Управление образования
администрации муниципального образования
городского округа «Сыктывкар»

# МЕТОДИЧЕСКИЕ ОСОБЕННОСТИ ПОДГОТОВКИ УЧАЩИХСЯ К ВЫПОЛНЕНИЮ ЗАДАНИЙ ПОВЫШЕННОГО УРОВНЯ СЛОЖНОСТИ ЕГЭ ПО ИНФОРМАТИКЕ И ИКТ

г. Сыктывкар
2016 г.

Управление образования
администрации муниципального образования
городского округа «Сыктывкар»

В сборнике представлены материалы по выполнению заданий повышенного уровня сложности 1 части ЕГЭ по информатике и ИКТ 2016 года.

Материалы публикуются в авторской редакции.

# Содержание

# Введение

В сентябре 2015 года муниципальным учреждением «Информационно-методический центр» г. Сыктывкара проведён анализ выполнения выпускниками заданий ЕГЭ по информатике и ИКТ в 2015 году.

В таблице представлена информация о заданиях, процент выполнения которых ниже 40%.

| № задания в ЕГЭ | Тема | Процент выполнения на ЕГЭ 2015 |
|---|---|---|
| №6 | Анализ алгоритмов | 39% |
| № 9 | Объем информации. Передача информации | 36% |
| № 10 | Комбинаторика | 29% |
| № 11 | Рекурсивные алгоритмы | 11% |
| № 12 | Адресация в сети | 36% |
| № 13 | Количество информации | 32% |
| № 14 | Алгоритмы с исполнителем | 18% |
| № 16 | Системы счисления | 18% |
| № 18 | Логические выражения. Отрезки, множества, функции | 7% |
| № 22 | Перебор вариантов (количество программ) | 11% |

В рамках подготовка учащихся к ЕГЭ по информатике и ИКТ в 2015 – 2016 учебном году для учащихся 11-х классов проведён разбор типовых задач, вызвавших затруднения на ЕГЭ в 2015 году.

По итогам практических занятий составлен сборник рекомендаций по подготовке к ЕГЭ по информатике и ИТК.

# Методические особенности подготовки учащихся к выполнению заданий повышенного и высокого уровней сложности ЕГЭ по информатике и ИКТ.

***Лапшина Татьяна Авенировна,***
*учитель информатики*
*МАОУ «Лицей № 1» г. Сыктывкара*

## Анализ алгоритмов[1]

***Задание № 6*** ЕГЭ по информатике и ИКТ 2016 года
*базовый уровень, время – 4 мин*

***Проверяемые элементы содержания:*** формальное исполнение алгоритма, записанного на естественном языке или умение создавать линейный алгоритм для формального исполнителя с ограниченным набором команд.

**Исполнитель Автомат**

***Теория:***

Цифры числа находятся в диапазоне от 0 до 9, первая цифра – от 1 до 9.
Сумма цифр числа находятся в диапазоне от 0 до 18.
Произведение цифр числа находятся в диапазоне от 0 до 81.

*Общий вид задачи:*

Автомат получает на входе n-значное число, из цифр этого числа по некоторым правилам получают новое число. Указать наибольшее или наименьшее число, которое может выдать автомат.

*Решение:*

1. Разбиваем число-результат полученный от автомата на два числа стоящие по возрастанию, убыванию, невозрастанию или неубыванию (в зависимости от условия задачи).
2. Из каждого числа получаем цифры. При произведении раскладываем на множители, при сумме – на слагаемые.
3. Для наибольшего числа первая цифра – бóльшая, последняя – мéньшая. Для наименьшего числа первая цифра – мéньшая, последняя – бóльшая.

***Примеры заданий:***

**№ 1.** Автомат получает на вход трёхзначное число. По этому числу строится новое число по следующим правилам.

1. Перемножаются первая и вторая, а также вторая и третья цифры.
2. Полученные два числа записываются друг за другом в порядке неубывания без разделителей.

*Пример.* Исходное число: 631. Произведение: 6 × 3 = 18; 3 × 1 = 3. Результат: 318.

Укажите наибольшее число, при обработке которого автомат выдаёт результат 621.

*Решение:*

1. Разбиваем число-результат, полученный от автомата 621 на два числа записанные в порядке неубывания 6 и 21.

[1] В работе используются материалы сайта https://inf-ege.sdamgia.ru/

2. Разбиваем числа на множители, так, чтобы был общий множитель: 6 = 2 × 3, 21 = 7 × 3, 3 – общая цифра, значит стоит в середине числа.
3. Из цифр получаем наибольшее число: 732

*Ответ:* 732.

**№ 2.** Автомат получает на вход четырёхзначное число. По этому числу строится новое число по следующим правилам:

1. Перемножаются первая и вторая, а также третья и четвёртая цифры исходного числа.
2. Полученные два числа записываются друг за другом в порядке убывания (без разделителей).

*Пример.* Исходное число: 2466. Произведения: 2 × 4 = 8; 6 × 6 = 36.
Результат: 368.

Укажите наименьшее число, в результате обработки которого автомат выдаст число 124.

*Решение:*

1. Разбиваем число-результат, полученный от автомата 124 на два числа записанные в порядке убывания 12 и 4.
2. Разбиваем числа на множители: 12 = 2 × 6, 4 = 4 × 1, разбиваем числа таким образом, чтобы одна из цифр бала минимальным множителем.
3. Из цифр получаем наименьшее число: 1426.

*Ответ:* 1426

**№ 3.** Автомат получает на вход трёхзначное число. По этому числу строится новое число по следующим правилам.

1. Складываются первая и вторая, а также вторая и третья цифры исходного числа.
2. Полученные два числа записываются друг за другом в порядке убывания (без разделителей).

*Пример.* Исходное число: 348. Суммы: 3 + 4 = 7; 4 + 8 = 12. Результат: 127.

Сколько существует чисел, в результате обработки которых автомат выдаст число 1715?

*Решение:*

1. Разбиваем число-результат, полученный от автомата 1715 на два числа записанные в порядке убывания 17 и 15.
2. Разбиваем числа на слагаемые (цифры): 17 = 9 + 8, 15 = 9 + 6 = 8 + 7
3. Из цифр получаем числа: 987, 789, 698, 896

*Ответ:* 4

**№ 4.** Автомат получает на вход четырёхзначное десятичное число. По этому числу строится новое число по следующим правилам.

1. Складываются первая и вторая, а также третья и четвёртая цифры.
2. Полученные два числа записываются друг за другом в порядке возрастания (без разделителей).

*Пример.* Исходное число: 8754. Суммы: 8 + 7 = 15; 5 + 4 = 9. Результат: 915.

Определите, сколько из приведённых ниже чисел могут быть получены, как результат работы автомата.  1419  1518  406  911.

*Решение:*

1. Разбиваем числа-результаты, полученные от автомата на два числа, записанные в порядке возрастания 1419 = 14 + 19, 1518 = 15 + 18, 406 = 4 + 06, 911 = 9 + 11.
2. Числа 19 и 06 не попадают в диапазон суммы (см. теорию). Числа 1419 и 406 не являются результатом работы автомата.
3. Результатом работы автомата являются два числа: 1518, 911

*Ответ:* 2

**№ 5.** Автомат получает на вход четырёхзначное десятичное число. По этому числу строится новое число по следующим правилам.

1. Складываются первая и вторая, а также третья и четвёртая цифры.
2. Полученные два числа записываются друг за другом в порядке убывания (без разделителей).

*Пример.* Исходное число: 5487. Суммы: 5 + 4 = 9; 8 + 7 = 15. Результат: 159.

Определите, сколько из приведённых ниже чисел могут быть получены, как результат работы автомата. 199 188 21 212.

*Ответ:* 2

**№ 6.** Автомат получает на вход четырёхзначное восьмеричное число. По этому числу строится новое число по следующим правилам.

1. Складываются первая и вторая, а также третья и четвёртая цифры.
2. Полученные два восьмеричных числа записываются друг за другом в порядке возрастания (без разделителей); основание системы счисления не пишется.

*Пример.* Исходное число: $6331_8$. Суммы: $6 + 3 = 11_8$; $3 + 1 = 4_8$. Результат: 411.

Определите, сколько из приведённых ниже чисел могут быть получены, как результат работы автомата. 812 617 1512 1213.

*Решение:*

1. Разбиваем числа-результаты полученный от автомата на два числа, записанные в порядке возрастания 812 = 8 + 12, 617 = 6 + 17, 1512 = 1 + 512, 1213 = 12 + 13

2. В восьмеричной системе счисления диапазон цифр от 0 до 7, диапазон суммы цифр от 0 до 14. Поэтому отбрасываем числа 812, 617, 1512.

*Ответ:* 1

**Смещение чисел**

***Теория:***

При сдвиге влево все биты числа в ячейке (регистре) сдвигаются на 1 бит влево, в младший бит записывается нуль, старший бит пропадает.

Если в старшем разряде двоичного числа нет единицы, то при смещении на одну позицию влево число удваивается.

*Например:*

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **1** | **1** | **0** | **1** | **1** | **0** | = 54 |
| **0** | **1** | **1** | **0** | **1** | **1** | **0** | **0** | = 108 |

Если единица есть (т.е. десятичное число не меньше 128), то выводится остаток от деления удвоенного числа на 256.

*Например:*

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **1** | **0** | **1** | **1** | **0** | **1** | **1** | **0** | = 182 |
| **0** | **1** | **1** | **0** | **1** | **1** | **0** | **0** | = 108 |

$$182 \times 2 = 364,$$
$$364 \div 256 = 1 \text{ (108 в остатке)}$$

При сдвиге вправо все биты числа в ячейке (регистре) сдвигаются на 1 бит вправо, в старший бит записывается нуль, а младший бит попадает в специальную ячейку – бит переноса, т.е. он теряется.

Следовательно,

– если число чётное, то при сдвиге мы получаем число, в два раза меньше исходного;

*Например:*

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **1** | **1** | **0** | **1** | **1** | **0** | = 54 |
| **0** | **0** | **0** | **1** | **1** | **0** | **1** | **1** | = 27 |

$$54 \div 2 = 27,$$

– если число нечётное, в два раза меньше ближайшего меньшего чётного числа.

*Например:*

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **0** | **0** | **1** | **1** | **0** | **1** | **1** | **1** | = 55 |
| **0** | **0** | **0** | **1** | **1** | **0** | **1** | **1** | = 27 |

Для числа 55 ближайшее меньшее чётное – 54, $54 \div 2 = 27$

**№ 1.** У исполнителя, который работает с положительными однобайтовыми двоичными числами, две команды, которым присвоены номера:

1. сдвинь вправо
2. прибавь 4

Выполняя первую из них, исполнитель сдвигает число на один двоичный разряд вправо, а выполняя вторую, добавляет к нему 4. Исполнитель начал вычисления с числа 191 и выполнил цепочку команд 112112. Запишите результат в десятичной системе.

*Решение:*

Выполнение команд начинаем с числа 191

1. (1. сдвинь вправо): 191 – число нечётное, в два раза меньше ближайшего чётного числа, т.е. $190 \div 2 = 95$
2. (1. сдвинь вправо): 95 – число нечётное, значит $94 \div 2 = 47$
3. (2. прибавь 4): $47 + 4 = 51$
4. (1. сдвинь вправо): 51 – число нечётное, значит $50 \div 2 = 25$
5. (1. сдвинь вправо): 25 – число нечётное, значит $24 \div 2 = 12$
6. (2. прибавь 4): $12 + 4 = 16$

*Ответ:* 16

**№ 2.** У исполнителя, который работает с положительными однобайтовыми двоичными числами, две команды, которым присвоены номера:

1. сдвинь влево
2. вычти 1

Выполняя первую из них, исполнитель сдвигает число на один двоичный разряд влево, а выполняя вторую, вычитает из него 1. Исполнитель начал вычисления с числа 91 и выполнил цепочку команд 112112. Запишите результат в десятичной системе.

*Решение:*

Выполнение команды начинается с числа 91

1. (1. сдвинь влево): 91 < 128, следовательно, число 91 увеличиваем в два раза => 182,
2. (1. сдвинь влево): 182 > 128, следовательно, 182 × 2 = 364, 364 ÷ 256 = 1 (остаток 108),
3. (2. вычти 1): 108 – 1 = 107,
4. (1. сдвинь влево): 107 < 128, следовательно, число увеличиваем в два раза => 214,
5. (1. сдвинь влево): 214 > 128, 214 × 2 = 428, 428 ÷ 256 = 1 (остаток 172),
6. (2. вычти 1): 172 – 1 = 171.

*Ответ:* 171

**Кодирование чисел**

**№ 1.** На вход алгоритма подаётся натуральное число N. Алгоритм строит по нему новое число R следующим образом:

1. Строится двоичная запись числа N.
2. К этой записи дописываются справа ещё два разряда по следующему правилу:
   а). Складываются все цифры двоичной записи, и остаток от деления суммы на 2 дописывается в конец числа (справа). Например, запись 11100 преобразуется в запись 111001;
   б). над этой записью производятся те же действия – справа дописывается остаток от деления суммы цифр на 2.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью искомого числа R.

Укажите минимальное число R, которое превышает 43 и может являться результатом работы алгоритма. В ответе это число запишите в десятичной системе.

*Решение:*

1. Переведём число 43 в двоичную систему счисления: $43_{10} = 101011_2$
2. Если в числе было нечётное количество единиц, то в конец допишется 10. Если чётное, то 00. Таким образом, нужно найти первое число, большее 43, у которого в двоичной записи чётное количество единиц, а на конце 10 или 00.
3. Имеем: $43_{10} = 101011_2$, найдём числа большее 43 оканчивающиеся в двоичной записи цифрами 00 или 1 0. Запишем последовательность чисел: 101100, 101101, 101110, …

$101100_2$, $101101_2$ – не удовлетворяют условию получения чисел (нечётное число единиц, 0 не может стоять на предпоследнем месте)

$46_{10} = 101110_2$ – удовлетворяет условию

*Ответ:* 46

**№ 2.** В некоторой информационной системе информация кодируется двоичными шестиразрядными словами. При передаче данных возможны их искажения, поэтому в конец каждого слова добавляется седьмой (контрольный) разряд таким образом, чтобы сумма разрядов нового слова, считая контрольный, была чётной. Например, к слову 110011 справа будет добавлен 0, а к слову 101100 – 1.

После приёма слова производится его обработка. При этом проверяется сумма его разрядов, включая контрольный. Если она нечётна, это означает, что при передаче этого слова произошёл сбой, и оно автоматически заменяется на зарезервированное слово 0000000. Если она чётна, это означает, что, сбоя не было или сбоев было больше одного. В этом случае принятое слово не изменяется.

Исходное сообщение

1100101 1001011 0011000

было принято в виде

1100111 1001110 0011000.

Как будет выглядеть принятое сообщение после обработки?

*Решение:*

Проверяем суммарный код у каждого полученного слова на чётность: 1100111 – нечётное, заменяем нулями. 1001110 – чётное, оставляем без изменений. 0011000 – чётное, оставляем без изменений.

*Ответ:* 0000000 1001110 0011000

**Исполнитель Кузнечик**

***Теория:***

Вводим две переменные: x – движение исполнителя вперёд, y – движение исполнителя назад. Составляем уравнение с двумя переменными и накладываем ограничение на одну из переменных. Значение второй переменной находим методом подбора.

**№ 1.** Исполнитель КУЗНЕЧИК живёт на числовой оси. Начальное положение КУЗНЕЧИКА – точка 0. Система команд Кузнечика:

Вперёд 7 – *Кузнечик* прыгает вперёд на 7 единиц,

Назад 5 – *Кузнечик* прыгает назад на 5 единиц.

Какое наименьшее количество раз должна встретиться в программе команда «Назад 5», чтобы Кузнечик оказался в точке 19?

*Решение:*

1. Вводим переменные: x раз – движение вперёд, y раз – движение назад.
2. Составляем уравнение: 7x – 5y = 19
3. Выразим переменную y через x: $y = \frac{7x-19}{5}$
4. Накладываем ограничения на переменную: x, y – целые положительные числа, значит: 7x – 19 > 5, следовательно 7x >= 24, x > 3
5. Подбираем значение переменной x, при x = 7 находим значение переменной y, y = 6

*Ответ:* 6

## Адреса сайтов для самоподготовки:

1. http://inf.ege.sdamgia.ru
2. http://somit.ru/14_2015.html
3. http://kpolyakov.spb.ru/school/ege.htm

# Объём информации. Передача информации.

***Полутова Дина Валентиновна****,*
*учитель информатики*
*МАОУ «Средняя общеобразовательная школа № 27»*
*г. Сыктывкара*

***Задание № 9*** ЕГЭ по информатике и ИКТ 2016 года.

***Проверяемые элементы содержания:*** умение определять скорость передачи информации при заданной пропускной способности канала, объем памяти, необходимый для хранения звуковой и графической информации.

## 1. Объем памяти для хранения графической информации

Объяснение материла можно предложить в форме диалога, когда учащиеся, основываясь на понятиях «Компьютерной графики», почти самостоятельно приходят к формулам, запоминание которых в большинстве случаев и вызывают наибольшую трудность.

Начать объяснение материала можно с постановки вопроса: «Как формируется изображение на экране монитора?» Как правило, учащиеся хорошо помнят из курса Информатики и ИКТ 10 класса, что изображение формируется из отдельных точек – пикселей, образующих графическую сетку – матрицу M × N пикселей.

Далее следует постановка вопроса: «Какое изображение «весит» больше: черно-белое или цветное?». Учащиеся, как правило, отвечают, что цветное, забывая о размере изображения. Если все-таки, возникнет вопрос о размерах выше обозначенных изображений, можно добавить, что изображения равны.

Следующий вопрос: «А почему цветное весит больше?». При затруднении демонстрируем примеры, задавая вопросы: «Сколько цветов в изображении? Какова длина кода, чтобы закодировать все цвета?»

*Пример 1*


*Пример 2*

*Пример 3*. Рассмотрите 8 – цветное изображение.


Делаем ***вывод 1***: Чем больше цветов, тем длиннее двоичный код одного пикселя, т.е. тем больше глубина цвета *i*.

Делаем ***вывод 2***: Объём памяти для хранения графической информации *I* зависит от глубины цвета *i*.

Вспоминаем «главную» формулу информатики: $N = 2^i$, где

N – количество возможных исходов события;

i – количество информации в сообщении об одном из исходов.

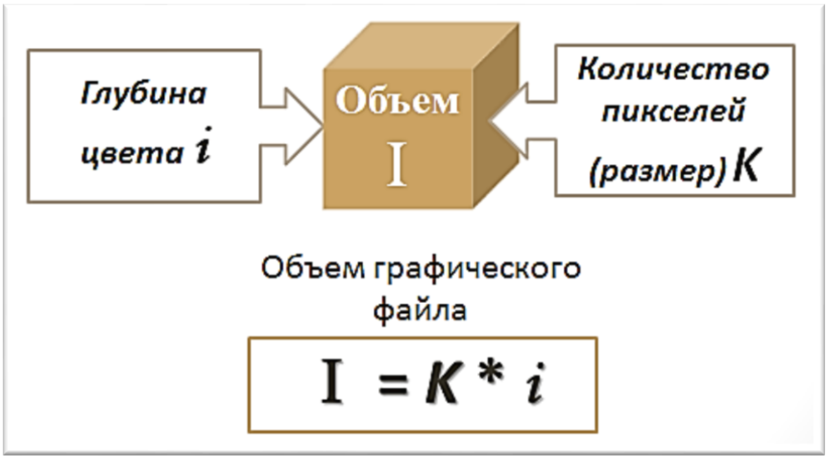
В случае с компьютерной графикой, где:

N – количество цветов в палитре;

i – количество бит для кодировки одного пикселя.

Далее снова следует постановка вопроса: «Какое изображение «весит» больше: черно-белое или цветное?». Учащиеся, как правило, отвечают, что уже отвечали на этот вопрос, и тут надо задать следующий вопрос: «А если цветное изображение размером 10 × 10, а черно-белое закрывает весь экран монитора?» Учащиеся делают вывод, что на объём изображения влияет размер изображения.

Записываем формулу:




Далее следует решить задачу на использование данных формул с указанных сайтов:

1. http://kpolyakov.spb.ru/school/ege.htm;
2. https://inf-ege.sdamgia.ru/.

## 2. Объем памяти для хранения звуковой информации

Вспоминаем, что звук – информация аналоговая. Для того, чтобы сохранить и передать звуковую информацию, требуется провести процесс дискретизации (*рис. 1*). В этом случае, запомнить требуются все точки получившегося дискретного графика. Вопрос: «Будет ли одинаково звучать звук аналоговый и звук дискретный?»

Ответ: «Нет, т.к. графики совсем разные»

Вопрос: «В каком случае звук аналоговый и дискретный будут практически идентичны друг другу?»

Ответ: «В случае, когда: 1) шаг дискретизации будет стремиться к нулю;

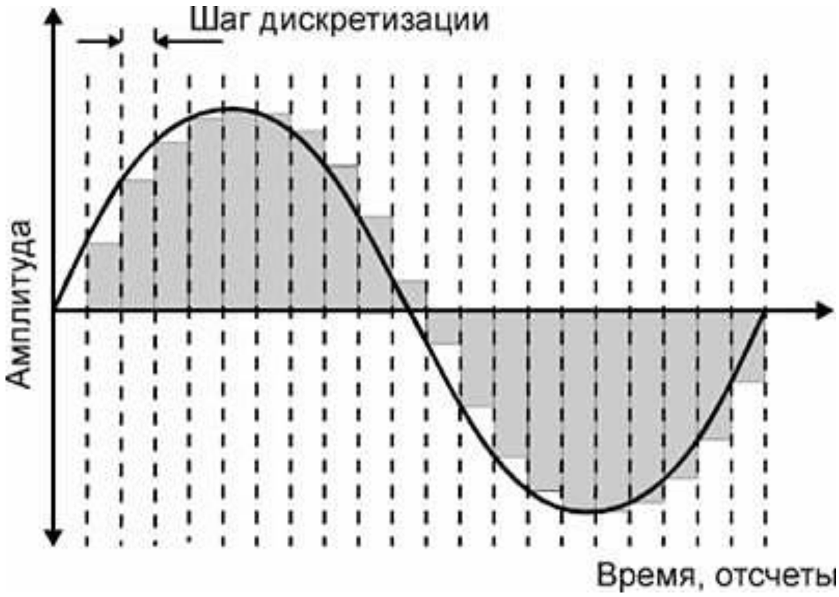



*Рис. 1*

Делаем ***вывод 1***:чем чаще производить замеры, тем больше объём звукового файла.

Частота дискретизации звука **ν** – это количество измерений громкости звука за одну секунду.

Единицы измерения: 1 Гц = 1/с; 1 кГц = 1000/с.

Ещё раз задаём вопрос: «Ещё в каком случае звук аналоговый и дискретный будут практически идентичны друг другу?»

Ответ: «В случае, когда: 1) шаг дискретизации будет стремиться к нулю; 2) «ступени» амплитуды будут стремиться к бесконечности».

Амплитуда, т.е. уровень громкости имеет название «глубина дискретизации звука» или иногда имеет название «разрешение звука». Проводим аналогию с графическим файлом. Там «глубина цвета», тут – «глубина дискретизации звука». Аналогично вспоминаем «главную» формулу информатики: $N = 2^i$, где

N – количество возможных исходов события;

i – количество информации в сообщении об одном из исходов.

В случае со звуковым файлом, где:

N – количество уровней громкости;

i – количество бит для кодировки одного уровня.

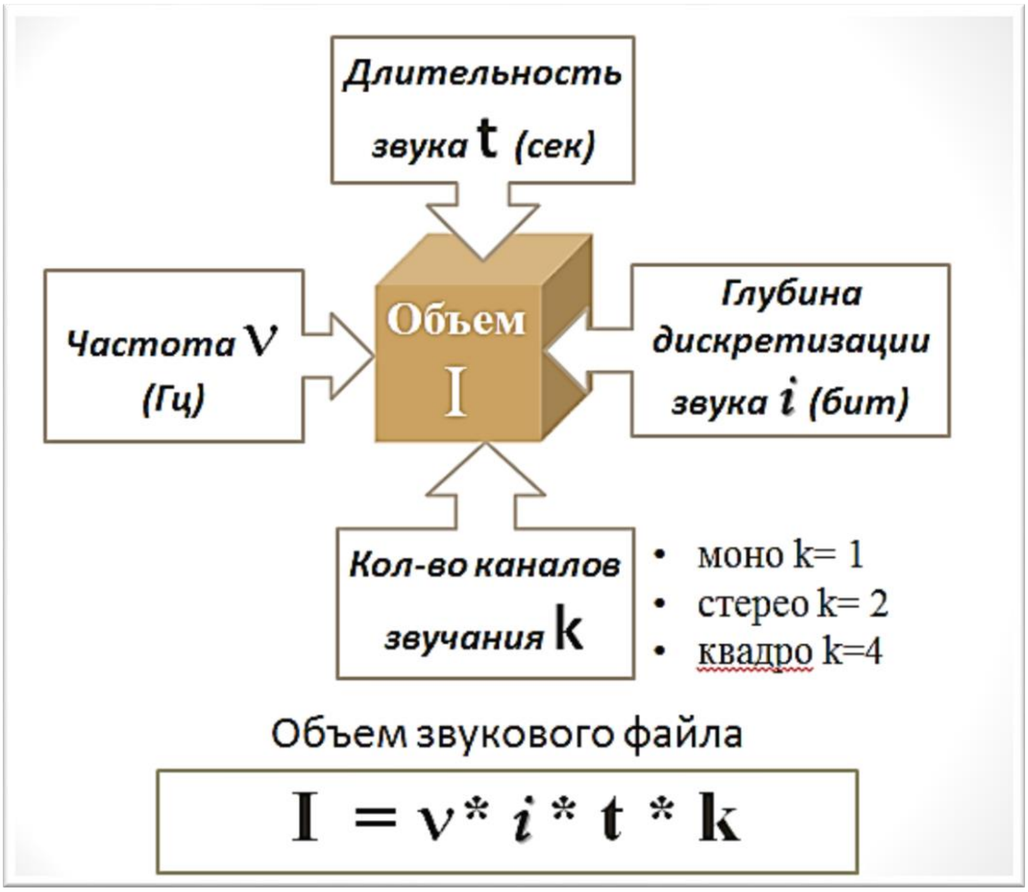
Делаем ***вывод 2***: чем больше глубина дискретизации звука, тем больше объём звукового файла.

Вопрос: «Какой звуковой файл будет иметь наибольший объём: аудиоклип длиной 3 минуты или аудиокнига длительностью 2 часа (при прочих равных условиях)?»

Делаем ***вывод 3***: объём звукового файла зависит от длительности звучания.

Вопрос: «Какой звуковой файл будет иметь наибольший объём: монофайл или стереофайл (при прочих равных условиях)?»

Делаем ***вывод 4***: объём звукового файла зависит от количества каналов звучания.




Далее следует решить задачу на использование данных формул с указанных сайтов:

1. http://kpolyakov.spb.ru/school/ege.htm;
2. https://inf-ege.sdamgia.ru/.

## 3. Определение скорости передачи информации при заданной пропускной способности канала[2]

Канал связи можно сравнить с обычной трубой для подачи воды или газа.

Пропускная способность (или скорость передачи) – 10 л/час

Сколько воды перекачается по трубе за 1 час?

Ответ: 10 л/мин × 60 мин = 600 л

**I = V × t**, отсюда выражаем скорость передачи информации $\boldsymbol{V = \frac{I}{t}}$

[2] В объяснении данной темы использованы материалы сайта Константина Полякова http://kpolyakov.spb.ru/school/ege.htm

## Количество информации.

***Задание № 13*** ЕГЭ по информатике и ИКТ 2016 года

***Проверяемые элементы содержания:*** умение подсчитывать информационный объем сообщения.

### 4. Объём памяти для хранения текстовой информации

Вспоминаем, что текст состоит из символов. Множество символов составляют алфавит языка. В зависимости от мощности алфавита (полного количества символов в алфавите), информационный вес символа может быть различным.

Вспоминаем, что информационный вес символа равен разрядности двоичного кода, которым закодирован этот символ.

| **Порядковый номер символа** | 1 | 2 |
|---|---|---|
| **Двоичный код** | 0 | 1 |

2 символа – 1 бит

| **Порядковый номер символа** | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| **Двоичный код** | 00 | 01 | 10 | 11 |

4 символа – 2 бита

| **Порядковый номер символа** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| **Двоичный код** | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |

8 символов – 3 бита

Анализируя таблицы, делаем ***вывод 1***: объём звукового файла зависит от информационного веса символа, т.е. количества бит (разрядности) для кодировки одного символа.

Выводим соотношение.

Вспоминаем «главную» формулу информатики: $N = 2^i$, где

N – количество возможных исходов события;

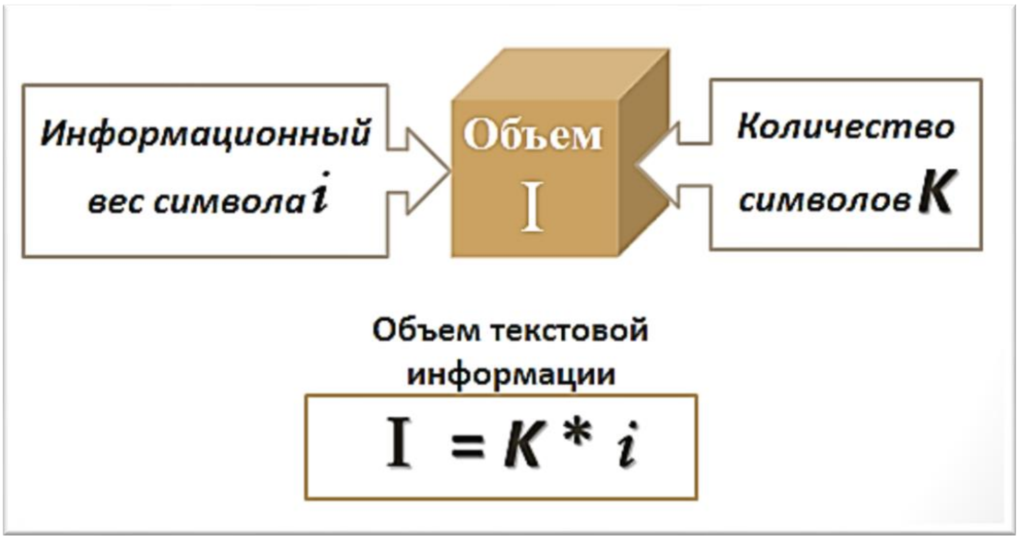
i – количество информации в сообщении об одном из исходов.

В случае с текстовой информацией, где:

N – мощность алфавита – полное количество символов в алфавите;

i – количество бит (разрядность) для кодировки одного символа.

Задаём вопрос: Что «весит» больше стихотворение А. Барто «Уронили мишку на пол…» или 4 тома романа Л.Н. Толстого «Война и мир»? Ответ очевиден, на его основании делаем ***вывод 2***: объём звукового файла зависит от количества символов в сообщении.




Далее следует решить задачу на использование данных формул с указанных сайтов:

1. http://kpolyakov.spb.ru/school/ege.htm;
2. https://inf-ege.sdamgia.ru/.

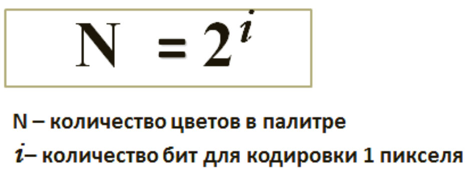
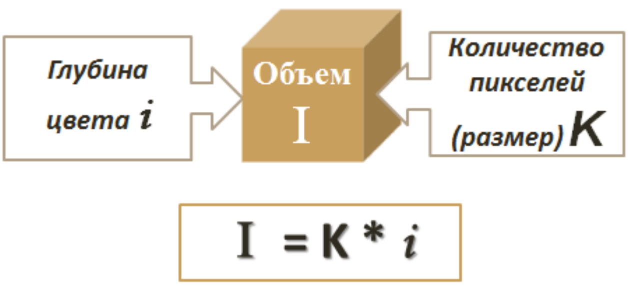
Далее вспоминаем единицы измерения количества информации:

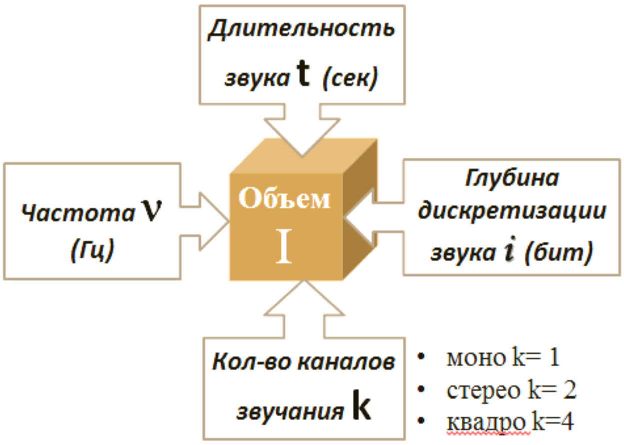
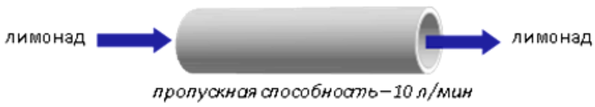
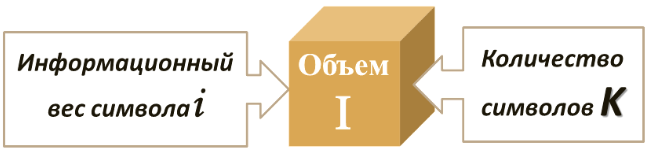
Бит – наименьшая единица измерения количества информации.

1 байт = 8 бит = $2^3$ бит
1 Кбайт = 1024 байт = $2^{10}$ байт = $2^{13}$ бит
1 Мбайт = 1024 Кбайт = $2^{10}$ Кбайт = $2^{20}$ байт = $2^{23}$ бит
1 Гбайт = 1024 Мбайт = $2^{10}$ Мбайт = $2^{20}$ Кбайт = $2^{30}$ байт = $2^{33}$ бит

*Примечание:* ребятам нужно рекомендовать запоминать крупные единицы измерения в степенях $2^x$. При решении задач, вычисления можно выполнить, сокращая показатели степени и этот способ более удобен.

После объяснения этих тем можно выдать учащимся памятку.

<table>
<tr><td colspan="2">Задание 9. Умение определять скорость передачи информации при заданной пропускной способности канала, объем памяти, необходимый для хранения звуковой и графической информации</td></tr>
<tr><td rowspan="2">Объём памяти для хранения ГРАФИЧЕСКОЙ информации</td><td>
</td></tr>
<tr><td>
</td></tr>
</table>

<table>
<tr>
<td rowspan="2">Объём памяти для хранения ЗВУКОВОЙ информации</td>
<td>$$N = 2^i$$<br>N – количество уровней громкости<br>$i$ – количество бит для кодировки 1 уровня</td>
</tr>
<tr>
<td>
<br>Объем звукового файла<br>$$I = v * i * t * k$$</td>
</tr>
<tr>
<td>Определение СКОРОСТИ ПЕРЕДАЧИ ИНФОРМАЦИИ при заданной пропускной способности канала</td>
<td>
<br>Сколько лимонада перекачается по трубе за 1 час?<br>Ответ: 10 л/мин · 60 мин = 600 л<br>Скорость передачи информации<br>$$I = V \cdot t \qquad V = \frac{I}{t}$$</td>
</tr>
<tr>
<td colspan="2">Задание 13. Умение подсчитывать информационный объём сообщения</td>
</tr>
<tr>
<td rowspan="2">Объём памяти для хранения ТЕКСТОВОЙ информации</td>
<td>
<br>$$N = 2^i$$<br>N – мощность алфавита - полное количество символов в алфавите<br>$i$ – количество бит для кодировки 1 символа</td>
</tr>
<tr>
<td>
<br>Объем текстовой информации<br>$$I = K * i$$</td>
</tr>
</table>

| | | | |
|---|---|---|---|
| Бит – наименьшая единица измерения количества информации.<br>1 байт = 8 бит = $2^3$ бит<br>1 Кбайт = 1024 байт = $2^{10}$ байт = $2^{13}$ бит<br>1 Мбайт = 1024 Кбайт = $2^{10}$ Кбайт = $2^{20}$ байт = $2^{23}$ бит<br>1 Гбайт = 1024 Мбайт = $2^{10}$ Мбайт = $2^{20}$ Кбайт = $2^{30}$ байт = $2^{33}$ бит | | | |
| $2^0 = 1$<br>$2^1 = 2$<br>$2^2 = 4$ | $2^3 = 8$<br>$2^4 = 16$<br>$2^5 = 32$ | $2^6 = 64$<br>$2^7 = 128$<br>$2^8 = 256$ | $2^9 = 512$<br>$2^{10} = 1024$ |

# Комбинаторика

***Тарабукина Любовь Николаевна,***
*учитель информатики и ИКТ,*
*МАОУ «Лицей № 1» г. Сыктывкара*

***Задание № 10*** ЕГЭ по информатике и ИКТ 2016 года.

***Проверяемые элементы содержания:*** знания о методах измерения количества информации.

**Часть 1**

***Уровень:*** базовый

***Теория:***

## Основные формулы комбинаторики

1) **Факториал** (произведение всех натуральных чисел от 1 до $n$ включительно)

$1! = 1$
$2! = 1 \times 2 = 2$
$3! = 1 \times 2 \times 3 = 6$
$4! = 1 \times 2 \times 3 \times 4 = 24$
$5! = 1 \times 2 \times 3 \times 4 \times 5 = 120$
$6! = 1 \times 2 \times 3 \times 4 \times 5 \times 6 = 5040$
…
$(n-1)! = 1 \times 2 \times 3 \times 4 \times 5 \times \ldots \times (n-2)(n-1)$
$n! = 1 \times 2 \times 3 \times 4 \times 5 \times \ldots \times (n-2)(n-1)n$
$(n+1)! = 1 \times 2 \times 3 \times 4 \times 5 \times \ldots \times (n-2)(n-1)n(n+1)$
…
Кроме того: $0! = 1$

2) **Перестановки, сочетания и размещения без повторений**

***Участники действий:*** множество, состоящее из $n$ различных объектов (*либо объектов, считающихся в контексте той или иной задачи различными*)

| |
|---|
| **Формула количества перестановок:** $P_n = n!$<br>*Типичная смысловая нагрузка: «Сколькими способами можно переставить n объектов?»* |
| **Формула количества сочетаний:** $C_n^m = \frac{n!}{(n-m)!\cdot m!}$<br>*Типичная смысловая нагрузка: «Сколькими способами можно выбрать m объектов из n?». Поскольку выборка проводится из множества, состоящего из n объектов, то справедливо неравенство $0 \le m \le n$* |
| **Формула количества размещений:** $A_n^m = (n-m+1)\cdot \ldots \cdot (n-1)n$<br>*Типичная смысловая нагрузка: «Сколькими способами можно выбрать m объектов (из n объектов)* ***и в каждой*** *выборке переставить их местами (либо распределить между ними какие-нибудь уникальные атрибуты)?»*<br>Исходя из вышеуказанного, справедлива следующая формула: $C_n^m \cdot P_m = A_n^m$ |

4) **Комбинаторное правило суммы и комбинаторное правило произведения**

– Если объект $A$ можно выбрать из некоторого множества объектов $m$ способами, а другой объект $B$ – $n$ способами, то выбор объекта $A$ или объекта $B$ (без разницы какого) возможен $m + n$ способами.
– Если объект $A$ можно выбрать из некоторого множества объектов $m$ способами и после каждого такого выбора объект $B$ можно выбрать $n$ способами, то упорядоченная пара объектов $(A; B)$ может быть выбрана $mn$ способами.
– Данные принципы справедливы и для большего количества объектов.
– Важная содержательная часть правил состоит в том, знак «плюс» понимается и читается как союз **ИЛИ**, а знак «умножить» – как союз **И**.

5) **Перестановки, сочетания и размещения с повторениями**

***Участники действий:*** множество, состоящее из объектов, среди которых есть одинаковые (*либо считающиеся таковыми по смыслу задачи*)

**Формула количества перестановок с повторениями:** $P_{n(\text{повт})} = \frac{n!}{n_1!\cdot n_2!\cdot n_3!\cdot\ldots\cdot n_k!}$, где $n_1 + n_2 + n_3 + \ldots + n_k = n$

*Типичная смысловая нагрузка: «Количество способов, которыми можно переставить n объектов, среди которых 1-й объект повторяется $n_1$ раз, 2-й повторяется $n_2$ раз, 3-й объект – $n_3$ раз, …, k-й объект – $n_k$ раз»*

Следует отметить, что в подавляющем большинстве задач в совокупности есть и уникальные (не повторяющиеся) объекты, в этом случае соответствующие значения $n_i$ равны единице, и в практических расчётах их можно не записывать в знаменатель.

## Решение задач

***Перестановки, сочетания и размещения без повторений.***

Выкладываем фрукты слева направо в следующем порядке: яблоко / груша / банан.

*Вопрос первый:* сколькими способами их можно переставить?

Одна комбинация уже записана выше и с остальными проблем не возникает:

яблоко / банан / груша
груша / яблоко / банан
груша / банан / яблоко
банан / яблоко / груша
банан / груша / яблоко

*Итого:* 6 комбинаций или 6 перестановок.

Хорошо, здесь не составило особого труда перечислить все возможные случаи, но как быть, если предметов больше? Уже с четырьмя различными фруктами количество комбинаций значительно возрастёт!

В справочном материале Основные формулы комбинаторики (*методичку удобно распечатать*) и в пункте № 2 найдём формулу количества перестановок.

3 объекта можно переставить $P_3 = 3! = 6$ способами.

*Вопрос второй:* сколькими способами можно выбрать:

а) один фрукт, б) два фрукта, в) три фрукта, г) хотя бы один фрукт?

а) Один фрукт можно выбрать, очевидно, тремя способами – взять либо яблоко, либо грушу, либо банан. Формальный подсчёт проводится по формуле количества сочетаний:

$$C_3^1 = \frac{3!}{2! \cdot 1} = \frac{6}{2 \cdot 1} = 3$$

Запись $C_3^1$ в данном случае следует понимать так: «сколькими способами можно выбрать 1 фрукт из трёх?»

б) Перечислим все возможные сочетания двух фруктов:

яблоко и груша;
яблоко и банан;
груша и банан.

Количество комбинаций легко проверить по той же формуле:

$$C_3^2 = \frac{3!}{1! \cdot 2!} = 3$$

Запись $C_3^2$ понимается аналогично: «сколькими способами можно выбрать 2 фрукта из трёх?».

в) И, наконец, три фрукта можно выбрать единственным способом:

$$C_3^3 = \frac{3!}{0! \cdot 3!} = 1$$

Кстати, формула количества сочетаний сохраняет смысл и для пустой выборки: $C_3^0 = \frac{3!}{3! \cdot 0!} = 1$ способом можно выбрать ни одного фрукта – собственно, ничего не взять и всё.

г) Сколькими способами можно взять ***хотя бы один*** фрукт? Условие «хотя бы один» подразумевает, что нас устраивает 1 фрукт (любой) или 2 любых фрукта или все 3 фрукта:

$C_3^1 + C_3^2 + C_3^3 = 3 + 3 + 1 = 7$ способами можно выбрать хотя бы один фрукт.

*Вопрос третий:* сколькими способами можно раздать по одному фрукту Даше и Наташе?

Для того чтобы раздать два фрукта, сначала нужно их выбрать. Согласно пункту «бэ» предыдущего вопроса, сделать это можно $C_3^2 = 3$ способами, перепишу их заново:

яблоко и груша;
яблоко и банан;
груша и банан.

Но комбинаций сейчас будет в два раза больше. Рассмотрим, например, первую пару фруктов:

яблоком можно угостить Дашу, а грушей – Наташу;
либо наоборот – груша достанется Даше, а яблоко – Наташе.

И такая перестановка возможна для каждой пары фруктов.

В данном случае работает формула количества размещений:

$$A_3^2 = 2 \cdot 3 = 6$$

Она отличается от формулы $C_3^2$ тем, что учитывает ***не только*** количество способов, которым можно выбрать несколько объектов, но и все перестановки объектов ***в каждой*** возможной выборке. Так, в рассмотренном примере, важно не только то, что можно просто выбрать, например, грушу и банан, но и то, как они будут распределены (размещены) между Дашей и Наташей.

Пожалуйста, внимательно прочитайте пункт № 2 методички Основные формулы комбинаторики и постарайтесь хорошо уяснить разницу между перестановками, сочетаниями и размещениями. В простейших случаях можно пересчитать все возможные комбинации вручную, но чаще всего это становится неподъёмной задачей, именно поэтому и нужно понимать смысл формул.

Также напоминаю, что сейчас речь идёт о множестве с ***различными*** объектами, и если яблоко/грушу/банан заменить на 3 яблока или даже на 3 очень похожих яблока, то в контексте рассмотренной задачи они всё равно будут считаться ***различными***.

Остановимся на каждом виде комбинаций подробнее:

## Решение задач

### *Перестановки*

***Перестановками*** называют комбинации, состоящие *из одних и тех же* ***n различных*** объектов и отличающиеся только порядком их расположения. Количество всех возможных перестановок выражается формулой $P_n = n!$

Отличительной особенностью перестановок является то, что *в каждой из них* участвует **ВСЁ** множество, то есть, **все *n*** объектов. Например, дружная семья:

***Общий вид задачи:***

Сколькими способами можно рассадить 5 человек за столом?

*Решение:* используем формулу количества перестановок:

$$P_5 = 5! = 120$$

*Ответ:* 120 способами

***Задача 2***

Сколько четырёхзначных чисел можно составить из четырёх карточек с цифрами 0, 5, 7, 9?

Для того чтобы составить четырёхзначное число нужно задействовать все четыре карточки (*цифры на которых* ***различны!***), и это очень важная предпосылка для применения формулы $P_n = n!$ Очевидно, что, переставляя карточки, мы будем получать различные четырёхзначные числа, … стоп, а всё ли тут в порядке? ☺

Хорошенько подумайте над задачей! Вообще, это характерная черта комбинаторных и вероятностных задач – в них НУЖНО ДУМАТЬ.

*Решение:* найдём количество всех возможных перестановок 4 карточек:

$$P_4 = 4! = 24$$

*Когда карточка с нулём располагается на 1-м месте, то число становится трёхзначным, поэтому данные комбинации следует исключить. Пусть ноль находится на 1-м месте, тогда оставшиеся 3 цифры в младших разрядах можно переставить* $P_3 = 3! = 6$ *способами.*

***Примечание:*** *т.к. карточек немного, то здесь несложно перечислить все такие варианты:*

0579
0597
0759
0795
0957
0975

Таким образом, из предложенного набора можно составить: 24 – 6 = 18 четырёхзначных чисел

*Ответ:* 18

***Задача 4:***

*Решение:* $C_{36}^{3} = \frac{36!}{33!\cdot 3!} = \frac{33!\cdot 34\cdot 35\cdot 36}{33!\cdot 3!} = \frac{34\cdot 35\cdot 36}{6} = 7140$ способами можно выбрать 3 карты из 36.

*Ответ:* 7140

***Задача 6:***

*Решение:* $A_{23}^{2} = 22 \cdot 23 = 506$ способами.

*Другой вариант решения:* $C_{23}^{2}$ способами можно выбрать двух человек из группы и $P_2 = 2! = 2$ способами распределить должности в каждой выборке. Таким образом, старосту и его заместителя можно выбрать $C_{23}^{2} \cdot P_2 = \frac{23!}{21!\cdot 2!} \cdot 2! = 22 \cdot 23 = 506$ способами.

*Ответ:* 506

## ***Сочетания***

***Сочетаниями*** называют различные комбинации из ***m*** объектов, которые выбраны из множества ***n*** различных объектов, и которые отличаются друг от друга хотя бы одним объектом. Иными словами, отдельно взятое сочетание – это уникальная выборка из ***m*** элементов, в которой *не важен их порядок* (расположение). Общее же количество таких уникальных сочетаний рассчитывается по формуле $C_n^m = \frac{n!}{(n-m)!\cdot m!}$

***Задача 3***

В ящике находится 15 деталей. Сколькими способами можно взять 4 детали?

*Решение:* прежде всего, снова обращаю внимание на то, что по логике условия, детали считаются ***различными*** – даже если они на самом деле однотипны и визуально одинаковы (*в этом случае их можно, например, пронумеровать*).

В задаче речь идёт о выборке из 4 деталей, в которой не имеет значения их «дальнейшая судьба» – грубо говоря, «просто выбрали 4 штуки и всё». Таким образом, у нас имеют место сочетания деталей. Считаем их количество:

$$C_{15}^{4} = \frac{15!}{(15-4)!\cdot 4!} = \frac{15!}{11!\cdot 4!} = (*)$$

Здесь, конечно же, не нужно ворочать огромные числа 11! = 39916800, 15! = 1307674368000. В похожей ситуации надо использовать следующий приём: в знаменателе выбираем наибольший факториал (в данном случае 11!) и сокращаем на него дробь. Для этого числитель следует представить в виде 15! = 11! × 12 × 13 × 14 × 15. Подробно: $(*) = \frac{11!\cdot 12\cdot 13\cdot 14\cdot 15}{11!\cdot 4!} = \frac{12\cdot 13\cdot 14\cdot 15}{4!} = \frac{12\cdot 13\cdot 14\cdot 15}{24} = 1365$ способами можно взять 4 детали из ящика.

Ещё раз: что это значит? Это значит, что из набора 15 различных деталей можно составить *одну тысячу триста шестьдесят пять* ***уникальных*** сочетания 4 деталей. То есть, каждая такая комбинация из четырёх деталей будет отличаться от других комбинаций хотя бы одной деталью.

*Ответ:* 1365 способами

Формуле $C_n^m = \frac{n!}{(n-m)! \cdot m!}$ необходимо уделить самое пристальное внимание, поскольку она является «хитом» комбинаторики. При этом полезно понимать и без всяких вычислений записывать «крайние» значения: $C_n^0 = 1$, $C_n^1 = n$, $C_n^{n-1} = n$, $C_n^n = 1$. Применительно к разобранной задаче:

$C_{15}^0 = 1$ – единственным способом можно взять ни одной детали;

$C_{15}^1 = 15$ способами можно взять 1 деталь (любую из пятнадцати);

$C_{15}^{14} = 15$ способами можно взять 14 деталей (при этом какая-то одна из 15 останется в ящике);

$C_{15}^{15} = 1$ – единственным способом можно взять все пятнадцать деталей.

Рекомендую внимательно ознакомиться с биномом Ньютона и треугольником Паскаля, по которому, к слову, очень удобно выполнять проверку вычислений $C_n^m$ при небольших значениях «эн».

***Задача 4****

В шахматном турнире участвует $\boldsymbol{k}$ человек и каждый с каждым играет по одной партии. Сколько всего партий сыграно в турнире?

Турнирная таблица размером $\boldsymbol{k} \times \boldsymbol{k}$ клеток, в которой результат каждой партии учитывается дважды и, кроме того, затушёвываются клетки «главной диагонали» (*т.к. участники не играют сами с собой*). Исходя из проведённых рассуждений, общее количество сыгранных партий рассчитывается по формуле $n = \frac{k \cdot k - k}{2}$. Такое решение полностью корректно (*см. соответствующий файл банка готовых решений*). Однако, на самом деле здесь можно руководствоваться самыми что ни на есть банальными сочетаниями: $C_k^2 = \frac{k!}{(k-2)! \cdot 2!} = \frac{1 \cdot 2 \cdot 3 \cdot \ldots \cdot (k-2)(k-1)k}{1 \cdot 2 \cdot 3 \cdot \ldots \cdot (k-2) \cdot 2} = \frac{(k-1)k}{2}$ различных пар можно составить из $\boldsymbol{k}$ соперников (*кто играет белыми, кто чёрными – не важно*).

Эквивалентной является задача о рукопожатиях: в отделе работает $\boldsymbol{k}$ мужчин и каждый с каждым здоровается за руку, сколько рукопожатий они совершают? К слову, шахматисты тоже пожимают друг другу руку перед каждой партией.

## Решение задач

### *Размещения*

Или «продвинутые» сочетания. ***Размещениями*** называют различные комбинации из $\boldsymbol{m}$ объектов, которые выбраны из множества $\boldsymbol{n}$ различных объектов, и которые отличаются друг от друга как составом объектов в выборке, так и их порядком. Количество размещений рассчитывается по формуле $A_n^m = (n-m+1) \cdot \ldots \cdot (n-1)n$.

***Задача 5***

Боря, Дима и Володя сели играть в «21». Сколькими способами им можно сдать по одной карте? (*колода содержит 36 карт*)

*Решение:* ситуация похожа на Задачу 4, но отличается тем, что здесь важно не только то, какие три карты будут извлечены из колоды, но и то, КАК они будут распределены между игроками. По формуле размещений: $A_{36}^3 = 34 \cdot 35 \cdot 36 = 42840$ способами можно раздать 3 карты игрокам.

Есть и другая схема решения, которая, с моей точки зрения, даже понятнее: $C_{36}^3 = \frac{36!}{33! \cdot 3!} = 7140$ способами можно извлечь 3 карты из колоды.

Теперь давайте рассмотрим, какую-нибудь одну *из семи тысяч ста сорока* комбинаций, например: король пик, 9 червей, 7 червей. Выражаясь комбинаторной терминологией, эти 3 карты можно «переставить» между Борей, Димой и Володей $P_3 = 3! = 6$ способами:

КП, 9Ч, 7Ч;
КП, 7Ч, 9Ч;
9Ч, КП, 7Ч;
9Ч, 7Ч, КП;
7Ч, КП, 9Ч;
7Ч, 9Ч, КП.

И аналогичный факт справедлив ***для любого*** уникального набора из трёх карт. А таких наборов, не забываем, мы насчитали $C_{36}^3 = 7140$. Не нужно быть профессором, чтобы понять, что найденное количество сочетаний следует умножить на шесть: $C_{36}^3 \cdot P_3 = 7140 \cdot 6 = 42840$ способами можно сдать по одной карте трём игрокам.

По существу, получилась наглядная проверка формулы $C_n^m \cdot P_n = A_n^m$, окончательный смысл которой мы проясним в следующем параграфе.

*Ответ:* 42840

## Решение задач

### *Правило сложения и правило умножения комбинаций*

1) *Знак «плюс» следует понимать и читать как союз ИЛИ*. Вспоминаем демонстрационную задачу с яблоком, грушей и бананом: $C_3^1 + C_3^2 + C_3^3 = 3 + 3 + 1 = 7$ способами можно выбрать хотя бы один фрукт.

То есть, можно взять 1 фрукт (любой из трёх) ***ИЛИ*** какое-нибудь сочетание двух фруктов ***ИЛИ*** все три фрукта. Заметьте, что сложение комбинаций предполагает безразличие выбора (*без разницы будет ли выбран один, два или 3 фрукта*).

Рассмотрим более основательный пример:

***Задача 7***

Студенческая группа состоит из 23 человек, среди которых 10 юношей и 13 девушек. Сколькими способами можно выбрать двух человек одного пола?

*Решение:* в данном случае не годится подсчёт $C_{23}^2$, поскольку общее количество сочетаний включает в себя и разнополые пары.

Условие «выбрать двух человек одного пола» подразумевает, что необходимо выбрать двух юношей или двух девушек, и уже сама словесная формулировка указывает на верный путь решения:

$C_{10}^2 = \frac{10!}{8! \cdot 2!} = \frac{8! \cdot 9 \cdot 10}{8! \cdot 2!} = \frac{9 \cdot 10}{2} = 45$ способами можно выбрать 2 юношей;

$C_{13}^2 = \frac{13!}{11! \cdot 2!} = \frac{11! \cdot 12 \cdot 13}{11! \cdot 2!} = \frac{12 \cdot 13}{2} = 78$ способами можно выбрать 2 девушек.

Таким образом, двух человек одного пола (без разницы – юношей ***или*** девушек) можно выбрать: $C_{10}^2 + C_{13}^2 = 45 + 78 = 123$ способами.

*Ответ:* 123

***Правило умножения комбинаций:***

2) *Знак «умножить» следует понимать и читать как союз И*.

Рассмотрим ту же студенческую группу, которая пошла на танцы. Сколькими способами можно составить пару из юноши и девушки?

$C_{10}^{1} = 10$ способами можно выбрать 1 юношу;

$C_{13}^{1} = 13$ способами можно выбрать 1 девушку.

Таким образом, одного юношу и одну девушку можно выбрать: $C_{10}^{1} \cdot C_{13}^{1} = 10 \cdot 13 = 130$ способами.

Когда из каждого множества выбирается по 1 объекту, то справедлив следующий принцип подсчёта комбинаций: «***каждый*** объект из одного множества может составить пару ***с каждым*** объектом другого множества».

То есть, Олег может пригласить на танец любую из 13 девушек, Евгений – тоже любую из тринадцати, и аналогичный выбор есть у остальных молодых людей. Итого: 10 × 13 = 130 возможных пар.

Следует отметить, что в данном примере не имеет значения «история» образования пары; однако если принять во внимание инициативу, то количество комбинаций нужно удвоить, поскольку каждая из 13 девушек тоже может пригласить на танец любого юношу. Всё зависит от условия той или иной задачи!

Похожий принцип справедлив и для более сложных комбинаций, например: сколькими способами можно выбрать двух юношей ***и*** двух девушек для участия в сценке КВН?

Союз ***И*** недвусмысленно намекает, что комбинации необходимо перемножить: $C_{10}^{2} \cdot C_{13}^{2} = 45 \cdot 78 = 3510$ возможных групп артистов.

Иными словами, ***каждая*** пара юношей (45 уникальных пар) может выступать с ***любой*** парой девушек (78 уникальных пар). А если рассмотреть распределение ролей между участниками, то комбинаций будет ещё больше.

Правило умножения комбинаций распространяется и на большее количество множителей:

***Задача 8***

Сколько существует трёхзначных чисел, которые делятся на 5?

*Решение:* для наглядности обозначим данное число тремя звёздочками: ***

Комбинации будем считать по разрядам – *слева направо*:

В разряд сотен можно записать любую из $C_{9}^{1} = 9$ цифр (1, 2, 3, 4, 5, 6, 7, 8 или 9). Ноль не годится, так как в этом случае число перестаёт быть трёхзначным.

А вот в разряд *десятков* («посерединке») можно выбрать любую из 10 цифр: $C_{10}^{1} = 10$.

По условию, число должно делиться на 5. Число делится на 5, если оно заканчивается на 5 либо на 0. Таким образом, в младшем разряде нас устраивают 2 цифры.

*Итого, существует:* $C_{9}^{1} \cdot C_{10}^{1} \cdot 2 = 9 \cdot 10 \cdot 2 = 180$ трёхзначных чисел, которые делятся на 5.

При этом произведение $C_{9}^{1} \cdot C_{10}^{1} \cdot 2$ расшифровывается так: «9 способами можно выбрать цифру в *разряд сотен* ***и*** 10 способами выбрать цифру в *разряд десятков* ***и*** 2 способами в *разряд единиц*».

Или ещё проще: «***каждая*** из 9 цифр в *разряде сотен* комбинируется ***с каждой*** из 10 цифр *разряда десятков* ***и с каждой*** из двух цифр в *разряде единиц*».

*Ответ:* 180

***Задача 9***

Сколько существует выигрышных комбинаций из 2 карт при игре в «21»?

Для тех, кто не знает: выигрывает комбинация 10 + ТУЗ (11 очков) = 21 очко и, давайте будем считать выигрышной комбинацию из двух тузов.

(*порядок карт в любой паре не имеет значения*)

***Задача 10***

У Васи дома живут 4 кота.

а). сколькими способами можно рассадить котов по углам комнаты?

б). сколькими способами можно отпустить гулять котов?

в). сколькими способами Вася может взять на руки двух котов (одного на левую, другого – на правую)?

*Решаем:* во-первых, вновь следует обратить внимание на то, что в задаче речь идёт о разных объектах. Это очень важное условие!

а) Рассаживанию подвергаются ***сразу все коты***

+ важно их расположение, поэтому здесь имеют место перестановки: $P_4 = 4! = 24$ способами можно рассадить котов по углам комнаты.

Повторюсь, что при перестановках имеет значение лишь количество различных объектов и их взаимное расположение. В зависимости от настроения Вася может рассаживать животных полукругом на диване, в ряд на подоконнике и т.д. – перестановок во всех случаях будет 24. Же-лающие могут для удобства представить, что коты разноцветные (например, белый, чёрный, рыжий и полосатый) и перечислить все возможные комбинации.

б) Сколькими способами можно отпустить гулять котов?

Предполагается, что коты ходят гулять только через дверь, при этом вопрос подразумевает безразличие по поводу количества животных – на прогулку могут выйти 1, 2, 3 или все 4 кота.

Считаем все возможные комбинации:

- $C_4^1 = 4$ способами можно отпустить гулять одного кота (любого из четырёх);
- $C_4^2 = \frac{4!}{2!\cdot 2!} = \frac{24}{2\cdot 2} = 6$ способами можно отпустить гулять двух котов (варианты перечислите самостоятельно);
- $C_4^3 = 4$ способами можно отпустить гулять трёх котов (какой-то один из четырёх сидит дома);
- $C_4^4 = 1$ способом можно выпустить всех котов.

Наверное, вы догадались, что полученные значения следует просуммировать:
$C_4^1 + C_4^2 + C_4^3 + C_4^4 = 4 + 6 + 4 + 1 = 15$ способами можно отпустить гулять котов.

в) Сколькими способами Вася может взять на руки двух котов?

Ситуация предполагает не только выбор 2 животных, но и их размещение по рукам: $A_4^2 = 3 \cdot 4 = 12$ способами можно взять на руки 2 котов.

Второй вариант решения: $C_4^2 = 6$ способами можно выбрать двух котов ***и*** $P_2 = 2! = 2$ способами посадить ***каждую*** пару на руки: $C_4^2 \cdot P_2 = 6 \cdot 2 = 12$.

*Ответ:* а) 24, б) 15, в) 12

Пусть у Васи дополнительно живёт 5 кошек ☺. Сколькими способами можно отпустить гулять 2 котов и 1 кошку?

$$C_4^2 \cdot C_5^1 = 6 \cdot 5 = 30$$

То есть, с ***каждой*** парой котов можно выпустить ***каждую*** кошку.

***Перестановки, сочетания и размещения с повторениями***

Перечисленные виды комбинаций законспектированы в пункте № 5 справочного материала Основные формулы комбинаторики, однако некоторые из них по первому прочтению могут быть не очень понятными. В этом случае сначала целесообразно ознакомиться с практическими примерами, и только потом осмысливать общую формулировку.

***Перестановки с повторениями***

В перестановках с повторениями, как и в «обычных» перестановках, участвует ***сразу всё множество объектов***, то есть одно но: в данном множестве один или большее количество элементов (объектов) повторяются. Встречайте очередной стандарт:

***Задача 12***

Сколько различных буквосочетаний можно получить перестановкой карточек со следующими буквами: К, О, Л, О, К, О, Л, Ь, Ч, И, К?

*Решение:* в том случае, если бы все буквы были различны, то следовало бы применить тривиальную формулу $\boldsymbol{P_n}$, однако совершенно понятно, что для предложенного набора карточек некоторые манипуляции будут срабатывать «вхолостую», так, например, если поменять местами любые две карточки с буквами «К» в любом слове, то получится то же самое слово. Причём, физически карточки могут сильно отличаться: одна быть круглой с напечатанной буквой «К», другая – квадратной с нарисованной буквой «К». Но по смыслу задачи даже такие карточки ***считаются одинаковыми***, поскольку в условии спрашивается о буквосочетаниях.

Всё предельно просто – всего: 11 карточек, среди которых буква:

К – повторяется 3 раза;
О – повторяется 3 раза;
Л – повторяется 2 раза;
Ь – повторяется 1 раз;
Ч – повторяется 1 раз;
И – повторяется 1 раз.

Проверка: 3 + 3 + 2 + 1 + 1 + 1 = 11, что и требовалось проверить.

По формуле количества перестановок с повторениями: $P_{11(\text{повт})} = \frac{n!}{n_1!\cdot n_2!\cdot n_3!\cdot\ldots\cdot n_k!} = \frac{11!}{3!\cdot 3!\cdot 2!\cdot 1!\cdot 1!\cdot 1!} = \frac{39916800}{6\cdot 6\cdot 2} = 554400$ различных буквосочетаний можно получить. Больше полумиллиона!

На практике вполне допустимо не записывать общую формулу и, кроме того, опускать единичные факториалы:

$$P_{11(\text{повт})} = \frac{11!}{3!\cdot 3!\cdot 2!} = \frac{39916800}{6\cdot 6\cdot 2} = 554400$$

Но предварительные комментарии о повторяющихся буквах обязательны!

*Ответ:* 554400

***Сочетания с повторениями***

Характерная особенность этого вида комбинаций состоит в том, что выборка проводится из нескольких групп, каждая из которых состоит из одинаковых объектов.

***Задача 14***

В студенческой столовой продают сосиски в тесте, ватрушки и пончики. Сколькими способами можно приобрести пять пирожков?

*Решение:* сразу обратите внимание на типичный критерий сочетаний с повторениями – по условию на выбор предложено не множество объектов как таковое, а ***различные виды*** объектов; при этом предполагается, что в продаже есть не менее пяти хот-догов, 5 ватрушек и 5 пончиков. Однако физические характеристики пирожков по смыслу задачи не существенны, и хот-доги / ватрушки / пончики в своих группах считаются одинаковыми.

Что может быть в выборке? Прежде всего, следует отметить, что в выборке обязательно будут одинаковые пирожки (т.к. выбираем 5 штук, а на выбор предложено 3 вида). Варианты тут на любой вкус: 5 хот-догов, 5 ватрушек, 5 пончиков, 3 хот-дога + 2 ватрушки, 1 хот-дог + 2 + ватрушки + 2 пончика и т.д.

Как и при «обычных» сочетаниях, порядок выбора и размещение пирожков в выборке не имеет значения – просто выбрали 5 штук и всё.

Используем формулу $C_{n(\text{повт})}^{m} = C_{n+m-1}^{m} = \frac{(n+m-1)!}{(n-1)!\cdot m!}$ количества сочетаний с повторениями: $C_{3(\text{повт})}^{5} = C_{3+5-1}^{5} = C_7^5 = \frac{7!}{2!\cdot 5!} = \frac{6\cdot 7}{2!} = 21$ способом можно приобрести 5 пирожков.

*Ответ:* 21

Какой вывод можно сделать из многих комбинаторных задач?

Порой, самое трудное – это разобраться в условии.

Аналогичный пример для самостоятельного решения:

***Задача 15***

В кошельке находится достаточно большое количество рублей, 2-, 5- и 10-рублёвых монет. Сколькими способами можно извлечь три монеты из кошелька?

В целях самоконтроля ответьте на пару простых вопросов:

1. Могут ли в выборке все монеты быть разными?
2. Назовите самую «дешёвую» и самую «дорогую» комбинацию монет.

*Решение:* используем формулу $C_{n(\text{повт})}^{m} = C_{n+m-1}^{m} = \frac{(n+m-1)!}{(n-1)!\cdot m!}$ сочетаний с повторениями: $C_{4(\text{повт})}^{3} = C_6^3 = \frac{6!}{3!\cdot 3!} = \frac{720}{6\cdot 6} = 20$ способами можно выбрать 3 монеты из кошелька.

*Ответ:* 20

***Размещения с повторениями***

Из множества, состоящего из ***n*** элементов, выбирается ***m*** элементов, при этом важен порядок элементов в каждой выборке. И всё бы было ничего, но довольно неожиданный прикол заключается в том, что любой объект исходного множества мы можем выбирать сколько угодно раз. Образно говоря, от «множества не убудет».

Когда так бывает? Типовым примером является кодовый замок с несколькими дисками, но по причине развития технологий актуальнее рассмотреть его цифрового потомка:

***Задача 16***

Сколько существует четырёхзначных пин-кодов?

*Решение:* на самом деле для разруливания задачи достаточно знаний правил комбинаторики: $C_{10}^1 = 10$ способами можно выбрать первую цифру пин-кода ***и*** $C_{10}^1 = 10$ способами – вторую цифру пин-кода ***и*** столькими же способами – третью ***и*** столькими же – четвёртую. Таким образом, по правилу умножения комбинаций, четырёхзначный пин-код можно составить: $C_{10}^1 \cdot C_{10}^1 \cdot C_{10}^1 \cdot C_{10}^1 = 10 \cdot 10 \cdot 10 \cdot 10 = 10000$ способами.

А теперь с помощью формулы. По условию нам предложен набор из ***n* = *10*** цифр, из которого выбираются ***m* = *4*** цифры и располагаются *в определённом порядке*, при этом цифры в выборке могут повторяться (*т.е. любой цифрой исходного набора можно пользоваться произвольное количество раз*). По формуле $A_{n(\text{повт})}^m = n^m$ количества размещений с повторениями: $A_{10(\text{повт})}^4 = 10^4 = 10000$

*Ответ:* 10000

***Задача 17***

Согласно государственному стандарту, автомобильный номерной знак состоит из 3 цифр и 3 букв. При этом недопустим номер с тремя нулями, а буквы выбираются из набора А, В, Е, К, М, Н, О, Р, С, Т, У, Х (*используются только те буквы кириллицы, написание которых совпадает с латинскими буквами*).

Сколько различных номерных знаков можно составить для региона?

Не так их, кстати, и много. В крупных регионах такого количества не хватает, и поэтому для них существуют по несколько кодов к надписи RUS.

*Решение:* $A_{10(\text{повт})}^3 = 10^3 = 1000$ способами можно составить цифровую комбинацию автомо-бильного номера, при этом одну из них (000) следует исключить: $A_{10(\text{повт})}^3 - 1 = 1000 - 1 = 999$.

$A_{12(\text{повт})}^3 = 12^3 = 1728$ способами можно составить буквенную комбинацию автомобильного номера.

По правилу умножения комбинаций, всего можно составить: $\left(A_{10(\text{повт})}^3 - 1\right) \cdot A_{12(\text{повт})}^3 = 999 \cdot 1728 = 1726272$ автомобильных номера (***каждая** цифровая комбинация сочетается **с каждой** буквенной комбинацией*).

*Ответ:* 1726272

***Можно использовать метод простого перебора.***

Некто составляет таблицу кодовых слов для передачи сообщений, каждому сообщению соответствует своё кодовое слово. В качестве кодовых слов используется 5-буквенные слова, в которых есть только буквы Л, Е, С, причём буква Л появляется ровно 1 раз. Каждая из других допустимых букв может встречаться в кодовом слове любое количество раз или не встречаться совсем. Сколько различных кодовых слов можно использовать?

*Рассуждение:*

1. Из условия берём длину слова (в данной задаче 5-буквенное слово), обозначаем -d;
2. Из условия берём количество букв, их – 3, это будет являться мощностью алфавита, обозначим – Q;
3. Необходимо найти количество различных кодовых слов, то есть количество комбинаций, составленных из количества предлагаемых букв, обозначим – К;
4. В условии сказано, что буква Л встречается ровно 1 раз, отсюда делаем вывод, что она может стоять или 1, или 2, или 3, или 4, или 5 (зависит от количества

букв в слове):

Первая: Л_ _ _ _
Вторая: _ Л _ _ _
Третья: _ _Л _ _
Четвёртая: _ _ _Л _
Пятая: _ _ _ _ Л

5. Всего 5 позиций.
6. Три буквы в мощности алфавита, из них одна заняла 1 место (Л), отсюда осталось в слове 4 места и 2 буквы для комбинации.
7. По формуле $К = d^N$, где d = 4, N = 2, получаем К = 42 = 16 комбинаций без буквы Л.
8. Каждой такой комбинации будет соответствовать пять вариантов расстановки буквы Л, значит количество кодовых слов находим N = 5 × 16 = 80.

Сколько существует различных символьных последовательностей длины 5 в трёхбуквенном алфавите {Л, Е, С}, которые содержат ровно две буквы Л?

*Рассуждение:*

1. Переберём возможные комбинации слов из 5 букв.
2. В условии сказано, что буква Л встречается 2 раза (не указывается, что они стоят вместе), отсюда делаем вывод, что одна буква Л может стоять на 1, 2, 3, 4 местах (зависит от количества букв в слове):
3. Вариант, когда буква Л стоит на первом месте:

   ЛЛ_ _ _
   Л_ Л _ _
   Л_ _ Л _
   Л_ _ _ Л

4. Всего 4 позиции.
5. Вариант, когда буква Л стоит на втором месте:

   _ЛЛ_ _
   _Л_ Л _
   _Л_ _ Л

6. Всего 3 позиции.
7. Вариант, когда буква Л стоит на третьем месте:

   _ _ЛЛ _
   _ _Л_ Л

8. Всего 2 позиции.
9. Вариант, когда сочетание ЛЛ стоит в конце

   _ _ _ЛЛ

10. Всего позиций 1.
11. Рассмотрели 4 возможных варианта, сложив количество позиций, получаем 10 = (4 + 3 + 2 + 1).
12. По формуле $К = d^N$ , где d = 3, N = 3, получаем К = 33 = 27 комбинаций без букв ЛЛ.
13. Каждой такой комбинации будет соответствовать 10 вариантов расстановки букв ЛЛ, значит, количество различных символьных последовательностей находим N = 10 × 27 = 270.

**Ссылки:**

– http://mathprofi.ru/zadachi_po_kombinatorike_primery_reshenij.htm
– http://kpolyakov.spb.ru/school/ege.htm

# Адресация в сети

***Пепина Анастасия Николаевна***,
*учитель информатики,*
*МОУ «Средняя общеобразовательная школа № 30»*
*г. Сыктывкара*

***Задание № 12*** ЕГЭ по информатике и ИКТ 2016 года.

***Проверяемые элементы содержания:*** знание базовых принципов организации и функционирования компьютерных сетей, адресации в сети.

**1. Теория**

1) При подключении компьютера к сети в параметрах настройки протокола TCP/IP должны быть указаны ***IP-адрес*** компьютера и ***маска сети***

- ***IP-адрес*** уникально идентифицирует узел (компьютер) в сети. Первая часть IP-адреса обозначает адрес сети, вторая часть – адрес узла (номер компьютера).
- ***Маска сети*** показывает, какая часть IP-адреса узла относится к адресу сети, а какая – к адресу узла в этой сети.

2) ***IP-адрес*** и ***маска*** состоят из четырёх групп десятичных чисел, разделённых точками (каждое из этих чисел находится в интервале ***0 … 255***)

IP-адрес: ***192.168.123.132***

Маска: ***255.255.255.0***

Десятичные IP-адреса и маски преобразовываются в ***32-разрядные двоичные числа***, разделённые точками на 4 группы – «***октеты***»

***192.168.123.132*** ***11000000.10101000.01111011.10000100***

***255.255.255.0*** ***11111111.11111111.11111111.00000000***

3) ***Полезные советы:***

В маске сети:

- ***всегда*** впереди стоят «***1***», а в конце «***0***». Например, ***255.255.224.0***
  ***11111111.11111111.11100000.00000000***
- ***старшие биты*** (слева), имеющие значение «***1***» отведены в IP-адресе компьютера ***для адреса сети***;
- ***младшие биты*** (справа), имеющие значение «***0***» отведены в IP-адресе компьютера ***для адреса компьютера в сети***;
- от количества «***0***» в маске зависит, сколько компьютеров можно подключить к данной сети.

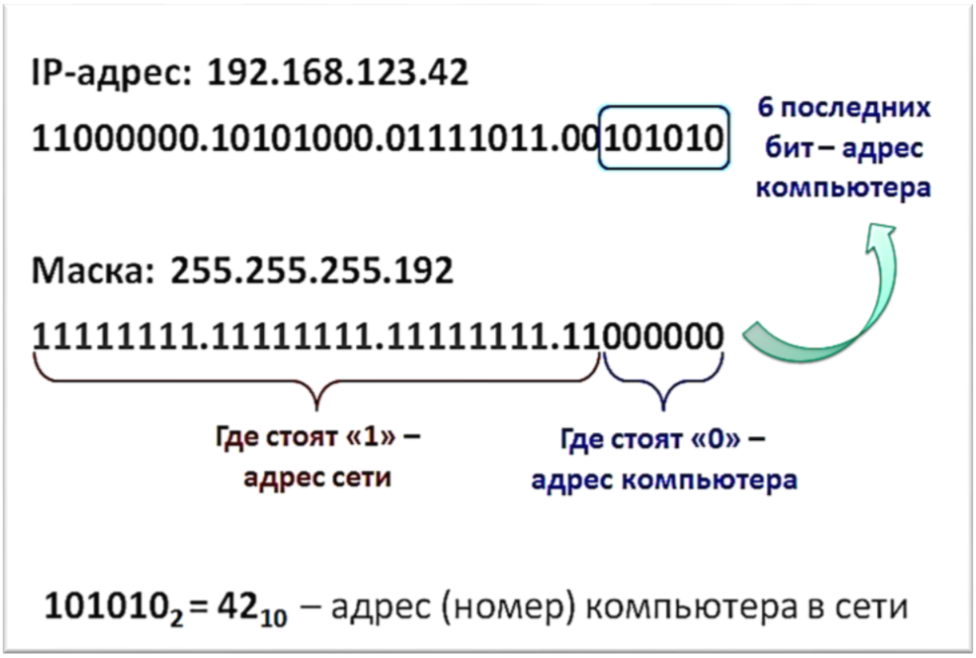
**2. Алгоритмы решения задач**

В 12-м задании ЕГЭ встречается 3 типа заданий. Разберём их.

***1) Алгоритм вычисления адреса (номера) компьютера в сети:***

1. Перевести каждое из чисел в маске и IP-адресе в двоичную систему (кроме $255_{10} = 11111111_2$)
2. Отсчитать в маске сети количество нулевых бит.
3. Отсчитать такое же количество последних бит в IP-адресе и перевести это число в десятичную систему.

*Пример:*




***2) Алгоритм вычисления адреса сети:***

1. Перевести каждое из чисел в IP-адресе и маске в двоичную систему.
2. Выполнить поразрядную ***конъюнкцию*** (умножить бит на бит) IP-адреса компьютера в сети и его маски, перевести каждый октет в десятичную систему.

*Пример* (сразу из двоичной системы):

```
  11000000.10101000.01111011.00101010
* 11111111.11111111.11111111.11000000
  -----------------------------------
  11000000.10101000.01111011.00000000
Адрес сети: 192.168.123.0
```

***3) Алгоритм определения числа компьютеров в сети***

1. Перевести в двоичную систему десятичные числа, не равные 0 и 255 (т.к. $255_{10} = 11111111_2$)
2. Отсчитать в маске количество нулевых бит ***n***.
3. Количество компьютеров в сети $\boldsymbol{K = 2^n - 2}$

*Примечание (необходимо запомнить!):* последнее число в IP-адресе не может принимать значения: ***0, 63, 64, 127, 128, 191, 192*** и ***255*** т.к. для адресации узлов сети не используются:

– адреса, в которых все биты, отсекаемые маской, равны ***0***;
– адреса, в которых все биты, отсекаемые маской, равны ***1***

*Пример:*
Маска сети: ***255.255.254.0***
$254_{10} = 11111110_2$
254.0    11111110.00000000
Общее количество нулевых бит – ***9***
Число компьютеров: $2^9 - 2 = 512 - 2 =$ ***510***

**3. Решение задач**

1) Сотруднику фирмы продиктовали по телефону IP-адрес компьютера. Сотрудник записал этот адрес, но не поставил разделительные точки: ***2153256182***. Восстановите IP-адрес.

*Решение:*

Нужно разделить «***2153256182***» на 4 группы чисел, каждое из которых от ***0*** до ***255***.

*Ответ:* ***215.32.56.182***

2) Определите номер компьютера в сети, если маска подсети ***255.255.248.0*** и IP-адрес компьютера ***112.154.133.208***

*Решение:*

1) Переведём в двоичную систему:

Маска: 255.255.248.0

11111111.11111111.11111000.00000000 (11 бит)

IP-адрес: 112.154.133.208

1110000.10011010.10000101.11010000

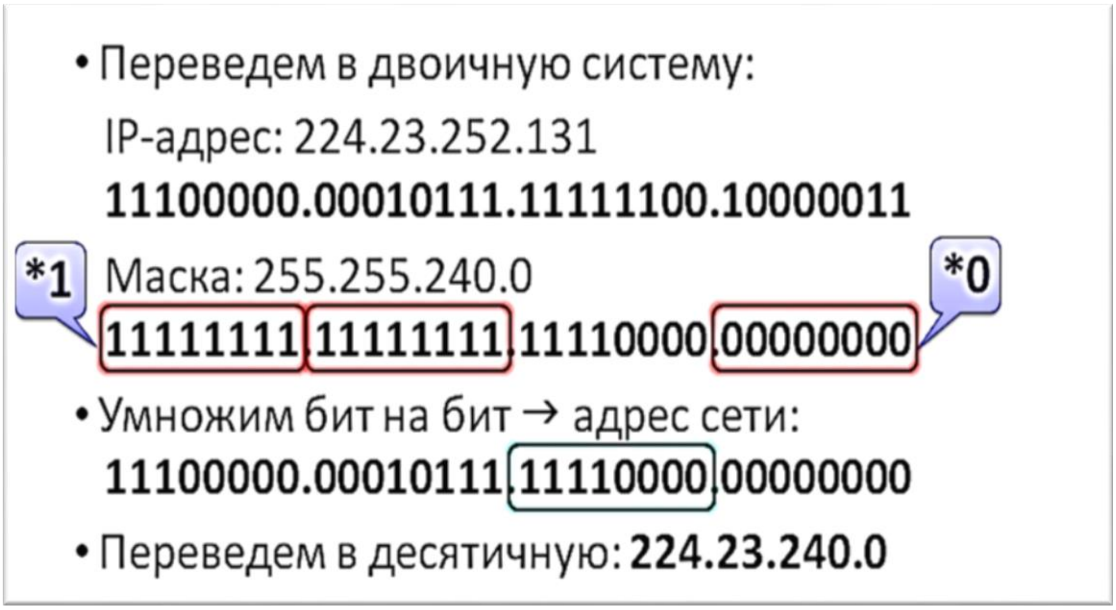
2) Отсчитаем последних 11 бит и переведём в десятичную систему: $10111010000_2$ = $\boldsymbol{1488_{10}}$.

*Ответ:* $\boldsymbol{1488_{10}}$

3) По заданным IP-адресу и маске сети определите адрес сети:

IP-адрес: ***224.23.252.131***. Маска: ***255.255.240.0***

*Решение:*




4) Для подсети используется маска ***255.255.255.128***. Сколько различных адресов компьютеров теоретически допускает эта маска, если два адреса (адрес сети и широковещательный) не используют?

*Решение:*

Переведём последнее число в маске в двоичную систему:

$\boldsymbol{128_{10} = 10000000_2}$ – содержит ***7*** нулевых бит

$\boldsymbol{2^7 - 2 = 128 - 2 = 126}$

*Ответ:* ***126***

## 4. Задания для самостоятельного решения

***Задание 1.***

В IP-адресе номером хоста называется то, что записывается на месте нулей маски подсети.

*Сколько различных хостов допускает маска подсети 255.255.252.0?*

***Задание 2.***

В терминологии сетей TCP/IP маской называется 32-разрядная двоичная последовательность. Маска определяет, какая часть IP-адреса относится к адресу сети, а какая – к адресу самого узла. Обычно маска записывается по тем же правилам, что и IP-адрес. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.

*Для узла с IP-адресом 218.44.150.15 адрес сети равен 218.44.148.0. Чему равен третий слева байт маски?*

***Задание 3.***

*Определите номер компьютера в сети, если IP-адрес компьютера – 192.112.25.5, а маска подсети – 255.255.240.0*

***Задание 4.***

*Определите номер компьютера в сети, если IP-адрес компьютера – 140.20.110.44, а маска подсети – 255.255.252.0*

***Задание 5.***

*Для узла с IP-адресом 185.12.107.15 адрес сети равен 185.12.104.0. Определите наименьшее возможное значение третьего слева байта маски.*

***Задание 6.***

*Для узла с IP-адресом 212.145.203.15 адрес сети равен 212.145.192.0. Определите наибольшее возможное значение третьего слева байта маски.*

***Задание 7.***

*Для узла с IP-адресом 111.81.208.27 адрес сети равен 111.81.192.0. Чему равно наименьшее возможное значение третьего слева байта маски? Ответ запишите в виде десятичного числа.*

***Задание 8.***

*Определите номер компьютера в сети, если IP-адрес компьютера – 146.146.146.146, а маска подсети – 255.255.252.0*

***Задание 9.***

*Сколько различных хостов допускает маска подсети 255.255.254.0?*

### *Ответы и решения к заданиям для самостоятельной работы*

### *К заданию 1*

*Решение:*

Для начала переведём каждый байт маски в двоичную систему счисления:

11111111.11111111.11111100.00000000

В IP-адресе номером хоста называется то, что записывается на месте нулей маски подсети. В нашей маске десять нулей, то есть в IP-адресе под хост используется 10 бит.

10 бит = $2^{10}$ = 1024 различных комбинаций. Однако номером хоста не могут быть все нули и все единицы, то есть мы должны исключить эти два варианта:

1024 – 2 = 1022 хоста.

*Ответ:* 1022

***К заданию 2***

*Решение:*

От нас требуется найти только третий слева байт, то есть переводить в двоичную систему весь IP-адрес и адрес сети не имеет смысла. Переведём в двоичную систему третий слева байт IP-адреса и адреса сети:

10010110 – IP-адрес
10010100 – Адрес сети

Адрес сети получается при применении поразрядной конъюнкции (умножения разрядов) маски и IP-адреса.

xxxxxxxx – Маска
10010110 – IP-адрес
--------
10010100 – Адрес сети

Таким образом, мы можем определить, где в маске находятся единицы. Если в разряде IP-адреса находится 1, и в соответствующем ему разряде адреса сети находится 1, то и в маске должна быть единица.

1xx1x1xx – Маска
10010110 – IP-адрес
--------
10010100 – Адрес сети

При этом в маске ВСЕГДА сначала идут только единицы, а потом только нули, то есть маску можно записать так:

111111xx – Маска
10010110 – IP-адрес
--------
10010100 – Адрес сети

Посмотрите на второй справа бит IP-адреса и адреса сети. Если в маске во втором справа бите будет находиться 1, то и в адресе сети должна быть единица, а в адресе сети указан нуль. Получается, что во втором справа бите маски может быть только нуль.

11111100 – Маска
10010110 – IP-адрес
--------
10010100 – Адрес сети

Переведём 11111100 в десятичную систему, $11111100_2 = 252_{10}$

*Ответ:* 252

***К заданию 3***

*Решение:*

Маска нужна для определения, какая часть IP-адреса является адресом сети, а какая – номером компьютера.

В двоичной системе счисления в IP-адресе на месте единиц маски записывается адрес сети, а на месте нулей – номер компьютера.

Переводить в двоичную систему весь IP-адрес и маску смысла нет – первые два байта маски равны 255, что в двоичной системе счисления равно восьми единицам, то есть в IP-адресе первые два байта отведены под адрес сети. Переведём в двоичную систему последние два байта маски и IP-адреса:

11110000.00000000 – маска
00011001.00000101 – IP-адрес

Под нулями маски в IP-адресе записано 1001.00000101 – это и есть номер компьютера. Уберём точку и переведём его в десятичную систему счисления:

$100100000101_2 = 2309_{10}$

*Ответ:* 2309

***К заданию 4***

*Решение:*

Маска нужна для определения, какая часть IP-адреса является адресом сети, а какая – номером компьютера.

В двоичной системе счисления в IP-адресе на месте единиц маски записывается адрес сети, а на месте нулей – номер компьютера.

Переводить в двоичную систему весь IP-адрес и маску смысла нет – первые два байта маски равны 255, что в двоичной системе счисления равно восьми единицам, то есть в IP-адресе первые два байта отведены под адрес сети. Переведём в двоичную систему последние два байта маски и IP-адреса:

11111100.00000000 – маска

01101110.00101100 – IP-адрес

Под нулями маски в IP-адресе записано 10.00101100 – это и есть номер компьютера. Уберём точку и переведём его в десятичную систему счисления:

$1000101100_2 = 556_{10}$

*Ответ:* 556

***К заданию 5***

*Решение:*

От нас требуется найти только третий слева байт, то есть переводить в двоичную систему весь IP-адрес и адрес сети не имеет смысла. Переведём в двоичную систему третий слева байт IP-адреса и адреса сети:

01101011 – IP-адрес

01101000 – Адрес сети

Примечание: числа 107 и 104 в двоичной системе счисления семизначны, но байт состоит из восьми бит, поэтому нужно дописать один незначащий нуль справа к IP-адресу и маске.

Адрес сети получается при применении поразрядной конъюнкции (умножения разрядов) маски и IP-адреса.

xxxxxxxx – Маска

01101011 – IP-адрес

--------

01101000 – Адрес сети

Мы можем определить, где в маске однозначно будут единицы, а где нули.

Если в IP-адресе и адресе подсети соответствующие разряды равны 1, то в маске тоже должна быть единица, потому что если в маске будет 0, тогда произведение 0 на 1 в IP-адресе даст 0 в адресе сети.

Если в IP-адресе находится единица, а в адресе сети – нуль, тогда в маске обязательно должен быть нуль, так как если там будет единица, то произведение разрядов маски и IP-адреса в адресе сети даст 1.

x11x1x00 – Маска

01101011 – IP-адрес

--------

01101000 – Адрес сети

В маске всегда сначала идут только единицы, а потом только нули, значит мы можем записать всё как:

11111x00 – Маска

01101011 – IP-адрес

--------

01101000 – Адрес сети

Последний икс в маске может быть равен как единице, так и нулю. Нам нужно найти наименьший байт маски, значит для наименьшего значения x должен быть равен нулю:

11111000 – Маска

01101011 – IP-адрес

--------

01101000 – Адрес сети

Переведём 11111000 в десятичную систему:

$11111000_2 = 248_{10}$

*Ответ:* 248

***К заданию 6***

*Решение:*

От нас требуется найти только третий слева байт, то есть переводить в двоичную систему весь IP-адрес и адрес сети не имеет смысла. Переведём в двоичную систему третий слева байт IP-адреса и адреса сети:

11001011 – IP-адрес

11000000 – Адрес сети

Адрес сети получается при применении поразрядной конъюнкции (умножения разрядов) маски и IP-адреса.

xxxxxxxx – Маска

11001011 – IP-адрес

--------

11000000 – Адрес сети

Мы можем определить, где в маске однозначно будут единицы, а где нули.

Если в IP-адресе и адресе подсети соответствующие разряды равны 1, то в маске тоже должна быть единица, потому что если в маске будет 0, тогда произведение 0 на 1 в IP-адресе даст 0 в адресе сети

Если в IP-адресе находится единица, а в адресе сети – нуль, тогда в маске обязательно должен быть нуль, так как если там будет единица, то произведение разрядов маски и IP-адреса в адресе сети даст 1.

11xx0x00 – Маска

11001011 – IP-адрес

--------

11000000 – Адрес сети

В маске всегда сначала идут только единицы, а потом только нули, значит мы можем записать всё как:

11xx0000 – Маска

11001011 – IP-адрес

--------

11000000 – Адрес сети

Оставшиеся два икса в маске могут быть как единицами, так и нулями. Нам нужно найти наибольший байт маски, для этого иксы должны быть равны единицам:

11110000 – Маска
11001011 – IP-адрес
--------
11000000 – Адрес сети

Переведём 11110000 в десятичную систему:

$11110000_2 = 240_{10}$

*Ответ:* 240

***К заданию 7***

*Решение:*

Адрес подсети вычисляется с помощью применения поразрядной конъюнкции к IP-адресу и маске, то есть при перемножении разрядов их двоичных записей.

Нас интересует только третий байт маски, поэтому мы можем полностью не переводить IP-адрес и адрес сети в двоичную систему, а перевести только третий байт:

20810 = 110100002 – IP-адрес
19210 = 110000002 – адрес сети

Адрес сети получается в результате применения поразрядной конъюнкции к IP и маске, то есть получение маски можно записать так:

xxxxxxxx – маска
11010000 – IP
--------
11000000 – адрес сети

Умножая разряды маски на разряды IP-адреса мы получаем адрес сети.

Обратите внимание на второй справа разряд IP-адреса и адреса сети:

xxxxxxxx – маска
11010000 – IP
--------
11000000 – адрес сети

Второй справа разряд маски нулём быть не может, так как в этом случае в адресе сети тоже должен быть ноль, значит этим иксом может быть только единица:

x1xxxxxx – маска
11010000 – IP
--------
11000000 – адрес сети

Теперь обратите на четвёртый справа разряд адреса сети и IP, в IP он равен 1, а в адресе сети – 0. То есть четвёртый правый разряд маски не может быть единицей, так как в этом случае в адресе сети тоже должна быть единица. Выходит, что четвёртый правый разряд маски – 0:

x1x0xxxx – маска
11010000 – IP
--------
11000000 – адрес сети

В маске ВСЕГДА сначала идут только единицы, а потом только нули, значит маску мы можем записать так:

11x00000 – маска
11010000 – IP

--------

11000000 – адрес сети

Остался последний икс. Он может быть и единицей, и нулём, так как в IP и адресе сети у нас нули. Но по условию нам нужен наименьший байт маски, соответственно на месте икса должен быть ноль:

11000000 – маска

11010000 – IP

--------

11000000 – адрес сети

Остаётся перевести число 110000002 в десятичную систему счисления.

$11000000_2 = 192_{10}$

*Ответ:* 192

***К заданию 8***

*Решение:*

Маска подсети показывает, какая часть IP-ареса является адресом сети, а какая – номером компьютера (хоста).

Для определения номера компьютера нужно перевести байты IP-адреса и маски подсети в двоичную систему счисления:

11111111.11111111.11111100.00000000 – Маска подсети

10010010.10010010.10010010.10010010 – IP-адрес

Номер компьютера в IP-адресе записан под нулями маски:

11111111.11111111.11111100.00000000 – Маска подсети

10010010.10010010.10010010.10010010 – IP-адрес

Таким образом, номер компьютера в двоичной системе счисления имеет вид: 1010010010.

$1010010010_2 = 658_{10}$

*Ответ:* 658

***К заданию 9***

*Решение:*

Маска подсети определяет, какая часть IP-адреса является адресом сети, а какая – номером хоста.

Переведём каждый байт маски в двоичную систему счисления:

11111111.11111111.11111110.00000000

На месте единиц маски в IP-адресе записан адрес сети, а на месте нулей – номер хоста. В двоичной записи маски девять нулей, значит в IP-адресе под номер хоста выделено девять разрядов.

Каждый разряд числа в двоичной системе счисления может принимать два варианта – 0 или 1. Таким образом, количество хостов для этой маски равно 29 = 512.

Однако, номером хоста не могут быть два случая – когда двоичная запись состоит из одних нулей или из одних единиц. То есть мы должны вычесть эти два случая.

512 – 2 = 510

*Ответ:* 510

# Алгоритмы с исполнителем[3]

***Лапшина Татьяна Авенировна***,
*учитель информатики*
*МАОУ «Лицей № 1» г. Сыктывкара*

***Задание № 14*** ЕГЭ по информатике и ИКТ 2016 года.

***Проверяемые элементы содержания:*** умение исполнить алгоритм для конкретного исполнителя с фиксированным набором команд.

**Исполнитель Редактор**

Исполнитель Редактор получает на вход строку цифр и преобразует её. Редактор может выполнять две команды, в обеих командах v и w обозначают цепочки цифр.

А) ***заменить*** (v, w).

Эта команда заменяет в строке первое слева вхождение цепочки v на цепочку w. Например, выполнение команды заменить (111, 27) преобразует строку 05111150 в строку 0527150. Если в строке нет вхождений цепочки v, то выполнение команды заменить (v, w) не меняет эту строку.

Б) ***нашлось*** (v).

Эта команда проверяет, встречается ли цепочка v в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь». Строка исполнителя при этом не изменяется.

Цикл

**ПОКА условие**
  **последовательность команд**
**КОНЕЦ ПОКА**
  выполняется, пока условие истинно.

В конструкции

**ЕСЛИ условие**
**ТО команда1**
**ИНАЧЕ команда2**
**КОНЕЦ ЕСЛИ**

выполняется команда1 (если условие истинно) или команда2 (если условие ложно).

***№ 1.*** Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 125 идущих подряд цифр «8»? В ответе запишите полученную строку.

```
НАЧАЛО
ПОКА нашлось (333) ИЛИ нашлось (888)
ЕСЛИ нашлось (333)
ТО заменить (333, 8)
ИНАЧЕ заменить (888, 3)
КОНЕЦ ЕСЛИ
КОНЕЦ ПОКА
КОНЕЦ
```

---

[3] В работе используются материалы сайта https://inf-ege.sdamgia.ru/

*Решение:*

При выполнении алгоритма тело цикла начинает проверять строчку с первого символа. Первые девять восьмёрок заменяются тремя тройками, затем три тройки заменяются одной восьмёркой. Следовательно, каждые девять восьмёрок заменяются одной. Найдём результат работы алгоритма:

1. 125 ÷ 9 = 13 (8 в остатке). Получили ряд из 21 восьмёрки (13 + 8 = 21)
2. 21 ÷ 9 = 2 (3 в остатке). Получили ряд из 5 восьмёрок (2 + 3 = 5)
3. Три восьмёрки заменяем тройкой, получаем результат: 388.

*Ответ:* 388

***№ 2.*** Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 127 идущих подряд цифр «9»? В ответе запишите полученную строку.

НАЧАЛО
ПОКА **нашлось** (333) ИЛИ **нашлось** (999)
ЕСЛИ **нашлось** (333)
ТО **заменить** (333, 9)
ИНАЧЕ **заменить** (999, 3)
КОНЕЦ ЕСЛИ
КОНЕЦ

*Решение:*

При выполнении алгоритма тело цикла начинает проверять строчку с первого символа. Первые девять девяток заменяются тремя тройками, затем три тройки заменяются одной девяткой. Следовательно, каждые девять девяток заменяются одной. Найдём результат работы алгоритма:

1. 127 ÷ 9 = 14 (1 в остатке). Получили ряд из 15 девяток (14 + 1 = 15)
2. 15 ÷ 9 = 1 (6 в остатке). Получили ряд из 7 девяток (1 + 6 = 7)
3. Шесть девяток заменяем двумя тройками, получаем результат: 339.

*Ответ:* 339.

***№ 3.*** Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 193 идущих подряд цифр «5»? В ответе запишите полученную строку.

```
НАЧАЛО
    ПОКА нашлось (333) или нашлось (555)
        ЕСЛИ нашлось (555)
            ТО заменить (555,3)
                ИНАЧЕ заменить (333,5)
        КОНЕЦ ЕСЛИ
    КОНЕЦ ПОКА
КОНЕЦ
```

*Ответ:* 55335

**Исполнитель Чертёжник**

***Теория:***

1. Для поиска наибольшего число находим наибольший общий делитель чисел.
2. Для поиска наименьшего число находим наименьшее общее кратное чисел.

***№ 1.*** Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду ***сместиться на*** (*a*, *b*), где *a*, *b* – целые числа. Эта команда перемещает Чертёжника из точки с координатами (*x*, *y*) в точку с координатами (*x* + *a*; *y* + *b*).

*Например, если Чертёжник находится в точке с координатами (4, 2), то команда* ***сместиться на*** *(2, -3) переместит Чертёжника в точку (6, -1).*

Цикл

```
ПОВТОРИ число РАЗ
        последовательность команд
КОНЕЦ ПОВТОРИ
```

означает, что ***последовательность команд*** будет выполнена указанное ***число*** раз (число должно быть натуральным).

Чертёжнику был дан для исполнения следующий алгоритм (буквами *n*, *a*, *b* обозначены неизвестные числа, *n* > 1):

```
НАЧАЛО
        сместиться на (60, 100)
        ПОВТОРИ n РАЗ
                сместиться на (a, b)
                сместиться на (33, 44)
        КОНЕЦ ПОВТОРИ
        сместиться на (13, 200)
        сместиться на (-1, 60)
КОНЕЦ
```

Укажите наибольшее возможное значение числа *n*, для которого найдутся такие значения чисел *a* и *b*, что после выполнения программы Чертёжник возвратится в исходную точку.

*Решение:*

1. Найдём координаты вектора, на который сместиться чертёжник после выполнения алгоритма:
   $$x = 60 + n \times (a + 33) + 13 – 1 = 72 + n \times (a + 33)$$
   $$y = 100 + n \times (b + 44) + 200 + 60 = 360 + n \times (b + 44)$$
2. Чертёжник возвращается в исходную точку, значит координаты вектора перемещения (0,0): $72 + n \times (a + 33) = 0$, $360 + n \times (b + 44) = 0$
3. Находим наибольший общий делитель: НОД(72,360) = 72.

*Ответ:* 72

***№ 2.*** Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду ***сместиться на*** (*a*, *b*), где *a*, *b* – целые числа. Эта команда перемещает Чертёжника из точки с координатами (*x*, *y*) в точку с координатами (*x* + *a*, *y* + *b*). Например, если Чертёжник находится в точке с координатами (4, 2), то команда ***сместиться на*** (2, -3) переместит Чертёжника в точку (6, -1).

Цикл

ПОВТОРИ *число* РАЗ

*последовательность команд*

КОНЕЦ ПОВТОРИ

означает, что последовательность команд будет выполнена указанное число раз (число должно быть натуральным).

Чертёжнику был дан для исполнения следующий алгоритм (количество повторений и смещения в первой из повторяемых команд неизвестны):

НАЧАЛО

**сместиться на (4, 6)**

ПОВТОРИ … РАЗ

**сместиться на (…, …)**

**сместиться на (-1, -2)**

КОНЕЦ ПОВТОРИ

**сместиться на (20, 30)**

КОНЕЦ

После выполнения этого алгоритма Чертёжник возвращается в исходную точку. Какое наибольшее число повторений могло быть указано в конструкции «ПОВТОРИ … РАЗ»?

*Решение:*

1. Обозначим количество повторений цикла и координаты точки за переменные *n, x, y*.
2. Найдём координаты вектора на который сместиться чертёжник после выполнения алгоритма: a = 4 + n × (x – 1) + 20 = 24 + n × (x – 1)
3. b = 6 + n × (y – 2) + 30 = 36 + n × (y – 2)
4. Чертёжник возвращается в исходную точку, значит координаты вектора перемещения (0,0): 24 + n × (x – 1) = 0, 36 + n × (y – 2) = 0
5. Находим наибольший общий делитель: НОД(24,36) = 12

*Ответ:* 12

***№ 3.*** Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду ***сместиться на*** (*a, b*) (где *a, b* – целые числа), перемещающую Чертёжника из точки с координатами (*x, y*) в точку с координатами (*x + a, y + b*). Если числа *a, b* положительные, значение соответствующей координаты увеличивается; если отрицательные, уменьшается.

*Например, если Чертёжник находится в точке с координатами (4, 2), то команда сместиться на (2, -3) переместит Чертёжника в точку (6, -1).*

Запись

**Повтори k раз**

**Команда1 Команда2 Команда3**

**Конец**

означает, что последовательность команд **Команда1 Команда2 Команда3** повторится **k** раз.

Чертёжнику был дан для исполнения следующий алгоритм:

**Повтори 4 раз**

**Команда1 Сместиться на (3, 3) Сместиться на (1, -2) Конец**

**Сместиться на (-8, 12)**

После выполнения этого алгоритма Чертёжник вернулся в исходную точку. Какую команду надо поставить вместо команды **Команда1**?

*Алгоритм решение:*
1. Обозначим координаты **Команда1** переменными *x, y*.
2. Найдём координаты вектора, на который сместиться чертёжник после выполнения алгоритма.
3. Чертёжник возвращается в исходную точку, значит координаты вектора перемещения (0, 0)
4. Решаем уравнения, находим координаты точки.

*Ответ:* (-2, -4)

**Исполнитель Робот**

***Теория:***

РОБОТ проверяет клетки группами по столбцам или рядам.
1. РОБОТ проверяет клетки, соответствующие условию внешнего цикла.
2. Если условие внешнего цикла выполняется, то РОБОТ переходит к первому действию внутри цикла.
3. Возвращается к началу внешнего цикла.
4. Зелёным цветом отмечены клетки, удовлетворяющие условию задачи.

***№ 1.*** Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости, включает в себя 4 команды-приказа и 4 команды проверки условия. Команды-приказы: ***вверх***, ***вниз***, ***влево***, ***вправо***. При выполнении любой из этих команд РОБОТ перемещается на одну клетку соответственно: вверх ↑, вниз ↓, влево ←, вправо →. Если РОБОТ начнёт движение в сторону находящейся рядом с ним стены, то он разрушится, и программа прервётся.

Другие 4 команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ: ***сверху свободно, снизу свободно, слева свободно, справа свободно***.

Цикл

```
ПОКА условие
    последовательность команд
КОНЕЦ ПОКА
```

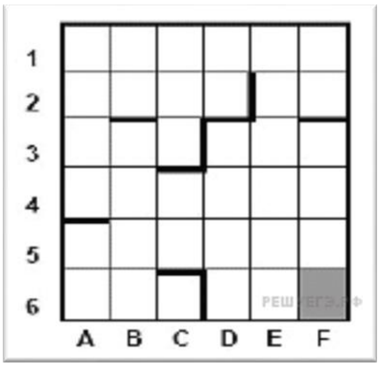
выполняется, пока условие истинно. В конструкции

```
ЕСЛИ условие
   ТО команда1
   ИНАЧЕ команда2
КОНЕЦ ЕСЛИ
```

выполняется *команда1* (если условие истинно) или *команда2* (если условие ложно).

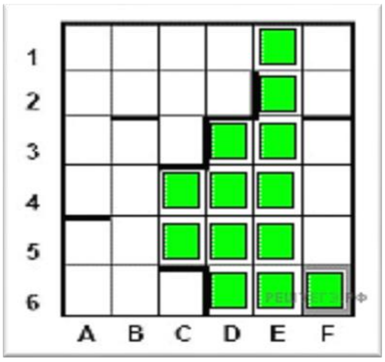
Сколько клеток лабиринта соответствуют требованию, что, начав движение в ней и выполнив предложенную программу, РОБОТ уцелеет и остановится в закрашенной клетке (клетка F6)?

```
НАЧАЛО
   ПОКА снизу свободно ИЛИ справа свободно
   ПОКА снизу свободно
   вниз
   КОНЕЦ ПОКА
   вправо
   КОНЕЦ ПОКА
КОНЕЦ
```



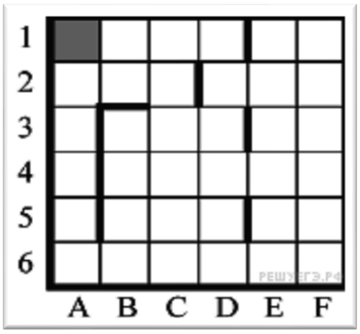
*Решение:*

1. РОБОТ проверяет, свободна ли клетка снизу или справа от него.
2. Если это так, то РОБОТ переходит к первому действию внутри цикла. В этом цикле пока у нижней стороны клетки, в которой находится РОБОТ, нет стены, он двигается вниз, далее делает один шаг вправо.
3. Возвращается к началу внешнего цикла.
4. Проверив все клетки движения РОБОТА, выясняем, что число клеток, удовлетворяющих условию задачи равно 13: все клетки столбца E, D3 – D6, C4, C5. Клетки F3 – F5 не удовлетворяют условию, т.к. РОБОТ разбивается при движении вправо.



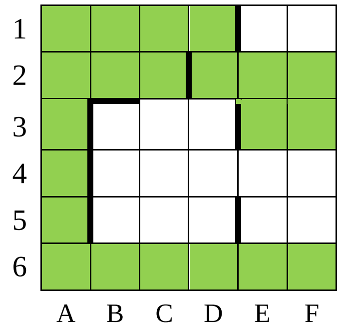

*Ответ:* 13

***№ 2***. Сколько клеток лабиринта соответствуют требованию, что, начав движение в ней и выполнив предложенную программу, РОБОТ уцелеет и остановится в закрашенной клетке (клетка A1)?

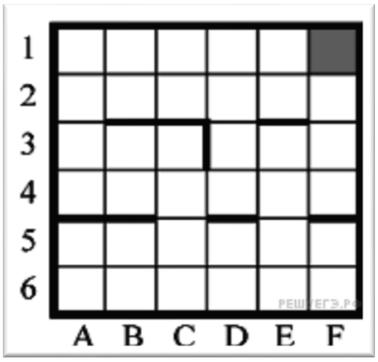



```
НАЧАЛО
    ПОКА слева свободно ИЛИ сверху свободно
        ЕСЛИ слева свободно
        ТО влево
        ИНАЧЕ вверх
        КОНЕЦ ЕСЛИ
    КОНЕЦ ПОКА
КОНЕЦ
```

*Ответ:* 21




***№ 3***. Сколько клеток лабиринта соответствуют требованию, что, начав движение в ней и выполнив предложенную программу, РОБОТ уцелеет и остановится в закрашенной клетке (клетка F1)?




```
НАЧАЛО
    ПОКА сверху свободно ИЛИ справа свободно
        ЕСЛИ сверху свободно
        ТО вверх
        ИНАЧЕ вправо
        КОНЕЦ ЕСЛИ
    КОНЕЦ ПОКА
КОНЕЦ
```

*Ответ:* 24

***Теория:***
Смотрим условие последнего цикла и проверяем клетки следующим образом:

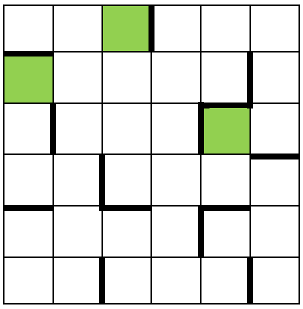
слева свободно – ищем клетки со стеной слева
снизу свободно – ищем клетки со стеной снизу
справа свободно – ищем клетки со стеной справа
сверху свободно – ищем клетки со стеной сверху

***№ 4***. Сколько клеток лабиринта соответствуют требованию, что, выполнив предложенную программу, РОБОТ остановится в той же клетке, с которой он начал движение?

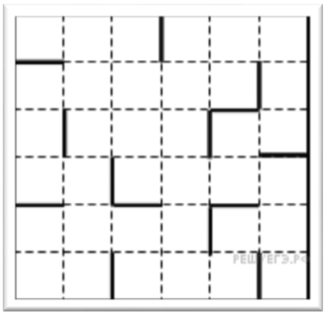
```
НАЧАЛО
    ПОКА <слева свободно> вниз
    ПОКА <снизу свободно> вправо
    ПОКА <справа свободно> вверх
    ПОКА <сверху свободно> влево
КОНЕЦ
```

*Ответ*: 3

1
2
3
4
5
6



A B C D E F



## Адреса сайтов для самоподготовки:

1. http://inf.ege.sdamgia.ru
2. http://somit.ru/14_2015.html
3. https://inf-ege.sdamgia.ru/

# Системы счисления

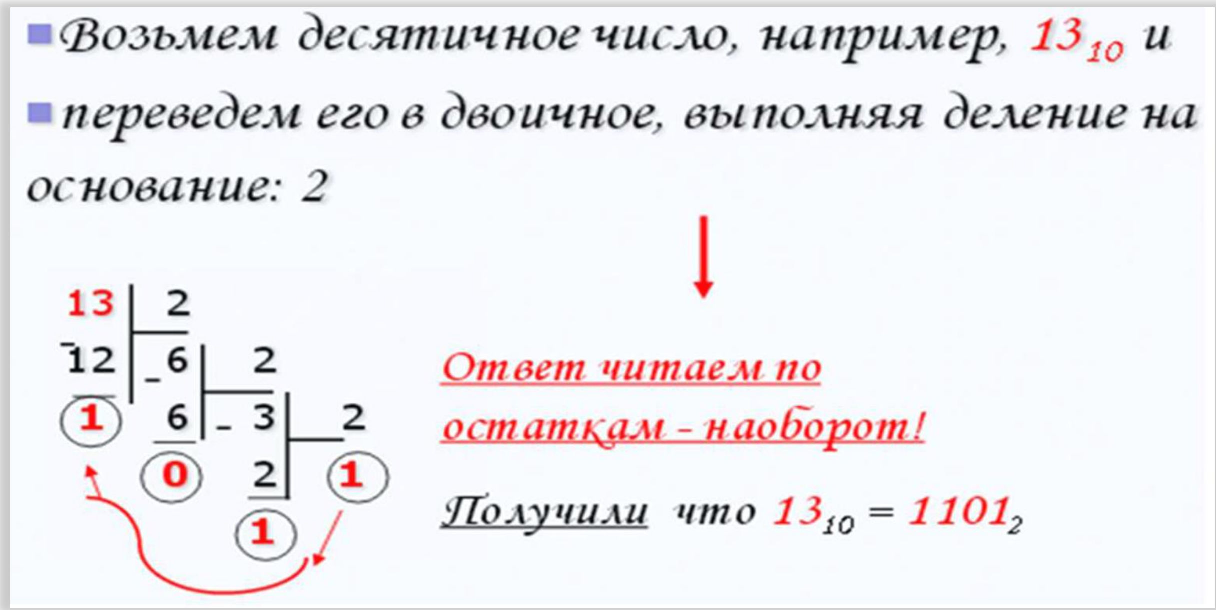
***Иванов Юрий Иванович***,
*учитель информатики*
*МАОУ «Средняя общеобразовательная школа № 28»*
*г. Сыктывкара*

***Задание № 16*** ЕГЭ по информатике и ИКТ 2016 года.

***Проверяемые элементы содержания:*** знание позиционных систем счисления.

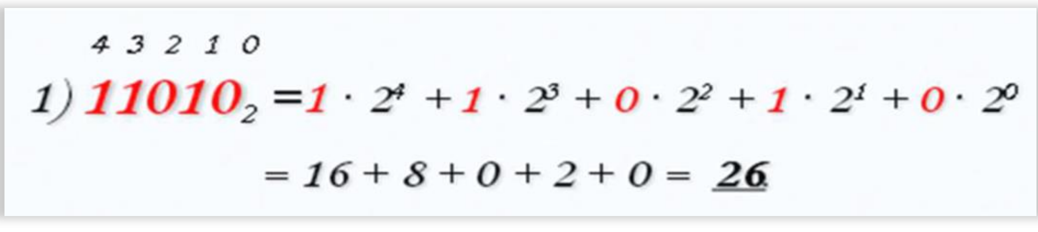
Перевод из десятичной системы счисления в любую другую происходит путём ***деления*** исходного числа ***на основание*** той системы, в которую переводим.

*Пример:*




Для перевода чисел из любой системы счисления в десятичную необходимо записать число в ***развёрнутой форме*** и ***вычислить*** его ***значение***.

*Пример:*




## Быстрый перевод числа из десятичной системы счисления в двоичную.

Чтобы быстро переводить числа между системами счисления, нужно хорошо знать числа «2 в степени» и запись первых 15 чисел в 2-й, 8-й и 16-й системах.

*Например*, $\mathbf{2^5 = 32}$, $\mathbf{2^9 = 256}$ и т.д.

Это позволит решать некоторые примеры на перевод буквально за секунды. Можно, конечно, долго и нудно делить число на «2». Но лучше решать по-другому, экономя драгоценное время на экзамене.

Если число, которое нужно перевести из десятичной системы, равно числу «***2 в степени***», то это число в двоичной системе ***содержит количество нулей, равное степени***.

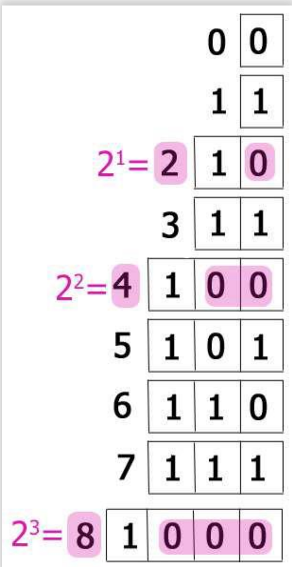
Впереди этих нулей добавляем «***1***».

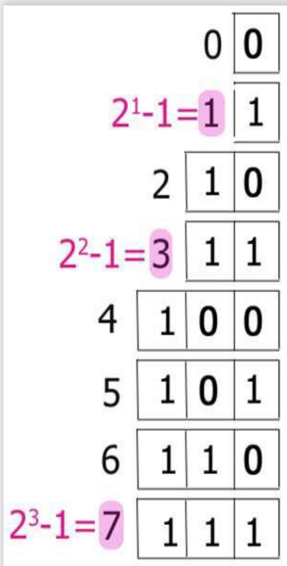
– Переведём число ***2*** из десятичной системы. ***2 = 2^1^***. Поэтому в двоичной системе число содержит ***1 нуль***. Впереди ставим «***1***» и получаем $\boldsymbol{10_2}$.
– Переведём ***4*** из десятичной системы. ***4*** $= \boldsymbol{2^2}$. Поэтому в двоичной системе число содержит ***2 нуля***. Впереди ставим «***1***» и получаем $\boldsymbol{100_2}$.
– Переведём ***8*** из десятичной системы. ***8*** $= \boldsymbol{2^3}$. Поэтому в двоичной системе число содержит ***3 нуля***. Впереди ставим «***1***» и получаем $\boldsymbol{1000_2}$.

Если число, которое нужно перевести, меньше числа «***2 в степени***» ***на 1***, то в двоичной системе это число состоит ***только из единиц***, количество которых ***равно степени***.

– Переведём ***3*** из десятичной системы. $\boldsymbol{3 = 2^2 - 1}$. Поэтому в двоичной системе число содержит ***2 единицы***. Получаем $\boldsymbol{11_2}$.
– Переведём ***7*** из десятичной системы. $\boldsymbol{7 = 2^3 - 1}$. Поэтому в двоичной системе число содержит ***3 единицы***. Получаем $\boldsymbol{111_2}$.

На рисунке квадратиками обозначено двоичное представление числа, а слева розовым цветом – десятичное.




Если число ***больше*** числа «***2 в степени***», то решаем так:

*Переводим сначала число «**2 в степени**» в двоичную систему. А потом прибавляем к нему разницу между числом «**2 в степени**» и **переводимым числом**.*

Например, переведём ***19*** из десятичной системы. Оно больше числа «***2 в степени***» на ***3***. Запишем 19 = 16 + 3. Тогда $16 = 2^4$, $16_{10} = 10000_2$, $3_{10} = 11_2$. Получили $19_{10} = 10000_2 + 11_2 = 10011_2$. Итак, $19_{10} = 10011_2$

Если число ***меньше*** числа «***2 в степени***», то удобнее пользоваться числом «***2 в степени – 1***». Решаем так:

*Переводим сначала число «**2 в степени – 1**» в двоичную систему. А потом **вычитаем** из него разницу между числом «**2 в степени – 1**» и **переводимым числом**.*

Например, переведём ***29*** из десятичной системы в двоичную. Оно больше числа «***2 в степени – 1***» на ***2***.

$29 = 31 – 2.$

$31_{10} = 2^5 – 1 = 11111_2$

$2_{10} = 10_2$

$29_{10} = 11111_2 – 10_2 = 11101_2$

Если разница между переводимым числом и числом «2 в степени» больше трёх, то можно разбить число на составляющие, перевести каждую часть в двоичную систему и сложить.

Например, перевести число ***528*** из десятичной системы в двоичную. ***528 = 512 + 16***. Переводим отдельно ***512*** и ***16***.

$512 = 2^9 \rightarrow 512_{10} = 1000000000_2$

$16 = 2^4 \rightarrow 16_{10} = 10000_2$

Теперь сложим столбиком:

$$\begin{array}{r} + \; 1000000000_2 \\ 10000_2 \\ \hline = \; 1000010000_2 \end{array}$$

*Ответ:* $528_{10} = 1000010000_2$

## Восьмеричная система счисления

В восьмеричной системе счисления основание равно 8, для записи чисел используются цифры от 0 до 7. Для записи каждой цифры восьмеричной системы счисления требуется максимум 3 разряда.

| A 8 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A 2 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

***Алгоритм перевода из двоичной в восьмеричную систему счисления:***

При переводе из 2-ой в 8-ую систему счисления надо число разбить на триады (по три разряда) и записать каждую триаду соответствующей ей цифрой восьмеричной системы счисления, недостающее число разрядов надо дополнить слева нулями.

*Примеры:*

1. $100111101_2 = 100\ 111\ 101_2 = 475_8$
2. $1100010_2 = 001\ 100\ 010_2 = 142_8$

***Алгоритм перевода из восьмеричной в двоичную систему счисления:***

Для перевода из 8-ой в 2-ую систему счисления используется обратное правило. Каждую цифру 8-ого числа надо записать тремя разрядами соответствующего ей двоичного кода.

*Примеры:*

– Перевод из 8-ой в 2-ую систему счисления

$$\boldsymbol{563_8 = 101\ 110\ 011_2}$$

– Перевод из 8-ой в 10-ую систему счисления (используем развёрнутую форму)

$$\boldsymbol{563_8 = 5 \times 8^2 + 6 \times 8^1 + 3 \times 8^0 = 320 + 48 + 3 = 371_{10}}$$

– Перевод из 10-ой в 8-ую систему счисления:

$$\boldsymbol{371_{10} = A_8\ ?}$$

| | | |
|---|---|---|
| 371 | 8 | |
| 3 | 46 | 8 |
| | 6 | 5 |

$\boldsymbol{371_{10} = 563_8}$

## Шестнадцатеричная система счисления

В шестнадцатеричной системе счисления основание системы равно 16. Для записи чисел используется 16 символов: цифры от 0 до 9 и далее буквы латинского алфавита от A до F.

Ниже представлена таблица соответствия кодов чисел четырёх систем счисления.

| 10-ая система счисления | 2-ая система счисления | 8-ая система счисления | 16-ая система счисления |
|---|---|---|---|
| 0 | 0000 | 0 | 0 |
| 1 | 0001 | 1 | 1 |
| 2 | 0010 | 2 | 2 |
| 3 | 0011 | 3 | 3 |
| 4 | 0100 | 4 | 4 |
| 5 | 0101 | 5 | 5 |
| 6 | 0110 | 6 | 6 |
| 7 | 0111 | 7 | 7 |
| 8 | 1000 | 10 | 8 |
| 9 | 1001 | 11 | 9 |
| 10 | 1010 | 12 | A |
| 11 | 1011 | 13 | B |
| 12 | 1100 | 14 | C |
| 13 | 1101 | 15 | D |
| 14 | 1110 | 16 | E |
| 15 | 1111 | 17 | F |

Для записи 1 цифры шестнадцатеричного числа в двоичной системе счисления требуется 4 разряда.

***Алгоритм перевода чисел из двоичной в шестнадцатеричную систему счисления:*** при переводе чисел из 2-ой в 16-ую систему счисления надо число разбить на тетрады (по четыре разряда) и записать каждую тетраду соответствующей ей цифрой шестнадцатеричной системы счисления, недостающее число разрядов надо дополнить слева нулями.

*Примеры:*

1. $1001\ 1110_2 = 9E_{16}$
2. $0010\ 0010_2 = 22_{16}$

***Алгоритм перевода чисел из шестнадцатеричной в двоичную систему счисления:*** для перевода из 16-ой в 2-ую систему счисления используется обратное правило. Каждую цифру шестнадцатеричного числа надо записать четырьмя разрядами соответствующего ей двоичного кода.

*Примеры:*

1. Перевод из 16-ой в 2-ую систему счисления: $\boldsymbol{173_{16} = 101110011_2}$
2. Перевод из 16-ой в 10-ую систему счисления: (используем развёрнутую форму)

$$\boldsymbol{173_{16} = 1 \times 16^2 + 7 \times 16^1 + 3 \times 16^0 = 256 + 112 + 3 = 371_{10}}$$

3. Перевод из 10-ой в 16-ую систему счисления:

```
371 | 16
----+-----
 3  | 23 | 16
    |----+----
    |  7 |  1
```

$371_{10} = A_{16}$?

*Получаем:* $371_{10} = A_{16}$?

## Примеры решения заданий ЕГЭ по теме «Системы счислений

**№ 1**

Сколько единиц в двоичной записи десятичного числа 255?

**1 способ:**

```
255 | 2
2     127 | 2
 5    12    63 | 2
 4     7   6   31 | 2
 15    6    3  2   15 | 2
 14    1    2  11  14  7 | 2
  1         1  10   1  6   3 | 2
                1      1   2   1
                           1
```

Выписываем конечный результат и остатки. Получаем: $11111111_2$. **В числе 8 единиц**.

Сколько единиц в двоичной записи десятичного числа 255?

2 способ (метод быстрого перевода):

Число 255 меньше числа "2 в степени" на 1.

$255 = 256 - 1 = 2^8 - 1$

**$11111111_2$**

**Ответ: 8 единиц**.

**№ 2**

**Запись десятичного числа в системах счисления с основаниями 3 и 5 в обоих случаях имеет последней цифрой 0. Какое минимальное натуральное десятичное число удовлетворяет этому требованию?**

При переводе числа из десятичной системы в другую, мы делим десятичное число на основание другой системы счисления.

Первый остаток от деления - это последняя цифра числа в этой системе счисления. Чтобы в остатке был 0, мы должны подобрать десятичное число, которое будет кратно основанию системы счисления, в которую переводим.

Для системы с основанием **3**, такими числами могут быть: **3, 6, 9** и т.д. Для системы с основанием **5 - 5, 10, 15** и т.д.

По заданию, **число должно быть минимально**, поэтому для системы с **основанием 3** - это число **3**, а с **основанием 5** - это число **5**.

Решение:

```
3|3     5|5
3 1     5 1
0       0
```

$3_{10}=10_3$ и $5_{10}=10_5$

Чтобы остаток числа был равен 0-ю в обеих системах счисления (с остатком 3 и 5), десятичное число должно быть кратно числам: **3 и 5**.

3*5=15 - это и есть искомое десятичное число.

```
15|3     15|5
15 5     15 3
 0        0
```

$15_{10}=50_3$ и $15_{10}=30_5$ **Ответ: 15**

## № 3

Запись числа $67_{10}$ в системе счисления с основанием **N** оканчивается на **1** и содержит **4 цифры**. Чему равно **основание** этой системы счисления **N**?

Начнем с двоичной системы. Для хранения числа **67** необходимо **7 цифр**, т.к. 64<**67**<128. $128=2^7$.

Троичная система. Для хранения числа **67** нужно **4 цифры**, т.к. 27<**67**<81. ($27=3^3$, $81=3^4$).

Следовательно, троичная система удовлетворяет условию: “*число содержит 4 цифры*”

Теперь необходимо проверить, удовлетворяет ли данная система условию: "**число оканчивается на 1**".

Для этого нужно перевести $67_{10}$ в троичную систему. Но полный перевод делать не надо, т.к. нас интересует только первый остаток, на него и будет оканчиваться **67** в троичной системе.

```
67 |3
6    22
 7
 6
 1
```

**Остаток равен 1**. Следовательно, и второе условие выполнено, поэтому троичная система подходит. Основание системы **равно 3**.

## № 4

В системе счисления с некоторым основанием десятичное число **49** записывается в виде **100**. Укажите это основание.

Обозначим основание искомой системы счисления как x.

$49=1*x^2+0*x^1+0*x^0=x^2 \Rightarrow x=\pm7.$

Основание системы счисления не может быть отрицательным, поэтому оно равно **7.**

**Ответ: 7.**

Дано $A=9D_{16}$, $B=237_8$. Какое из чисел **C**, записанных в двоичной системе, отвечает условию $A<C<B$?

1) $10011010_2$ 2) $10011110_2$ 3) $10011111_2$ 4) $11011110_2$

**1 способ**

Нужно **A и B** перевести в двоичную систему счисления. Сначала переведем **A**. Каждая цифра 16-чной системы соответствует 4 цифрам двоичной системы.

**A** содержит 2 цифры: $9_{16}$ и $D_{16}$.

$9=1001_2$. $D_{16}=13_{10}=8_{10}+5_{10}=1000_2+101_2=1101_2$. **$A=1001\ 1101_2$**

Переведем B:

Каждая цифра 8-чной системы соответствует 3 цифрам двоичной системы.

$B=237_8$ $2_8=010_2$ $3=011_2$ $7=111_2$

$237_8=10\ 011\ 111_2=1001\ 1111$ (для удобства сравнения разделим по четыре цифры, т.к. число **A** представлено так)

$A<C<B$: $1001\ 1101_2<\_\_\_\_\_\_\_\_\_\__2<1001\ 1111_2$.

**Ответ: 2**

Дано $A=9D_{16}$, $B=237_8$. Какое из чисел C, записанных в двоичной системе, отвечает условию $A<C<B$?

1) $10011010_2$ 2) $10011110_2$ 3) $10011111_2$ 4) $11011110_2$

**2 способ**

Переведем A и B в 10-чную систему счисления.

$A=9D_{16} = 16 * 9 + 13 = 144 + 13 = 157$ ($D_{16}=13_{10}$)

$B=237_8=2*8^2+3*8+7=128+24+7=159$

$A<C<B$: $157<C<159$. следовательно, C=158.

Переводим в двоичную систему:

158=128+30.

$128=10000000_2$.

$30=31-1=11111_2-1_2=11110_2$.

$$158=\begin{array}{r} 10000000_2 \\ +\ \ 11110_2 \\ \hline 10011110_2 \end{array}$$

**Ответ: 2**

## № 5

Чему равно *x*, если выполнено равенство **$25_x + 18_{3x} = 12_{6x}$**?

Если таких значений ***X*** несколько, перечислите их через запятую в порядке возрастания

1. Запишем числа в виде суммы разрядных слагаемых:

$25_x=2 \cdot x^1+5 \cdot x^0$,

$18_{3x}=1 \cdot (3x)^1+8 \cdot (3x)^0$,

$12_{6x}=1 \cdot (6x)^1+2 \cdot (6x)^0$,

2. Подставим полученные выражения в данное уравнение и решим его:

$2 \cdot x^1+5 \cdot x^0+1 \cdot (3x)^1+8 \cdot (3x)^0 = 1 \cdot (6x)^1+2 \cdot (6x)^0$,

5x+13 = 6x+2,

13-2=6x-5x

11=x

x = 11.

**Ответ: 11.**

Значение арифметического выражения:

**$9^8 + 3^5 - 9$** — записали в системе счисления с основанием **3**. Сколько цифр «**2**» содержится в этой записи?

Понятно, что возводить числа в эти степени и решать «в лоб» не получится. На ЕГЭ по информатике запрещено пользоваться калькуляторами. Требуется «разглядеть», что наше выражение можно записать через степени тройки:

$$3^{16} + 3^5 - 3^2$$

В троичной системе счисления это будет выглядеть так:

**10000000000000000 + 100000 — 100 = 10000000000100000 — 100**

Вычитая в столбик в троичной системе счисления, получим **10000000000022200.**

Или количество двоек в выражении **$3^5-3^2$** будет равно разнице степеней **5-2=3.**

**Ответ: 3 двойки**

# Методические особенности подготовки учащихся к выполнению заданий повышенного уровня сложности ЕГЭ по информатике и ИКТ


**СОСТАВИТЕЛЬ:**

**Назаревич Надежда Васильевна**
методист МУ «Информационно-методический центр»


Материалы предоставлены в МУ «Информационно-методический центр»
общеобразовательными организациями
МО ГО «Сыктывкар»


Печатается в авторской редакции
Оригинал-макет подготовлен и распечатан
в МУ «Информационно-методический центр»
167000, Республика Коми, г. Сыктывкар, ул. Южная, 15.
тел. (8212) 24-00-30. E-mail: mu_imc@mail.ru
Подписано в печать 25.12.2016 г.
Бумага офсет. Формат 60х84/8. Гарнитура Arial Narrow, Monotype Corsiva, Times New Roman