

**Дидактический материал для
подготовки к государственной
итоговой аттестации в форме ОГЭ для
9 класса. Часть 1.**

Никитина Татьяна
Евгеньевна, учитель
информатики

Аннотация:

Данный дидактический материал предназначен для отработки и закрепления знаний по информатике при подготовке к государственной итоговой аттестации в форме ОГЭ. Представляет собой образец разбора типовых заданий ОГЭ по предмету. Содержит справочные сведения, направленные на усвоение и закрепление материала.

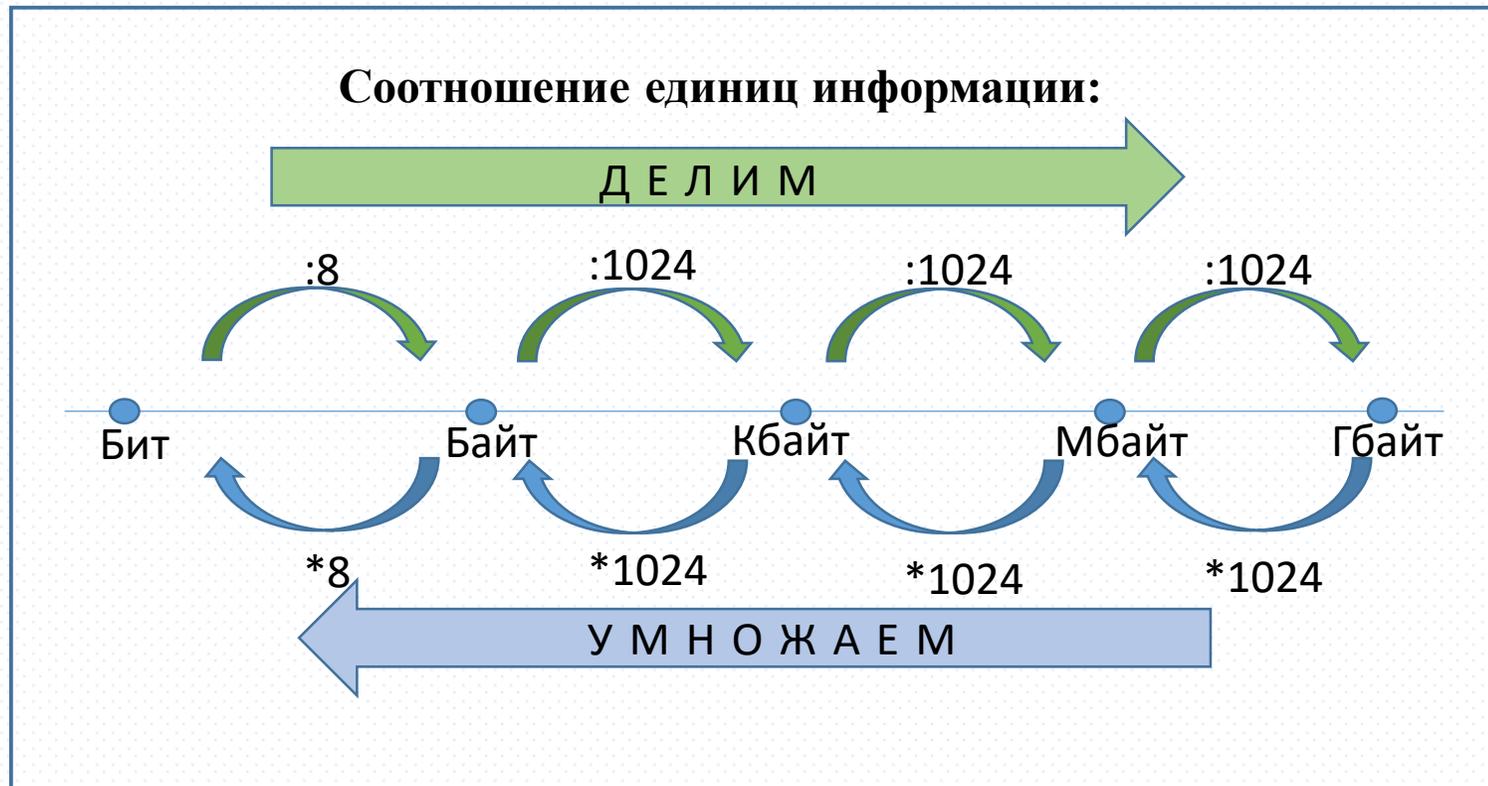


Задание №1. Количественные параметры информационных объектов.

Для начала вспомним некоторый теоретический материал. Соотношение единиц информации и кодировочные таблицы. Запомним числа **8** и **1024**! Бит – самая маленькая единица измерения.

Кодировочные таблицы:

1. КОИ-8 (1 символ кодируется 8 битами или 1 байтом)
2. Unicode (1 символ кодируется 16 битами или 2 байтами)
3. Остальные – по подобию первых двух!



ЗАПОМНИТЕ!!!

Каждый символ в компьютере занимает определенное количество памяти! Буква, цифра, знак препинания и **ДАЖЕ ПРОБЕЛ!** Количество битов определяется кодировочной таблицей. См выше



Разберем подробнее два типа задания.



Пример 1.

В кодировке КОИ-8 каждый символ кодируется 8 битами. Андрей написал текст (в нем нет лишних пробелов):

«Чад, Куба, Катар, Швеция, Эстония, Танзания, Сальвадор — страны».



Ученик вычеркнул из списка название одной из стран. Заодно он вычеркнул ставшие лишними запятые и пробелы — два пробела не должны идти подряд.

При этом размер нового предложения в данной кодировке оказался на 11 байтов меньше, чем размер исходного предложения. Напишите в ответе вычеркнутое название страны.

1 Помечаем себе в черновик, что кодировка КОИ-8, а это значит, что каждый символ кодируется 8 битами или 1 байтом.



1 символ – 1 байт

2 Смотрим, на сколько байтов сообщение оказалось меньше.



На 11 байт. Значит, если 1 символ 1 байт, тогда 11 байтов – это 11 символов.

3 Теперь главное не забыть, что пробел и запятая - это тоже символы и они так же, как и буквы, занимают по 1 байту информации.



Делаем следующие действия:
 $11 - 1(\text{запятая}) - 1(\text{пробел}) = 9$

В итоге осталось 9 символов. Ищем слово, которое состоит из 9 букв. Это **САЛЬВАДОР**

Пример 2.

В кодировке Unicode каждый символ кодируется 16 битами. Паша хотел написать текст (в нём нет лишних пробелов):

«Скользя по утреннему снегу,
Друг милый, предадимся бегу
Нетерпеливого коня
И навестим поля пустые...»

Одно из слов ученик написал два раза подряд, поставив между одинаковыми словами один пробел. При этом размер написанного предложения в данной кодировке оказался на 14 байт больше, чем размер нужного предложения. Напишите в ответе лишнее слово.

Эта задача обратная первой. Здесь нам нужно найти лишнее слово.

1 16 бит – 2 байт. Один символ – 2 байта!

2 В условии сказано, что он добавил лишний пробел. Это еще 2 байта на символ.

3 Все сообщение стало больше на 14 байт. Следовательно, $14:2=7$ (символов)

4 Из общего числа символов вычитаем один пробел.

$7-1(\text{пробел})=6$ (символов на слово)

5 Ищем слово из 6 букв. Лишнее слово - **ПУСТЫЕ**



Ответ: **пустые**

Задание 2. Кодирование и декодирование информации



Кодирование — это форма представления информации, удобная для её хранения, передачи и обработки. При кодировании символам исходного алфавита ставятся в соответствие так называемые **коды**.



Расшифровка или декодирование представляет собой восстановление исходного сообщения из последовательности кодов.



Декодирование может быть однозначным и многозначным. Если при расшифровке возможно получить несколько различных вариантов исходного сообщения, то такое **декодирование неоднозначно**.



При расшифровке с единственным возможным результатом сообщения имеет место **однозначное декодирование**.



Иногда процесс декодирования проще выполнять **с конца** зашифрованного сообщения.



Разберем пример:

От разведчика было получено сообщение:

001001110110100

В этом сообщении зашифрован пароль – последовательность русских букв.

В пароле использовались только буквы

А, Б, К, Л, О, С;

каждая буква кодировалась двоичным словом по следующей таблице:

А	Б	К	Л	О	С
01	100	101	111	00	110

Расшифруйте сообщение. Запишите в ответе пароль.



РЕШЕНИЕ:

Распределим цифры закодированного сообщения по кодовым словам букв, представленных в таблице. Будем «двигаться» слева направо:

00 100 111 01 101 00
↓ ↓ ↓ ↓ ↓ ↓
О Б Л А К О

Ответ: ОБЛАКО

Задание 3. Алгебра логики

Конъюнкция – истина тогда, когда оба высказывания **ИСТИННЫ!**
Обозначения:
 \wedge , И, and, &

Дизъюнкция – ложь тогда, когда оба высказывания **ЛОЖНЫ!**
Обозначения:
 \vee , ИЛИ, or, |

Инверсия – делает истинное высказывание ложным, и наоборот.
Обозначения:
НЕ, \bar{A} , \neg

Рассмотрим пример:

Для какого числа X истинно высказывание:
 $(X < 8)$ И НЕ $(X < 7)$?



Во-первых, у нас есть частица **НЕ** и конъюнкция. По правилам приоритетности, сначала избавимся от частицы НЕ. При этом знак **СТРОГОГО** неравенства поменяется и станет **НЕСТРОГИМ!!!** Это правило действует в **ОБЕ СТОРОНЫ!!!** Далее применяем операцию **КОНЪЮНКЦИЯ**. Смотрим в табличку выше и видим, что **ИСТИНА** будет тогда, когда **ОБЯЗАТЕЛЬНО** оба высказывания **ИСТИННЫ**. Подбираем число X и записываем ответ.

$\rightarrow (X < 8)$ И $(X \geq 7)$ \rightarrow ответ 7

ЗАПОМНИ!!!

Было НЕ $<$ - станет \geq	Было НЕ $>$ - станет \leq
Было НЕ \geq - станет $<$	Было НЕ \leq - станет $>$

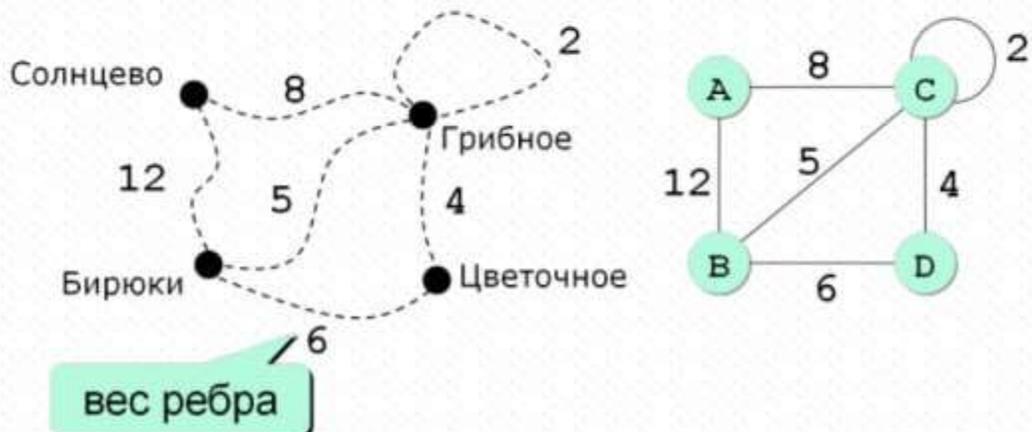
Задание 4. Графы. Поиск кратчайшего пути (перебор)

Иногда очень трудно структурировать информацию описанными структурами из-за сложных «взаимоотношений» между объектами. Тогда можно использовать графы:

 **Граф** – это набор вершин и связей между ними, называющихся рёбрами:



Взвешенные графы и весовая матрица
У взвешенных графов указан «**вес ребра**»:

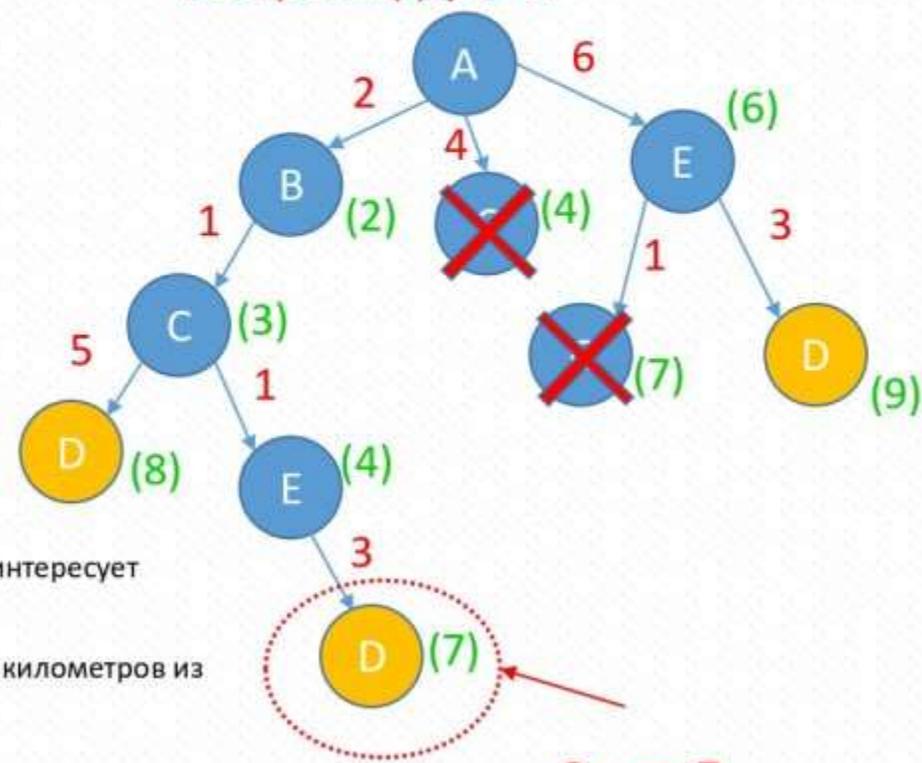


 Из взвешенных графов получается весовая матрица, обратное преобразование тоже возможно.

Определение кратчайшего пути между пунктами **A** и **D**:

	A	B	C	D	E
A		2	4		6
B	2		1		
C	4	1		5	1
D			5		3
E	6		1	3	

Построим дерево!



 - Вершина, которая нас не интересует

5 – вес ребра или количество километров из одного пункта в другой

(8) – результат сложения путей

Ответ: 7

Задание 5. Команды исполнителя

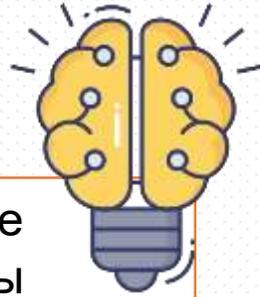
Для этого задания никаких теоретических знаний не нужно. Только логическое мышление и здравый смысл, а также знания по математике за 6 класс. Тут мы рассмотрим примеры и варианты решения данной задачи.

ЗАДАЧА 1.

У исполнителя Квадратор две команды, которым присвоены номера:

1. возведи в квадрат
2. прибавь 3

Составьте алгоритм получения **из числа 1 числа 25**, содержащий не более 5 команд. В ответе запишите только номера команд.



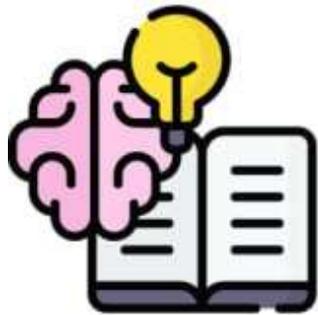
Итак, здесь все просто. Нам известны обе команды. Поэтому сразу начинаем продумывать первую команду. Единицу мы не можем возвести в квадрат. **Поэтому первое действие +3.** Далее размышляем: если мы еще раз прибавим 3 и потом возведем в квадрат, получится перебор (49). **Поэтому второе действие это возведение в степень.** Получаем 16. Таким образом, дальше можно понять, что остальные операции - только сложение. Это легкий пример. Разберем теперь с одной неизвестной командой.



1. $1+3=4$
2. $4^2=16$
3. $16+3=19$
4. $19+3=22$
5. $22+3=25$

А теперь внимание: в ответе нужно написать **НОМЕРА КОМАНД!!!** Смотрим вот сюда  А теперь вот сюда 
Первое действие +3, а это **2 команда!** Второе действие возвести в степень – **1 команда.** Таким образом, ответ будет 21222

Задание 5. Команды исполнителя



ЗАДАЧА 2.

У исполнителя Альфа две команды, которым присвоены номера:

1. прибавь 1
2. умножь на b

(b – неизвестное натуральное число; $b \geq 2$)

Первая из них увеличивает число на экране на 1, вторая умножает его на b .

Алгоритм для исполнителя Альфа – это последовательность номеров команд.

Найдите значение числа b , при котором из числа 6 по алгоритму **11211** будет получено число **82**.

Данный пример можно решить таким же способом, как в предыдущей карточке. А можно с помощью уравнения.

Итак, нам известна команда **11211**. Смотрим на номера. Первым действием будет +1, далее +1, потом *2 и еще два раза +1. Нам нужно из числа 6 получить число 82.

Объединим все действия в уравнение с одним неизвестным значением b .

1. $(6+1)$
2. $(6+1)+1$
3. $((6+1)+1)*b$
4. $((((6+1)+1)*b)+1)$
5. $(((((6+1)+1)*b)+1)+1)$

Раскрываем все скобки и получаем следующее:

$$(((6+1)+1)*b)+1=82$$

$$8b+2=82$$

$$8b=80$$

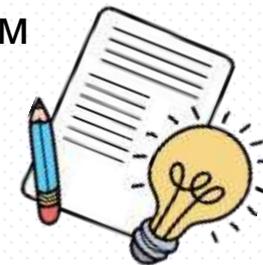
$$b=10$$

Ответ: 10



Задание 6. Анализируем программу.

В этом задании нам предлагают программу, которая записана аж на 5 языках программирования, так как в каждой школе изучается тот или иной язык. У нас, как правило, это Паскаль и Питон. Но это совершенно неважно. Нам нужно найти одну строчку с условием и внимательно прочитать задание. Прежде чем приступить к разбору, вспомним обозначения логических операций.



Бейсик → `IF s < 5 OR t > 5 THEN`
Паскаль → `if (s < 5) or (t > 5)`
С++ → `if (s < 5 || t > 5)`
Алгоритмический язык → `если s < 5 или t > 5`
Питон → `if (s < 5) or (t > 5):`

Смотрим на эти знаки

В данном случае представлена дизъюнкция. А это значит, что истина будет тогда, когда хотя бы одно условие будет принимать значение истина! Ложь – когда оба условия ложны.

Бейсик → `IF s >= 8 AND t >= 10 THEN`
Паскаль → `if (s >= 8) and (t >= 10)`
С++ → `if (s >= 8 && t >= 10)`
Алгоритмический язык → `если s >= 8 и t >= 10`
Питон → `if (s >= 8) and (t >= 10):`

Смотрим на эти знаки

А здесь уже конъюнкция. Истина будет лишь в единственном случае, когда оба условия истинны. Все остальное – ложь (если хотя бы одно условие принимает значение ложь!)

После того как мы определили, какая логическая операция представлена в задаче, смотрим на самую последнюю строку в задании! **Например, сколько было запусков, при которых программа напечатала «YES»?**

А теперь приступим к разбору примера.



Все задание достаточно объемное, поэтому мы разберем только программу на одном языке. Это будет Паскаль.

Паскаль

```
var s, t: integer;
begin
  readln(s);
  readln(t);
  if (s < 5) or (t > 5)
  then
    writeln("YES")
  else
    writeln("NO")
  end.
```



(2,5): 2 меньше 5 – да, 5 больше 5 – нет.
(5,2): 5 меньше 5 – нет, 2 больше 5 – нет.
(4,4): 4 меньше 5 – да, 4 больше 5 – нет.
(2,-2): 2 меньше 5 – да, -2 больше 5 – нет.
(3,1): 3 меньше 5 – да, 1 больше 5 – нет.

```
if (s < 5) or (t > 5)
```

Итак, первое, куда смотрим – это 5 строка программы. Далее определяем, что применяется дизъюнкция.

Второе – читаем задание после всех программ.

✓ Было проведено 9 запусков программы, при которых в качестве значений переменных вводились следующие пары чисел (s, t):
(2, 5); (5, 2); (4, 4); (2, -2); (3, 1); (8, 3); (9, -7); (7, 7); (4, 6).

Сколько было запусков, при которых программа напечатала «YES»?

Разберемся, что же это за пары чисел.

s t

(2,5) – здесь 2 – это переменная s, а 5 – это переменная t.

То же самое и в других парах чисел.

Далее я рекомендую записать все точки в столбец и подставлять значения вместо букв (s и t).



Так как это **дизъюнкция**, то нам достаточно хотя бы одного ответа **ДА**.

Теперь считаем количество и записываем ответ. ✓

Ответ: 6

* - даже если получилось два ДА, пара чисел одна, а мы считаем именно их. Поэтому эта строчка будет иметь значение 1!

Теперь разберем пример с конъюнкцией на примере программы, написанной на Питоне.

Python

```
s = int(input())
t = int(input())
if (s >= 8) and (t >= 10):
    print("YES")
else:
    print("NO")
```

Делаем все то же самое, что и в предыдущем примере. Только тут смотрим на 3 строчку. Видим союз **and** – значит это конъюнкция!

✓ Было проведено 9 запусков программы, при которых в качестве значений переменных вводились следующие пары чисел (s, t):
(6, 8); (9, 10); (9, 5); (-10, 9); (5, 6); (9, -10); (10, 9); (8, 10); (17, 10).

Сколько было запусков, при которых программа напечатала «YES»?

Точно так же помним, что s – это первое число, t – это второе число.
Выписываем пары чисел.

```
if (s >= 8) and (t >= 10):
```

(6, 8): 6 больше или равно 8 – нет, 8 больше или равно 10 – нет.
(9, 10): 9 больше или равно 8 – **да**, 10 больше или равно 10 – **да**.
(9, 5): 9 больше или равно 8 – да, 5 больше или равно 10 – нет.
(-10, 9): -10 больше или равно 8 – нет, 9 больше или равно 10 – нет.
(5, 6): 5 больше или равно 8 – нет, 6 больше или равно 10 – нет.
(9, -10): 9 больше или равно 8 – да, -10 больше или равно 10 – нет.
(10, 9): 10 больше или равно 8 – да, 9 больше или равно 10 – нет.
(8, 10): 8 больше или равно 8 – **да**, 10 больше или равно 10 – **да**.
(17, 10): 17 больше или равно 8 – **да**, 10 больше или равно 10 – **да**.

Так как в этом примере **конъюнкция**, то нам нужно найти такие пары чисел, в которых будет **два ДА!!!**

Ответ: 3



Задание 7. Адресация в Интернете

Данное задание очень простое и требует от вас знания определенной последовательности адреса. Разберемся.

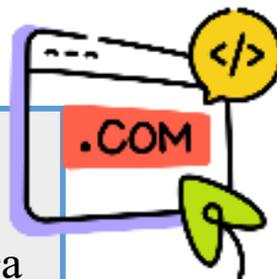
Адрес документа в Интернете состоит из следующих частей:

На первом месте всегда пишется ПРОТОКОЛ

- **http** (для web-страниц)
- **ftp** (для передачи файлов)
- **https** (защищенный протокол)



На втором месте записывается доменное имя сайта



Имя файла (всегда заканчивается расширением. Например, pdf, xls, jpg, docx и др.)

https://school.city.ru/inform/Lesson15.doc

:// - символ-разделитель протокола от остальной части



/ - слэш разделяет каталоги

Каталог на сервере (папка, где лежит файл)



Решим пример

Доступ к файлу **tiger.doc**, находящемуся на сервере **zoo.org**, осуществляется по протоколу **http**. Фрагменты адреса файла закодированы буквами от А до Ж. Запишите последовательность этих букв, кодирующую адрес указанного файла в сети Интернет.

- А) .doc
- Б) zoo
- В) /
- Г) ://
- Д) tiger
- Е) .org
- Ж) http

Итак, если вы внимательно прочтаете задание и запомните правильную последовательность, то сможете его решить за 1 минуту!

Ж Г Б Е В Д А
http://zoo.org/tiger.doc

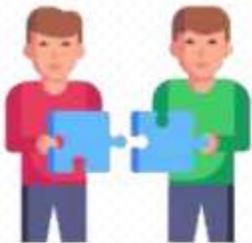
1. Протокол

2. Далее ://

3. Сервер

4. / (слэш)

5. Файл



Теперь, подставляем буквы под получившийся адрес. Получаем ЖГБЕВДА

Задание 8. Поиск информации. Формулирование запросов.

Данное задание можно решить двумя способами: с помощью формулы и кругов Эйлера-Вена. Рассмотрим каждый.

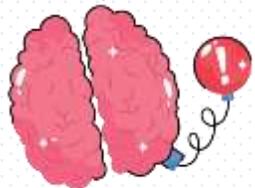
В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» — символ «&».

Запрос	Количество страниц(тыс.)
Сербия & Хорватия	500
Сербия Хорватия	3000
Сербия	2000



Какое количество страниц (в тысячах) будет найдено по запросу *Хорватия*? Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Будем использовать формулу и подставлять значения.



$$N_A = N_{A|B} + N_{A\&B} - N_B$$

Предположим, что А – это Сербия, а В – Хорватия.

Тогда, А|В – это Сербия|Хорватия, а А&В – Сербия&Хорватия

Подставляем:

$$2000 = 3000 + 500 - N_B \text{ (заменяем } N_B \text{ на } X \text{ – так вам будет проще)}$$

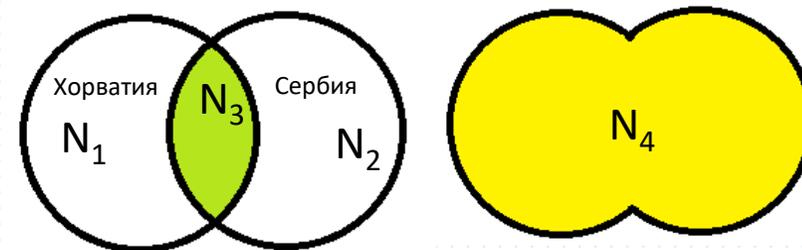
$$2000 = 3500 - X$$

$$2000 - 3500 = -X$$

$$X = 1500$$

Ответ: 1500

Решение с помощью кругов Эйлера-Вена



Распишем все области на кругах.

N_1 – это множество страниц *Хорватия*

N_2 – это множество страниц *Сербия*

N_3 – это множество страниц

Хорватия&Сербия

N_4 – это множество страниц *Хорватия|Сербия*

В задании сказано, что нужно найти количество страниц по запросу *Хорватия*. Это N_1 .

Что нужно сделать? Из общего количества страниц (N_4) отнять множество страниц *Сербия* (N_2) и прибавить множество страниц *Хорватия&Сербия* (N_3).

Получаем следующее:

$$N_1 = N_4 - N_2 + N_3 \longrightarrow \text{(данные берем из таблицы)}$$

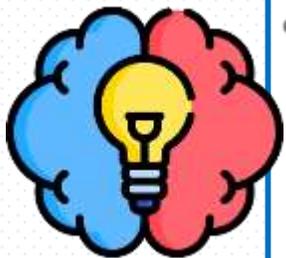
$$N_1 = 3000 - 2000 + 500$$

$$N_1 = 1500$$



Ответ: 1500

Разберем еще один пример, который может показаться вам сложным. На деле же это не так.



В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для логической операции «И» — символ «&».

В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

Запрос	Найдено страниц (в тысячах)
Сириус & Вега	260
Вега & (Сириус Арктур)	467
Сириус & Вега & Арктур	131

Какое количество страниц (в тысячах) будет найдено по запросу Вега & Арктур ?

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Какая же здесь сложность может возникнуть? Если вы внимательно посмотрите на таблицу и попытаетесь подставить значения в формулу, то поймете, что здесь больше действий, чем обычно. Но на самом деле все просто.

Сириус & Вега	260
Вега & (Сириус Арктур)	467
Сириус & Вега & Арктур	131

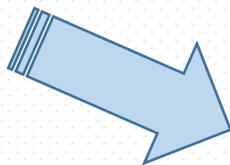
Напоминаю, **&** - это умножение!

И если в каждом множестве убрать повторяющиеся значения со знаком **&**, то получится все упростить и решить по формуле.

Подчеркну это в нашей таблице.



Найти: Вега & Арктур



Таким образом, получаем привычный вид задания, которое можно решить по уже известной нам формуле.

$$N_A = N_{A|B} + N_{A\&B} - N_B$$

Задание 9. Анализирование информации, представленной в виде схем

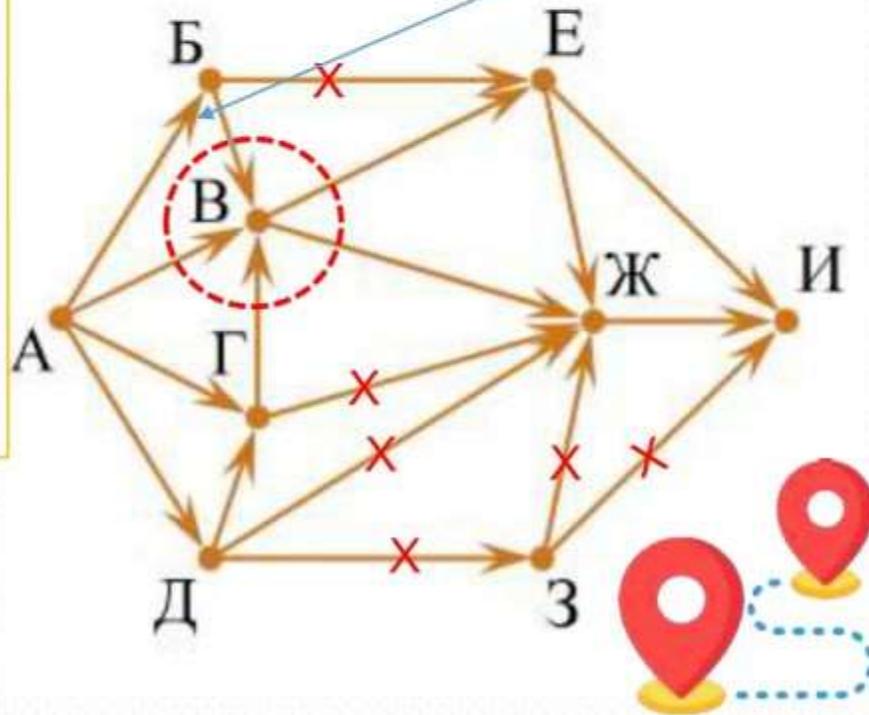
На первый взгляд это задание кажется простым. Но именно в этом задании допускаются много ошибок. Здесь нужно быть внимательным.

Рассмотрим два задания, которые немного отличаются друг от друга.

На рисунке — схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, З и И. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город И, проходящих через город В?

1 Нужно подсчитать все пути из А в И и обязательно заехать в город В! Значит, те дороги, которые не ведут в В, мы зачеркнем.

 Город, в который нужно заехать
 Вычеркнутая дорога

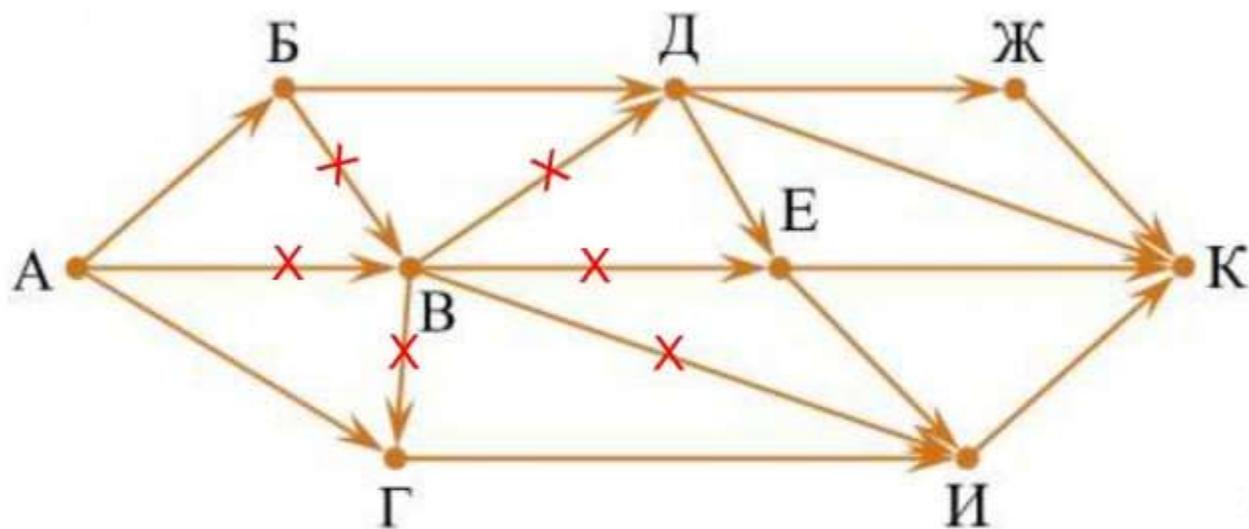


2 Теперь построим вспомогательную таблицу. Будем считать количество стрелок в каждом городе.

В город Б одна дорога из А (одна стрелка).	$B=A(1)=1$
В городе В стрелки из АБ и Г . *на этом этапе Г неизвестно, найдем его ниже и потом запишем	$V=A(1)+B(1)+G^*(2)=4$
В городе Г два пути — из А и Д . *на этом этапе Д неизвестно, найдем его ниже и потом запишем	$G=A(1)+D^*(1)=2$
В Д один путь из А .	$D=A(1)=1$
В Е два пути, но один не считаем.	$E=B(4)=4$
В Ж пять путей, но три не считаем.	$Zh=B(4)+E(4)=8$
 З не считаем, т.к. все дороги зачеркнуты	
В И три дороги, но из З не считаем.	$I=E(4)+Zh(8)=12$
Ответ:	12

Во втором примере задание похоже, но в какой-то город, наоборот, заезжать нельзя!

На рисунке — схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из пункта А в пункт К, не проходящих через пункт В?



1 Теперь мы зачеркнем все дороги, которые ведут в город **В**. А также те, которые выходят и **В**.

2 Как только мы сделали первый этап, переходим к построению вспомогательной таблицы.

В город Б одна дорога из А .	$B=A(1)=1$
В обходим стороной 	-
В город Г тоже одна дорога из А .	$\Gamma=A(1)=1$
В город Д две дороги, но одну мы зачеркнули.	$D=B(1)=1$
В город Е две дороги, но одну мы зачеркнули.	$E=D(1)=1$
В город И три дороги, но из В мы не считаем.	$I=E(1)+\Gamma(1)=2$
В город Ж одна дорога – из Д .	$Ж=D(1)=1$
В город К четыре дороги.	$K=Ж(1)+D(1)+E(1)+I(2)=5$
Ответ:	5

Задание 10. Сравнение чисел в различных системах счисления

В данном задании вам нужно иметь примитивные знания математики и запомнить один простой алгоритм. Как правило, в заданиях требуется перевести три числа из разных систем счисления в одну – десятичную.

АЛГОРИТМ:

1. Расставьте разряды над числом, двигаясь справа-налево, начиная с 0.
2. Поочередно умножаем каждую цифру числа на основание системы счисления и возводим в степень номера разряда.
3. Таким образом суммируем все цифры.



Любое число в нулевой степени дает нам единицу! ЗАПОМНИ!!!

$$n^0=1$$

Разберем пример.

Среди приведенных ниже трех чисел, записанных в различных системах счисления, найдите **максимальное** и запишите его в ответе в десятичной системе счисления. В ответе запишите только число, основание системы счисления указывать не нужно.

$$23_{16}, 32_8, 11110_2$$

Система счисления

Это можно сразу зачеркнуть и не считать, т.к. умножение на ноль всегда дает ноль!

1) $23_{16} = 2 \cdot 16^1 + 3 \cdot 16^0 = 32 + 3 = 35_{10}$ 3) $11110_2 = 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 16 + 8 + 4 + 2 + 0 = 30_{10}$

2) $32_8 = 3 \cdot 8^1 + 2 \cdot 8^0 = 24 + 2 = 26_{10}$

Читаем внимательно задание и видим, что в ответ нужно записать максимальное число. Сравниваем получившиеся числа и записываем ответ. Получаем ответ **35**. **Основание системы счисления писать не надо!**

Список литературы:

1. Босова, Л.Л. Информатика, 9 класс / Л. Л. Босова, А. Ю. Босова. – М.:БИНОМ. Лаборатория знаний, 2022. – 208 с.
2. Босова, Л.Л. Подготовка к ОГЭ по информатике. 9 кл. / Л.Л. Босова, Тарапата В.В., Босова А.Ю. [и др.]: Изд. Просвещение, 2023. – 112 с.
3. Блог об Информатике, Математике и Физике!: офиц. сайт. – URL: <https://code-enjoy.ru/>
4. Образовательный портал для подготовки к экзаменам: офиц. сайт. – URL: <https://inf-oge.sdamgia.ru/>
5. Online Test Pad - Онлайн тесты, опросы, кроссворды: офиц. сайт. – URL: <https://onlinetestpad.com/ru>
6. Векторные иконки и стикеры — PNG, SVG, EPS, PSD и CSS.: офиц. сайт. – URL: <view-source:https://www.flaticon.com/ru/>

