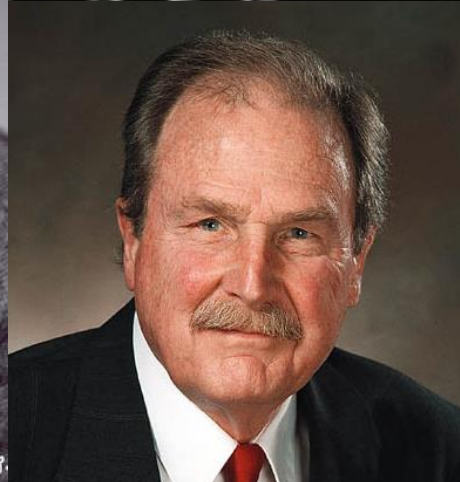
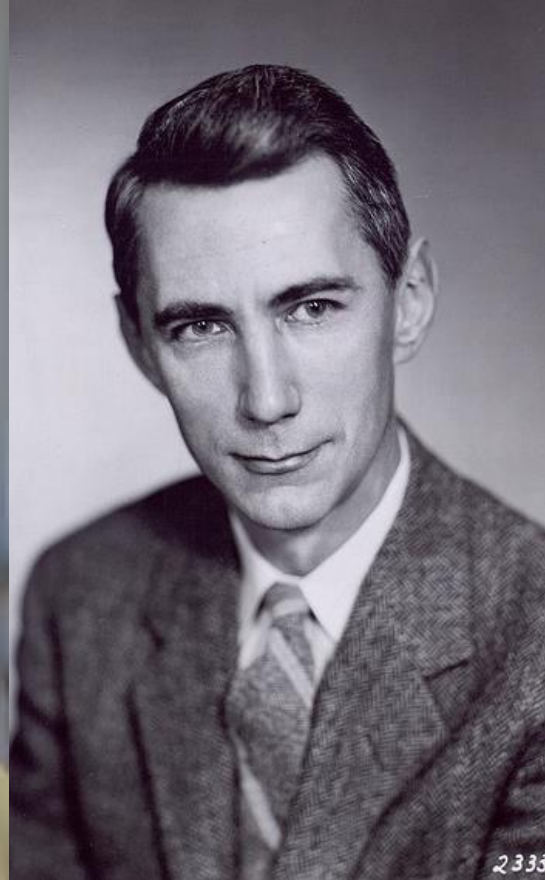
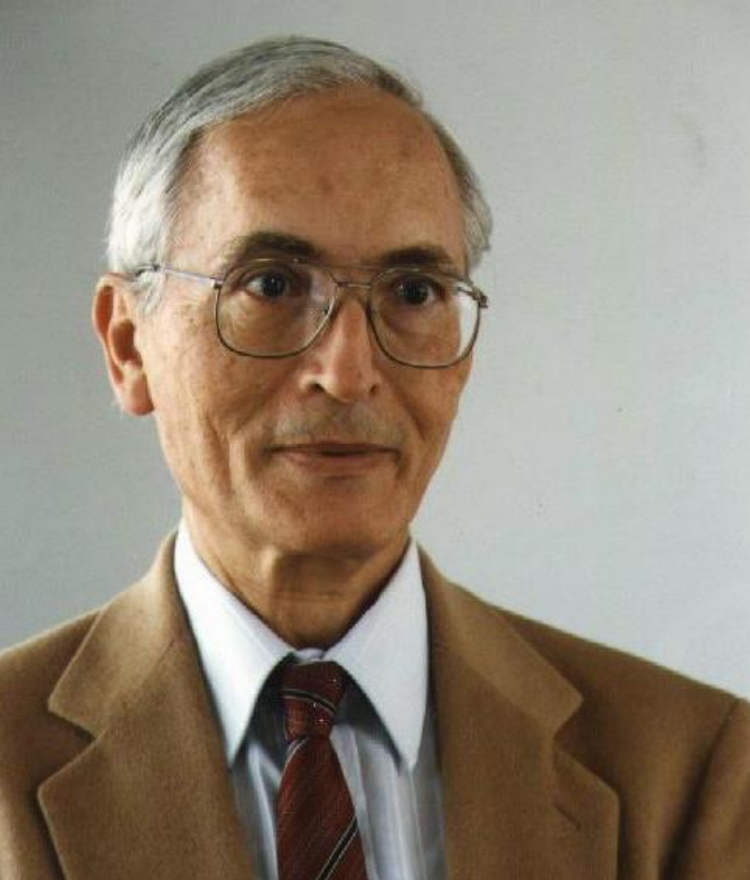




ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ

Лекция 5

28 сентября 2018 г.



ТЕОРИЯ КОДИРОВАНИЯ



ХАРАКТЕРИСТИКИ КОДА

Опр.: Средняя длина кода $K = \sum_{i=1}^N n_i p_i$

n_i - длина кода для i -го символа A
 p_i - вероятность появления i -го символа
 I_A и I_B – энтропии A и B

Опр.: Минимальная длина кода $K_{min} = \frac{I_A}{I_B}$

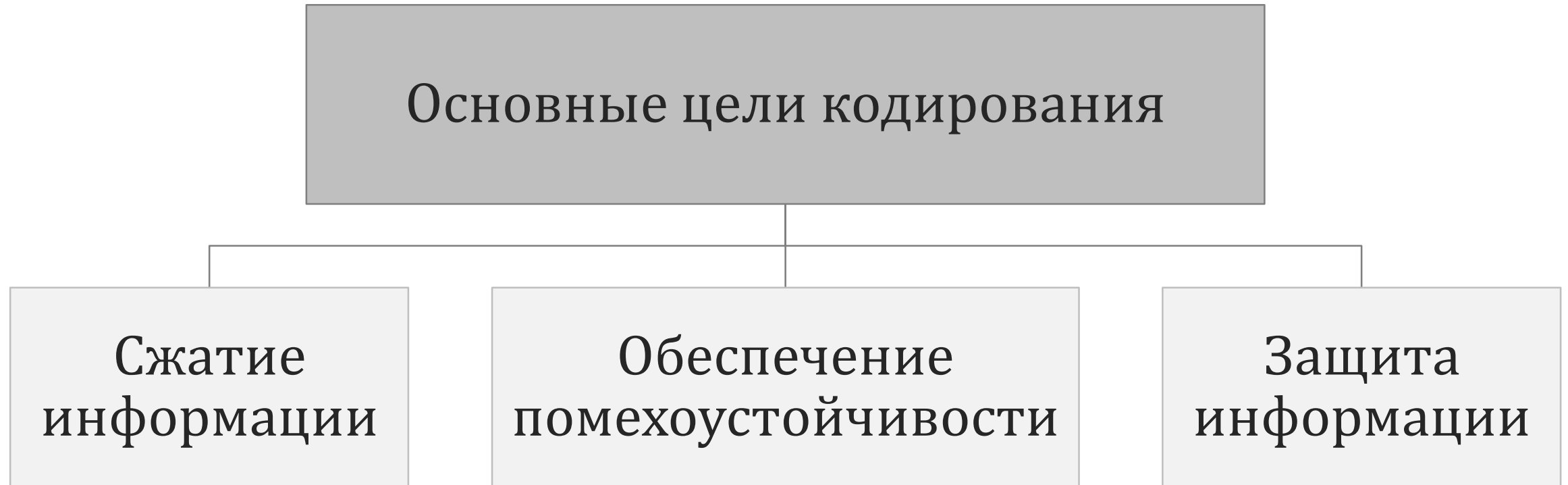
Опр.: Относительная избыточность кода $Q = \frac{K - K_{min}}{K_{min}}$

Первая теорема Шеннона: При отсутствии шумов всегда возможен такой вариант кодирования сообщения, при котором относительная избыточность кода будет сколь угодно близка к нулю.

Первая теорема Шеннона для алфавита {0, 1}: При отсутствии помех средняя длина двоичного кода может быть сколь угодно близкой к средней информации, приходящейся на знак первичного алфавита.



КОДИРОВАНИЕ ИНФОРМАЦИИ



КОД ФИКСИРОВАННОЙ ДЛИНЫ (FLC)

- В современных компьютерах символы кодируются с помощью байтов
- Один байт – 8 бит
- **ASCII** – кодовая таблица, кодирующая все основные печатные символы

32		00100000	56	8	00111000	80	P	01010000	104	h	01101000
33	!	00100001	57	9	00111001	81	Q	01010001	105	i	01101001
34	"	00100010	58	:	00111010	82	R	01010010	106	j	01101010
35	#	00100011	59	;	00111011	83	S	01010011	107	k	01101011
36	\$	00100100	60	<	00111100	84	T	01010100	108	l	01101100
37	%	00100101	61	=	00111101	85	U	01010101	109	m	01101101
38	&	00100110	62	>	00111110	86	V	01010110	110	n	01101110
39	'	00100111	63	?	00111111	87	W	01010111	111	o	01101111
40	(	00101000	64	@	01000000	88	X	01011000	112	p	01110000
41	)	00101001	65	A	01000001	89	Y	01011001	113	q	01110001
42	*	00101010	66	B	01000010	90	Z	01011010	114	r	01110010
43	+	00101011	67	C	01000011	91	[	01011011	115	s	01110011
44	,	00101100	68	D	01000100	92	\	01011100	116	t	01110100
45	-	00101101	69	E	01000101	93	]	01011101	117	u	01110101
46	.	00101110	70	F	01000110	94	^	01011110	118	v	01110110
47	/	00101111	71	G	01000111	95	_	01011111	119	w	01110111
48	0	00110000	72	H	01001000	96	`	01100000	120	x	01111000
49	1	00110001	73	I	01001001	97	a	01100001	121	y	01111001
50	2	00110010	74	J	01001010	98	b	01100010	122	z	01111010
51	3	00110011	75	K	01001011	99	c	01100011	123	{	01111011
52	4	00110100	76	L	01001100	100	d	01100100	124		01111100
53	5	00110101	77	M	01001101	101	e	01100101	125	}	01111101
54	6	00110110	78	N	01001110	102	f	01100110	126	-	01111110
55	7	00110111	79	O	01001111	103	g	01100111	127	□	01111111



КОД ПЕРЕМЕННОЙ ДЛИНЫ (VLC)

- Пример VLC-кода – азбука Морзе
- Принцип энтропийного (статистического) кодирования – построить VLC таким образом, чтобы **часто используемые** символы первичного алфавита кодировались самими **короткими кодами**, а редкие – длинными.
- Зная распределение частот, мы можем подсчитать минимальное среднее число необходимых бит на символ – это и есть энтропия.

Опр.: Энтропия – минимальное количество бит на один символ первичного алфавита (в среднем).

International Morse Code

1. The length of a dot is one unit.
2. A dash is three units.
3. The space between parts of the same letter is one unit.
4. The space between letters is three units.
5. The space between words is seven units.

A ● —
B — ● ● ●
C — ● — ●
D — ● ●
E ●
F ● ● — ●
G — — ●
H ● ● ● ●
I ● ●
J ● — — —
K — ● —
L ● — ● ●
M — —
N — ●
O — — —
P ● — — ●
Q — — ● —
R ● — ●
S ● ● ●
T —

U ● ● —
V ● ● ● —
W ● — —
X — ● ● —
Y — ● — —
Z — — ● ●

1 ● — — — —
2 ● ● — — —
3 ● ● ● — —
4 ● ● ● ● —
5 ● ● ● ● ●
6 — ● ● ● ●
7 — — ● ● ●
8 — — — ● ●
9 — — — — ●
0 — — — — —



ПРЕФИКСНЫЕ КОДЫ

Опр.: Префиксный код в теории кодирования — код со словом переменной длины, удовлетворяющий условию Роберта Фано.

Опр.: Условие Фано: неравномерный код может быть однозначно декодирован, если никакой из кодов не совпадает с началом (префиксом) какого-либо другого, более длинного кода.

Не префиксный код!

International Morse Code

1. The length of a dot is one unit.
2. A dash is three units.
3. The space between parts of the same letter is one unit.
4. The space between letters is three units.
5. The space between words is seven units.

A ● —
B — ● ● ●
C — ● — ●
D — ● ●
E ●
F ● ● — ●
G — — ●
H ● ● ● ●
I ● ●
J ● — — —
K — ● —
L ● — ● ●
M — —
N — ●
O — — —
P ● — — ●
Q — — ● —
R ● — ●
S ● ● ●
T —

U ● ● —
V ● ● ● —
W ● — —
X — ● ● —
Y — ● — —
Z — — ● ●

1 ● — — — —
2 ● ● — — —
3 ● ● ● — —
4 ● ● ● ● —
5 ● ● ● ● ●
6 — ● ● ● ●
7 — — ● ● ●
8 — — — ● ●
9 — — — — ●
0 — — — — —



ПРЕФИКСНЫЙ КОД ШЕННОНА-ФАНО

Алгоритм 1: Построение кода Шеннона-Фано

0. Строим таблицу частот символов и упорядочиваем ее
1. Если в таблице одна строка, то ничего не делаем
2. Делим таблицу на две части так, чтобы сумма частот в первой части была как можно ближе к сумме частот из второй части
3. К кодам символов из первой части таблицы дописываем 0, а из второй – 1.
4. Выполняем алгоритм отдельно для первой и второй частей таблицы.



ПРЕФИКСНЫЙ КОД ХАФФМАНА

Алгоритм 2: Построение кода Хаффмана

0. Для каждого символа из алфавита строим вершину с весом, равным частоте этого символа.

1. Если количество вершин равно 1, то заканчиваем.

2. Упорядочиваем вершины по убыванию веса.

3. Создаем новую вершину и «подцепляем» к ней две вершины с самыми низкими частотами. Частоту новой вершины задаем как сумму частот двух исходных. На ребре, ведущем к левому сыну пишем 0, а к правому – 1.

4. Удаляем **из списка** «подцепленные» вершины и переходим на шаг 1.

Код символа будет записан на ребрах, находящихся на пути из корня (последней оставшейся вершины) к вершине с этим символом.



УПРАЖНЕНИЯ

1. Определите является ли префиксным следующий двоичный код первых 5 букв латинского алфавита:

a	b	c	d	e
000	110	01	001	10

Декодируйте сообщение 1100000100110, записанное с использованием приведенного кода.

2. Для символов первичного алфавита $A=\{\text{«+»}, \text{«-»}, \text{«*»}, \text{«/»}\}$ с известными вероятностями появления символов $P=\{0,4; 0,2; 0,3; 0,1\}$:
- а) Вычислите среднюю и минимальную длины кода и относительную избыточность, если символы кодируются следующим образом: «+» - 0, «-» - 100, «*» - 1010, «/» - 1100
 - б) Постройте префиксный код по алгоритму Шеннона-Фано и подсчитайте K, I_A, Q
 - в) Постройте префиксный код по алгоритму Хаффмана и подсчитайте K, I_A, Q





БЛОКОВОЕ КОДИРОВАНИЕ

Пример 1

P1	P2	H
0.5	0.5	1
0.7	0.3	0.88
0.99	0.01	0.08

Пример 2

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 -> 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 2 1 2 1 2 1 2 1 2 1 2 1 2 -> 1 1 -1 1 -1 1 -1 1 -1 1 -1 1 -1 1

Пример 3

- T O B E O R N O T T O B E O R T O B E O R N O T $H = 2.38$
- T O B E O R N O T T O B E O R T O B E O R N O T $H = 1.98$
- T O B E O R N O T T O B E O R T O B E O R N O T $H = 2.5$

Основная задача блоковых алгоритмов – найти баланс между длиной блока и энтропией.



АРИФМЕТИЧЕСКОЕ КОДИРОВАНИЕ

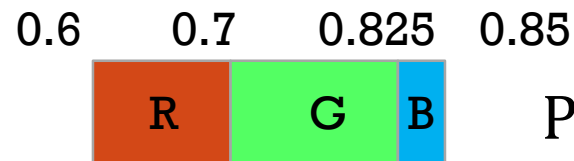
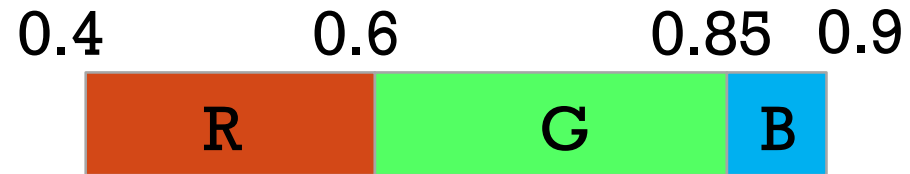
В отличие от КПД, которые присваивают код каждому отдельному символу, арифметическое кодирование присваивает одно кодовое слово всей кодируемой последовательности.



АРИФМЕТИЧЕСКОЕ КОДИРОВАНИЕ

Символ	Вероятность
R	0.4
G	0.5
B	0.1

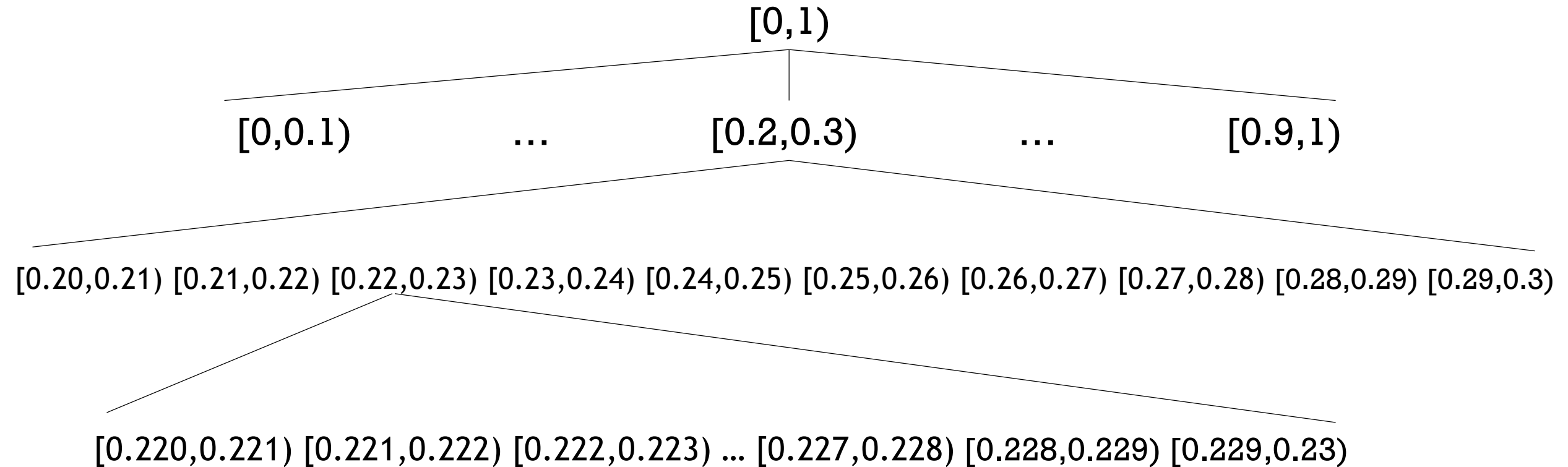
Входная строка: GGB



Результат кодирования –
любая точка из
полуинтервала $[0.825, 0.5)$



ПРИНЦИП РАБОТЫ АРИФМЕТИЧЕСКОГО КОДИРОВАНИЯ



УПРАЖНЕНИЯ

1. Вычислить длину арифметического кода для сообщения AAB, полученного от д. с. в. X со следующим распределением вероятностей $P(X=A)=1/3$, $P(X=B)=2/3$.
2. Декодировать арифметический код 0.35 для последовательности значений д. с. в. из предыдущего упражнения. Остановиться после декодирования третьего символа.

