

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

## ОГЛАВЛЕНИЕ

Задание 1. Автоматизация производственной деятельности .....	3
Задание 2. Логика .....	4
Задание 3. Хим-информатика .....	6
Задание 4. Измерение данных.....	7
Задание 5. Булева алгебра .....	8
Задание 6. Теория игр .....	10
Задание 7. Администрирование данных .....	11
Задание 8. Комбинаторика .....	12
Задание 9. Поиск простых решений сложных задач .....	12

## Задание 1. Автоматизация производственной деятельности

**Текст задачи.** При проектировании системы очистки нефти техническим заданием был определен этап сравнения нефтяных смесей по заданным критериям качества с участков добычи Звезда и Сияние. За один временной промежуток проводится десять итераций сравнения, в каждой из которых участвуют по 1589,883 литра с обоих участков. В случае, если топливо с участка набирает два очка качества на итерации и менее – оно попадает в хранилище LV-426, в остальных случаях – в хранилище HV-22. После обработки данных оба участка получают рейтинговые баллы в зависимости от результатов сравнения:

- Более высокое качество – 3 балла.
- Одинаковое качество – 1 балл.
- Менее высокое качество – 0 баллов.

Произведите (1) расчет количества литров нефтяной смеси в хранилищах HV-22 и LV-426 по итогам обработки тестового набора данных и (2) расчет рейтинговых баллов, которые заработали участки Сияние и Звезда. Тестовый набор данных представлен ниже, он имеет следующий синтаксис: Имя цистерны с участка Сияние, его очки качества: очки качества нефтяной смеси в цистерне с участка Звезда, ее имя.

```
["СияниеА 0:3 ЗвездаВВ", "СияниеВА 2:1 ЗвездаЗ", "СияниеВБ 1:1 ЗвездаУ",  
"СияниеОО 4:1 Звезда-ГШ", "СияниеНГ 2:3 ЗвездаРт", "Ф-Сияние 2:3  
ЗвездаДв", "СияниеВос 3:2 ЗвездаТр", "СияниеПШ 3:3 ЗвездаДШ",  
"СияниеХЛ 4:4 ЗвездаАВ", "СияниеМ-ЕТ 0:0 Звезда-воС"]
```

**Решение задачи.** Для решения необходимо задать четыре переменные счетчика – star, shine, HV\_22 и LV\_426 для записи в них необходимых параметров. Далее необходимо в исходном списке отделить символы, которые могут быть преобразованы в числовые от остальных. Для этого применяется соответствующий метод (**split()** в Python или аналогичный в других языках) – обращаться нужно к элементам в диапазоне длинны исходного списка. Так как разделителем выступит символ пробела, то все элементы будут декомпозированы на три, например: ["СияниеА", "0:3", "ЗвездаВВ"], что даст возможность обращаться к числам вторых элементов по индексам двумерного списка. Далее с помощью условных операторов необходимо последовательно прибавлять нужные значения к ранее объявленным переменным.

Пример программной реализации на Python:

```
lst = ["СияниеА 0:3 ЗвездаВВ", "...", "СияниеМ-ЕТ 0:0 Звезда-воС"]  
  
oil = 1589.883  
star, shine, HV_22, LV_426 = 0, 0, 0, 0  
for elem in range(len(lst)):  
    s = lst[elem].split()
```

```
if int(s[1][0]) > 2:
    HV_22 += oil
else:
    LV_426 += oil
if int(s[1][2]) > 2:
    HV_22 += oil
else:
    LV_426 += oil
if int(s[1][0]) > int(s[1][2]):
    shine += 3
if int(s[1][0]) < int(s[1][2]):
    star += 3
if int(s[1][0]) == int(s[1][2]):
    shine += 1
    star += 1

print(f'HV-22: {HV_22} литра(ов), LV-426: {LV_426} литра(ов)')
print(f'Звезда: {star} баллов, Сияние: {shine} баллов')
```

## Задание 2. 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 печенье 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 3. Хим-информатика

**Текст задания.** В химическую лабораторию несколько раз в день отгружают партии образцов для анализа их пригодности к производству определенного вида синтетического каучука. Недавнее внедрение базы данных существенно упростило работу, однако, ввиду большого объема присылаемых образцов, менеджеру требуется сразу видеть итоговый результат обработки по всем партиям одного заказчика. Вам поручили разработать метод расчета качества партий: на вход подается список, содержащий маркировку каждого образца как подходящего – 'good' или не подходящего – 'bad'. Если партия содержит один или два 'good'-образца, то метод должен вернуть '**Партия второго класса**', если более двух 'good'-

образцов – **'Партия первого класса'**. Если же партия вовсе не содержит подходящих образцов, то метод должен возвращать значение **'Неподходящий материал'**. Важным требованием является то, что все массивы могут иметь разную длину, а предлагаемое решение не должно быть чувствительно к регистру. Коллеги прислали выгрузку тестовых данных по трем партиям и попросили так же производить расчет количества примесей – они маркируются иначе чем 'good' или 'bad'.

Партия №1
[['bad', 'bAd', 'bad'], ['bad', 'goody', 'bad'], ['bad', 'bAd', 'bad']]
Партия №2
[['gOOd', 'bad', 'BAD', 'bad', 'bad'], ['dad', 'bAd', 'bad'], ['GOOD', 'bad', 'bad', 'bAd']]
Партия №3
[[hope, 'bAd', 'BAD', 'good', 'bad', 'GOOD'], ['bad'], ['gOOd', 'BAD']]

**Решение задачи.** Для решения задачи необходимо обладать знаниями о работе с функциями (процедурами), списками, в том числе двумерными, а также циклами и условными операторами. Ниже представлен пример программной реализации на Python:

```
def main(data):
    a, b = 0, 0
    for i in data:
        for st in i:
            st = st.lower()
            if st == 'good':
                a += 1
            elif st != 'bad':
                b += 1
    if a == 0:
        print('Неподходящий материал')
    elif a > 2:
        print('Партия первого класса')
    else:
        print('Партия второго класса')
    print(f'Количество примесей: {b}')
```

#### Задание 4. Измерение данных

**Текст задачи.** Между двумя рабочими участками месторождения планируется установить дополнительный канал обмена стерео аудио-сигналами (сообщениями) для ежедневных отчетных сеансов связи. Установите требуемую пропускную способность данного канала в кибибайтах

из расчета, что сеансы будут проводиться не более 51 минут. Ниже представлены требования к моно-сигналу в расчете на 1 секунду:

- Частота дискретизации: 63 Гц
- Глубина дискретизации: 17 бит
- Объем метаданных: 47 байт на каждые 5 Кибибайт аудио

**Решение задачи.** Пропускной способностью канала является показатель отношения объема передаваемых данных ко времени передачи, т. е., проще говоря – сколько данных можно передать за установленный отрезок времени. Так же стоит заметить, что показатели нам предоставлены для моно-сигнала, а расчет произвести необходимо для стерео-сигнала, т. е. получившиеся показатели увеличить в два раза. Для начала переведем все данные в единую шкалу – секунды и биты.

$$\text{Время для передачи данных} = 51 * 60 = 3060 \text{ секунд}$$

Для определения количества передаваемых данных нужно произвести умножение частоты дискретизации на ее глубину в отведенный период

$$\text{Объем данных} = 63 * 17 * 3060 = 3\,277\,260 \text{ бит}$$

Теперь вычислим объем метаданных

$$\text{Объем данных} = \frac{47 * 8 * 3\,277\,260}{5 * 1024} = 240\,674 \text{ бит}$$

Таким образом общий объем данных, нуждающихся в передаче, составит

$$\text{Общий объем данных} = (3\,277\,260 + 240\,674) * 2 = 7\,035\,868 \text{ бит}$$

И наконец определим пропускную способность в кибибайтах в секунду:

$$\text{Пропускная способность} = \frac{7\,035\,868}{8 * 1024 * 3060} = 0,28 \text{ Кибибайт/сек}$$

### Задание 5. Булева алгебра

**Текст задачи.** Евдокия и Сымбатай пытались разработать решение для проверки своих гипотез которое можно было бы быстро использовать прямо во время совещаний. Однако на утро было обнаружено, что программа не сохранилась, а из бумажных записей удалось разобрать далеко не все. Тем не менее им удалось восстановить связность факторов, исследуемой ими первичной модели, выраженную формулой  $Z \rightarrow (X \vee Y \wedge Z) \equiv X$  и часть таблицы истинности.

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

К текущему совещанию придется вручную закончить восстановление соответствия факторов (X, Y, Z) и их значений. Для дальнейшей работы необходимо разработать программу, которая бы автоматически составляла таблицу истинности для данной размерности таблицы по имеющейся формуле.

**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

Ниже представлена программная реализация на Python:

```
x, y, z, f = 0, 1, 1, 0

def func(x, y, z):
    return (x, y, z, (x <= (x or y and z)) == x)
```

## Задание 6. Теория игр

**Тест задачи.** Компании Самотлор и БалтикаПромНефть стремятся выйти на новые рынки и конкурируют между собой по количеству добытой нефти. На текущем этапе происходит расчет перспектив добычи в миллионах баррелей за заданный период времени. Подготовленный аналитический отчет указывает на четыре возможные стратегии игроков – они представлены в платежной матрице:

		Стратегии БалтикаПромНефти			
		1	2	3	4
Стратегии Самотлора	1	3	3	2	2
	2	2	5	3	3
	3	4	3	4	3
	4	2	4	5	4

Ввиду того, что часть стратегий может являться неподходящей для обоих игроков имеет смысл провести доминирование<sup>1</sup> и исключить таковые из анализа. Найдите оптимальные стратегии<sup>2</sup> по объемам добычи для двух игроков и цену игры<sup>3</sup>.

**Примечание.** Ответом являются полученные аналитически корректирующие коэффициенты показателей добычи игроков, указанных в платежной матрице.

**Решение задачи.** Найдем наилучшую стратегию для компании Самотлора – для этого запишем минимальное число в каждой строке: (2, 2, 3, 2). Максимальным из них является 3 – данное число является нижней ценой игры и относится к стратегии **С3**. Далее для БалтикаПромНефти выделим максимальные значения выигрыша: (4, 5, 5, 4) – минимальным будет число 4. Таким образом определена верхняя цена игры, относящаяся к стратегиям **Б1** и **Б4**.

Ввиду того, что нижняя и верхняя цены не совпадают можно сделать вывод о невозможности нахождения решения в чистых стратегиях. Цена игры находится в следующем диапазоне  $3 < v < 4$

Заметим, что игра обладает большой размерностью – матрица 4\*4. Попробуем осуществить доминирование – исключение из нее не выгодных стратегий. Их можно увидеть, сравнивая столбцы или строки между собой – в случае если все элементы какого-то столбца больше или равны, чем элементы другого столбца, то его можно исключить из рассмотрения. Аналогичная ситуация возникает при сравнении строк – удаляются строки, значения которых меньше или равны другим строкам. В результате получим матрицу:

<sup>1</sup> Доминирование – ситуация, при которой определенная стратегия одного из игроков позволяет ему получить больший выигрыш при любых действиях конкурента.

<sup>2</sup> Оптимальной называется стратегия, обеспечивающая игроку максимально возможный средний выигрыш или минимально возможный средний проигрыш при многократном повторении игры.

<sup>3</sup> Цена игры – общее значение выигрыша одного игрока и проигрыша другого в седловой точке игры.

$$\begin{pmatrix} 3 & 3 & 2 & 2 \\ 2 & 5 & 3 & 3 \\ 4 & 3 & 4 & 3 \\ 2 & 4 & 5 & 4 \end{pmatrix} \rightarrow \begin{pmatrix} 3 & 2 \\ 2 & 3 \\ 4 & 3 \\ 2 & 4 \end{pmatrix} \rightarrow \begin{pmatrix} 4 & 3 \\ 2 & 4 \end{pmatrix}$$

Произведем решение аналитическим методом. Средний выигрыш компании Саматлора, при условии (а) использования им оптимальной смешанной стратегии  $x^* = (x_1^*, x_2^*)$  и (б) использовании чистой стратегии, соответствующей первому столбцу компанией БалтикаПромНефть равен цене игры:

$$4x_1^* + 2x_2^* = v$$

Аналогично можно рассчитать средний выигрыш при использовании компаний БалтикаПромНефть чистой стратегии, соответствующей второму столбцу платежной матрицы. Учтем также, что сумма корректирующих коэффициентов равна 100%, т. е. единице:  $x_1^* + x_2^* = 1$ . Исходя из этого построим систему линейных уравнений и представим ее решение.

$$\begin{cases} 4x_1^* + 2x_2^* = v \\ 3x_1^* + 4x_2^* = v \\ x_1^* + x_2^* = 1 \end{cases} \quad \begin{cases} x_1^* = 2/3 \\ x_2^* = 1/3 \\ v = 10/3 \end{cases}$$

Определим смешанную стратегию компании БалтикаПромНефть, здесь при любой чистой стратегии компании Саматлора средний искомый проигрыш данного игрока равен цене игры:

$$\begin{cases} 4y_1^* + 3y_2^* = 10/3 \\ 2y_1^* + 4y_2^* = 10/3 \\ y_1^* + y_2^* = 1 \end{cases} \quad \begin{cases} y_1^* = 1/3 \\ y_2^* = 2/3 \end{cases}$$

Игра решена, смешанные оптимальные стратегии определены.

### Задание 7. Администрирование данных

**Текст задачи.** В процессе аутентификации оператора газовой установки каждую новую смену ему присваивается уникальный идентификационный номер, варьирующийся в диапазоне от 9 до 21 символа. В данном номере лишь последние четыре символа являются постоянными и идентифицируют самого оператора, давая возможность введения пароля для входа в систему. Остальные содержат техническую информацию, которую не следует демонстрировать. Службе технической поддержки поступила задача скрыть все символы, кроме последних четырёх при помощи символа #. Предусмотрите так же возможность сбоев в системе, в результате которых идентификатор может содержать менее четырех символов, которые не должны заменяться. В качестве тестовых значений используйте следующие:

321313nfgnfgn1311	jfowiejf923b
jvsodjv2344	k2

**Решение задачи.** Принципиальным моментом решения была его лаконичность – согласно условия наибольший балл получался в случае записи кода в функцию, тело которой содержит лишь одну строку. Для решения необходимо иметь знания в области работы с функциями, строковыми переменными и срезами. Ниже представлен пример кода на Python.

```
def maskify(cc):
    return (len(cc) - 4) * '#' + cc[-4:]
```

### Задание 8. 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 были исключены) 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.  $55 * 220 = 12100$  вариантов.

### Задание 9. Поиск простых решений сложных задач

**Текст задания.** Для заброски геологоразведочной партии необходимо предусмотреть возможность форсирования водных преград. Местность, которую необходимо пересечь изобилует мелкими реками, протоками и ручьями и как раз для таких случаев инженеры компании предложили специальные транспортные капсулы. В них встроен специальный механизм сцепки: если колонна таких транспортных средств наткнется на водное препятствие, то первая капсула останавливается, вторая цепляется к ней и

переваливается через верхнюю часть кузова, затем вторая и т. д. пока не будет образовано подобие моста, имеющего зацепы на двух берегах. Такой на первый взгляд сложный механизм был предусмотрен, чтобы достаточно легкие капсулы не уносило течением. Когда последнее транспортное средство окажется на противоположном берегу, то мост разбирается таким же способом как он был построен. Процесс будет повторяться необходимое количество раз, т. к. почти наверняка возникнет необходимость преодоления нескольких подобных препятствий. На выходе необходимо установить в каком порядке пришла колонна. Для этого разработайте функцию, которая на вход получает два строковых аргумента: последовательно выстроенную колонну капсул и карту местности. В качестве тестовых значений используйте следующие:

Колонна капсул	Местность
jpoqavp	----- .. ----- . . . . -----

Условные обозначения:

- ' - ' – твердая поверхность.
- ' . ' – водная преграда.

Пример перемещения колонны капсул по местности:

abcdefg ----- .. -----	abcdef g ----- .. -----	ab gfedc ----- .. -----
a b gfedc ----- .. -----	fedc gab ----- .. -----	cdefgab ----- .. -----

**Примечание.** Количество капсул может быть разным, но их всегда достаточно, чтобы преодолеть водные преграды. Рельеф никогда не начинается и не заканчивается с водных преград. Капсулы не могут проехать вперед иначе как по «мосту».

**Решение задачи.** Задача относится к задачам повышенной сложности и кажется довольно громоздкой, если ее решать с помощью «грубой силы», т. е. «в лоб» – описывая движение капсул. Если же сместить фокус внимания на местность – то решение может быть простым и лаконичным.

```
def ant_bridge(ants, terrain):
    n_ants = len(ants)
    terrain = terrain.replace('-.', '..')
    terrain = terrain.replace('.-', '..')
    count = terrain.count('.') % n_ants

    return ants[-count:] + ants[:-count]
```