

**Всероссийская олимпиада школьников по информатике,
профиль «Искусственный интеллект»
Вологодская область, 2025-26 учебный год
II (муниципальный) этап,
7 - 8 классы
Методические рекомендации по разбору задач**

Задача 1. Матрица ошибок

Наибольшая точность равна 1, она получается при $FP=0$.

Наименьшая точность получается при такой матрице:

$TP=16$ $FN=0$

$FP=48$ $TN=35$

При этом точность равна $TP/(TP+FP) = 16/(16+48) = 0.25$

При дополнительном условии "точность=полнота" наибольшая полнота получается при такой матрице:

$TP=51$ $FN=24$

$FP=24$ $TN=0$

При этом полнота равна $TP/(TP+FN) = 51/(51+24) = 0.68$

При дополнительном условии "точность=полнота" наименьшая полнота получается при такой матрице:

$TP=16$ $FN=24$

$FP=24$ $TN=35$

При этом полнота равна $TP/(TP+FN) = 16/(16+24) = 0.4$

Задача 2. Результаты олимпиады

Рассмотрим сначала один из способов решения без программирования. Возьмём для примера Microsoft Excel (можно использовать и аналоги из других офисных пакетов). Отсортируем данные по столбцу score по убыванию и удалим строки, где score равен нулю или пуст. Далее выделяем все четыре столбца, выбираем меню "Данные" – "Удалить дубликаты", снимаем выделение со всех столбцов, кроме userId, и нажимаем ОК. В итоге останется 203 уникальных строки с данными (возможно, Excel напишет 204, посчитав ещё и строку с заголовками). Таким образом, 203 – ответ на первый вопрос.

Закроем без сохранения и затем заново откроем наш файл. Удалим столбцы problem и score. Отсортируем данные по userId и вторым уровнем по language. Через меню "Данные" – "Удалить дубликаты" удалим дубликаты по двум столбцам сразу. Осталось найти количество таких строк, что в них значение userId уникально, а language равен "python". Для этого в ячейке C2 напишем такую формулу: " $\text{=ЕСЛИ(И(A2<>A1;A2<>A4;B2=\text{"python"})$;1;0)".

Продублируем её по всему столбцу и найдём сумму значений в этом столбце – она равна 120. Это будет ответ на второй вопрос.

Перейдём к вопросам 3 и 4. Закроем без сохранения и вновь откроем файл. Заменяем в столбце language значение «с++» на 1, а все остальные – на 0.

Отсортируем данные по userId.

Теперь сделаем так, чтобы баллы за задачу 1 были в столбце E, за вторую – в столбце F, и так далее. Для этого в ячейке E2 напишем формулу « $\text{=ЕСЛИ}(\$A2=1;\$D2;\text{""})$ » и протянем её на пять столбцов вправо. Заменяем в следующем столбце условие « $\$A2=1$ » на « $\$A2=2$ », потом на « $\$A2=3$ », и так далее до « $\$A2=5$ ». Затем протянем формулу в каждом столбце до конца вниз.

Дальше на вкладке «Данные» выберем «Промежуточный итог» и поставим такие настройки: «При каждом изменении в: userId», операция – «Максимум», «Добавить итоги по:» и отметим галочками столбец language и наши новые 5 столбцов с баллами по задачам:

	A	B	C	D	E	F	G	H	I
1	problem	userId	language	score	П1	П2	П3	П4	П5
2	1	115650513	0	68	68				
3	1	115650513	0	100	100				
4	2	115650513	0	100		100			
5	3	115650513	0	100			100		
6	4	115650513	0	60				60	
7	5	115650513	0	100					100
8	1	115720036	0	100	100				
9	3	115720036	0	0			0		
10	3	115720036	0	0			0		
11	4	115720036	0	0				0	
12	4	115720036	0	0				0	
13	4	115720036	0	0				0	
14	4	115720036	0	25				25	

Промежуточные итоги

При каждом изменении в:

userId

Операция:

Максимум

Добавить итоги по:

problem

userId

language

score

П1

П2

☒ Заменить текущие итоги

☐ Конец страницы между группами

☒ Итоги под данными

Убрать все

OK

Отмена

Теперь добавим новый столбец для суммы баллов. Для строк, в которых записаны итоговые (максимальные) значения, найдём сумму максимальных

баллов за все задачи. Для этого в ячейку J2 запишем такую формулу и протащим её по столбцу:

=ЕСЛИ(A2="";СУММ(E2:I2);"")

Должно получиться примерно так:

1	2	3		A	B	C	D	E	F	G	H	I	J
			1	problem	userId	language	score	П1	П2	П3	П4	П5	Итог
</													

Для ответа на 3-й вопрос осталось сосчитать в столбце I количество значений, которые больше или равны 200. Для этого, например, можно скопировать данные из этого столбца на новый лист (при вставке выбрать «Вставить значения»), отсортировать их, выделить подходящие и посмотреть на их количество. Ответ равен 52.

Для ответа на 4-й вопрос надо выбрать такие строки с итоговыми значениями 200 и более, что у них в столбце language стоит единица (то есть есть решение на C++). Для этого можно в новом столбце справа написать формулу:

=ЕСЛИ(И(A2="";C2=1;J2>=200);1;"")

После этого найдём сумму в этом столбце. Она равна 23 – это и есть ответ на 4-й вопрос. Excel-файл с решениями к 3 и 4 вопросам задачи прилагается.

Также данную задачу можно было решив, написав программу. Для каждого участника в словарях запоминаем максимальный балл по каждой задаче,

а также набор языков, на которых он сдавал. Затем по этим данным получаем ответы на вопросы задачи. Пример такого решения на языке Python:

```
with open('log.csv') as fp:
    fp.readline()
    points = dict()
    langs = dict()
    while True:
        try:
            line = fp.readline().strip()
        except EOFError:
            break
        if line == "":
            break
        problem, userid, lang, score = line.split(';')
        problem = int(problem.strip())
        userid = int(userid.strip())
        lang = lang.strip()
        score = score.strip()
        if score == "":
            score = "0"
        score = int(score)
        if userid not in points:
            points[userid] = [0, 0, 0, 0, 0, 0]
        points[userid][problem] = max(points[userid][problem], score)
        if userid not in langs:
            langs[userid] = set()
        langs[userid].add(lang)

    results = dict()
    ans1 = 0
    ans2 = 0
    ans3 = 0
    ans4 = 0
    for userid in points:
        results[userid] = sum(points[userid])
        if results[userid] > 0:
            ans1 += 1
        if len(langs[userid]) == 1 and "python" in langs[userid]:
            ans2 += 1
        if results[userid] >= 200:
            ans3 += 1
            if "c++" in langs[userid]:
                ans4 += 1
    print(ans1)
    print(ans2)
    print(ans3)
    print(ans4)
```

При желании при написании кода участники могли также воспользоваться библиотекой pandas, рекомендация по установке которой есть в требованиях.

Задача 3. Верный диагноз

Пусть 0 - программа ошиблась, 1 - ответила верно. У нас есть 4 варианта, на которых будет верный ответ:

011 101 110 111

Вероятности этих вариантов – $(1-p_1) \cdot p_2 \cdot p_3$, $p_1 \cdot (1-p_2) \cdot p_3$, $p_1 \cdot p_2 \cdot (1-p_3)$, $p_1 \cdot p_2 \cdot p_3$.

Вероятность, что возник хотя бы один из них, равен их сумме (так как по условию события независимые). Пример решения на языке Python:

```
p1 = float(input())
p2 = float(input())
p3 = float(input())
print((1-p1)*p2*p3 + p1*(1-p2)*p3 + p1*p2*(1-p3) + p1*p2*p3)
```

Задача 4. Меры центральной тенденции

В первой подзадаче можно выполнить простой перебор вариантов (например, четырьмя вложенными циклами).

Рассмотрим теперь полное решение. Выделим три варианта.

1). $x=y$. Пусть есть не менее трёх чисел, равных x . После сортировки у нас может получиться так: $d \ x \ x \ x \ e$. Возьмём $e = 10^9$, тогда $d = 5z-3x-e$, и такое d будет гарантированно меньше x .

2). $x < y$. После сортировки получится: $x \ x \ y \ d \ e$. Тогда возьмём $d = y+1$, $e = 5z-2x-y-d$. Если $e \leq d$, то этот вариант не подходит (увеличивать d бессмысленно, так как тогда e станет лишь меньше). Тогда решений нет. Например, решений нет при $x=1$, $y=3$, $z=2$.

3). $x > y$. После сортировки получится $s \ d \ y \ x \ x$. Тогда возьмём $d = y-1$, $s = 5z-2x-y-d$. Если $s \geq d$, то этот вариант не подходит (уменьшать d бессмысленно, так как тогда s станет лишь больше). Тогда решений нет. Например, решений нет при $x=3$, $y=1$, $z=2$.

Пример решения на языке Python:

```
x = int(input())
y = int(input())
z = int(input())
if x == y:
    print(5*z-3*x-1000000000, x, x, x, 1000000000)
elif x < y:
    d = y + 1
    e = 5*z-2*x-y-d
    if e <= d:
```

```

        print(-1)
    else:
        print(x, x, y, d, e)
else:
    d = y - 1
    c = 5*z-2*x-y-d
    if c >= d:
        print(-1)
    else:
        print(c, d, y, x, x)

```

Задача 5. Ближайшие соседи

Подзадачу 1 можно написать “в лоб” со сложностью в районе $O(n^4)$ – перебираем k , вложенным циклом перебираем точки, ещё двумя вложенными циклами ищем k ближайших.

В подзадаче 2 для поиска k ближайших точек к заданной можно применить встроенную сортировку – это даст сложность $O(n^3 \log(n))$.

В подзадаче 3 внешним циклом по i будем перебирать точку, для которой будем прогнозировать цвет, а внутренним циклом будем перебирать k . Тогда сортировку для каждого i будет достаточно делать только один раз. Ответы будем накапливать в массиве ans. Приведём пример такого решения на языке Python:

```

n = int(input())
pt = []
for _ in range(n):
    pt.append([int(t) for t in input().split()])
ans = [0] * n
for i in range(n):
    p = pt.copy()
    p.sort(key = lambda q : (q[0]-pt[i][0])**2 + (q[1]-pt[i][1])**2)
    color = [0, 0]
    for k in range(1, n):
        color[p[k][2]] += 1
        if k % 2 == 1:
            curcolor = 0 if color[0] > color[1] else 1
            if curcolor == pt[i][2]:
                ans[k] += 1
print(*ans[1::2], sep='\n')

```