011101111011111.

0100011011000001010011100101110111.

101100010111001000010111111101.

The **Kolmogorov Complexity** of a f nite string is the length of the shortest program which will print that string.

001001001001.

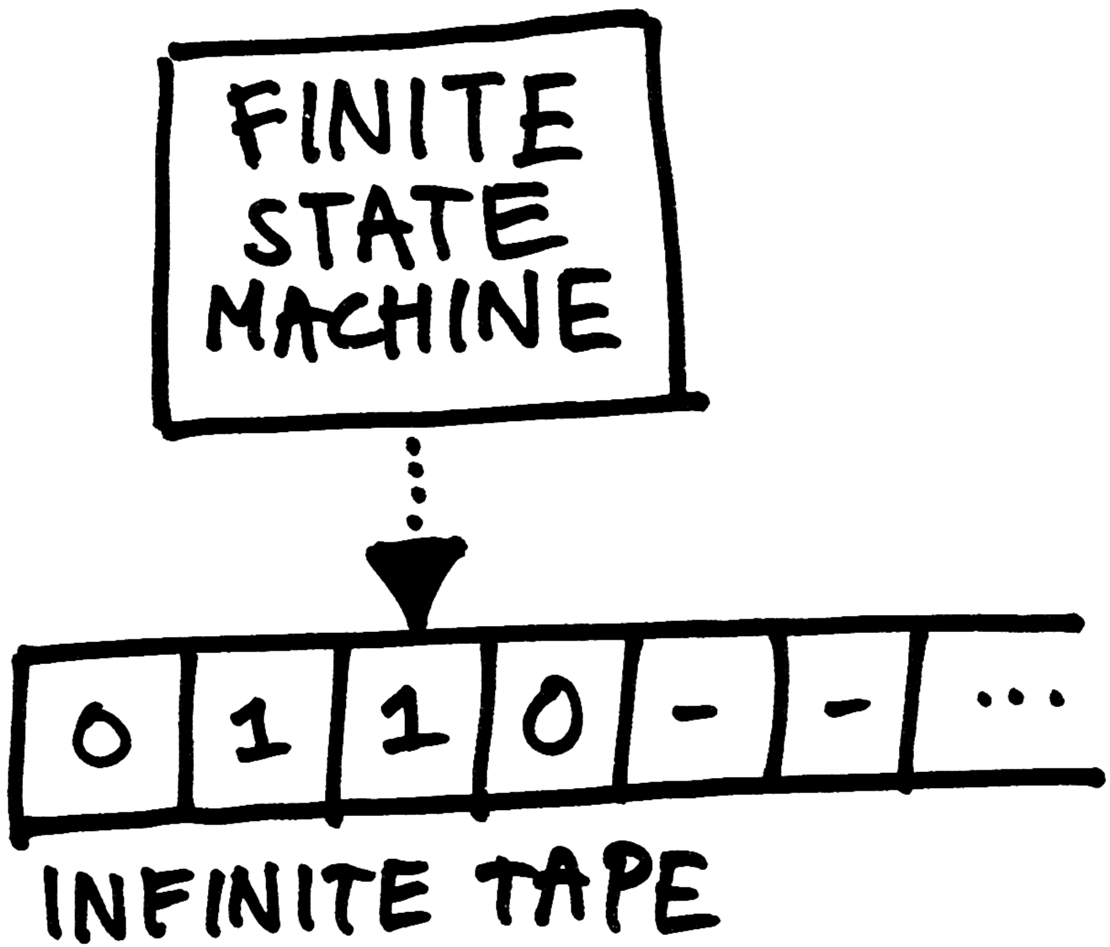
1100111100111111001111111100.

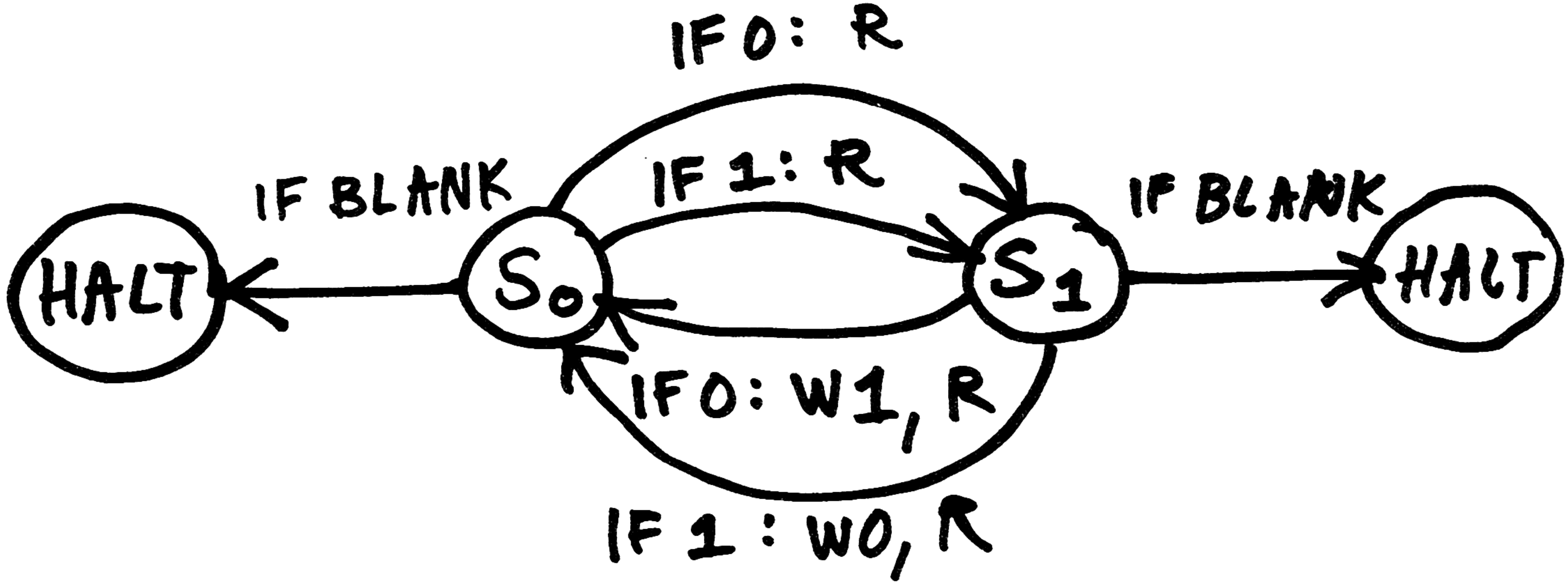
01011011101111011111.

0100011011000001010011100101110111.

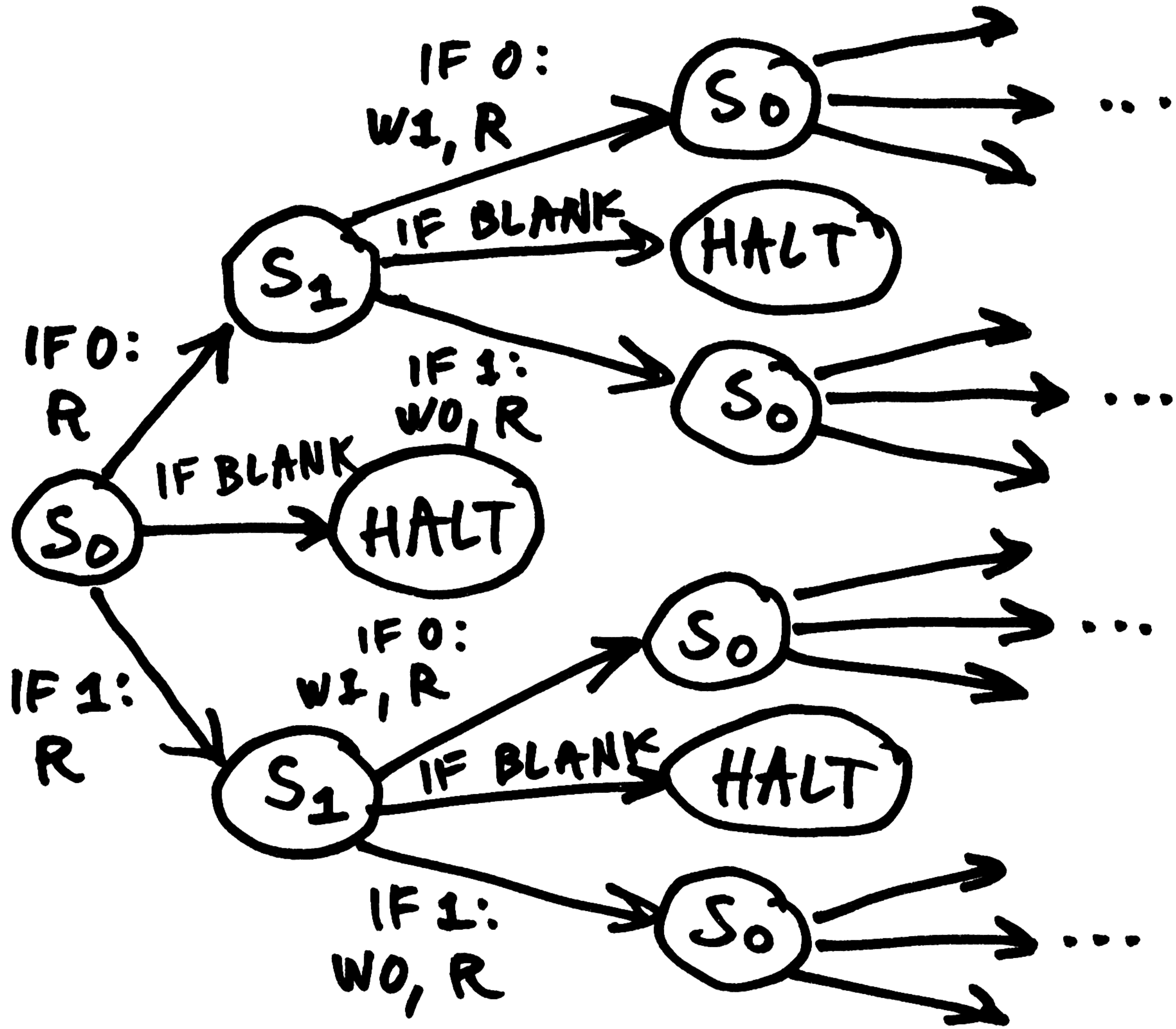
101100010111001000001011111101.

The Turing Machine





S_i	IF 0	IF 1	IF BLANK	S_{i+1}
S_0	R	R	HALT	S_1
S_1	W1, R	W0, R	HALT	S_0



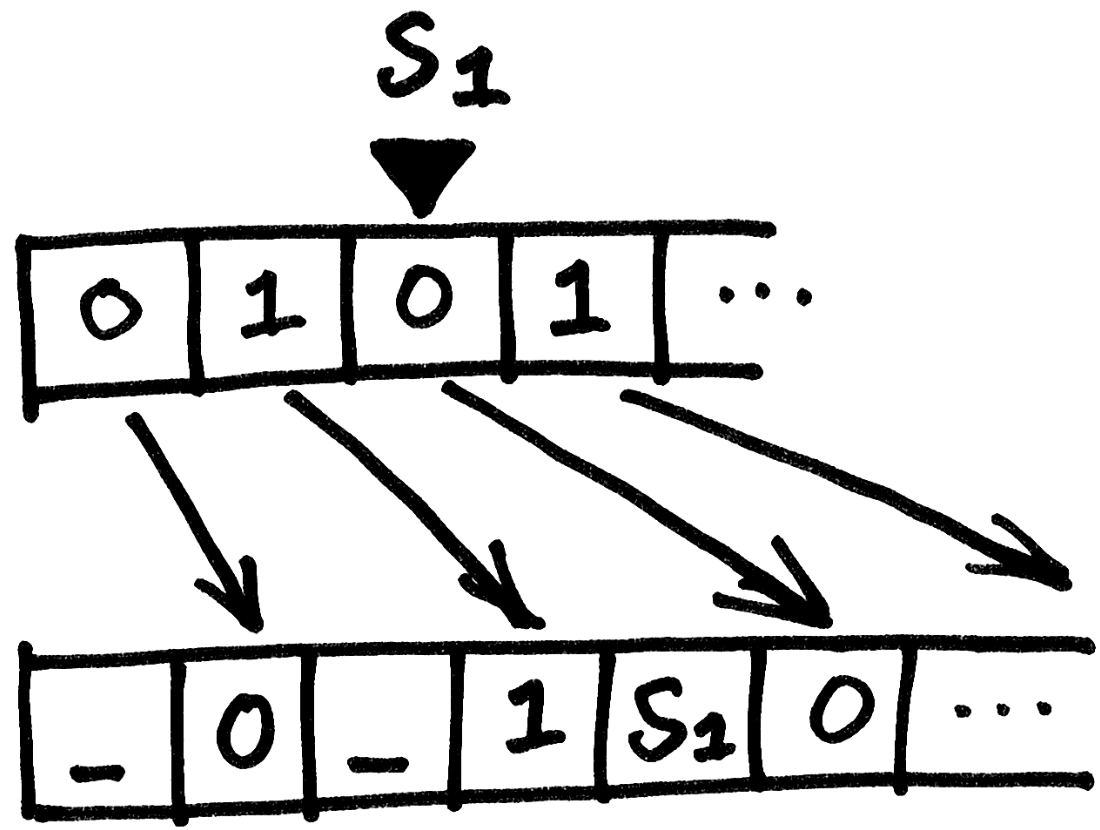
S0
IF 0: R
IF BLANK
HALT
IF 1: R
S1
IF 0: W1, R
IF BLANK
HALT
IF 1: W0, R
S0
S0
...
S1
IF 0: W1, R
IF BLANK
HALT
IF 1: W0, R
S0
S0
...

Theorem:

There are universal machines.

MACHINE("MACHINE")

Sproof:



Consequence:

The Kolmogorov complexity of a string on two different universal machines differs only by the length of the longest simulation program:

$$K_{M_1}(x) - K_{M_2}(x) == c(M_1, M_2)$$

(And constants are sublinear.)

The Kolmogorov Complexity of a finite string is the length of the shortest program which will print that string.

001001001001001001 … 001001001.

11001111001111110011111111 … 00.

01011011101111011111 … 11111111.

010001101100000101001110010 … 1.

10110001011100100001011111 … 01.

Theorem:

Most strings don't have any structure.

Proof:

There are 2^n strings of length n, and

$$1 + 2 + 4 + 8 + 16 + \ldots + 2^{n-1}$$

programs of length $< n$.

```
PrintString(n, k, i):
    construct all string of length n
    select the ones containing k 1s
    print the ith of those strings.
```

0011
0101
1001
0110
1010 ← THIS ONE!
1100

$$\overbrace{\texttt{1 0 1 0}}^{n\,=\,10,}\ \texttt{,}\ \overbrace{\texttt{1 1}}^{k\,=\,3,}\ \texttt{,}\ \overbrace{\texttt{1 1 0 1}}^{i\,=\,13.}$$

$$\overbrace{\texttt{11}}^{\texttt{1}}\ \overbrace{\texttt{00}}^{\texttt{0}}\ \overbrace{\texttt{11}}^{\texttt{1}}\ \overbrace{\texttt{00}}^{\texttt{0}}\ \overbrace{\texttt{01}}^{\texttt{,}}\ \overbrace{\texttt{11}}^{\texttt{1}}\ \overbrace{\texttt{11}}^{\texttt{1}}\ \overbrace{\texttt{01}}^{\texttt{,}}\ \overbrace{\texttt{11}}^{\texttt{1}}\ \overbrace{\texttt{11}}^{\texttt{1}}\ \overbrace{\texttt{00}}^{\texttt{0}}\ \overbrace{\texttt{11}}^{\texttt{1}}$$

$n = 10, \qquad k = 3, \qquad i = 13.$

```
1 0 1 0 , 1 1 , 1 1 0 1
11 00 11 00 01 11 11 01 11 11 00 11
```

8 + 2 + 4 + 2 + 8

Stirling's approximation for a binomial coeff cient is

$$\ln \binom{n}{k} == n\, H_2\!\left(\frac{k}{n}\right)$$

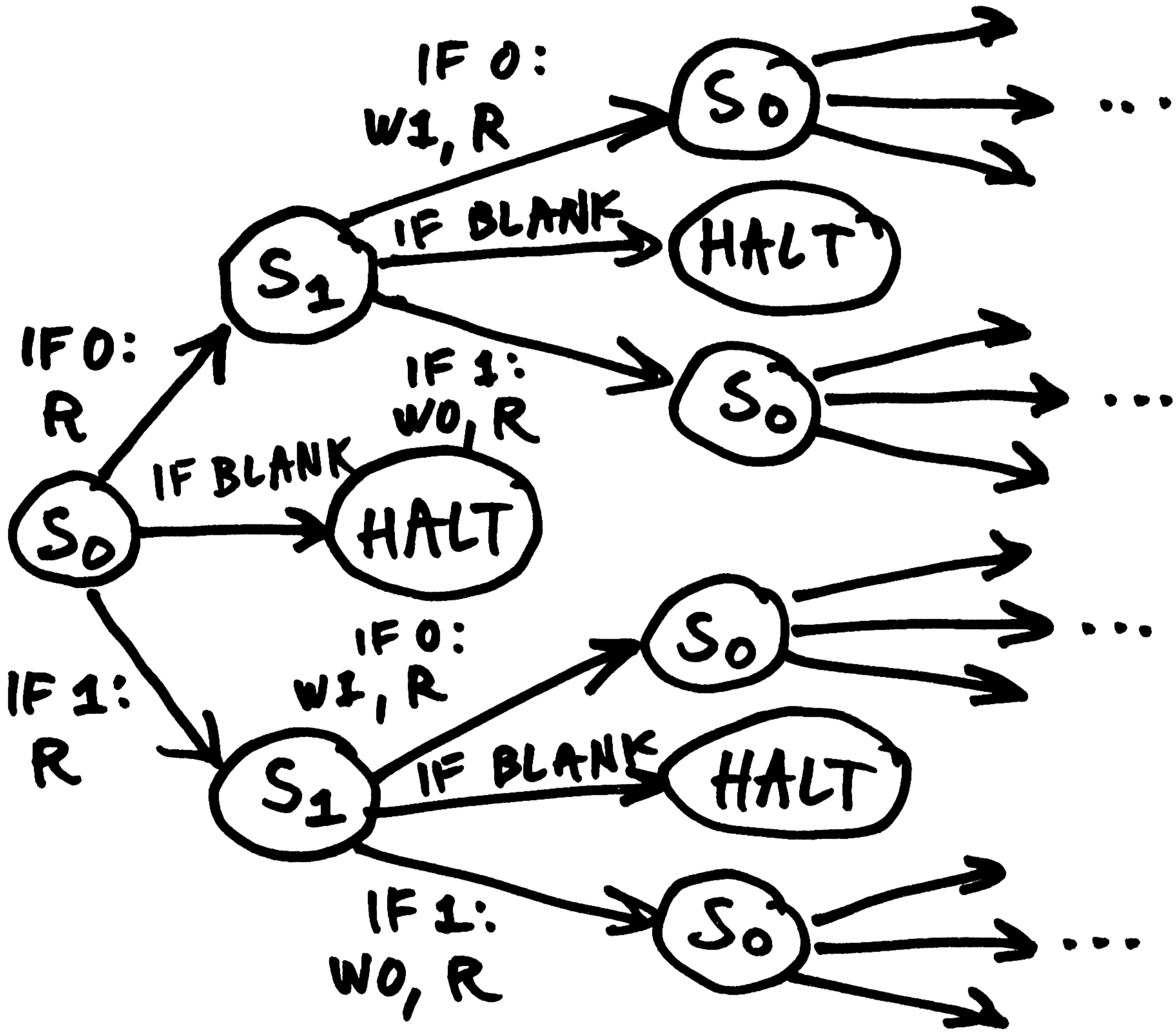
H_2 is here the binary entropy function **measured in nats** (natural units, 1.44 bits).

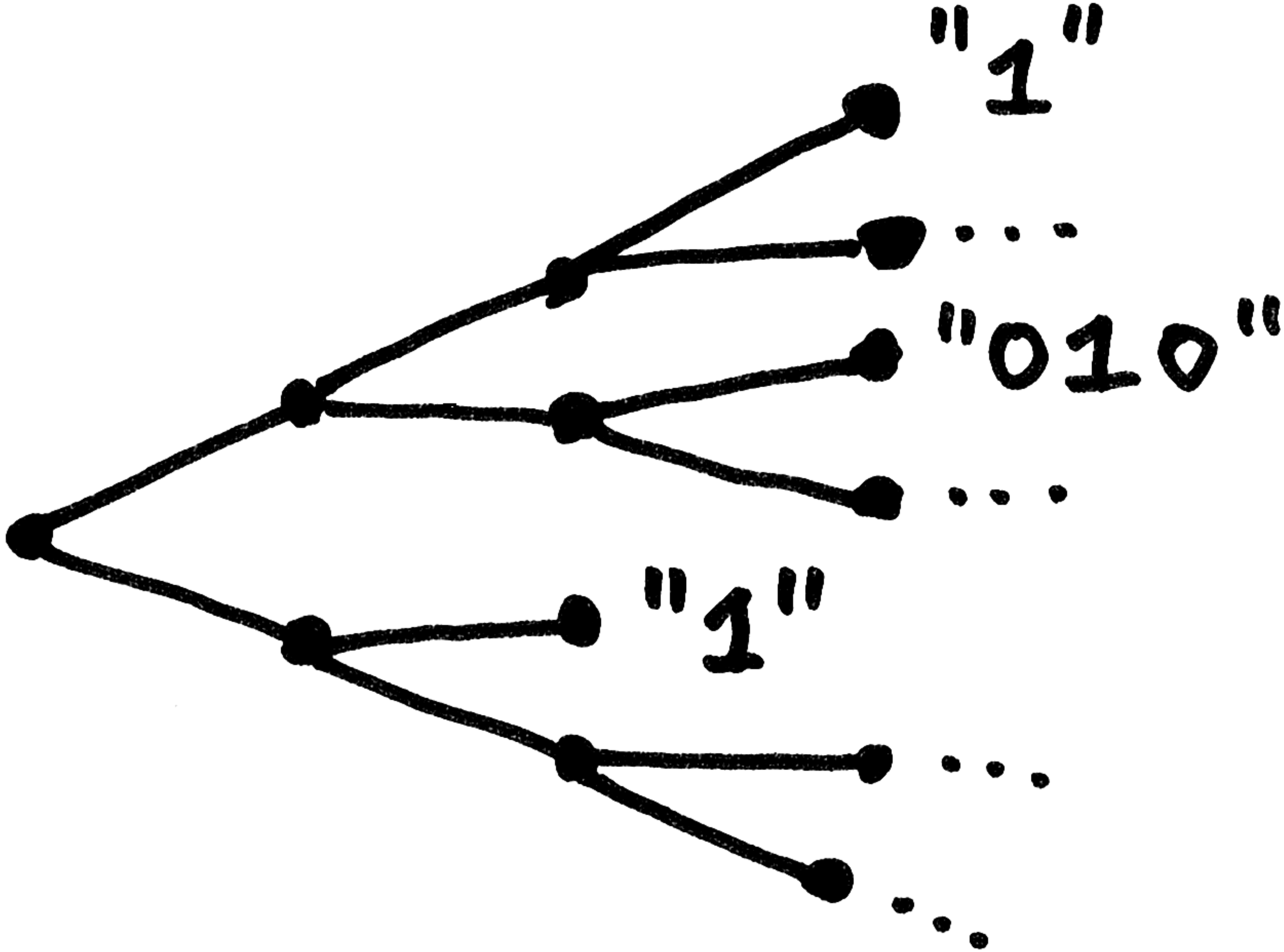
The error grows slower than a linearly.

So

$$K(x) \qquad 2n\, H_2\left(\frac{k}{n}\right) \pm \quad (n)$$

where   is sublinear.

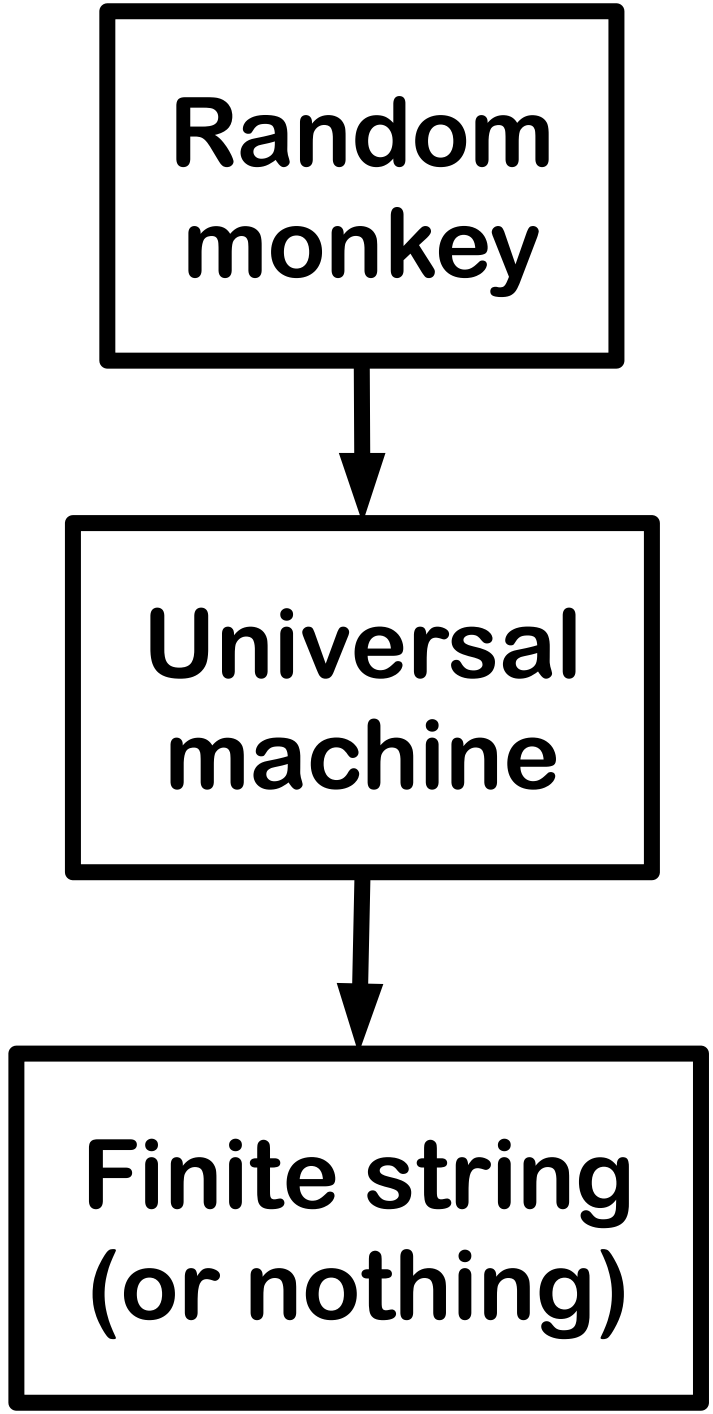

IF 0:
R
IF 1:
R
IF BLANK
HALT
S0
S1
IF 0:
W1, R
IF BLANK
IF 1:
W0, R
S1
IF 0:
W1, R
IF BLANK
IF 1:
W0, R
S0
HALT
...

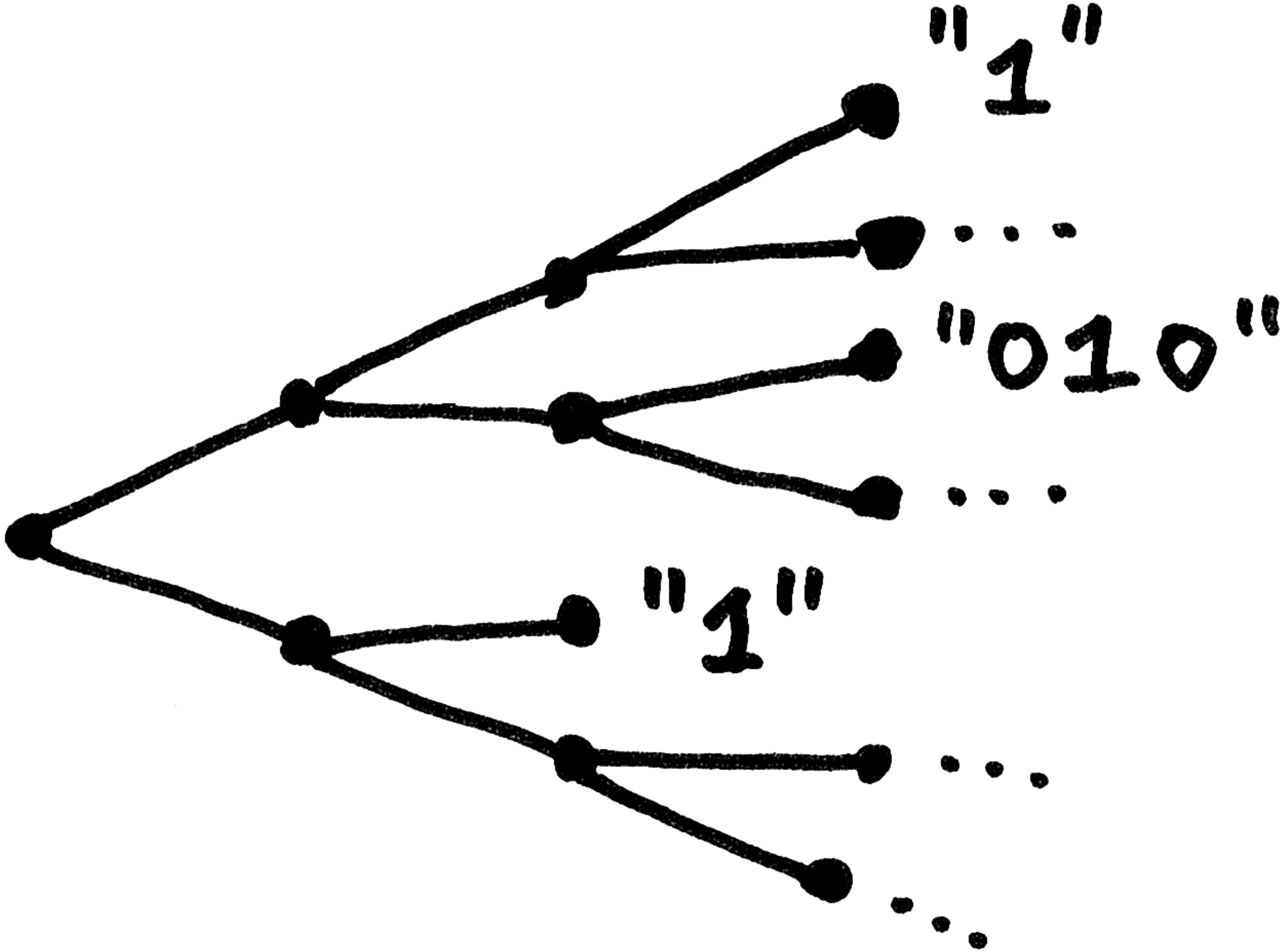

"1"
...
"010"
...
"1"
...
...

For coin flipping sequences, Kolmogorov complexity is equal to Shannon entropy, plus or minus a sublinear term.

For other sequences, Kolmogorov complexity is smaller than the Shannon entropy of the string **if modeled as as a coin flipping sequence**.

Conclusion: Coin flipping is the worst.


Random monkey
Universal machine
Finite string (or nothing)



Ray J. Solomonoff: “A formal theory of inductive inference,” *Information and control*, 1964.

Again, it doesn't matter which universal machine you use.

The universal probability of a string x is close to $2^{-K(x)}$.

For long strings, f nding the probability of the shortest description is thus as good as summing up the probability of all descriptions.