

**Методические указания для учащихся 8-9 классов  
по Отраслевой олимпиаде школьников «Газпром»  
профиль «Информационные и коммуникационные технологии»**

## ОГЛАВЛЕНИЕ

Задание 1. Биоинформатика .....	3
Задание 2. Цифровая социология .....	4
Задание 3. Булева алгебра .....	5
Задание 4. Алгоритмы.....	6
Задание 5. Логика .....	7
Задание 6. Разработка метода оптимизации.....	9
Задание 7. Системное администрирование .....	11
Задание 8. Разработка алгоритма сложной сортировки .....	12
Задание 9. Комбинаторика .....	13
Задание 10. Теория игр .....	14

## Задание 1. Биоинформатика

**Текст задачи.** Для исследования местности в области подготовки перспективного месторождения была отправлена группа экологов, задача которых изучить местную флору. Для этого ей был собран набор образцов для проведения анализа их дезоксирибонуклеиновой кислоты (ДНК), являющейся основной молекулой хранения информации в биологических системах. Она состоит из четырех оснований нуклеиновых кислот: гуанина («G»), цитозина («C»), аденина («A») и тимина («T»). Как выяснилось в цепочках ДНК элементы отвечают следующим закономерностям: аденин и тимин дополняют друг друга, как и гуанин с цитозином. В результате анализа появились три цепочки одной из сторон ДНК:

№	Сторона 1	Сторона 2
1.	AAAA	?
2.	ATCCCGATA	?
3.	CACGGTATT	?

Вам необходимо получить вторую дополняющую ее сторону. Кроме того, для перехода с позиции стажера на позицию младшего научного сотрудника руководитель группы будет следить не только за финальным результатом, но и ходом его достижения. Поэтому он предложил отказаться от использования условных операторов в ходе решения.

**Решение задачи.** Для нахождения второй стороны ДНК достаточно подать на вход элементы первой стороны, заменив имеющиеся элементы на «парные». Так как в задаче нежелательным является использование условных операторов, делающих программу излишне громоздкой, можно прибегнуть к сочетанию циклов и словарей.

Пример программной реализации на Python

```
def dna_to_rna(dna):
    dna_dict = { 'A': 'T', 'T': 'A', 'C': 'G', 'G': 'C' }
    result = []
    for letter in dna:
        new_letter = dna_dict[letter]
        result.append(new_letter)
    return ''.join(result)
```

## Задание 2. Цифровая социология

**Текст задачи.** Вы аналитик данных и имеете задачу изучения отношения к компании Нефтепром-Сибирь на основе сообщений в корпоративной социальной сети. Применяемые способы анализа множества отдельных сообщений не всегда дают удобную для восприятия, а главное достаточную картину. Поэтому, погружаясь в тематику было принято решение модернизировать их, внедрив анализ тенденций. Для этого на первом этапе необходимо выделять два типа сущностей: субъектов взаимодействия и связи между ними, которые могут быть заданы как напрямую, так и не явно. На основе одного из сообщений составьте блок-схему таких субъектов и установите связи между ними. В комментариях к решению опишите принципы, по которым они были выделены.

Доставленные в Новый Уренгой операторы газодобычи не смогли документально подтвердить приказы о командировании из-за сбоя в информационной системе. Трехчасовая задержка позволила выявить и устранить уязвимость в работе сетевых протоколов и сообщить начальникам участков и в региональный офис информацию о задержке.

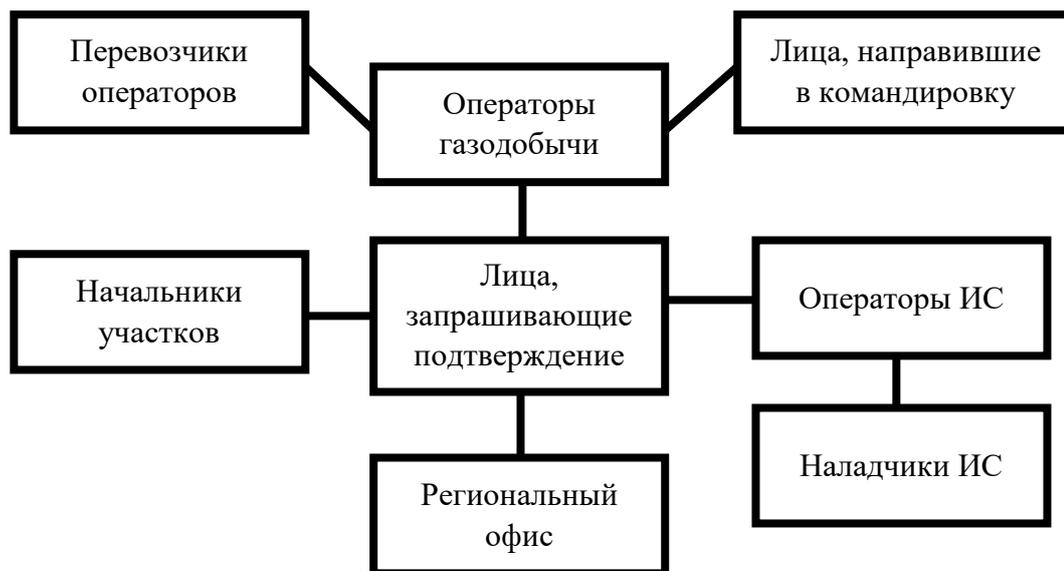
**Примечание:** субъект взаимодействия – это сущность, являющаяся источником деятельности в некотором процессе

**Решение задачи.** В данной задаче необходимо опираться строго на текст сообщения, не добавляя тех элементов, которые в нем не упоминаются. В первую очередь необходимо понять – кто или что является субъектом, а что им являться не может. Так субъектом может быть какое-либо физическое лицо или организационная система – например какой-либо отдел компании. В свою очередь субъектом не может являться процесс, такой как *задержка*, или атрибут какой-либо системы – например *уязвимость*.

Составим список субъектов взаимодействия, идя последовательно по сообщению (имена субъектов можно давать с некоторой условностью, главное – прослеживаемость связи между ними и текстом):

	<b>Субъект взаимодействия</b>	<b>Слово(а) в тексте</b>
1.	Перевозчики операторов	Доставленные
2.	Операторы газодобычи	Операторы газодобычи
3.	Лица, запрашивающие подтверждение	не смогли ... подтвердить
4.	Лица, направившие в командировку	Приказы о командировании
5.	Операторы информационной системы	Сбоя в информационной системе
6.	Наладчики информационной системы	Выявить и устранить уязвимость
7.	Начальники участков	Начальникам участков
8.	Региональный офис	Региональный офис

Далее составим требуемую блок-схему:



Также стоит обратить внимание, что некоторые связи могут как бы «напрашиваться» – например связь между *лицами, направившими в командировку* и *перевозчиками операторов*. Однако прямой отсылки к данной связи в сообщении нет, а значит она не является достоверной.

### Задание 3. Булева алгебра

**Текст задачи.** Степан и Федор пытались проверить одну из своих гипотез перед завтрашним докладом на совещании. На утро Степан обнаружил, что записи часто велись ручкой поверх ранее сделанных записей и разобрать можно далеко не все. Тем не менее ему удалось восстановить связность исследуемых им факторов, выраженную в формуле:  $Z \rightarrow (X \vee Y \wedge Z) \equiv X$  и часть таблицы истинности.

...	...	...	<b>F</b>
<b>1</b>	<b>1</b>	...	<b>1</b>
<b>1</b>	...	...	<b>1</b>

Восстановите соответствие факторов (X, Y, Z) и их значений.

**Решение задачи.** Для начала построим таблицу истинности для заданной схемы  $Z \rightarrow (X \vee Y \wedge Z) \equiv X$

X	Y	Z	$Y \wedge Z$	$X \vee Y \wedge Z$	$Z \rightarrow (X \vee Y \wedge Z)$	F
0	0	0	0	0	1	0
0	0	1	0	0	0	1
0	1	0	0	0	1	0
0	1	1	1	1	1	0
1	0	0	0	1	1	1
1	0	1	0	1	1	1
1	1	0	0	1	1	1
1	1	1	1	1	1	1

По заданной таблице нам нужны те строки, где  $F = 0$ . Их получилось 3:

X	Y	Z	F
0	0	0	0
0	1	0	0
0	1	1	0

Далее расставим столбцы так, чтобы таблица соответствовала предложенной в задании:

Y	Z	X	F
1	1	0	0
1	0	0	0

#### Задание 4. Алгоритмы

**Текст задачи.** Разработка перспективного месторождения газа в пограничной зоне потребовала обеспечения дополнительных мер информационной безопасности. С этой целью был привлечен коллектив специалистов по шифрованию, однако его руководитель позаботился о том, чтобы никто из его членов не знал весь алгоритм, поэтому разделил его на элементы.

Вам поручили разработать часть метода дешифрования, производящего сортировку входящих строк. Каждое слово будет содержать число от 1 до 9, означающее его позицию в предложении. Во входящих сообщениях гарантированно не будет пустых, либо некорректных строк.

В качестве тестовой строки можно использовать следующую:  
нефт5 двадца3и П1анируется то4н добы2а

## Решение задачи.

Пример программной реализации на Python

```
def order(sentence):
    words = sentence.split()
    numbers = range(1, 10)
    result = []
    for number in numbers:
        str_number = str(number)
        for word in words:
            if str_number in word:
                result.append(word)
    return ' '.join(result)
```

## Задание 5. Логика

**Текст задачи.** На новом участке для нефтяников было последовательно установлено пять контейнеров для проживания пяти вахтовиков. Каждый из них имеет любимый фильм/сериал, любимое блюдо и область интересов, которой он занимается в свободное от работы время. Запишите все, что удалось узнать про жителя белого контейнера, если известно:

- Известно, что Иван живет в красном контейнере.
- Житель синего контейнера занимается киберспортом.
- Анна смотрит сериал «Ликвидация».
- Тот, кто все время ест печенье еще и смотрит «Доктора Живаго».
- Живущий в желтом контейнере предпочитает пить кумыс.
- Зеленый контейнер стоит рядом с тем, в котором смотрят фильм «Папа».
- Дамир терпеть не может солянку.
- Тот, кто любит пасьянсы живет по соседству с увлекающимся киберспортом.
- В зеленом контейнере смотрят сериал «Бермудский треугольник».
- Контейнер Анны стоит сразу справа от контейнера Дианы.
- Любитель хинкали любит создавать трехмерные модели.
- Семен очень любит фрукты.
- Первым номером маркирован белый контейнер.
- Сосед любителя солянки, с интересом играет на флейте.
- Сосед того, кто постоянно ест фрукты, ведет походный дневник.
- Иван живет между зеленым и синим контейнерами.
- В центральном контейнере смотрят сериал «Три мушкетера».

**Решение задачи.** Задачу можно решать с помощью таблиц. Построим сначала пустую таблицу с номерами контейнеров, прочитав утверждения, заполним те ячейки, информация о которых указана прямо в утверждениях.

Номер	1	2	3	4	5
Цвет	Белый				
Имя					
Фильм			Три мушкетера		
Блюдо					
Хобби					

Далее сопоставляем утверждения:

- Известно, что Иван живет в красном контейнере.
- Иван живет между зеленым и синим контейнерами.
- Анна смотрит сериал «Ликвидация».
- Контейнер Анны стоит сразу справа от контейнера Дианы.

Отсюда:

- Иван живет в третьем или четвертом контейнере.
- Анна живет во втором, четвертом или пятом контейнере.

Предположим, что Иван живет в третьем контейнере, зеленый контейнер стоит слева от него, синий – справа. Если в ходе решения придем к противоречию, попробуем другое размещение. Используем утверждения:

- Зеленый контейнер стоит рядом с тем, в котором смотрят фильм «Папа».
- Житель синего контейнера занимается киберспортом.
- В зеленом контейнере смотрят сериал «Бермудский треугольник».
- Живущий в желтом контейнере предпочитает пить кумыс.

Номер	1	2	3	4	5
Цвет	Белый	Зеленый	Красный	Синий	Желтый
Имя			Иван	Диана	Анна
Фильм	Папа	Бермудский треугольник	Три мушкетера		Ликвидация
Блюдо					Кумыс
Хобби				Киберспорт	

Значит появляется возможность учесть, что Диана ест печеньки и смотрит «Доктора Живаго». Мы не использовали утверждения:

- Любитель хинкали любит создавать трехмерные модели.
- Сосед того, кто постоянно ест фрукты, ведет походный дневник.
- Сосед любителя солянки, с интересом играет на флейте.
- Дамир терпеть не может солянку
- Тот, кто любит пасьянсы живет по соседству с увлекающимся киберспортом.
- Семен очень любит фрукты.

Можно сделать следующие выводы:

- Дамир не любит солянку, значит предпочитает хинкали
- Семен предпочитает фрукты, а значит Иван – солянку
- На флейте играют в зеленом контейнере
- Дамир живет в белом контейнере

Оставшиеся утверждения без труда приведут к нужному результату:

Номер	1	2	3	4	5
Цвет	Белый	Зеленый	Красный	Синий	Желтый
Имя	Дамир	Семен	Иван	Диана	Анна
Фильм	Папа	Бермудский треугольник	Три мушкетера	Доктор Живаго	Ликвидация
Блюдо	Хинкали	Фрукты	Солянка	Печеньки	Кумыс
Хобби	3D модели	Флейта	Походный дневник	Киберспорт	Пасьянс

Таким образом в белом контейнере проживает Дамир, он любит хинкали, часто пересматривает фильм «Папа» и увлекается трехмерным моделированием.

Возможно, вам удалось найти другое решение, не противоречащее условиям. В остальных вариантах приведены задачи такого же типа.

### Задание 6. Разработка метода оптимизации

**Текст задачи.** Геологическая партия, проводившая разведку полезных ископаемых, сбилась с ориентиров и попала на такой участок местности, где навигационная аппаратура дала сбой. Общий анализ направления движения показал, что в целом выдаваемый ей маршрут верен, однако он выдает

последовательность направлений, часть из которых является взаимнообратной. Например, сразу после рекомендации идти на север, аппаратура рекомендует идти в противоположную сторону - на юг, а сразу после запада отправляет на восток.

Идти в одном направлении, а затем сразу же возвращаться в противоположном по пересеченной местности в не располагающих к этому погодных условиях партия двигаться не может. При этом выдаваемая последовательность направлений довольно велика и наличие одной ошибки при ручном расчете приведет к тому, что выйти на нужную точку не получится. К счастью у одного из членов команды есть ноутбук и он может разработать программу, которая очистит лишние направления и позволит за приемлемое время добраться до пункта назначения.

В качестве теста работоспособности берется часть выдаваемого массива строковых значений:

```
север, запад, юг, запад, восток, восток, юг
```

Напишите программу, возвращающую только нужные направления движения. **Примечание:** не все пути можно сделать проще, так путь *север* → *запад* → *юг* → *восток* нельзя считать взаимнообратным, так как *север* и *запад*, *запад* и *юг*, *юг* и *восток* не являются прямо противоположными друг другу.

### Решение задачи.

Пример программной реализации на Python

```
opposite = {
    'север': 'юг',
    'юг': 'север',
    'запад': 'восток',
    'восток': 'запад'
}
def dir_reduc(plan):
    new_plan = []
    for d in plan:
        if new_plan and new_plan[-1] == opposite[d]:
            new_plan.pop()
        else:
            new_plan.append(d)
    return new_plan
```

## Задание 7. Системы счисления в системном администрировании

**Текст задачи.** В рабочем поселке вахтовиков произошел сбой сети, в результате которого одно из устройств комплекса добычи газа стало возвращать не стандартное отображение IP-адреса устройства и маски подсети, а его двоичное представление. Простой оборудования комплекса, который не может работать без этого устройства приведет к убыткам. Нужно срочно восстановить десятичное представление (1) IP-адреса, (2) маски подсети и (3) адреса сети с соответствующей пунктуацией для ручной передачи данных значений в комплекс добычи.

IP-адрес: 10110010101100001110011010101010

Маска подсети: 11111111111111111111111100000000

**Решение задачи.** Для преобразования IP-адреса и маски подсети из двоичного вида в десятичный необходимо каждый из них разделить на октеты – сочетания 8 чисел, разделенных точками. Таким образом получим:

IP-адрес: 10110010.10110000.11100110.10101010

Маска подсети: 11111111.11111111.11111111.10000000

Далее каждый разряд каждого октета преобразуется путем суммирования значений ненулевых октетов. Для примера возьмем последний октет IP-адреса:

Двоичный вид							
1	0	1	0	1	0	1	0
Перевод в десятичный вид							
128	0	32	0	8	0	2	0

Получим сумму всех десятичных чисел: 170. Аналогично вычисляются оставшиеся октеты IP-адреса и октеты маски подсети. Таким образом получим:

IP-адрес: 178.176.230.170

Маска подсети: 255.255.255.128

Для определения адреса сети необходимо выполнить поразрядную конъюнкцию (логическое умножение) IP-адреса и маски подсети в двоичном виде. Так как, согласно таблице истинности операции конъюнкции только  $1 * 1 = 1$ , а  $1 * 0$  и  $0 * 0$  имеют значение 0, то данную операцию можно сократить до последнего октета, т. к. первые три из них будут полностью повторять IP-адрес.

Двоичный вид							
1	0	1	0	1	0	1	0
1	0	0	0	0	0	0	0
Поразрядная конъюнкция							
1	0	0	0	0	0	0	0
Перевод в десятичный вид							
128	0	0	0	0	0	0	0

Таким образом адресом сети будет: 178.176.230.128

### Задание 8. Разработка алгоритма сложной сортировки

**Текст задачи.** Руководство терминала по переработке полезных ископаемых поставило задачу маркировки входящих экземпляров и поручило Вам разработать первую часть маркера. Было сформулировано следующее техническое задание: на вход приходит значение масс множества элементов породы, задача состоит в его сортировке. Сортируется входящий массив по возрастанию массы полезного вещества в породе.

Ранее было математически доказано: значение массы полезного вещества вычисляется как сумма цифр общей массы элемента породы. Например, если масса породы составил 192, то масса содержащегося в ней полезного вещества:  $1 + 9 + 2 = 12$ . На выходе метод возвращает отсортированный массив значений масс элементов породы (не их полезного вещества!).

В качестве теста работоспособности используйте следующее строковое множество значений масс элементов породы, пришедшее на вход:

"601 110 1194 3612 743 94 361 203 940"

**Примечание.** Одинаковый вес полезного вещества в разных элементах породы демонстрирует их эквивалентность, т. е. порядок их размещения между собой не имеет значения.

### Решение задачи.

Пример программной реализации на Python

```
def order_weight(string):
    weights = string.split(' ')
    order_weights = {}
    for weight in weights:
        sum = 0
        for digit in weight:
            sum += int(digit)
        order_weights[weight] = sum
    weights.sort()
    weights.sort(key = lambda x: order_weights[x])
    return ' '.join(weights)
```

### Задание 9. Комбинаторика

**Текст задачи.** В конце финансового года проводится традиционный аудит деятельности нескольких дочерних предприятий нефтегазового холдинга. В текущем году ими стали 13 нефтеперерабатывающих заводов (НПЗ) и 15 газотранспортных узлов (ГТУ). Какое количество вариантов выбора объекта аудита существует в начале второй недели, если в течение одной недели аудиторская фирма может посетить не более 2 НПЗ и 3 ГТУ?

### Решение задачи.

Всего необходимо проверить 13 НПЗ и 15 ГТУ. После первой недели осталось 11 НПЗ (так как  $13 - 2 = 11$ , т.е. на первой неделе проверили 2 НПЗ, и мы их исключаем из общего числа) и 12 перерабатывающих ( $15 - 3 = 12$ ). Для решения можно воспользоваться формулой числа сочетаний:

$$C_{11}^2 = \frac{11!}{2! * (11 - 2)!} = \frac{11 * 10 * 9!}{2 * 9!} = 55$$
$$C_{12}^3 = \frac{12!}{3! * (12 - 3)!} = \frac{12 * 11 * 10 * 9!}{6 * 9!} = 220$$

Иначе, из 11 НПЗ мы можем пятьюдесятью пятью способами взять по два. Из 12 ГТУ мы можем двумястами двадцатью способами взять по три. Каждое из 11 НПЗ может комбинироваться с каждым из 12 ГТУ. Перемножим эти числа и получим количество вариантов выбрать объекты для проверки на вторую неделю.  $55 * 220 = \mathbf{12100}$  вариантов.

Задачи в других вариантах аналогичны, поэтому приводится решение только для одного варианта.

### Задание 10. Теория игр

Два подразделения нефтедобывающей компании конкурируют между собой по количеству добытой нефти. В зависимости от наличия свободных резервуаров каждое из них может добыть такое количество тонн, которое определено следующей платежной матрицей:

$$\text{Добыча} = \begin{pmatrix} 12 & 22 \\ 32 & 2 \end{pmatrix}$$

Определите оптимальные стратегии по объемам добычи для обоих подразделений, а также цену игры.

Ответ задачи.

Оптимальные смешанные стратегии  $X = \left(\frac{3}{4}; \frac{1}{4}\right)$   $Y = \left(\frac{1}{2}; \frac{1}{2}\right)$

Цена игры  $v = 17$