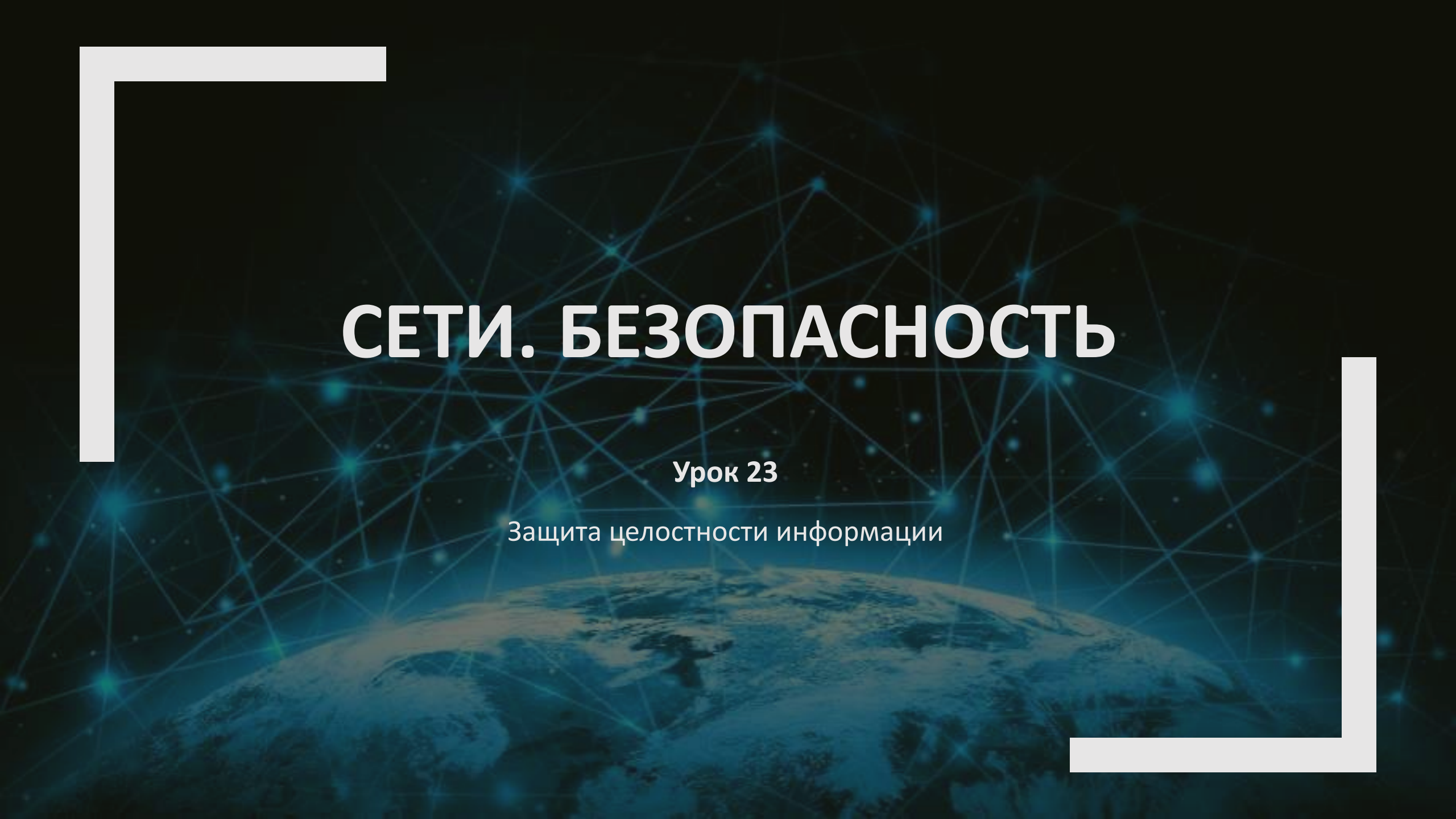


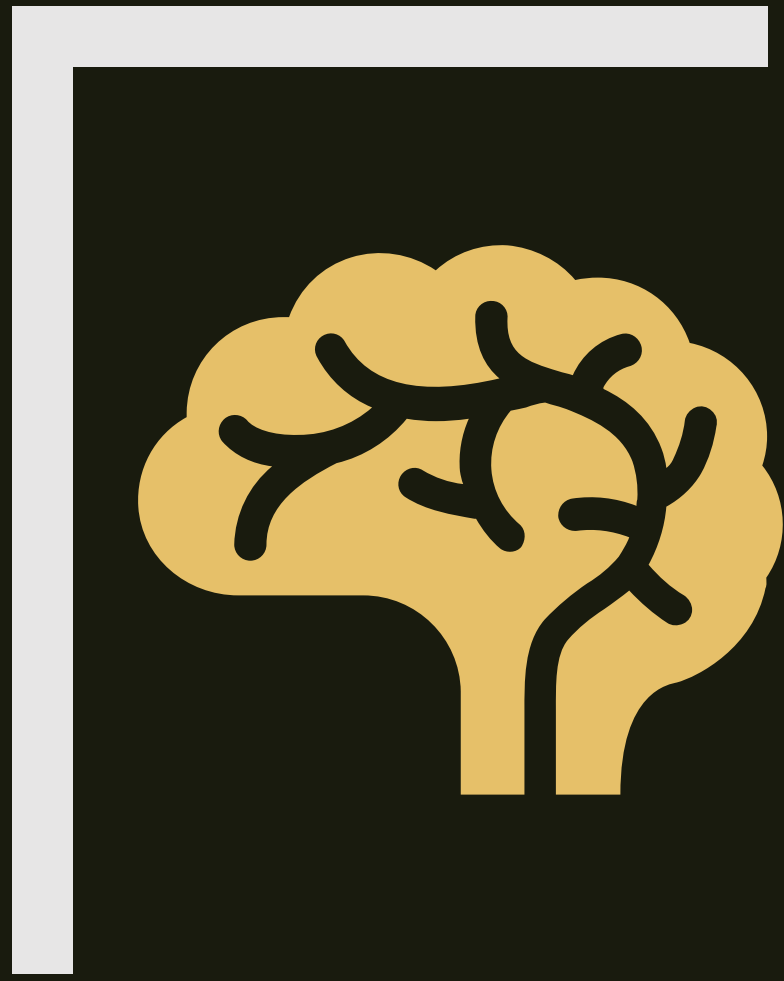
СЕТИ. БЕЗОПАСНОСТЬ

Урок 23

Защита целостности информации

Memory line

Base64



НАЛИЧИЕ ОШИБОК

CRC – контроль чётности

Добавляем **девятый** бит 000100001

Бит чётности или контрольный разряд формируется при выполнении операции «Исключающее-ИЛИ» поразрядно. *(1 если единиц нечётно)*

- **10111101** содержит 6 '1' битов. Бит чётности будет 0, получаем 101111010.
- **01110011** содержит 5 '1' битов. Бит чётности будет 1, получаем 011100111.
- **00000000** содержит 0 '1' битов. Бит чётности будет 0, получаем 000000000.

Развитие CRC -> Хеш-сумма

Хеш-сумма – математическая функция от входной строки

CRC32: F6DE2FEA

MD5: *026f8e459c8f89ef75fa7a78265a0025*

SHA-1: 7DD987F846400079F4B03C058365A4869047B4A0

Хеш-сумма

Хеш-сумма – основные свойства:

Функционал:

- Произвольная длина входных данных
- Фиксированная длина результата
- Однозначность результата

Качество

- Сильная зависимость результата от входных данных
- Непредсказуемость результата

Стойкость

- Необратимость
- Стойкость к коллизиям первого рода: невозможно подобрать сообщение под известный хеш
- Стойкость к коллизиям второго рода: невозможно подобрать пару сообщений с одинаковым хешом

КОДИРОВАНИЕ ХЕММИНГА 15/11

Магия… делай раз

Итак, есть некоторый закодированный текст:

11001010 11100101 11010001…

Выделяем из него первые 11 бит:

11001010 11100101 11010001…

Обратите внимание, что в один блок попадает одна целая буква и 3/8ых от второй буквы… То есть некоторые символы будут разорваны между несколькими блоками кодирования

Магия… делай два

Готовим место под кодирование. Нам нужно 15 ячеек:

Из них первая, вторая, четвёртая и восьмая ячейки являются защитными, а остальные информационными:

Записываем наш блок в информационные ячейки:

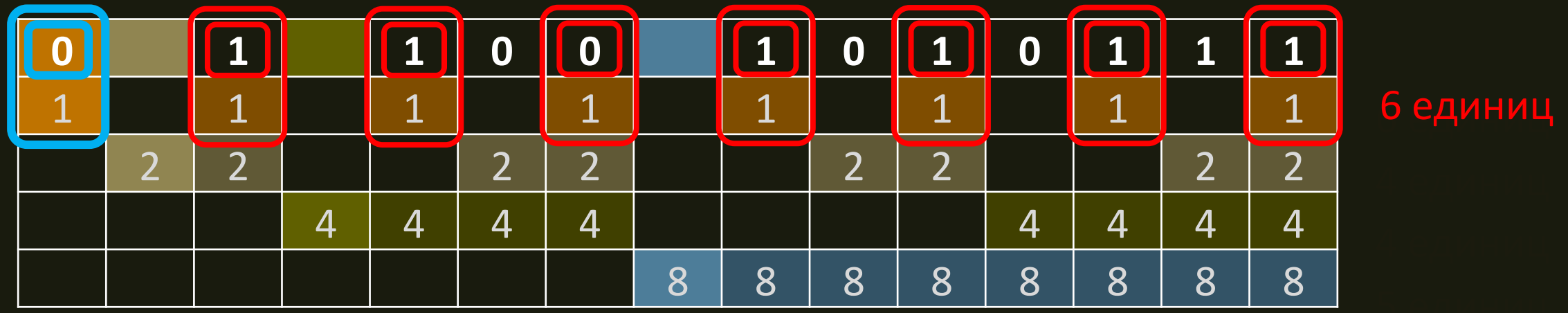
		1		1	0	0		1	0	1	0	1	1	1

Магия… готовь три

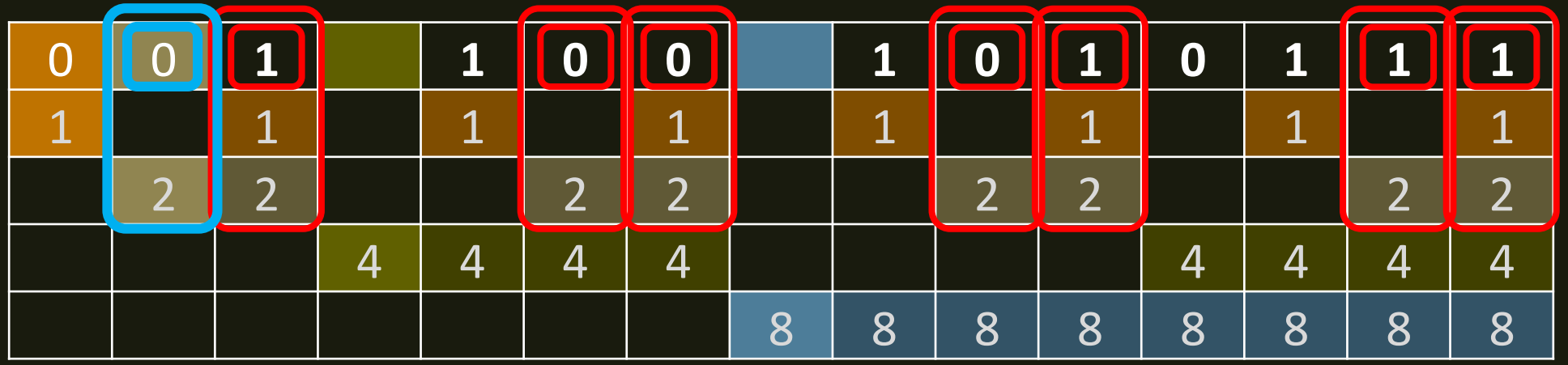
		1		1	0	0		1	0	1	0	1	1	1
1		1		1		1		1		1		1		1
	2	2			2	2			2	2			2	2
			4	4	4	4					4	4	4	4
							8	8	8	8	8	8	8	8

Магия… считаем единицы. Четыре

Магия… Ставим бит. Пять



Магия… Повторяем….



Магия… Готово….

0	0	1	0	1	0	0	1	1	0	1	0	1	1	1
1		1		1		1		1		1		1		1
	2	2			2	2			2	2			2	2
			4	4	4	4					4	4	4	4
							8	8	8	8	8	8	8	8

6 единиц

4 единиц

4 единиц

5 единиц

О нет! Помеха…

0	0	1	0	1	0	0	1	1	1	1	0	1	1	1
1		1		1		1		1		1		1		1
	2	2			2	2			2	2			2	2
			4	4	4	4					4	4	4	4
							8	8	8	8	8	8	8	8

6 единиц

4 единиц

4 единиц

5 единиц

Магия… Пересчитываем….

0	0	1	0	1	0	0	1	1	1	1	0	1	1	1
1		1		1		1		1		1		1		1
	2	2			2	2			2	2			2	2
			4	4	4	4					4	4	4	4
							8	8	8	8	8	8	8	8

8	4	2	1
1	**0**	**1**	**0**

Магия… Пересчитываем….

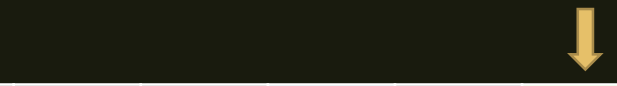
$1010_2 = 10_{10}$

0	0	1	0	1	0	0	1	1	1	1	0	1	1	1
1		1		1		1		1		1		1		1
	2	2			2	2			2	2			2	2
			4	4	4	4					4	4	4	4
							8	8	8	8	8	8	8	8

8	4	2	1
1	**0**	**1**	**0**

Магия… Пересчитываем….

$1010_2 = 10_{10}$



0	0	1	0	1	0	0	1	1	0	1	0	1	1	1
1		1		1		1		1		1		1		1
	2	2			2	2			2	2			2	2
			4	4	4	4					4	4	4	4
							8	8	8	8	8	8	8	8

8	4	2	1
1	0	1	0

Магия… Ещё разок….

0	0	1	0	1	0	0	1	1	0	1	0	1	1	1
1		1		1		1		1		1		1		1
	2	2			2	2			2	2			2	2
			4	4	4	4					4	4	4	4
							8	8	8	8	8	8	8	8

8	4	2	1

Проверка самостоятельно

0 0 1 0 1 0 0 1 1 1 1 0 1 1 1

Алгоритм

- 0 0 1 0 1 0 0 1 1 1 1 0 1 1 1
- 0 0 1 0 1 0 0 1 1 0 1 0 1 1 1
- 0 0 1 0 1 0 0 1 1 0 1 0 1 1 1
- 1 1 0 0 1 0 1 0 1 1 1
- 1 1 0 0 1 0 1 0 1 1 1 + … 000000
- 1100 1010 111+…00000
- CA …
- «К…»

Хемминг

1) **Без кода Хемминга.**

Если пересылать информацию блоками по m' бит с повторной пересылкой в случае обнаружения ошибки, то получим, что в среднем нам придётся переслать *D* бит:

$$D = Lm'\frac{1}{1-P_r}$$

Где $P_r = (1-(1-p)^{m'})(1-\varepsilon)$ — вероятность повторной передачи равная вероятности ошибки умноженной на вероятность того, что мы её заметим. Коэффициент раздувания равен

$$k(m,p,\varepsilon) = \frac{D}{M} = \frac{k_\varepsilon(m)}{\varepsilon + (1-\varepsilon)(1-p)^{k_\varepsilon(m)m}}$$

2) **С кодом Хемминга.**

При кодировании методом Хемминга слова длины m' получается слово длины *n* бит:

$$2^n = 2^{m'}(n+1),\quad k_\varepsilon(m)m = n - \log_2(n+1) \text{ (eq:hnm)}$$

Для отдельного блока вероятность безошибочной передачи равна $P_0 = (1-p)^n$. Вероятность одинарной ошибки $P_1 = np^1(1-p)^{n-1}$. Вероятность того, что произошло более чем одна ошибка, и мы это заметили

$$P_r = (1-P_0-P_1)(1-\varepsilon) = 1-\varepsilon-(1-\varepsilon)(1-p)^{n-1}(np+1-p)$$

— в этом случае требуется повторная передача кадра. Количество передаваемых данных:

$$D_H = Ln\frac{1}{1-P_r} = \frac{Ln}{\varepsilon + (1-\varepsilon)(1-p)^{n-1}(np+1-p)}$$

И коэффициент раздувания

$$k_H(m,p,\varepsilon) = \frac{n}{m\big(\varepsilon + (1-\varepsilon)(1-p)^{n-1}(np+1-p)\big)},$$

где $n(m)$ неявно определённая с помощью ((**eq:hnm**)) функция. Удобно записать соответствующие коэффициенты полезного содержания:

$$KPS = KPS_\varepsilon(n)\big(\varepsilon + (1-\varepsilon)(1-p)^n\big)$$

$$KPS_H = KPS_\varepsilon(m')\frac{m'}{n}\big(\varepsilon + (1-p)^{n-1}(np+1-p)(1-\varepsilon)\big),\ m' = n - \log_2(n+1) \text{ (eq:kps)}$$

Легко обнаружить что при $n > 3444$ и $p = 10^{-6}$ код Хемминга оказывается эффективнее, то есть $KPS_H/KPS > 1$

Практика