

## Основные понятия информатики

Человек слышит то, что понимает.

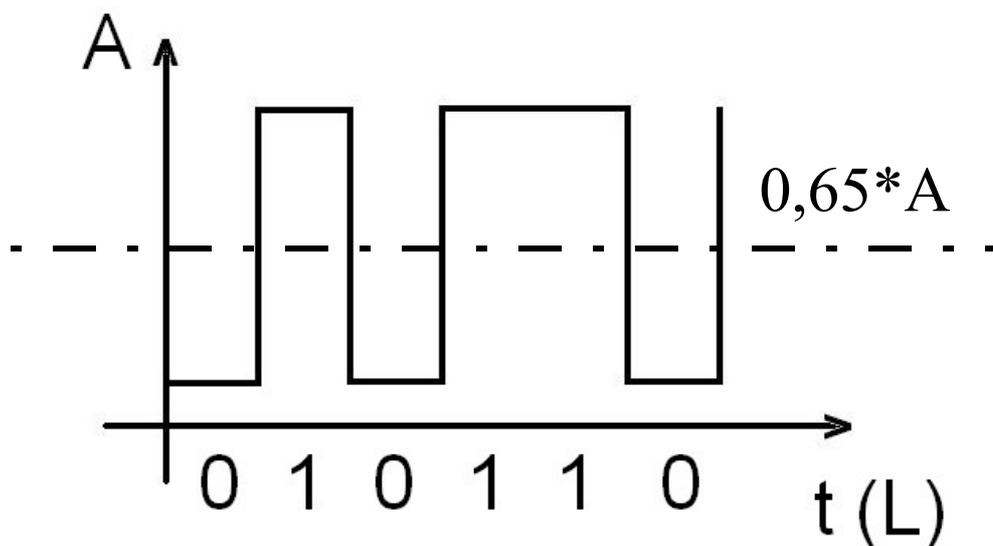
Гете

### Измерение и кодирование информации

**Информация** – специфический атрибут реального мира, представляющий собой его объективное отражение в виде совокупности сигналов и проявляющейся при взаимодействии с «приемником» информационных сигналов, позволяющим выделять, регистрировать эти сигналы из окружающего мира и по тому или иному критерию их идентифицировать.

**Сигнал** (от лат. *signum* – знак) – физический процесс (явление), несущий сообщение (информацию) о событии или состоянии объекта наблюдения.

Микросхемы компьютера работают на основе дискретного, двоичного сигнала:



$A$  – амплитудное значение.

$0,65*A$  – порог срабатывания триггера.

$t$  – развертка сигнала во времени.

$L$  – развертка сигнала в пространстве.

**Триггер** – электронная схема, обладающая двумя устойчивыми состояниями.

Переход из одного устойчивого состояния в другое происходит скачкообразно под воздействием управляющих сигналов. При этом также скачкообразно изменяется уровень напряжения на выходе триггера.

Триггеры служат основой для построения регистров, счетчиков и других элементов, обладающих функцией хранения.

**Измерение информации** – синтаксический (объемный, алфавитный, вероятностный) и семантический подходы к измерению информации обязаны своим возникновением науке кибернетики (Норберт Винер, 1948).

**Наименьшая единица** - бит (англ. binary digit - двоичное число) - единица измерения информационной емкости и количества информации, а также и еще одной величины – информационной энтропии (Википедия).

На физическом уровне бит является ячейкой памяти, которая в каждый момент времени находится в одном из двух состояний. Эти состояния можно обозначить как «0» или «1».

Байт (byte)	8 бит.
Слово (word)	2 байта или 16 бит.
Двойное слово (double word)	4 байта или 32 бита.

1 Килобайт=2<sup>10</sup> Байт =2<sup>10</sup> Байт=1024 Байт  
1 Мегабайт=2<sup>20</sup> Байт =2<sup>20</sup> Байт=1024 Килобайт  
1 Гигабайт=2<sup>30</sup> Байт =2<sup>30</sup> Байт=1024 Мегабайт

**Информационный объем сообщения** - количество двоичных разрядов (символов 0 или 1), используемое для кодирования этого сообщения. Количество возможных состояний двоичного сигнала из N разрядов соответствует в математике формула комбинаторики - количество возможных размещений A из n элементов с повторениями взятых по m раз.

$$A_n^m = n^m$$

В данном случае, в информатике N возможных размещений из 2 элементов (0, 1) взятых по N раз. Количество возможных состояний байта определяется формулой комбинаторики:

$$N = 2^H.$$

Например, для 1 байта это возможные размещения из двух элементов (0, 1), взятых по 8 раз. Таким образом, байт это такой двоичный сигнал, который может хранить одно из 256 возможных значений (состояний, элементов). Бит – хранит одно из 2 возможных значений. Определяя N через

$N$ , получаем формулу Ральфа Хартли. Количество информации, которое вмещает один символ  $N$ -элементного алфавита:

$$H = \log_2 N.$$

По-другому, это количество информации, полученное при выборе одного предмета из  $N$  равнозначных предметов

Двоичным кодом на основе 1 байта можно представлять 256 символов кодовой таблицы ASCII.

10-е число	2-е число	Символ ASCII	Код символа
0	00000000	Управляющи	0
1	00000001	е символы	1
2	00000010	...	2
3	00000011	...	3
...	...	...	...
13	...	Enter	13
...	...	...	...
32	100000	Пробел	32
...	...	...	...
48	110000	0	48
49	110001	1	49
50	110010	2	50
...	...	...	...
65	1000001	A	65
66	1000010	B	66
67	1000011	C	67
...	...	...	...
255	11111111	Пробел	255

Ввод символа по коду выполняется клавишей **Alt + Цифры** на дополнительной, цифровой, расширенной клавиатуре. Это же действие на языке программирования Pascal, Basic и др. выполняется функцией **chr**. Например, `chr(13)` - это Enter.

Повторим: Двоичным числом 00110001 можно представлять цифру 1 (это символ) или значение числа 49 (это же состояние байта – десятичное число).

### Формула Шеннона

Для равновероятных состояний информационного сигнала объема  $N$  и имеющего одно и  $N$  возможных, равновероятных состояний можно выполнить следующие преобразования формулы Хартли. Из формулы

вероятности  $P(A)=m/n$  (отношение  $m$  количества ожидаемого события  $A$  ко всем событиям  $n$ ) для равновероятных состояний информационного сигнала получим:

$$p(i)=1/N; N=1/p(i).$$

Учитывая, что сумма вероятностей всех возможных событий равна 1 и  $\log_2 (1/p(i)) = -\log_2 p(i)$ , получаем формулу Клода Шеннона.

Информационная двоичная энтропия для независимых случайных не равновероятных событий  $x$  с  $n$  возможными состояниями (от 1 до  $n$ ,  $p$  - функция вероятности) рассчитывается по формуле:

$$H(x) = - \sum_{i=1}^n p(i) \log_2 p(i).$$

Двоичным кодом на основе 1 байта можно представлять положительные десятичные числа в диапазоне от 0 до 255.

Двоичным кодом со смещенным порядком на основе 1 байта можно представлять десятичные числа в диапазоне от -128 до +127.

## Системы счисления

В позиционной системе счисления любое вещественное число любой системы такого счисления может быть представлено в следующем виде для получения значения числа:

$$A_q = \pm(a_{n-1} * q^{n-1} + a_{n-2} * q^{n-2} + \dots + a_1 * q^1 + a_0 * q^0 + a_{-1} * q^{-1} + a_{-2} * q^{-2} + \dots + a_{-m} * q^{-m}).$$

Здесь  $A$  - само число;

$q$  - основание системы счисления;

$a_i$  - цифры принадлежащие алфавиту данной системы счисления;

$n$  - количество целых разрядов числа;

$m$  - количество дробных разрядов числа;

Эту формулу используем для преобразования из различных систем счисления в десятичную систему счисления и привычного выполнения арифметических действий в понятной десятичной системе.

Например:

$$10101,01_2 = 1 * 2^4 + 0 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 + 0 * 2^{-1} + 1 * 2^{-2}$$

Обратное преобразование из десятичной в другие системы счисления выполняется отдельно для целой части числа делением на основание новой



## Представление чисел в цифровых автоматах

1) Естественная форма с фиксированной запятой.

Знак, целая часть, запятая, дробная часть.

2) Нормальная форма с плавающей запятой (нормализованное число).

Знак числа.

1 группа цифр – 1 цифра от 1 до 9, но не 0.

2 группа цифр – Мантисса (дробная часть < 1).

Знак порядка числа.

3 группа цифр – Порядок (целое число)

А) Целые числа без знака.

Б) Целые числа со знаком.

Чтобы получить **внутреннее представление целого положительного числа**  $N$ , хранящегося в  $K$ -разрядной ячейке, необходимо:

1. Перевести число  $N$  в двоичную систему счисления;

2. Полученный результат дополнить слева незначащими нулями до  $K$  разрядов.

Пример: Получить внутреннее представление целого числа 1607 в 2-х байтовой ячейке.

Решение:  $N=1607=11001000111_2$ .

Внутреннее представление этого числа будет: 0000 0110 0100 0111.

Шестнадцатеричная форма внутреннего представления числа: 0647.

**Дополнительным кодом** двоичного числа  $X$  в  $N$ -разрядной ячейке является число, дополняющее его до значения  $2^N$  (или  $N$ -разрядной ячейке - до значения  $2^N$ ).

Получение дополнительного кода:

1. Получить внутреннее представление положительного числа  $N$  (прямой код);

2. Получить обратный код этого числа заменой 0 на 1 и 1 на 0 (обратный или инвертированный код);

3. К полученному числу прибавить 1.

**Положительное число в прямом, обратном и дополнительном кодах не меняют свое изображение.**

Знаковые и беззнаковые двоичные числа.

### Прямой код

Первый разряд слева – **знак числа** (0 это плюс, 1 это минус).

$N-1$  разрядов для записи значащих цифр.

Пример: прямой код  $1101_2$  числа  $-5_{10}$ .

Преобразование числа  $1101_2$  в десятичный вид:

$$A_{10} = (-1)^1 * [1*2^2 + 0*2^1 + 1*2^0] = -1 * [4+0+1] = -5$$

Для представления отрицательных чисел используют обратный и дополнительный коды. Замена вычитания сложением с обратным или дополнительным кодом.

**Обратный код** получаем *инвертированием* всех цифр двоичного числа.

**0101** прямой код. **1010** обратный код. Старший разряд – знак числа.

Пример: обратный код числа  $0101_2$  ( $5_{10}$ ) будет  $1010_2$  ( $-5_{10}$ ).

Преобразование числа  $1010_2$  в десятичный вид:

$$A_{10} = 1 * (-2^3 + 1) + [0 * 2^2 + 1 * 2^1 + 0 * 2^0] = -7 + 2 = -5$$

**Дополнительный код**

Пример дополнительного кода для двух разрядного десятичного числа:

$$S = 84 + (-32) = 52$$

$$S = 84 + (100 - 32) = 84 + 68 = 152$$

Вычитание 32 заменяем сложением с дополнительным кодом 68.

1 выходит за разрядность числа и отбрасывается.

Получение дополнительного двоичного кода:

**Инверсия числа + 1 в младший разряд.**

Старший бит это знак числа.

**Пример вычитания:**

$$00000101 (\text{это } 5) - 00000100 (\text{это } 4)$$

Преобразуем:

$$1 \text{ шаг - инверсия числа } 4: \quad 11111011$$

$$2 \text{ шаг - } + 1 \text{ в младший разряд: } 11111100$$

$$(\text{т.е. из } 1 \text{ шага } 11111011 + 00000001 = 11111100)$$

Дополнительный код числа 4 это 11111100

Вычитание заменяем на сложение с дополнительным кодом.

Вычитание это сложение с дополнительным кодом:

$$00000101$$

+

$$11111100$$

-----

$$10000001$$

1 - отбрасывается.

**Дополнительный код на примере числа 5.**

0101 - это пять. Обратный код - 1010

Дополнительный код для числа 5:

$$1010 + 0001 = 1011$$

Десятичная запись отрицательного числа, представленного в дополнительном коде как 1011 будет иметь вид:

$$A_{10} = 1*(-2^3) + [0*2^2 + 1*2^1 + 1*2^0] = -8 + 0 + 2 + 1 = -5$$

Вычитание числа 5 заменяем на сложение с дополнительным кодом.

Десятичная запись отрицательного числа, представленного в обратном коде как 1010 будет иметь вид:

$$A_{10} = 1*(-2^3 + 1) + [0*2^2 + 1*2^1 + 0*2^0] = -7 + 0 + 2 + 0 = -5$$

Вычитание числа 5 заменяем на сложение с обратным кодом.

Для положительных чисел представление числа полностью совпадает с представлением в прямом и дополнительном кодах.

Положительные числа в прямом, обратном и дополнительном кодах изображаются одинаково - двоичными кодами с цифрой 0 в знаковом разряде.

Отрицательные числа при вводе в машину автоматически преобразуются в обратный или дополнительный код и в таком виде хранятся, перемещаются и участвуют в операциях.

Ещё пример:

Известен дополнительный код числа = 10110011

Преобразуем дополнительный код в десятичный вид:

$$\begin{aligned} A_{10} &= 1*(-2)^7 + 0 + 1*2^5 + 1*2^4 + 0 + 0 + 1*2^1 + 1 = \\ &= -128 + 32 + 16 + 2 + 1 = -77 \end{aligned}$$

Обратное преобразование:

1 шаг. Отнимем 1

$$10110011 - 00000001 = 10110010$$

2 шаг. Инвертируем

$$01001101$$

Преобразуем исходное число для дополнительного кода в десятичный вид:

$$\begin{aligned} A_{10} &= 0 + 1*2^6 + 0 + 0 + 1*2^3 + 1*2^2 + 0 + 1*2^0 = \\ &= 64 + 8 + 4 + 1 = 77 \end{aligned}$$

### Смещенный код

Пример для 8 разрядов двоичного числа. Порядок значений от -128 до +127 представляется смещенным порядком двоичного числа от 0 до 255.

# Представление символьной и графической информации в цифровых автоматах

## Представление символьной информации

Каждому символу соответствует отдельно бинарное слово-код.

## Представление графической информации

Модель RGB. По 8 бит (24 бит).

Красный. Зеленый. Синий.

Полноцветный режим (True Color) 16,5 млн.цветов.

Модель RGB. Для экрана.

Модель CMYK. Для бумаги.

## Архитектура ЭВМ

Классическая функциональная схема ЭВМ впервые предложена Дж. фон Нейманом.

### Принципы Джон фон Неймана (1903-1957)

1) ЭВМ работает на электронной основе в 2-ой системе (1,0).

2) В ее состав входят:  
арифметическое устройство (процессор),  
запоминающие устройства (ОЗУ, ПЗУ),  
устройства ввода/вывода.

3) В системе команд должны быть команды условного и безусловного переходов (ветвления).

4) ЭВМ состоит из двух основных частей:  
**линейно-адресуемой памяти**, слова которой хранят команды и элементы данных и **процессора**, выполняющего эти команды.

1. Принцип двоичного кодирования - для представления данных и команд используется двоичная система счисления.

2. Принцип однородности памяти - программы и данные хранятся в одной памяти и одной и той же системе счисления.

3. Адресуемость памяти - основная память состоит из пронумерованных ячеек, а процессору в любой момент времени доступна любая ячейка.

4. Принцип последовательного программного управления, когда команды выполняются последовательно, одна за другой.

5. Принцип условного перехода, когда команды не выполняются одна за другой, в программе появляются команды условного перехода, которые меняют последовательность выполнения команд.

**Процессор** – устройство для выполнения арифметико-логических операций и управления устройствами компьютера.

Обмен информацией между отдельными устройствами ЭВМ производится по трем многоуровневым шинам, соединяющим все модули: **шине данных, шине адресов и шине управления.**

Разрядность шины данных задается разрядностью процессора, т. е. количеством двоичных разрядов, которые процессор обрабатывает за один такт.

Данные по шине данных могут передаваться как от процессора к какому-либо устройству, так и в обратную сторону, т.е. шина данных является двунаправленной. К основным режимам работы процессора с использованием шины передачи данных можно отнести следующие: запись/чтение данных из оперативной памяти и из внешних запоминающих устройств, чтение данных с устройств ввода, пересылка данных на устройства вывода.

Подключение отдельных модулей компьютера к магистрали на физическом уровне осуществляется с помощью **контроллеров**, а на программном обеспечивается **драйверами**. Контроллер принимает сигнал от процессора и дешифрует его, чтобы соответствующее устройство смогло принять этот сигнал и отреагировать на него. За реакцию устройства процессор не отвечает - это функция контроллера. Поэтому внешние устройства ЭВМ заменяемы, и набор таких модулей произволен.

## **Запоминающие устройства ПЭВМ**

Static RAM. Dynamic RAM.

Регистровая память – кратковременное хранение.

Основная память – ОЗУ, ПЗУ.

ОЗУ - Random Access Memory.

ПЗУ – Read Only Memory

Характеристики:

Информационная емкость.

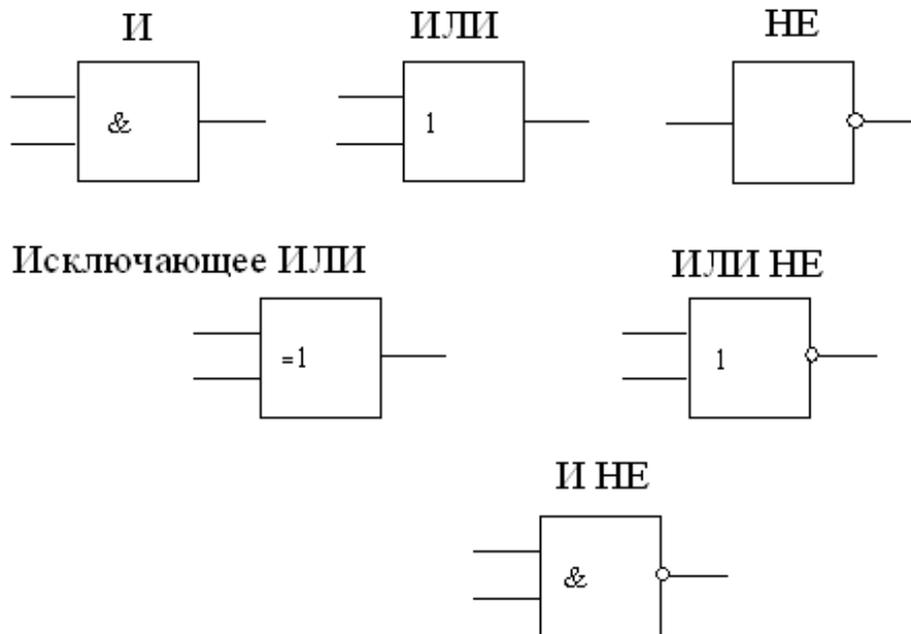
Время доступа.

Кэш-память - хранение копий, буфер между устройствами с различным быстродействием.

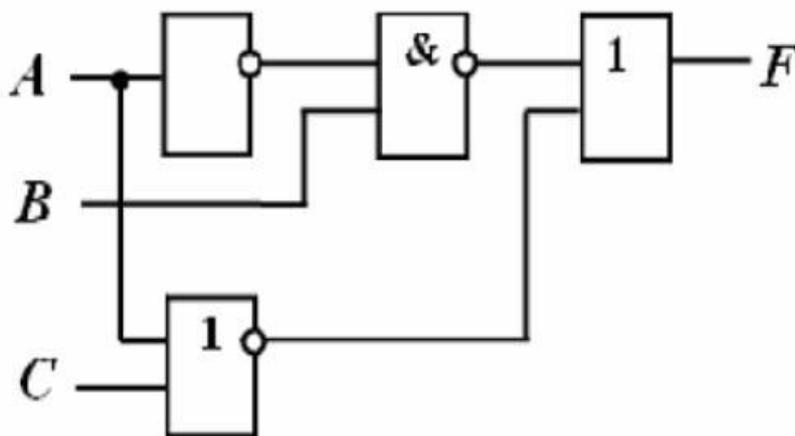
**Прерывание** - это прекращение выполнения текущей команды или текущей последовательности команд для обработки некоторого события специальной программой - обработчиком прерывания, с последующим возвратом к выполнению прерванной программы. Событие может быть вызвано особой ситуацией, сложившейся при выполнении программы, или сигналом от внешнего устройства. Прерывание используется для быстрой реакции процессора на особые ситуации, возникающие при выполнении программы и взаимодействии с внешними устройствами.

Механизм прерывания обеспечивается соответствующими аппаратно-программными средствами компьютера.

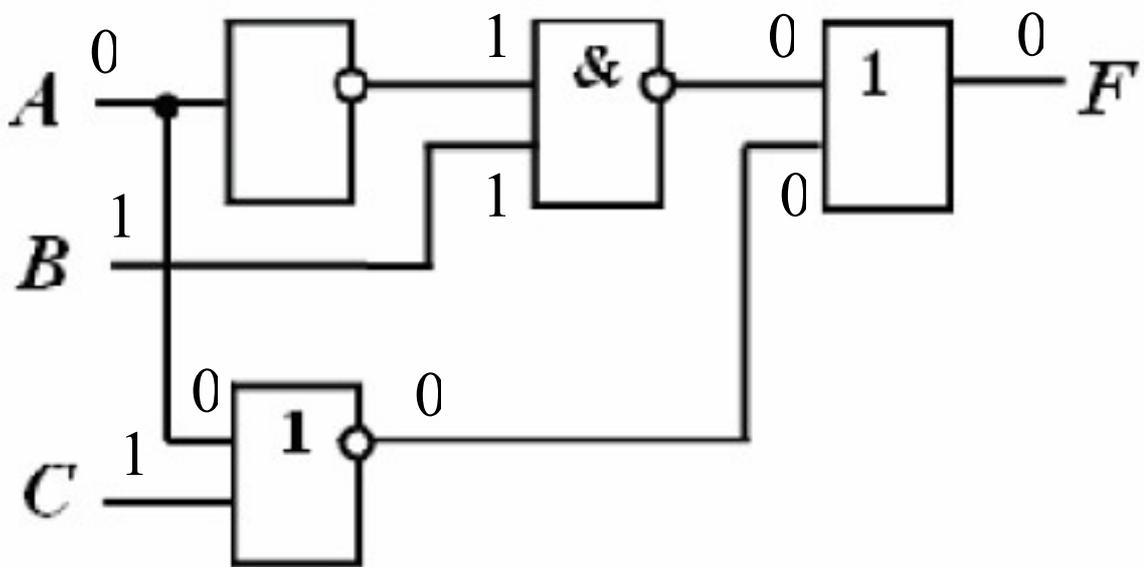
## Логические элементы



Пример задачи: Определить при каких входных значениях  $A$ ,  $B$ ,  $C$  значение  $F$  на выходе логической схемы будет равно 0.



**Решение:**



Придумать свой вариант логической схемы и свои значения переменных. Определить результат на выходе для своей логической схемы.

### **Задания для самопроверки**

1. Автоматическое устройство осуществило перекодировку информационного сообщения на русском языке длиной в N символов, первоначально записанного в 2-байтном коде Unicode, в 8-битную кодировку ANSI. На сколько бит уменьшилась длина сообщения?

Для вычисления N своего варианта воспользуйтесь формулой:

$N = n \cdot 10 + 11$ , где n - это номер по списку в журнале.

2. Для регистрации на сайте некоторой страны пользователю требуется придумать пароль. Длина пароля – ровно 8 символов. В качестве символов используются десятичные цифры и 12 различных букв местного алфавита, причём все буквы используются в двух начертаниях: как строчные, так и заглавные (регистр буквы имеет значение!). Под хранение каждого такого пароля на компьютере отводится минимально возможное и одинаковое целое количество байтов, при этом используется посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством битов. Определите объём памяти, который занимает хранение 50 паролей. Проверьте решения задачи для другого количества символов и паролей.

3. В программе используется одномерный целочисленный массив A с индексами от 0 до 9. Ниже представлен фрагмент программы, записанный

на разных языках программирования, в котором значения элементов сначала задаются, а затем меняются.

```
S:=0;
for i:=0 to 9 do
  A[i]:=9-i;
for i:=0 to 4 do
begin
  k:=A[i]; A[i]:=A[9-i]; A[9-i]:=k; S:=S+k;
end;
```

Чему будут равны элементы этого массива и значение переменной S после выполнения фрагмента программы?

4. У исполнителя «Утроитель» две команды, которым присвоены номера:

1. прибавь 1,
2. умножь на 3.

Первая из них увеличивает число на экране на 1, вторая – утраивает его. Запишите порядок команд в программе преобразования числа 1 в число 18, содержащей не более 5 команд, указывая лишь номера команд.

```
умножь на 3
прибавь 1
умножь на 3
прибавь 1
прибавь 1,
```

Например, 21211 – это программа, которая преобразует число 1 в 14.

Если таких программ более одной, то запишите несколько найденных примеров возможных программ.

5. Дан целочисленный массив из N элементов. Элементы массива могут принимать целые значения от  $-1000$  до  $1000$ . Опишите на русском языке или на одном из языков программирования алгоритм, позволяющий найти и вывести минимальное значение среди элементов массива, которые имеют чётное значение и не делятся на три. Гарантируется, что в исходном массиве есть хотя бы один элемент, значение которого чётно и не кратно трем. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из них.

```
const
  N=___;
var
  a: array [1..N] of
integer;
  i, j, min: integer;
begin
  for i:=1 to N do
```

```
readln(a[i]);
```

```
...
```

```
end.
```

Для вычисления  $N$  воспользуйтесь формулой:  $N = n \cdot 10 + 11$ , где  $n$  - это номер по списку в журнале.

6. Найдутся ли десятизначные числа, делящиеся на 11, в записи каждого из которых использованы все цифры от 0 до 9? Рассматривая признаки делимости числа на 11, написать программу для нахождения ответа задачи по одному из признаков делимости. Предусмотреть возможность изменения разрядности проверяемого числа.

**Признаки делимости на 11**

*Признак 1:* Число делится на 11 тогда и только тогда, когда модуль разности между суммой цифр, занимающих нечётные позиции, и суммой цифр, занимающих чётные места делится на 11. Или число делится на 11, если знакопеременная сумма его цифр (последняя цифра со знаком +) делится на 11. Например, **9163627** делится на 11, так как 22 (28-6) делится на 11

*Признак 2:* Число делится на 11 тогда и только тогда, когда на 11 делится сумма чисел, образующих группы по две цифры (начиная с единиц). Например, 103785 делится на 11, так как 132 (85+37+10) делится на 11.

*Признак 3:* Число делится на 7, 11 или 13, если знакопеременная сумма чисел, образованных тройками его цифр, взятыми с конца (последнее число со знаком +), делится на 7, 11 или 13 соответственно.

7. Какое десятичное число и какой символ может представлять двоичное число 00110001?

Найти другие примеры тройки значений в последовательности:

Двоичное число    Его десятичное значение. Соответствующий символ в кодовой таблице.

### **Использованные первоисточники:**

1. [www.wikipedia.ru](http://www.wikipedia.ru)
2. [www.intuit.ru](http://www.intuit.ru)
3. [www.fepo.ru](http://www.fepo.ru)
4. Демонстрационный вариант ЕГЭ 2012 г. ИНФОРМАТИКА и ИКТ, 11 класс.
5. Демонстрационный вариант ЕГЭ 2013 г. ИНФОРМАТИКА и ИКТ, 11 класс.
6. Яшина Е.Ю, Яшин Е.А. Арифметические основы ЭВМ. ЭУП. Гимназия №5, Давлеканово. 2004.