

Итоги и анализ результатов ЕГЭ по информатике и ИКТ

БЕЛЯНЧЕВА СВЕТЛАНА ЮРЬЕВНА,
СТАРШИЙ МЕТОДИСТ ЦЕНТРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
ГАУ ДПО ЯО ИРО

Содержательные разделы курса информатики и ИКТ

2022, 2021

Информация и её кодирование

Моделирование и компьютерный эксперимент

Системы счисления

Логика и алгоритмы

Элементы теории алгоритмов

Программирование

Архитектура компьютеров и компьютерных сетей

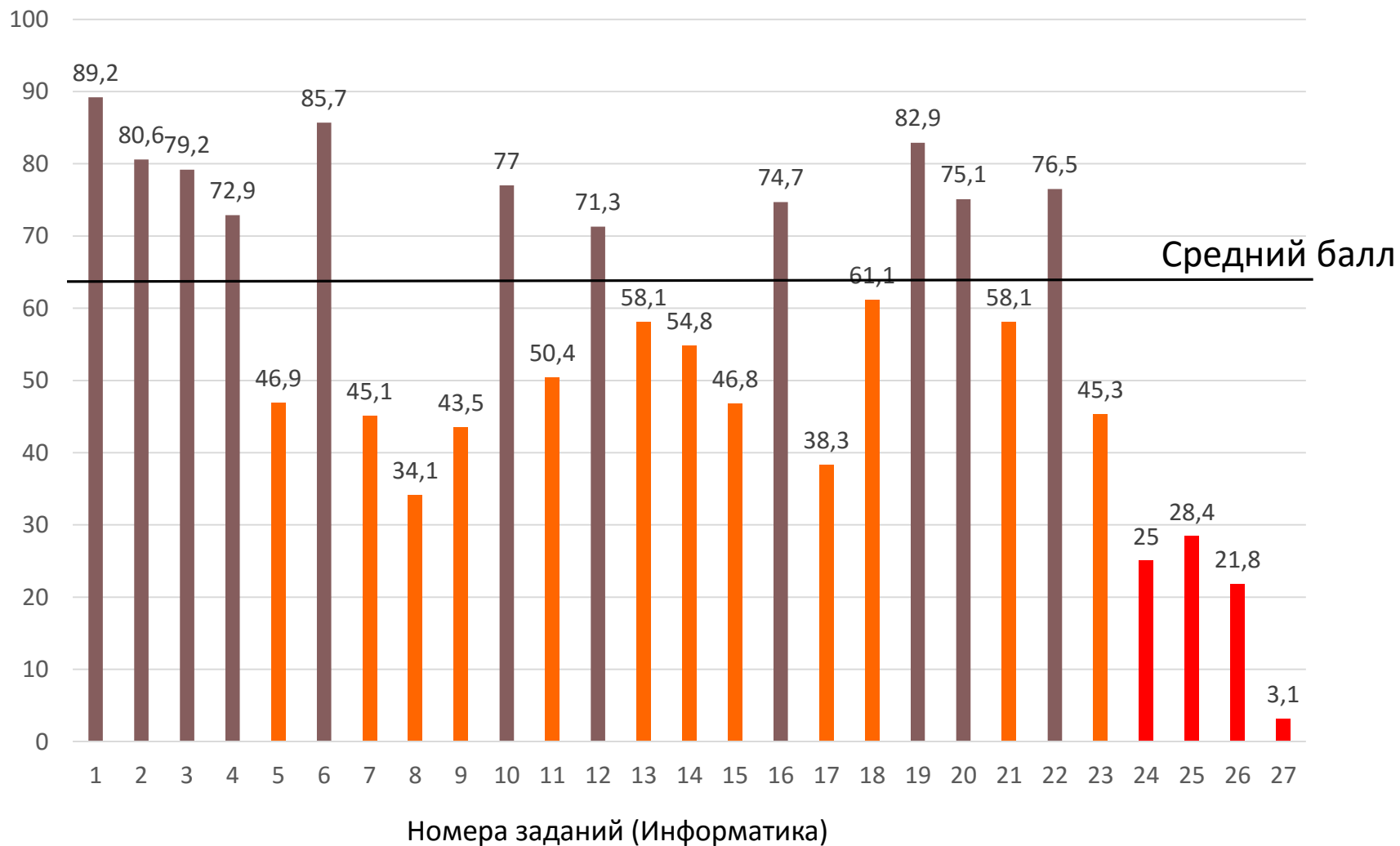
Обработка числовой информации

Технологии поиска и хранения информации

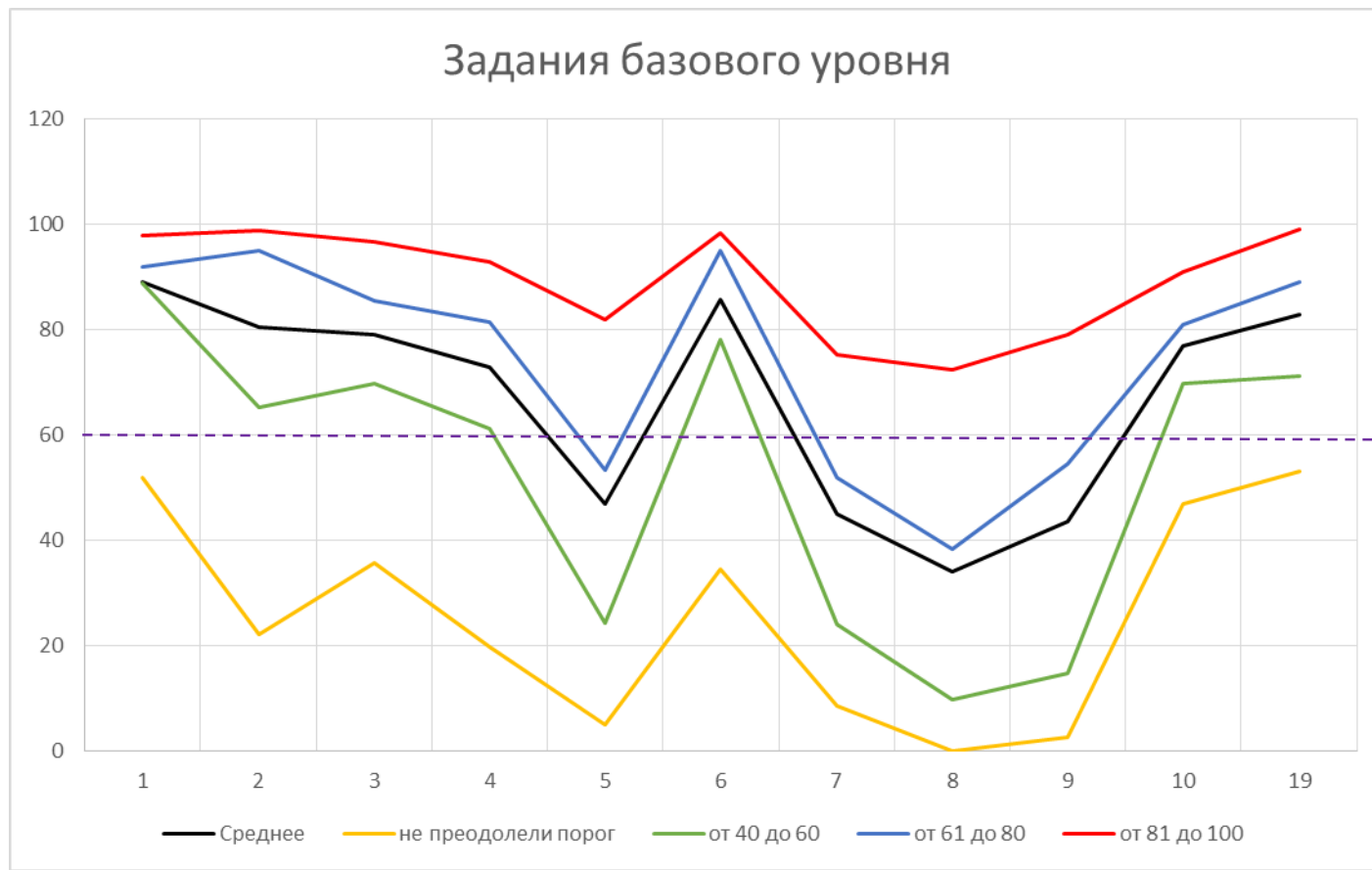
Динамика результатов ЕГЭ по информатике за последние 3 года

Категория	Ярославская область		
	2020	2021	2022
Число участников	984	999	1097
Средний балл	65,7	67,4	64,0
Меньше 40 баллов, %	7,52	4,40	8,83
От 61 до 80 баллов, %	36,59	37,24	35,40
От 81 до 99 баллов, %	25,51	25,23	22,84
100 баллов, чел.	15	8	3

Средний процент выполнения заданий ЕГЭ по учебному предмету «Информатика и ИКТ» по региону 2022 год



Статистический анализ выполнения заданий КИМ в 2022 году



Задание 5

На вход алгоритма подаётся натуральное число N . Алгоритм строит по нему новое число R следующим образом.

1. Строится двоичная запись числа N .
2. Далее эта запись обрабатывается по следующему правилу:

а) если сумма цифр в двоичной записи числа N четная, то к этой записи справа дописывается 0, а затем два левых разряда заменяются на 10;

б) если сумма цифр в двоичной записи числа N нечетная, то к этой записи справа дописывается 1, а затем два левых разряда заменяются на 11.

Полученная запись является двоичной записью искомого числа R . Укажите минимальное число N , после обработки которого с помощью этого алгоритма получается число R , больше 40. В ответе запишите это число в десятичной системе счисления.

```
for n in range(1,100):
    s = bin(n)[2:]
    if s.count('1')%2 == 0 :
        s = '10' + s[2:] + '0'
    else :
        s = '11' + s[2:] + '1'
    r = int(s,2)
    if r > 40:
        print(n)
```

IDLE Shell 3.9.5

File Edit Shell Debug Options Window Help

Python 3.9.5 (tags/v3.9.5:0a7dcdb, M
D64)] on win32

Type "help", "copyright", "credits"

>>>

==== RESTART: C:/Users/svbel/AppDat

16

19

>>

Задание 7

Для хранения сжатого произвольного растрового изображения размером 640×256 пикселей отведено 170 Кбайт памяти без учёта размера заголовка файла. Файл оригинального изображения, больше сжатого на 35%. Для кодирования цвета каждого пикселя используется одинаковое количество бит, коды пикселей записываются в файл один за другим без промежутков. Какое максимальное количество цветов можно использовать в изображении?

Сжатый файл 170 Кбайт – 100 %
Оригинал 135 % - $170 \cdot 1,35$ Кбайт

$$640 \cdot 256 \cdot i \leq 170 \cdot 1,35 \cdot 2^{13}$$

$$i \leq \frac{170 \cdot 1,35 \cdot 2^{13}}{640 \cdot 256}$$

$$i \leq 11,47$$

$$2^{11} = 2048$$

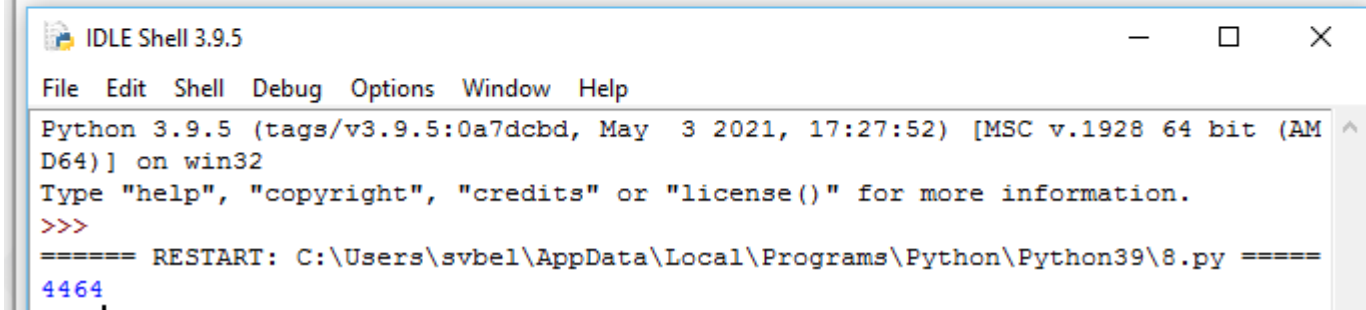
```
>>> 170*1.35*2**13/(640*256)
11.475000000000001
>>> |
```

Задание 8

Определите количество пятизначных чисел, записанных в девятеричной системе счисления, в записи которых ровно одна цифра 1, при этом никакая четная цифра не стоит рядом с цифрой 1.

```
from itertools import *
k=0 # счетчик всех слов
q = ['01', '10', '21', '12', '41', '14', '61', '16', '81', '18'] # все варианты, когда четная
# цифра стоит рядом с 1

for x in product('012345678', repeat=5):
    s=''.join(x)
    if s[0] != '0' and s.count('1') == 1 and all(sub not in s for sub in q) :
        k+=1
print(k)
```



The screenshot shows the IDLE Shell 3.9.5 window. The code is executed, and the output is 4464. The window title is "IDLE Shell 3.9.5" and the menu bar includes "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The status bar shows "Python 3.9.5 (tags/v3.9.5:0a7dcbd, May 3 2021, 17:27:52) [MSC v.1928 64 bit (AMD64)] on win32". The prompt is ">>>" and the output is "4464".

Задание 9

Откройте файл электронной таблицы, содержащей в каждой строке четыре натуральных числа. Определите количество строк таблицы, содержащих числа, для которых выполнены оба условия:

- наибольшее из четырех чисел меньше суммы трех других;
 - четыре числа можно разбить на две пары чисел с равными суммами.
- В ответе запишите только число.

КРИТИКОМ

НАИМЕНЬШИЙ(массив;k)
Возвращает k-ое наименьшее значение в множестве данных (например, пятое наименьшее).

буфер обмена | шрифт | выравнивание | число

K3 : =ЕСЛИ(И(И3<F3+G3+H3;F3+I3=G3+H3);1;

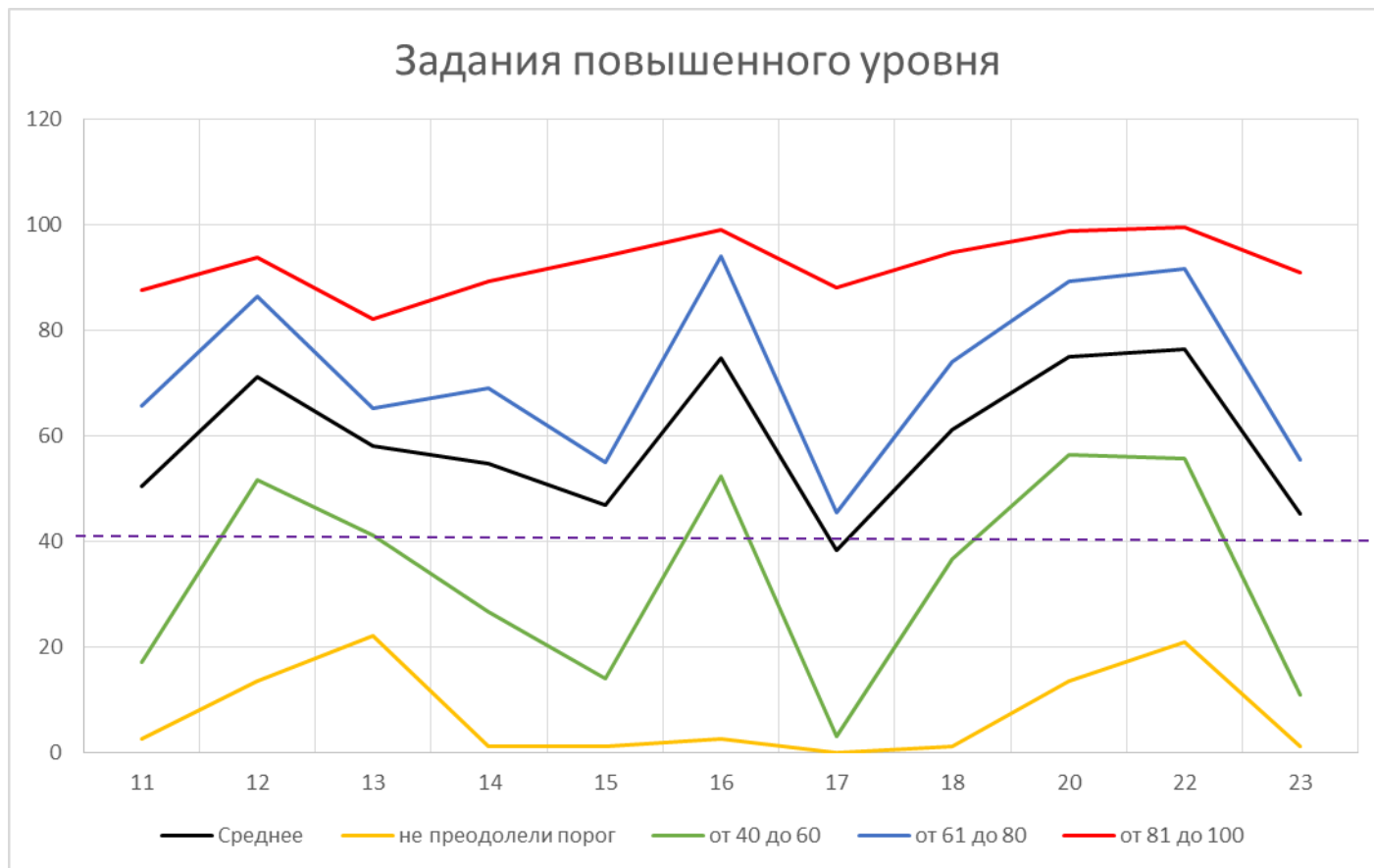
	A	B	C	D	E	F	G	H	I	J	K	L
1	59	65	56	23		23	56	59	65		0	
2	9	15	57	13		9	13	15	57		0	
3	85	39	51	5		5	39	51	85		1	

SMALL Результат функции
Возвращает k-ое наименьшее значение выборки.

K3 = =IF(AND(I3<F3+G3+H3;F3+I3=G3+H3);1;0)

	A	B	C	D	E	F	G	H	I	J	K	L
1	59	65	56	23		23	56	59	65		0	
2	9	15	57	13		9	13	15	57		0	
3	85	39	51	5		5	39	51	85		1	

Статистический анализ выполнения заданий КИМ в 2022 году

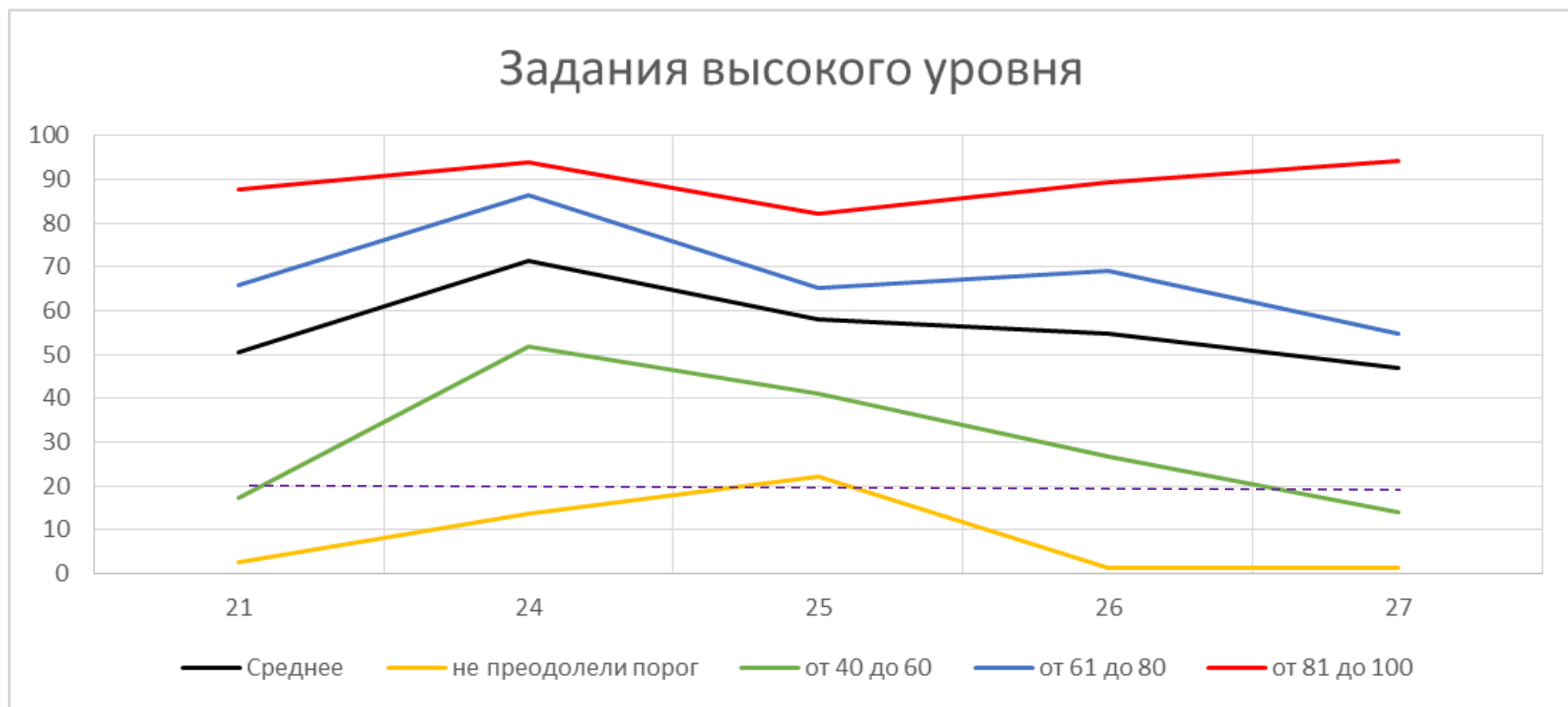


Задание 17

В файле содержится последовательность натуральных чисел. Элементы последовательности могут принимать целые значения от -1 до $100\,000$ включительно. Определите количество пар последовательности, в которых остаток от деления хотя бы одного из элементов на 20 равен минимальному элементу последовательности. В ответе запишите количество найденных пар, затем максимальную из сумм элементов таких пар. В данной задаче под парой подразумевается два идущих подряд элемента последовательности.

```
a = [int(x) for x in open('17.txt')]
m = min(a) #минимальное число из файла
ans = [] #пустая последовательность для хранения сумм подходящих пар
for i in range(len(a)-1):
    if abs(a[i])%20 == m or abs(a[i+1])%20 == m: #проверка условия для модулей чисел
        ans.append(a[i]+a[i+1]) #сохраняем сумму пар в последовательность
print(len(ans), max(ans)) #выводим длину и максимальное значение последовательности
```

Статистический анализ выполнения заданий КИМ в 2022 году



Задание 21

Умение построить дерево игры по заданному алгоритму и найти выигрышную стратегию.

Для игры, описанной в задании 19, найдите минимальное значение s , при котором одновременно выполняются два условия:

- у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети;
- у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Задание 23

Умение анализировать результат исполнения алгоритма, содержащего ветвление и цикл. Исполнитель преобразует число на экране.

У исполнителя есть две команды, которым присвоены номера:

1. Вычти 1
2. Найди целую часть от деления на 2

Сколько существует программ, для которых при исходном числе 30 результатом является число 1, и при этом траектория вычислений содержит число 9?

```
def numProg( start, x ):
    if x > start: return 0
    if x == start: return 1
    return numProg(start-1,x) + numProg(start // 2,x)

print( numProg(30,9)*numProg(9,1) )
# начал со значения start, мы можем попасть в x либо через значение start-1,
# либо через start // 2
```

IDLE Shell 3.9.5

File Edit Shell Debug Options Window Help

Python 3.9.5 (tags/v3.9.5:0a7dcbd, May 3 2021, 17:27:52) [MSC v.1928 64 bit (AMD64)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

```
>>>
==== RESTART: C:/Users/svbel/AppData/Local/Programs/Python/Python39/2323.py ====
322
>>> |
```

Задание 24

Умение создавать собственные программы (10–20 строк) для обработки символьной информации

Текстовый файл состоит из символов A, B, C, D и E.

Определите максимальное количество идущих подряд символов вида
согласная + гласная

в прилагаемом файле.

Для выполнения этого задания следует написать программу.

```
s = open('24.txt').readline()
s = s.replace('C', 'B').replace('D', 'B').replace('E', 'A')
s = s.replace('BA', '*').replace('B', ' ').replace('A', ' ')
print(max(len(c) for c in s.split()))
```

Задание 25

Умение создавать собственные программы (10–20 строк) для обработки целочисленной информации

Назовем маской последовательность цифр, в которой также могут встречаться следующие символы:

- символ «?» означает ровно одну произвольную цифру;
- символ «*» означает любую последовательность цифр произвольной длины; в том числе «*» может задавать и пустую последовательность.

Среди натуральных чисел, не превышающих 10^8 , найдите все числа, соответствующие маске $123*678$, делящиеся на 13 без остатка.

В ответе запишите в первом столбце таблицы все найденные числа в порядке возрастания, а во втором столбце – соответствующие им результаты деления этих чисел на 13.

Количество строк в таблице для ответа избыточно.

Задание 25

Среди натуральных чисел, не превышающих 10^8 , найдите все числа, соответствующие маске $123*678$, делящиеся на 13 без остатка.

В ответе запишите в первом столбце таблицы все найденные числа в порядке возрастания, а во втором столбце – соответствующие им результаты деления этих чисел на 13.

```
#123A678 <= 100000000
#123AA678 <= 100000000

s = ''
for i in '0123456789':
    s = '123'+i+'678'
    d = int(s)
    if d % 13 == 0:
        print(d, d // 13)
s = ''
for i in '0123456789':
    for j in '0123456789':
        s = '123'+i+j+'678'
        d = int(s)
        if d % 13 == 0:
            print(d, d // 13)
```

```
File Edit Shell Debug Optio
Python 3.9.5 (tags/v3.9
D64)] on win32
Type "help", "copyright
>>> 123678/13
9513.692307692309
>>>
==== RESTART: C:\Users\
1237678 95206
12300678 946206
12313678 947206
12326678 948206
12339678 949206
12352678 950206
12365678 951206
12378678 952206
12391678 953206
....
```

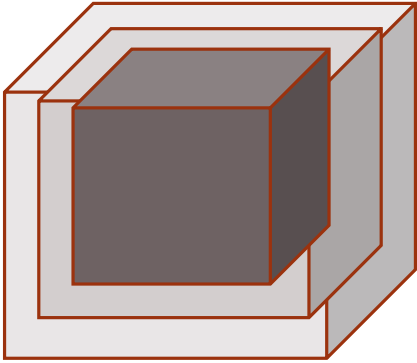
Задание 26

Умение обрабатывать целочисленную информацию с использованием сортировки

В магазине для упаковки подарков есть N кубических коробок. Самой интересной считается упаковка подарка по принципу матрешки – подарок упаковывается в одну из коробок, та в свою очередь в другую коробку и т.д. Одну коробку можно поместить в другую, если длина её стороны хотя бы на 7 единиц меньше длины стороны другой коробки.

Определите наибольшее количество коробок, которое можно использовать для упаковки одного подарка и максимально возможную длину сторон самой маленькой коробки, где будут находиться подарки. Размер подарка позволяет поместить его в самую маленькую коробку.

Запишите в ответе два целых числа: сначала наибольшее количество коробок, которое можно использовать для упаковки одного подарка, затем максимально возможную длину стороны самой маленькой коробки в таком наборе.



Типовой пример организации данных во входном файле

5
43
40
32
40
30

Пример входного файла приведен для пяти коробок и случая, когда минимально допустимая разница между длинами сторон коробок, подходящих под упаковку «матрешкой», составляет 3 единицы. При таких исходных данных условию задачи удовлетворяет набор коробок с длинами сторон 30, 40, 43 или 32, 40, 43 соответственно, т.е. коробок равно 3, а длина стороны самой маленькой коробки равна 32.

Типовой пример имеет иллюстративный характер. Для выполнения задания используйте данные прилагаемых файлов.

Одну коробку можно поместить в другую, если длина её стороны хотя бы на 7 единиц меньше длины стороны другой коробки.

Определите наибольшее количество коробок, которое можно использовать для упаковки одного подарка и максимально возможную длину сторон самой маленькой коробки, где будут находиться подарки.

Запишите в ответе два целых числа: сначала наибольшее количество коробок, которое можно использовать для упаковки одного подарка, затем максимально возможную длину стороны самой маленькой коробки в таком наборе.

```
f = open('26.txt')
n = int(f.readline())
a = [int(x) for x in f]# формируем список чисел из файла
a = sorted(set(a), reverse = 1)#сортируем числа по убыванию,
                               #метод set() исключает повторения

ans = [a[0]]
for i in range(1, len(a)):
    if ans[-1] - a[i] >= 7:
        ans.append(a[i])

print(len(ans), ans[-1])
```

Задание 27

Умение создавать собственные программы (20–40 строк) для анализа числовых последовательностей

У медицинской компании есть N пунктов приёма биоматериалов на анализ. Все пункты расположены вдоль автомагистрали и имеют номера, соответствующие расстоянию от нулевой отметки до конкретного пункта. Известно количество пробирок, которое ежедневно принимают в каждом из пунктов. Пробирки перевозят в специальных транспортировочных контейнерах вместимостью не более 40 штук. Каждый транспортировочный контейнер упаковывается в пункте приема и вскрывается только в лаборатории. Компания планирует открыть лабораторию в одном из пунктов. Стоимость перевозки биоматериалов равна произведению расстояния от пункта до лаборатории на количество контейнеров с пробирками. Общая стоимость перевозки за день равна сумме стоимостей перевозок из каждого пункта в лабораторию. Лабораторию расположили в одном из пунктов приема биоматериалов таким образом, чтобы общая стоимость доставки биоматериалов из всех пунктов была минимальной.

Определите минимальную общую стоимость доставки биоматериалов из всех пунктов приема.

Входные данные.

Дано два входных файла (А и В), каждый из которых в первой строке содержит число N ($1 \leq N \leq 10\,000\,000$) – количество пунктов приема биоматериалов. В каждый из следующих N строк находится два числа: номер пункта и количество пробирок в этом пункте (все числа натуральные, количество пробирок в каждом пункте не превышает 1000).

Пункты перечислены в порядке их расположения вдоль дороги, начиная с нулевой отметки.

В ответе укажите два числа: сначала значение искомой величины для файла А, затем для файла В.

Пример организации данных в файле:

```
6
1 100
2 200
5 4
7 3
8 2
10 190
```

При таких исходных данных и вместимости транспортировочного контейнера, составляющей 96 пробирок, компании выгодно открыть лабораторию в пункте 2. В этом случае сумма транспортировочных затрат составит:

$$1 \cdot 2 + 3 \cdot 1 + 5 \cdot 1 + 6 \cdot 1 + 8 \cdot 2$$

Пример организации данных в файле:

6

1 100

2 200

5 4

7 3

8 2

10 190

При таких исходных данных и вместимости транспортировочного контейнера, составляющей 96 пробирок, компании выгодно открыть лабораторию в пункте 2. В этом случае сумма транспортировочных затрат составит:

$$1 \cdot 2 + 3 \cdot 1 + 5 \cdot 1 + 6 \cdot 1 + 8 \cdot 2$$

Пункт	1	2	5	7	8	10
Кол. Конт.	2	3	1	1	1	2

Стоимость в первом пункте:

$$S_1 = 3 \cdot 1 + 4 \cdot 1 + 6 \cdot 1 + 7 \cdot 1 + 9 \cdot 2$$

Пример организации данных в файле:

6

1 100

2 200

5 4

7 3

8 2

10 190

При таких исходных данных и вместимости транспортировочного контейнера, составляющей 96 пробирок, компании выгодно открыть лабораторию в пункте 2. В этом случае сумма транспортировочных затрат составит:

$$1 \cdot 2 + 3 \cdot 1 + 5 \cdot 1 + 6 \cdot 1 + 8 \cdot 2$$

Пункт	1	2	5	7	8	10
Кол. Конт.	2	3	1	1	1	2
Пункт до		Пункты после				

Стоимость во втором пункте:

$$S_2 = S_1 + \text{разница} \cdot \text{число конт. до} - \text{разница} \cdot \text{число конт. после}$$


```

f = open('27A.txt')
n = int(f.readline())
a = []
for i in range(n):
    s,k = map(int, f.readline().split())
    #количество контейнеров
    if k%40 == 0: c = k//40
    else: c = k//40+1
    a.append([s,c])

#решение A
ms = 10**20
for i in range(n):
    s = 0
    for j in range(n):
        s += abs(a[i][0] - a[j][0])*a[j][1]
    ms = min(ms,s)
print(ms)

#решение B
#префиксные суммы количества контейнеров в первых пунктах
bags = [0]*n
bags = a[0]*[1]
for i in range(n):
    bags[i] = bags[i-1] + a[i][1]
#стоимость доставки в нулевом пункте
s = 0
for j in range(n):
    s += abs(a[0][0] - a[j][0])*a[j][1]
ms = s
for i in range(1,n):
    dif = a[i][0] - a[i-1][0] #расстояние между двумя пунктами
    s = s + dif*bags[i-1] #для предыдущих пунктов дороже, для следующих дешевле
    ms = min(ms,s)
print(ms)

```