

Заключительный этап 7 и 8 класса (приведен один из вариантов заданий)

1. Системы счисления. 1 балл.

[Система неравенств]

Известно, что некоторое целое положительное число f может быть получено в результате вычисления следующего выражения:

$$f = 2000_N - 100_N, \text{ где } N - \text{неизвестное основание позиционной системы счисления.}$$

Дана система неравенств:

$$\begin{cases} 258_{16} \leq f < 7D0_{16} \\ 3E8_{16} \geq f > C8_{16} \end{cases}$$

Определите, для каких N эта система неравенств будет верна.

В ответе запишите через пробел все подходящие основания систем счисления N в порядке возрастания.

Ответ: 7 8

Решение:

Сначала объединим условия системы неравенств:

1. $7D0_{16} > 3E8_{16}$, следовательно $f \leq 3E8_{16}$
2. $C8_{16} < 258_{16}$, следовательно $258_{16} \leq f$

3. Получаем неравенство $258_{16} \leq f \leq 3E8_{16}$

Переведем шестнадцатеричные числа в десятичную систему счисления:

$$258_{16} = 2 * 16^2 + 5 * 16 + 8 = 512 + 80 + 8 = 600$$

$$3E8_{16} = 3 * 16^2 + 14 * 16 + 8 = 768 + 224 + 8 = 1000$$

Теперь, используя формулу развернутой записи числа для f , получаем:
 $f = 2 * N^3 - N^2$

Далее подберем такие целые положительные N , для которых выполняется неравенство $258_{16} \leq 2 * N^3 - N^2 \leq 3E8_{16}$. Для данного неравенства подходят только два целых значения N : 7 и 8.

Можно предложить также программное решение:

```
# Переберем все основания систем счисления до достаточно большого числа
for n in range(2, 37):
    # Запишем число f в многочленной форме
    f = 2 * n ** 3 - n ** 2
    # Проведем проверку из условия задачи
    if int('258', 16) <= f < int('7d0', 16) and int('c8', 16) < f <= int('3e8', 16):
        print(n)
```

2. Кодирование информации. 1 балл.

[Новый шифр]

Петя придумал новый шифр для текстов, записанных с использованием 32-х символов русского алфавита (без буквы ё):

1. Для шифрования используется таблица из черных и белых квадратов размером 4*8, например, такая:

■	□	□	□	□	□	□	□
□	■	□	□	□	□	□	□
□	□	■	□	□	□	□	□
□	□	□	■	□	□	□	□

2. Каждой букве алфавита сопоставляется число от 1 до 32. Так же каждой клетке таблицы сопоставляется число от 1 до 32 (нумерация идёт слева-направо сверху-вниз). Например, черная клетка во второй строке будет иметь номер 10.

3. Строится новый алфавит для шифрования следующим образом. Просматриваем ячейки таблицы в порядке возрастания их номеров. Если встречается ячейка черного цвета, то буква, соответствующая порядковому номеру ячейки, перемещается в новом алфавите со своей позиции в конец.

4. В результате получается новый алфавит, который записывается под исходным алфавитом. Для приведенного примера таблицы получится следующий результат:

а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
б	в	г	д	е	ж	з	и	к	л	м	н	о	п	р	с	у	ф	х	ц	ч	ш	щ	ъ	ь	э	ю	я	а	й	т	ы

Для шифрования каждая буквы исходного текста меняется на букву, находящуюся под ней.

Тогда, используя таблицу из примера можно зашифровать фразу «шифр удался», и зашифрованный текст будет выглядеть следующим образом: «ькчу цебнфы».

Петя использовал таблицу, приведенную ниже, чтобы зашифровать некоторое слово:

□	■	□	□	■	□	■	□
■	■	□	□	□	■	□	□
□	□	■	■	□	□	■	■
□	■	□	□	■	□	■	□

У него получилось слово «ваывазшэ». Каким было исходное слово до шифрования?

Ответ: барбадос

Решение:

Исходя из таблицы, нужно перемещать в конец символы со следующими номерами: 2, 5, 7, 9, 10, 14, 19, 20, 23, 24, 26, 29 и 31.

Составив алфавит для шифрования, перемещая соответствующие символы, получим:

а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
а	в	г	е	з	к	л	м	о	п	р	с	ф	х	ш	ъ	ы	э	я	б	д	ж	и	й	н	т	у	ц	ч	щ	ь	ю

Поскольку для шифрования каждая буква слова заменялась с верхней строчки на букву под ней, то для расшифровки нам нужно брать букву из нижней строчки и заменять её на букву над ней.

Получаем из «ваывазшэ» слово «барбадос», это и есть ответ на задачу.

Можно предложить также программное решение:

```
# Введем эталонный русский алфавит без буквы ё
```

```
alph = 'абвгдежзийклмнопрстуфхцчшщъыьэюя'
```

```
# Организуем два списка: tmp_start - для букв алфавита, которые не перемещаются в конец
```

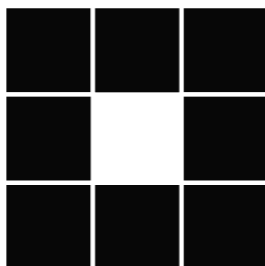
```

# tmp_end - для букв алфавита перемещенных в конец
tmp_start = []
tmp_end = []
# Закодируем картинку из условия в виде списка нулей (белых клеток) и единиц (четных
клеток)
code = [0, 1, 0, 0, 1, 0, 1, 0,
        1, 1, 0, 0, 0, 1, 0, 0,
        0, 0, 1, 1, 0, 0, 1, 1,
        0, 1, 0, 0, 1, 0, 1, 0]
# Смоделируем создание алфавита для кодирования, описанное в условии
tmp = []
for i in range(32):
    if code[i] == 1:
        tmp_end.append(alph[i])
    else:
        tmp_start.append(alph[i])
alph_new = ''.join(tmp_start + tmp_end)
# Слово, которое нужно раскодировать
word = 'ваывазшэ'
# В новом алфавите находим место (индекс) из word и берем букву, стоящую на этой позиции
в старом алфавите
res = []
for x in word:
    res.append(alph[alph_new.find(x)])
# Выводим результат
print(''.join(res))

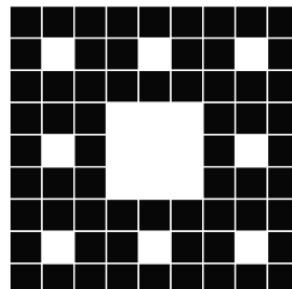
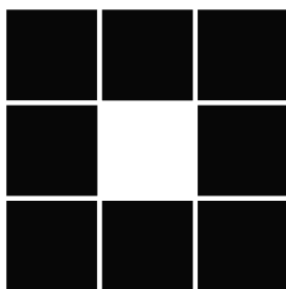
```

3. Кодирование графической информации. 2 балла. [Ковёр Серпинского]

Петя и Вася узнали новый для себя фрактал – ковёр Серпинского, это фигура, которая меняется на каждом шаге. За один шаг все черные квадраты заменяются на 8 черных квадратов и один белый квадрат меньшего размера, расположенные следующим образом:



На картинке изображены нулевой, первый и второй шаги построения ковра Серпинского.



Сколько будет квадратов самого маленького размера (суммарно и черных, и белых) на 5-ом шаге? Например, на втором шаге 72 самых маленьких квадрата (64 черных и 8 белых). В ответе укажите целое число.

Ответ: 36864

Решение:

Согласно условию изменения фигуры «Ковёр Серпинского», каждый черный квадрат заменяется на 8 черных и 1 белый. На нулевом шаге мы имеем один черный квадрат.

На каждом следующем шаге самых маленьких квадратов будет:

Белых: $w = b'$

Черных: $b = b' * 8$

Суммарно: $w + b = b' * 9$, где

w – количество самых маленьких белых квадратов на данном шаге,
 b – количество самых маленьких черных квадратов на данном шаге,
 b' – количество черных квадратов на предыдущем шаге.

Количество черных квадратов на каждом шаге легко узнать. Поскольку каждый черный квадрат заменяется на 8 черных квадратов меньшего размера, то: на шаге 0 – 1 черный квадрат,

на шаге 1 – 8 черных квадратов,
на шаге 2 – 64 черных квадрата,
...
на шаге n – 8^n черных квадратов.

Таким образом, найдём формулу для нахождения общего числа самых маленьких квадратов на шаге n:

$$ANS(n) = 8^{n-1} * 9$$

Подставим номер шага из задачи

$$ANS(5) = 8^{5-1} * 9 = 8^4 * 9 = 36864, \text{ это и есть ответ на задачу.}$$

4. Алгебра логики. 3 балла.

[Полином Жегалкина]

Петя узнал о новом представлении логических функций – полином Жегалкина. В этом представлении функция задается выражением, в котором используются не привычные булевы операции (НЕ, И, ИЛИ), а операции «И($\wedge$)» и «Исключающее ИЛИ($\oplus$)» и константа «истина (1)».

Для двух указанных операций представлены таблицы истинности ниже:

И($\wedge$)			Исключающее ИЛИ($\oplus$)		
A	B	$A \wedge B$	A	B	$A \oplus B$
0	0	0	0	0	0
0	1	0	0	1	1
1	0	0	1	0	1
1	1	1	1	1	0

Примечание: В полиноме Жегалкина «Исключающее ИЛИ($\oplus$)» имеет наименьший приоритет.

Пример: Дана функция в полиноме Жегалкина: $1 \oplus A \oplus A \wedge B$.

Таблица истинности для данной функции:

A	B	$A \wedge B$	$1 \oplus A$	$1 \oplus A \oplus A \wedge B$
0	0	0	1	1
0	1	0	1	1
1	0	0	0	0
1	1	1	0	1

Как видно, при построении таблицы истинности функции сначала построили столбец для операции «И», как имеющей наивысший приоритет ($A \wedge B$). Затем для первой операции «Исключающее ИЛИ» ($1 \oplus A$) и затем для операции «Исключающее ИЛИ» между результатами в построенных перед этим двух столбцах.

Для данной функции будет три набора аргументов с истинным значением функции.

Вам даны несколько функций, представленных в полиноме Жегалкина:

- $C \oplus B \oplus B \wedge C \oplus A \oplus A \wedge B \wedge C$
- $1 \oplus C \oplus B \oplus A \oplus A \wedge C \oplus A \wedge B$
- $1 \oplus C \oplus B \oplus A$
- $1 \oplus C \oplus A \wedge B \wedge C$

Помогите Пете найти функции, для каждой из которых будет не менее 5 различных наборов аргументов, при которых функция даёт истину.

В ответе укажите номера подходящих функций через пробел в порядке возрастания.

Ответ: 1 4

Решение:

Построим для каждой функции таблицу истинности и узнаем количество наборов аргументов, при которых функция даёт истину.

При построении можно не расписывать столбцы для каждой операции «исключающее ИЛИ», поскольку данная операция является также «сложением по модулю 2», иначе говоря, «исключающее ИЛИ» для нескольких переменных даёт истину тогда и только тогда, когда количество слагаемых, равных истине – нечетно.

Так же для операции «исключающее ИЛИ» не важен порядок, а значит слагаемые можно менять местами для удобства.

- $C \oplus B \oplus B \wedge C \oplus A \oplus A \wedge B \wedge C$

A	B	C	$B \wedge C$	$A \wedge B \wedge C$	$A \oplus B \oplus C \oplus B \wedge C \oplus A \wedge B \wedge C$
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	0	1
0	1	1	1	0	1
1	0	0	0	0	1
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	1	1	1

Для данной функции 5 наборов аргументов дают истину.

2. $1 \oplus C \oplus B \oplus A \oplus A \wedge C \oplus A \wedge B$

A	B	C	$A \wedge B$	$A \wedge C$	$1 \oplus A \oplus B \oplus C \oplus A \wedge B \oplus A \wedge C$
0	0	0	0	0	1
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	1
1	0	0	0	0	0
1	0	1	0	1	0
1	1	0	1	0	0
1	1	1	1	1	0

Для данной функции 2 набора аргументов дают истину.

3. $1 \oplus C \oplus B \oplus A$

A	B	C	$1 \oplus A \oplus B \oplus C$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Для данной функции 4 набора аргументов дают истину.

4. $1 \oplus C \oplus A \wedge B \wedge C$

A	B	C	$A \wedge B \wedge C$	$1 \oplus C \oplus A \wedge B \wedge C$
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	0	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	1
1	1	1	1	1

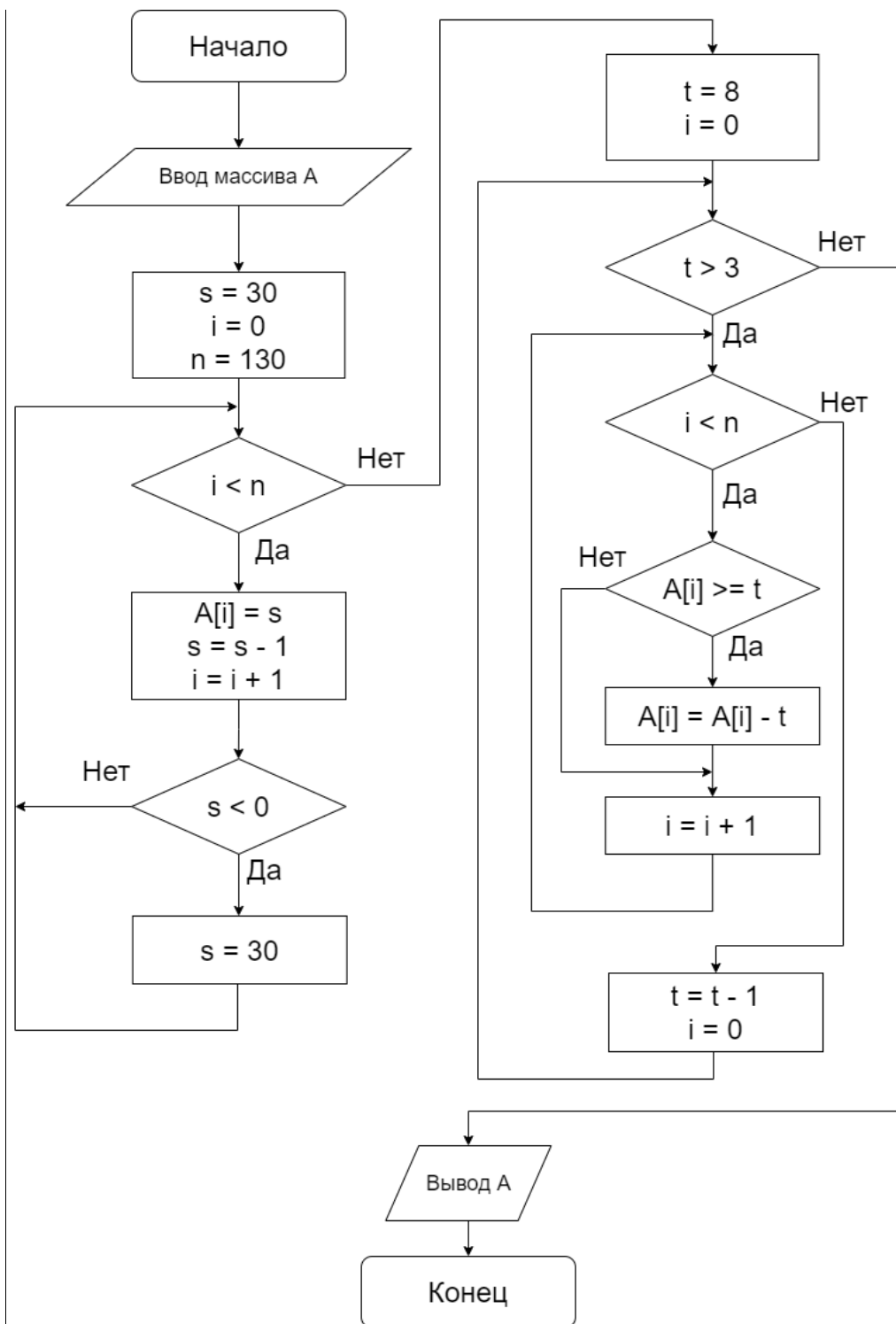
Для данной функции 5 наборов аргументов дают истину.

Таким образом, только функции 1 и 4 имеют не менее 5 наборов аргументов, при которых функция имеет истинное значение.

5. Анализ алгоритма, заданного в виде блок-схемы. 3 балла.

[Обработка массива]

Вам дана блок-схема алгоритма:



Известно, что на вход был подан массив А из 130 элементов, равных 0. Нумерация элементов массива начинается с нуля. Укажите, сколько нулей в массиве А будет после окончания работы описанного выше алгоритма?
В ответе укажите одно число – искомое количество нулей.

Ответ: 67

Решение:

Проанализируем левую часть блок-схемы. В данной части алгоритма массив А заполняется числами, посмотрим несколько чисел массива после выполнения этой части алгоритма:

30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

Очевидно, что дальше последовательность будет содержать повторения приведенного фрагмента. Причем, поскольку всего чисел в последовательности должно быть 130, то повторений приведенного фрагмента будет 4 и еще шесть первых чисел из пятого повторения приведенного фрагмента:

30 29 28 27 26 25

Теперь приступим к правой части алгоритма. В ней берется некое число t, меняющееся от 8 до 4 включительно. Для каждого t алгоритм проходит по всему массиву и вычитает из каждого числа t, если $t \leq A[i]$. Составим таблицу для каждого t для первого фрагмента из 31 числа:

Фрагмент (первые 31 элемент)	t	Количество нулей
30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0	8	2
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 6 5 4 3 2 1 0 0 6 5 4 3 2 1 0	7	4
9 8 7 6 5 4 3 2 1 0 5 4 3 2 1 0 0 5 4 3 2 1 0 0 0 5 4 3 2 1 0	6	7
4 3 2 1 0 4 3 2 1 0 0 4 3 2 1 0 0 0 4 3 2 1 0 0 0 0 4 3 2 1 0	5	11
0 3 2 1 0 0 3 2 1 0 0 0 3 2 1 0 0 0 0 3 2 1 0 0 0 0 0 3 2 1 0	4	16

Следовательно, во всех остальных фрагментах произойдёт то же самое, поскольку они одинаковы.

Вычислим общее число нулей в последовательности по завершении выполнения алгоритма (необходимо помнить, что в последовательности 4 полных фрагмента и еще 6 чисел):

$$16 * 4 + 3 = 67$$

Это и есть ответ на задачу.

Можно предложить также программное решение:

Запишем блок-схему в виде алгоритма на Python

```
a = [0] * 130
```

```
s = 30
```

```
n = 130
```

```
for i in range(n):
```

```
    a[i] = s
```

```
    s = s - 1
```

```
    if s < 0:
```

```
        s = 30
```

```
t = 8
```

```
while t > 3:
```

```
    for i in range(n):
```

```
        if a[i] >= t:
```

```
            a[i] -= t
```

```
        t -= 1
```

```
# Найдем количество нулей в списке
```

```
print(a.count(0))
```

6. Анализ алгоритма, заданного в виде программного кода. 2 балла

[Три символа]

Петя придумал алгоритм, который рисует картинки с использованием трех символов: *, + и #.

алг

нач цел x, y

 x := 0

 нц пока x < 10

 y := 0

 нц пока y < 10

 если F(x, y) >= 0 и G(x, y) >= 0

 то

 Вывод '*'

 все

 если G(x, y) < 0

 то

 Вывод '+'

 все

 если F(x, y) < 0 и G(x, y) >= 0

 то

 Вывод '#'

 все

 y := y + 1

 кц

 Вывод '\n'

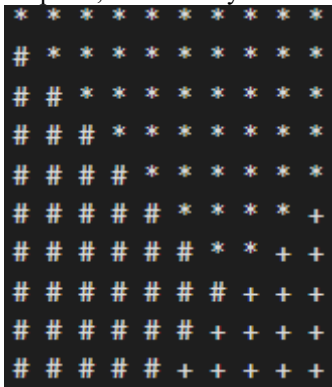
 x := x + 1

кц

кон

Во время исполнения алгоритма происходит вычисление значений двух функций: F(x,y) и G(x,y). Известно, что каждая из них представляет собой выражение вида (a*x + b*y + c), где a, b и c могут быть различными числами.

Петя дал Васе список выражений, которые можно подставить вместо $F(x, y)$ и $G(x, y)$, помогите Васе выбрать два выражения, которые необходимо подставить в алгоритм, чтобы получить следующую картинку:



Список выражений, который Петя дал Васе.

1. $-10 * x + 10 * y + 0$
2. $-4 * x + 5 * y + 0$
3. $-7 * x - 7 * y + 91$
4. $-6 * x + 6 * y + 20$
5. $x + y - 10$

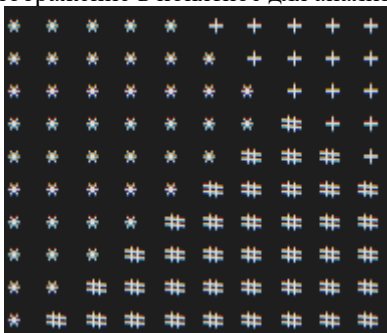
В ответе укажите два числа - сначала номер выражения, которое скрывается за $F(x, y)$, а потом за $G(x, y)$ без запятых и пробелов.

Примечание: Вывод '\n' – вывод символа переноса строки.

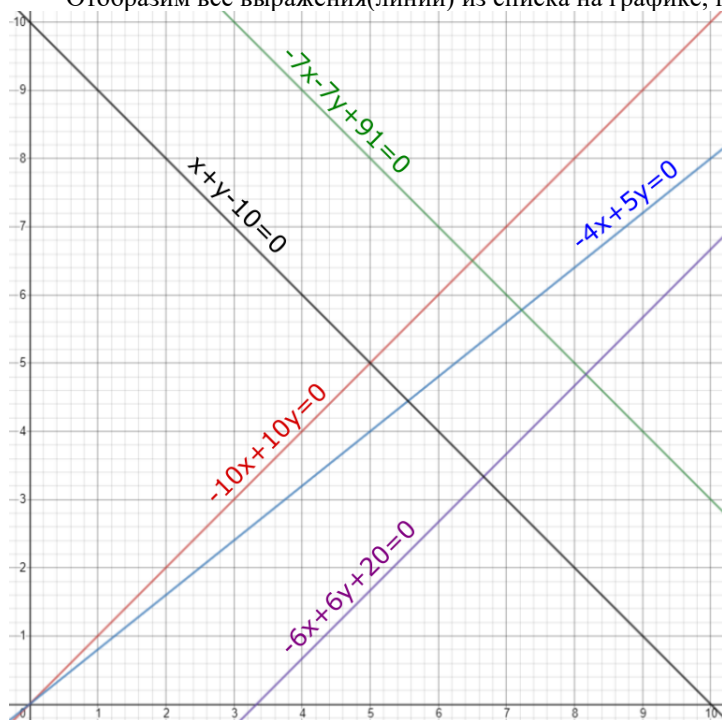
Ответ: 13

Решение:

Несложно заметить, что получившая картинка – ни что иное, как 1 четверть декартовой системы координат, повернутая на 90 градусов по часовой стрелке, а выражения $F(x, y)$ и $G(x, y)$ задают некоторые прямые в этой системе координат, разделяющие плоскость на 4 части. Вернем изображение в понятное для анализа положение:



Отобразим все выражения(линии) из списка на графике, получаем:



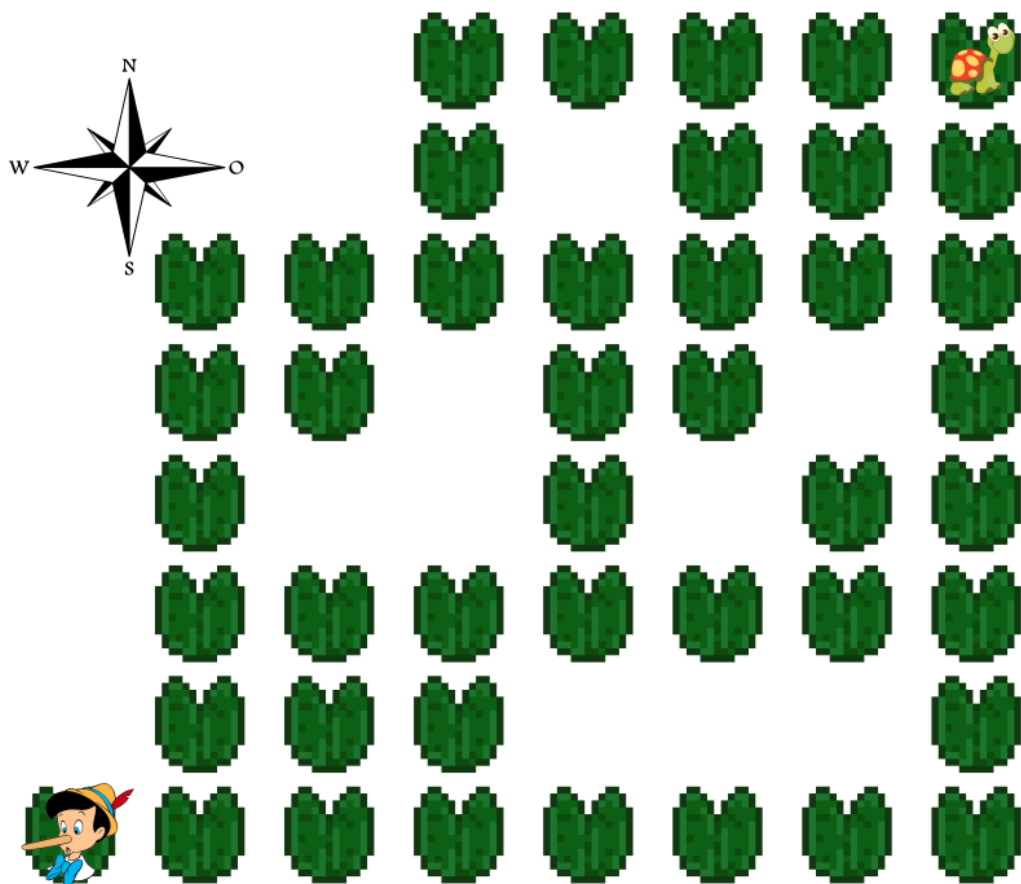
Далее попробуем сопоставить каждой целой координатной точке символ из рисунка, получаем:


```
# Сравниваем полученную строку с эталоном
if t == etalon:
    print(f, g)
```

7. Информационное моделирование. 2 балла.

[И снова Тортилла]

Когда Буратино пришел к мудрой черепахе Тортилле, она отправилась по кувшинкам за Золотым Ключиком. Но когда собралась идти обратно, заметила, что те кувшинки, по которым она шла сюда, переместились и обратный путь теперь совсем другой и очень витиеватый. Тортилла начинает движение в правом верхнем углу и может идти только на юг и на запад. Помогите Тортилле узнать, сколько у неё вариантов составить маршрут до нижней левой кувшинки. В ответе укажите целое число.



Ответ: 99

Решение:

Чтобы решить данную задачу нужно посчитать количество вариантов дойти от старта до каждой кувшинки как сумму количества вариантов дойти до кувшинки сверху и кувшинки справа от текущей кувшинки (если такие имеются). До первой кувшинки считаем, что есть только 1 вариант дойти. Далее будем считать ответ для каждой кувшинки, складывая количество вариантов дойти до тех кувшинок, из которых в неё можно попасть:

			1	1	1	1	1
			1		3	2	1
7	7	7	6	6	3	1	
14	7		12	6			1
14			12		1	1	
28	14	14	14	2	2	1	
56	28	14					1
99	99	43	15	1	1	1	1

В левом нижнем углу и есть ответ на поставленную задачу.

8. Поиск и фильтрация данных. 1 балл.

[База студентов]

Петя и Вася получили доступ к базе данных, содержащей результаты опроса студентов Университета. В базе данных для каждого студента хранятся следующие данные: пол студента (мужской, женский), выбранный формат обучения (очное, дистанционное), а также его уровень образования (бакалавриат, специалитет или магистратура). По каждому из указанных трех параметров у каждого студента выбрано ровно одно значение.

Известно количество записей, полученных в ответ на ряд запросов к этой базе:

1. Пол студента = «женский» и формат обучения = («очное» или «дистанционное») и уровень образования = «специалитет» - 14 записей
2. Пол студента = «женский» и формат обучения = «дистанционное» и уровень образования = «специалитет» - 0 записей
3. Пол студента = «мужской» и формат обучения = «дистанционное» и уровень образования = («бакалавриат» или «специалитет» или «магистратура») – 42 записи
4. Пол студента = «мужской» и формат обучения = («очное» или «дистанционное») и уровень образования = («бакалавриат» или «специалитет» или «магистратура») – 74 записи
5. Пол студента = («мужской» или «женский») и формат обучения = «очное» и уровень образования = «специалитет» - 21 записей
6. Пол студента = («мужской» или «женский») и формат обучения = «очное» и уровень образования = («бакалавриат» или «магистратура») – 66 записей
7. Пол студента = «женский» и формат обучения = («очное» или «дистанционное») и уровень образования = («бакалавриат» или «магистратура») – 80 записей

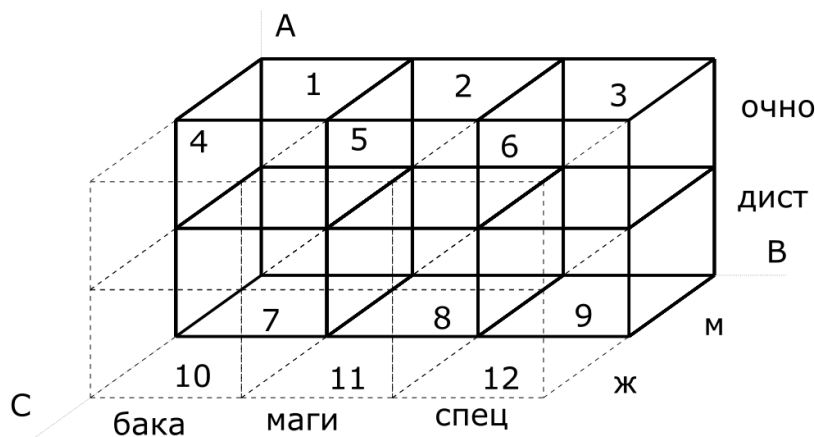
Определите, сколько студентов выбрали дистанционный формат обучения?

В ответе укажите одно число.

Ответ: 81

Решение:

Как видно из запросов, в них участвуют значения трех полей, следовательно, все записи можно представить как области пространства значений полей с заданной на нем декартовой системой координат, где в качестве осей будут выступать поля: «Пол студента», «Формат обучения» и «Уровень образования». Как видно из условия, поля «Пол студента» и «Формат обучения» имеют два варианта значения, а «Уровень образования» три варианта значений. Следовательно, любая запись в базе данных будет принадлежать одной из 12 областей:



Тогда первому запросу соответствуют области 6,12

Второму запросу соответствует область 12

Третьему запросу соответствуют области 7,8,9

Четвертому запросу соответствуют области 1,2,3,7,8,9

Пятому запросу соответствует область 3,6

Шестому запросу соответствуют области 1,2,4,5

Седьмому запросу соответствуют области 4,5,10,11

Для определения количества студентов на дистанционном формате обучения необходимо найти, сколько запросов соответствуют областям 7,8,9,10,11,12.

- 1) Из первых двух запросов легко найти, что в области 12 – 0 записей, а в области 6 – 14 записей
- 2) Из первого пункта и пятого запроса видно, что в области 3 – 7 записей
- 3) Из второго пункта и третьего запроса видно, что суммарно в областях 1,2 – 25 записей
- 4) Из третьего пункта и шестого запроса видно, что суммарно в областях 4,5 – 41 запись
- 5) Из четвертого пункта и седьмого запроса видно, что суммарно в областях 10,11 – 39 записей
- 6) Из первого и пятого пунктов, а также третьего запроса видно, что в областях 7,8,9,10,11,12 суммарно $42+39+0=81$ запись

Это и будет правильным ответом.

9. Обработка информации в электронных таблицах. 2 балла

[Запутанные формулы]

Петя показал Васе фрагмент таблицы:

	A	B	C
1	=B1*B2	7	=A2*B4 / 11
2	=B2+10	12	=B3*B5
3	=C3	10	=A1+C2-8
4	=-(C4 + B4*(B3-B1)*(B3-B1))		=C1+B4*(B2-B1)
5	=C5*1/(B5-B1)	15	=B4*B4*(B2-B3)
6			
7		=СУММ(A3:A5)*СУММ(A3:A5)	
8			

Петя также задал Васе вопрос, какое целое положительное число должно быть в ячейке B4, чтобы после выключения режима отображения формул в ячейке B7 оказалось число **900**?

Помогите Васе найти ответ на этот вопрос. В ответе укажите одно число – искомый ответ на задачу.

Ответ: 32

Решение:

Поскольку в ячейке B7 формула $(A3+A4+A5)^2$ после выключения режима отображения формул оказалась равна 900, тогда $A3+A4+A5 = \pm 30$.

Следующим шагом найдём несколько значений ячеек, которые уже известны:

	A	B	C
1	=7*12=84	7	=A2*B4/11
2	=12+10=22	12	=10*15=150
3	=226	10	=84+150-8=226
4	=(C4+B4*(B3-B1)*(B3-B1)) x		=C1+B4*(B2-B1)
5	=C5*1/(B5-B1)	15	=B4*B4*(B2-B3)

В оставшихся формулах заменим ячейку B4 на x, а остальные известные ячейки на числа и уже посчитанные формулы с x:

	A	B	C
1	84	7	=22*x/11=2*x
2	22	12	150
3	226	10	226
4	=(7*x+x*(10-7)*(10-7))=-7*x-9*x=-16*x x		=2*x+x*(12-7)=2*x+x*5=7*x
5	=2*x*x*1/(15-7)=2*x*x/8=x*x/4	15	=x*x*(12-10)=2*x*x

Из формулы $A3+A4+A5 = \pm 30$ и известных формул в ячейках A3, A4 и A5 получим равенство:

$$\frac{x^2}{4} - 16 * x + 226 = \pm 30$$

Теперь есть два пути решения.

Первый, аналитический – решив два квадратных уравнения (для 30 и -30 в правой части), получить три возможных значения x: $32+4*\sqrt{15}$; $32-4*\sqrt{15}$; 32

Из данных ответов нам подходит только 32, поскольку по условию требуется целое положительное число.

Второй путь – перебрать значения в ячейке B4 вручную или используя макрос, например, такой:

```
f = 0
i = 1
While f = 0
    Cells(4, 2) = i
    If Cells(7, 2) = 900 Then
        Cells(1, 5) = i
        f = 1
    End If
    i = i + 1
Wend
```

10. Информационное моделирование. 1 балл

[В путь за диаграммами]

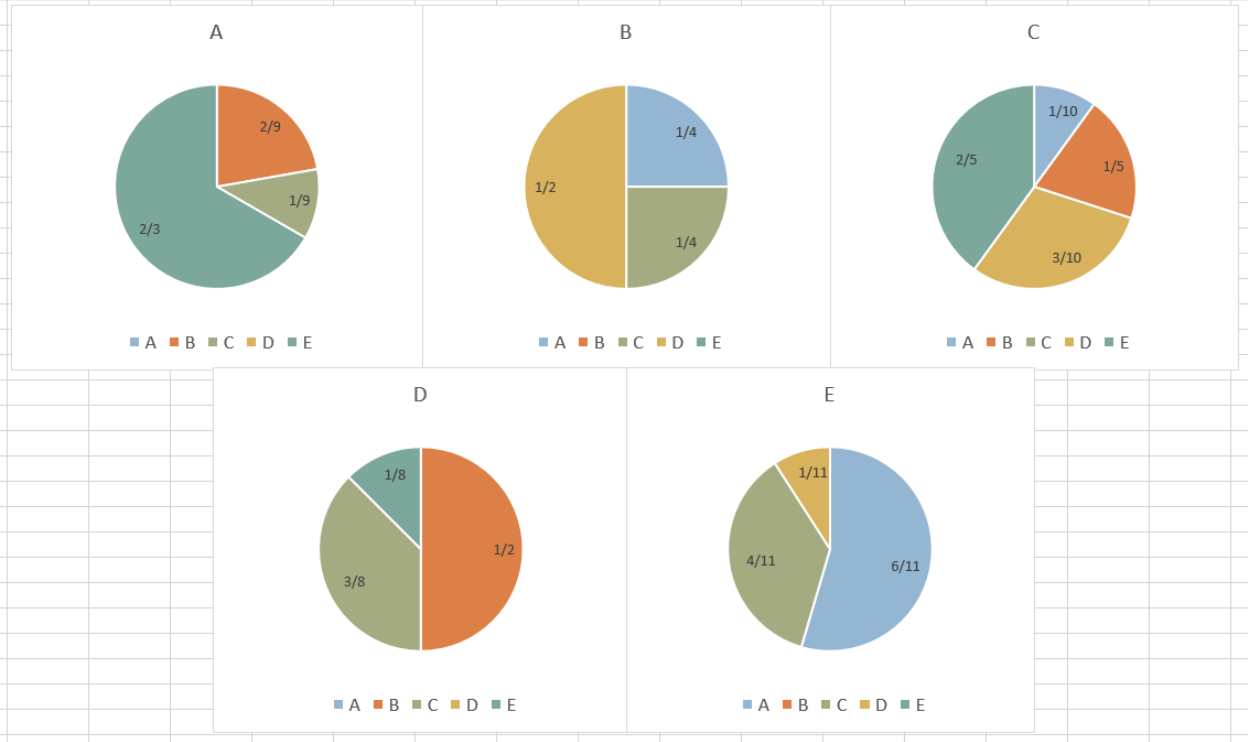
Петя любит изучать дорожные карты. Он рассматривает дорожную сеть, соединяющую 5 населенных пунктов (A, B, C, D, E) с указанием длин дорог, соединяющих два населенных пункта. Петя решил описать эту сеть в виде таблицы:

	A	B	C	D	E
A					
B					

C					
D					
E					

В такой таблице на пересечении некоторой строки и столбца записывается длина пути, если дорога между двумя населенными пунктами есть, в противном случае – пусто. Если есть дорога, соединяющая некоторый пункт X с пунктом Y, то есть и дорога, соединяющая пункт Y с пунктом X, той же длины. Ни один пункт не соединен дорогой с самим собой (главная диагональ таблицы всегда пустая), то есть нет дороги из A в A, из B в B и т.д.

Вася очень любит круговые диаграммы и знает, что по этой таблице можно составить 5 круговых диаграмм (по одной для каждой строки). Вася решил узнать, как хорошо Петя разобрался в дорожной карте и показал Пете получившиеся диаграммы:



A также Вася подсказал Пете, что дорога, соединяющая пункты A и B, имеет длину 14. Вася спросил Петю, какая минимальная длина пути между населенными пунктами A и E?

Помогите Пете, в ответе укажите одно целое число – найденную длину кратчайшего пути.

Ответ: 35
Решение:

Заметим, что раз для каждой пары пунктов X и Y есть ровно одна дорога в обе стороны, то таблица будет симметрична относительно главной диагонали (строка A равна строке-столбцу A, строка B равна строке-столбцу B и т.д.).

Нам известна из условия длина дороги между пунктами A и B, занесём её в таблицу:

	A	B	C	D	E
A		14			
B	14				
C					
D					
E					

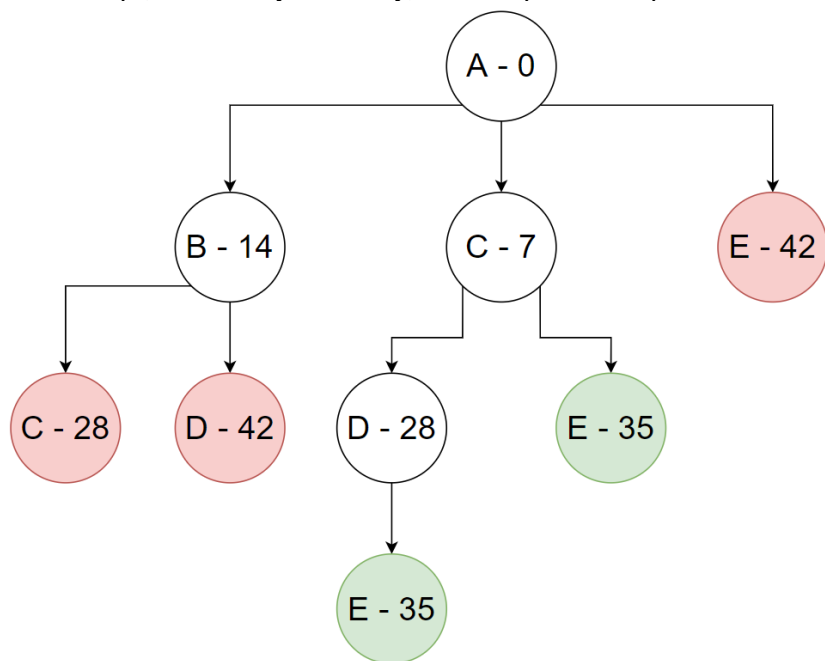
Далее, зная хотя бы одну длину дороги в строке, можно по диаграмме узнать протяженность остальных, составляя пропорции. Заполним строки и столбцы A и B:

	A	B	C	D	E
A		14	7		42
B	14		14	28	
C	7	14			
D		28			
E	42				

Теперь осталось только заполнить строки и столбцы C и D:

	A	B	C	D	E
A		14	7		42
B	14		14	28	
C	7	14		21	28
D		28	21		7
E	42		28	7	

Теперь, имея полную таблицу, найдём кратчайшее расстояние от А до Е:



Пояснение: красным выделены неэффективные маршруты. Маршрут от А до С длиной 28 менее выгоден, чем маршрут длины 7, поэтому мы его не рассматриваем. Маршрут от А до D длиной 42 не выгоден, так как прямой маршрут от А до Е имеет ту же длину. Ну и наконец прямой маршрут от А до Е длиной 42 менее выгоден, чем найденные два маршрута длиной 35.

Это и есть ответ на поставленную задачу – 35.