

Ты с ДВ? Присылай задачи сюда
<https://t.me/rodyainfsupport>

Привет всем чатам, по которым гуляет ссылка. Люблю вас, удачи на экзамене, надеюсь у вас все получится

Стрим Информатика Роди, Алексея Кабанова и Максима Рейма проходит тут:

Youtube — <https://www.youtube.com/live/aRg6OyHj114>

VK — https://vkvideo.ru/video-205865487_456240581

РАЗБОР ДАЛЬНЕГО ВОСТОКА

Задание 5 — двоичная сс

Задание 7 — голосовое сообщение как на досроке, база

Задание 8 — база

Задача 8.1

Определите количество пятизначных чисел, записанных в девятеричной системе счисления, в записи которых ровно две цифры 3, и при этом никакая нечётная цифра не стоит рядом с цифрой 2.

```
1 from itertools import product
2
3 count = 0
4 for x in product('012345678', repeat=5):
5     x = ''.join(x)
6     if x[0] != '0':
7         if x.count('3') == 2:
8             for c in '1357': x = x.replace(c, '1')
9             if '12' not in x and '21' not in x:
10                count += 1
11 print(count)
```

Задание 11 — минимальная мощность алфавита (либо автокод, либо $N = 2^{i-1} + 1$)

Задание 12 — база

	λ	0	1
q0	$\lambda, R, q1$		
q1	0, R, q2	0, R, q1	1, R, q1
q2	0, R, q3		
q3	$\lambda, S, q3$		

Задание 14 — стандартная с f-строкой

Задание 15 — ДЕЛ и отрезки в одной задаче

(обычное решение ДЕЛ и добавляем на отрезок $15 \leq x \leq 30$)

15

№ 4986 (Уровень: Базовый)

Обозначим через $\text{ДЕЛ}(n, m)$ утверждение «натуральное число n делится без остатка на натуральное число m »; дан отрезок $B = [50; 70]$.

Для какого наибольшего натурального числа A формула

$\text{ДЕЛ}(x, A) \vee (\text{ДЕЛ}(x, 23) \rightarrow \neg(x \in B))$

тождественно истинна (т.е. принимает значение 1) при любом натуральном значении переменной x ?

```
python
def f(x, A):
    return (x % A == 0) or ((x % 23 == 0) <= (not
(50 <= x <= 70)))

for A in range(1, 1000):
    if all(f(x, A)==1 for x in range(1, 1000)):
        print(A)
```

**Задание 18 — УГЛОВЫЕ КЛЕТКИ КРАТНЫЕ ТРЕМ ТОЛЬКО
БЕРУТСЯ, ЕСЛИ В УГЛУ ЧИСЛО НЕ КРАТНО 3 ТО ЭТО НЕ МОЖЕТ
БЫТЬ ОТВЕТОМ**

Задание 19-21 — база

Задание 22 —

Задача №22

Задача 22.1

В файле содержится информация о совокупности N вычислительных процессов, которые могут выполняться параллельно или последовательно. Приостановка выполнения процесса не допускается. Будем говорить, что процесс B зависит от процесса A , если для выполнения процесса B необходимы результаты выполнения процесса A . В этом случае процессы A и B могут выполняться только последовательно.

Информация о процессах представлена в файле в виде таблицы. В первом столбце таблицы указан идентификатор процесса (ID), во втором столбце таблицы – время его выполнения в миллисекундах, в третьем столбце перечислены с разделителем «;» ID процессов, от которых зависит данный процесс. Если процесс независим, то в таблице указано значение 0.

Определите **максимальное** количество процессов, которые начнут выполняться не ранее 15-й секунды (время начала 15 секунд также учитывается).

Типовой пример организации данных в файле

Для приведённой таблицы процесс 3 начинается на 8-й мс, заканчивается на 9-й мс.

Типовой пример имеет иллюстративный характер. Для выполнения задания используйте данные из прилагаемого файла.

Задание 23

Два хода: 1) Прибавь 1; 2) Поменять цифры в разряде единиц и десятков местами, если разряд десятков меньше разряда единиц

```
python
def f(c, e):
    p = str(c)
    if c > e: return 0
    if c == e: return 1
    if p[1] < p[2]:
        return f(c+1, e) + f(int(p[0]+p[2]+p[1]), e)
    else:
        return f(c+1, e)
print(f(100, 150))
```

№ 23



Исполнитель преобразует число на экране. У исполнителя есть две команды, которые обозначены номерами:

1. Прибавь 3
2. Поменяй местами

Первая из этих команд увеличивает число на экране на 3. **Вторая команда может применяться только к числу, у которого цифра разряда десятков по значению меньше цифры, стоящей в разряде единиц**, и действует, заменяя число на экране числом, в котором цифры двух младших разрядов поменялись местами.

Программа для исполнителя — это последовательность команд.

Сколько существует программ, для которых при исходном числе 10 результатом является число 169?

```
1 def f(x, end): 4 usages
2     if x > end: return 0
3     if x == end: return 1
4     h = []
5     h.append(f(x + 3, end))
6     if len(str(x)) == 2:
7         x = str(x)
8         if x[0] < x[1]:
9             h.append(f(int(x[1] + x[0]), end))
10    if len(str(x)) == 3:
11        x = str(x)
12        if x[1] < x[2]:
13            h.append(f(int(x[0] + x[2] + x[1]), end))
14    return sum(h)
15
16 print(f(10, 169))
```

Задание 24

• ЗАДАНИЕ 24

Обработка символьной строки

Задание выполняется с использованием прилагаемых файлов.

условие

Текстовый файл состоит из десятичных цифр и знаков арифметической операции `+` и `*` (сложения и умножения).

Определите **максимальное количество символов** в непрерывной последовательности, которая является корректным арифметическим выражением с целыми неотрицательными числами (без знака). В этом выражении никакие два знака арифметических операций не стоят рядом, в записи чисел отсутствуют незначащие (ведущие) нули.

В ответе укажите количество символов.

egeedu.me

```
1 from re import *
2
3 s = open('24.txt').readline()
4 s = s.replace('*', '-')
5 number = r'(0|[1-9][0-9]*)'
6 reg = rf'{number}(-{number})*'
7
8 print(max(len(x.group()) for x in finditer(reg, s)))

## Демонстрация
s = open('24_20813.txt').readline()

m = 0
warn = ['**', '*-', '--', '-*'] \
      + ['*0'+c for c in '0789'] \
      + ['-0'+c for c in '0789']
for left in range(len(s)):
    for right in range(left+m, len(s)):
        c = s[left:right+1]
        if c[0] in '-*' or any(w in '-' + c for w in warn):
            break
        elif c[-1] not in '**':
            m = max(m, len(c))
print(m)
```

Задание 25 — база на простые множители, должны меть цифры 1 и 6

№ 25. Обработка целочисленной информации (Простые множители)

Напишите программу, которая перебирает целые числа, большие 7_513_048, в порядке возрастания и ищет среди них числа, представленные в виде произведения ровно двух простых множителей, не обязательно различных, каждый из которых содержит в своей записи хотя бы одну цифру **1** и одну цифру **6**.

В ответе для первых 5 найденных чисел запишите само число и наибольший из его простых множителей в соответствующие столбцы таблицы.

```
python
def p(x):
    for i in range(2, int(x**0.5)+1):
        if x%i==0:
            return [i]+p(x//i)
    return [x]

for x in range(7_513_049,7_600_000):
    d = p(x)
    if len(d)==2:
        if ('1' in str(d[0]) and '6' in str(d[0])) and \
            ('1' in str(d[1]) and '6' in str(d[1])):
            print(x,max(d))
```

Задание 26 — задача на МФЦ

```
f = open('26.txt')

K = int(f.readline())
N = int(f.readline())

a = []
for s in f:
    st,end = [int(x) for x in s.split()]
    a.append([st,end])
a.sort()

#время окончания работы с клиентом
okna = [0]*(K+1)
count=0
last = 0
for st, end in a:
    for j in range(1,K+1):
        if okna[j]<st:
            okna[j] = end
            count+=1
            last = j
            break
print(count,last)
```

Задание 26



Задание выполняется с использованием прилагаемых файлов.

В центр обработки информации приходят запросы на сервер, имеющий ограниченный запас памяти. Для каждого запроса дана время регистрации, идентификатор клиентского устройства и объём данных. Если в какой-то момент приходит информация, а свободного объёма памяти сервера не хватает для сохранения этой информации, сервер делает резервную копию и отправляет её в облако, после чего память сервера обнуляется, новая информация добавляется на сервер.

Входные данные

Первая строка входного файла содержит два натуральных числа: N - количество строк, K вместимость специального раздела памяти сервера в Кб. Каждая из следующих N строк содержит информацию об одном выполненном запросе: время регистрации в формате ЧЧ:ММ:СС и два натуральных числа: C – идентификатор клиентского устройства, S – объём данных запроса в Кб.

Выходные данные

Два целых положительных числа: сначала идентификатор клиентского устройства, который отправил наибольший объём запросов, а затем максимальный суммарный объём двух резервных копий, которые отправлялись в облако до 12 часов дня.

Логика решения — (Экселем) 1 вопрос – сводная таблица чик-чик (либо сортировать по возр. Идентификаторов, суммировать размеры файлов если айди одинаковые, иначе начинается с нуля)

2 вопрос - Сортировка строчек по возрастанию времени, рядом столбец с размером сервера, приплюсовываем файлики и пишем условие что если суммарный размер больше чем объём памяти, то сбрасывается до размера одного файла

Фильтром выбираем тайминги до 12:00:00 (можно разбить часы минуты секунды на три столбца), посмотреть сколько раз сбрасывалась память на одиночный файл, и посмотреть две резервные копии, выбрать 2 МАКС

```
python
f = open('26.txt').readline()

N = int(f.readline())
K = int(f.readline())
a = []

for s in f:
    t, c, s = s.split()
    c, s = int(c), int(s)
    a.append([t,c,s])

a.sort()
server = 0
backup = []
d = [0]*1_000_000
for t,c,s in a:
    d[c]+=s
    server+=s
    if server>K:
        if int(t[:2])<12: backup.append(server-t)
        server = t
backup.sort()
print(max(d), backup[-1]+backup[-2])
```

Задание 27 — звезды с характеристиками