Коды

Σ – алфавит, любое непустое конечное мн-во

$\Sigma^0 = \{\varepsilon\}$ пустая строка

$\bigcup_{k=0}^{\infty} \Sigma^k = \Sigma^*$ – слова над Σ

$\alpha\beta$ – конкатенация двух строк

$\cdot : \Sigma^* \times \Sigma^* \to \Sigma^*$

$\alpha \in \Sigma^k, \beta \in \Sigma^l \Rightarrow \gamma = \alpha\beta \in \Sigma^{k+l}$

$$\gamma[i] = \begin{cases} \alpha[i], & i \le k \\ \beta[i-k] & i > k \end{cases}$$

1) $(\alpha\beta)\gamma = \alpha(\beta\gamma)$ ассоциативность
2) $\alpha\varepsilon = \varepsilon\alpha = \alpha$ нейтральный

} моноид (свободный моноид над Σ) (нет нарушающих отношений)

Код – функция: $\varphi: \Sigma^* \to \Pi^*$

однозначный / декодируемый код $\Leftrightarrow \alpha \ne \beta \Rightarrow \varphi(\alpha) \ne \varphi(\beta)$

декод. ф
$Zip: \Sigma^* \to \Sigma^*$

недекод. ф
$jpeg: \Sigma^* \to \Sigma^*$

Код, в котором каждый символ кодируется отдельно – разделенный (гомоморфный)

$\varphi: \Sigma \to \Pi^*$

$\varphi(c_1, c_2, c_3, \ldots c_n) = \varphi(c_1)\varphi(c_2)\varphi(c_3)\ldots\varphi(c_n)$

Кодирование с целью сжатия

Теорема:

$\nexists$ кода $\Sigma^* \to \Sigma^*$, который не увеличивает $\forall$ текст, а некоторые уменьшает, однозн. декод.

$\overset{str}{S}$ $\quad \Sigma = \{c_1, c_2, c_3 \ldots c_k\}$ $\quad p_i$ = кол-во вхождений c_i в S

$\underset{\text{двоич. код}}{\varphi: \Sigma \to \mathbb{B}^*}$ $\quad l_i$ – $len(\varphi(c_i))$

$len(\varphi(S)) = \sum_{i=1}^{k} l_i p_i \longrightarrow min$

Префиксный код

φ – префикс. код $\Leftrightarrow$ $\forall a, b \in \Sigma$ $\varphi(a)$ не префикс $\varphi(b)$

$\Sigma = \{a, b, c\}$

a	0
b	01
c	11

не преф

$\triangle$

a	0
b	10
c	11

преф

Задача: префиксный код, $\sum l_i p_i \to min$

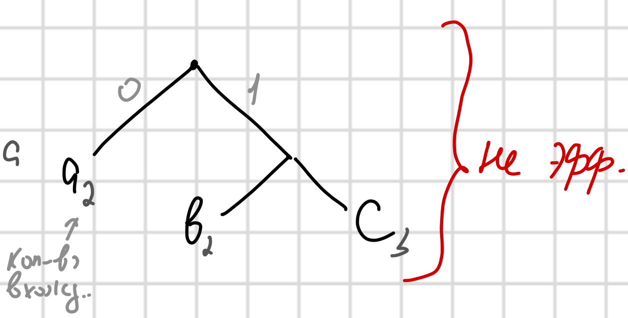
префиксный код однозначно декод-ый

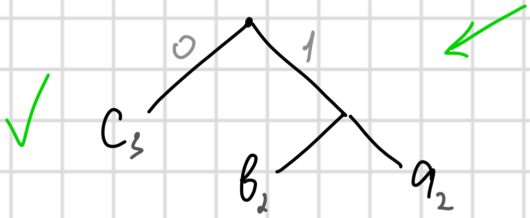
для $\triangle$ возьмем

0	0	10	10	11	11	11
a	a	b	b	c	c	c

$t = \varphi(S) \qquad S = ?$

дерево кода



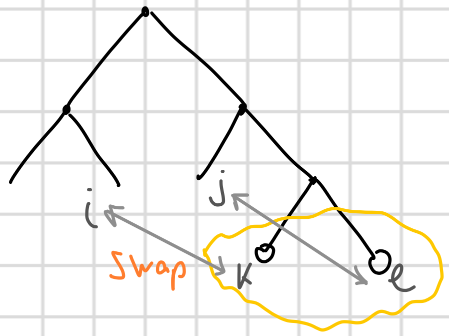


Лемма:

$\exists$ дерево оптимального кода два симв. с $\min p_i$ явл братьями на макс. глубине.

▷ Пусть оптимальное дерево кода. p_i, p_j – min

p_k, p_l – сам. глубокие



$$\longrightarrow \begin{cases} p_i \le p_k \\ p_j \le p_l \end{cases} \Rightarrow \begin{cases} l_i \le l_k \\ l_j \le l_l \end{cases}$$

Было $\sum\limits_t l_t p_t \overset{\triangle}{=} \sum\limits_{t \ne j,i,k,l} l_t p_t + p_i l_i + p_j l_j + p_k l_k + p_l l_l$

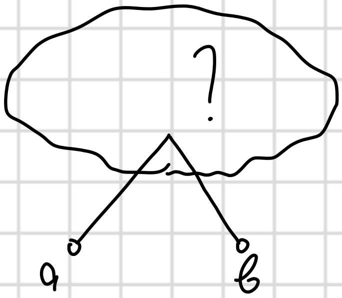
Стало $\sum\limits_t l'_t p_t \overset{\square}{=} \sum\limits_{t \ne i,j,k,l} l_t p_t + p_i l_k + p_j l_l + p_k l_i + p_l l_j$

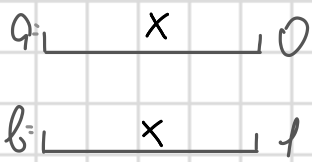
$\triangle - \square = p_i(l_i - l_k) + p_j(l_j - l_l) - p_k(l_i - l_k) - p_l(l_j - l_l) =$

$= \underbrace{\underset{\le 0}{(p_i - p_k)}\underset{\le 0}{(l_i - l_k)}}_{\ge 0} + \underbrace{(p_j - p_l)(l_j - l_l)}_{\ge 0} \ge 0 \Rightarrow$

$\triangle \ge \square$ ◀

a	b	c	...
2	2	3	





т.к. они не зависят от fр.

объединим a и b → x. ab встр. $p_a + p_b$ раз

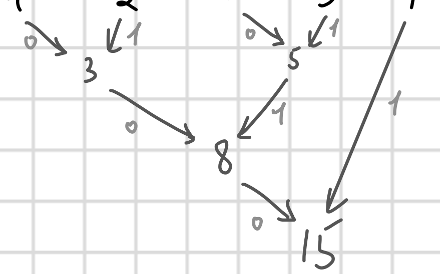
aabbccc → xxxxccc

Код Хаффмана

$$\sum_{a \text{ и } b \to x} p_i l_i = \sum_{i \neq x} p_i l_i + p_x l_x = \sum_{i \neq x} p_i l_i + p_a(l_a - 1) + p_b(l_b - 1) =$$

$$= \sum_{\substack{i \\ a \text{ и } b \not\to x}} p_i l_i - p_a - p_b$$

a	b	c	d	e
1	2	2	3	7



a – 000
b – 001
c – 010
d – 011
e – 1

Нер-во Крафта-МакМиллана

Теорема:

$S, C_1 \ldots C_K$

можно построить одн. дек. двоич. код с длинами код. слов l_i $\Leftrightarrow$

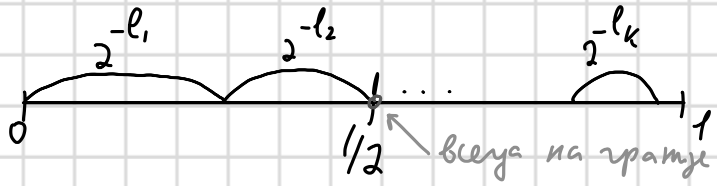
$\Leftrightarrow \sum_{i=1}^{K} 2^{-l_i} \leq 1$

▷ Достаточность

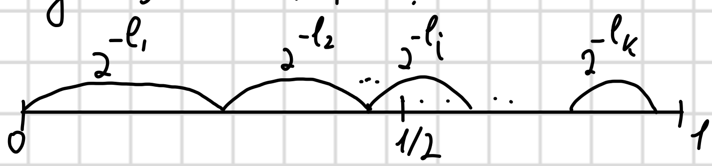
$\exists\ l_1 \le l_2 \le \dots \le l_k$

$2^{-l_1} \le 2^{-l_2} \le \dots \le 2^{-l_k}$

		l	2^{-l}	
a	0	1	1/2	$1 \le 1$
b	10	2	1/4	
c	11	2	1/4	
a	000	3	1/8	$1 \le 1$
b	001	3	1/8	
c	010	3	1/8	
d	011	3	1/8	
e	1	1	1/2	
		1	1/2	≥ 1
		3	1/8	
		3	1/8	
		3	1/8	
		3	1/8	
		3	1/8	



предположим обратное:

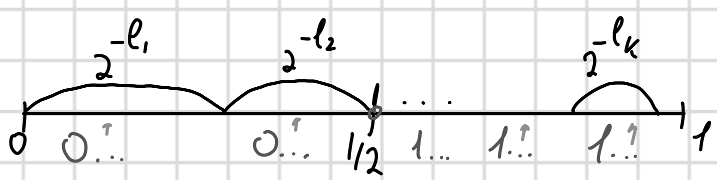


$$\begin{cases} \text{было: } 2^{-l_1} + \dots + 2^{-l_{i-1}} < \frac{1}{2} & \cdot 2^{l_i} \\ \text{стало: } 2^{-l_1} + \dots + 2^{-l_{i-1}} + 2^{-l_i} > \frac{1}{2} & \cdot 2^{l_i} \end{cases}$$

$$\begin{cases} 2^{l_i - l_1} + 2^{l_i - l_2} + \dots + 2^{l_i - l_{i-1}} < 2^{l_i - 1} \\ \underbrace{2^{l_i - l_1} + \dots + 2^{l_i - l_{i-1}}}_{C} + 1 > 2^{l_i - 1} \end{cases}$$

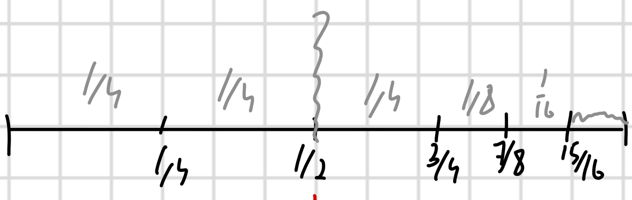
$$\begin{cases} C < 2^{l_i - 1} \\ C + 1 > 2^{l_i - 1} \end{cases}$$

означает, что между двумя соседними целыми числами лежит ещё одно, а такое невозможно

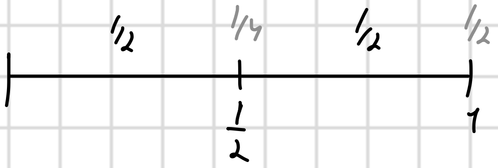


$$\sum 2^{-(l_i - 1)} = \sum 2^{-l_i + 1} = 2 \cdot \sum 2^{-l_i} \le 1$$

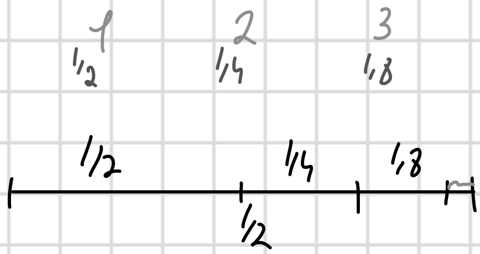
$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$
2	2	2	3	4



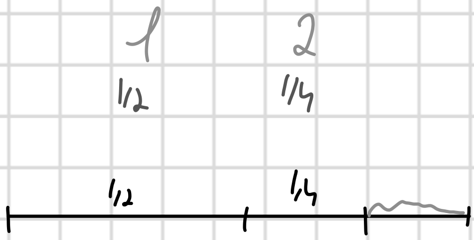
0	0	0	0	0
$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$



00	01	1	1	1
$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$



00	01	10	11	11
$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$



00	01	10	110	1111
$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$