

# Сегодня в уроке

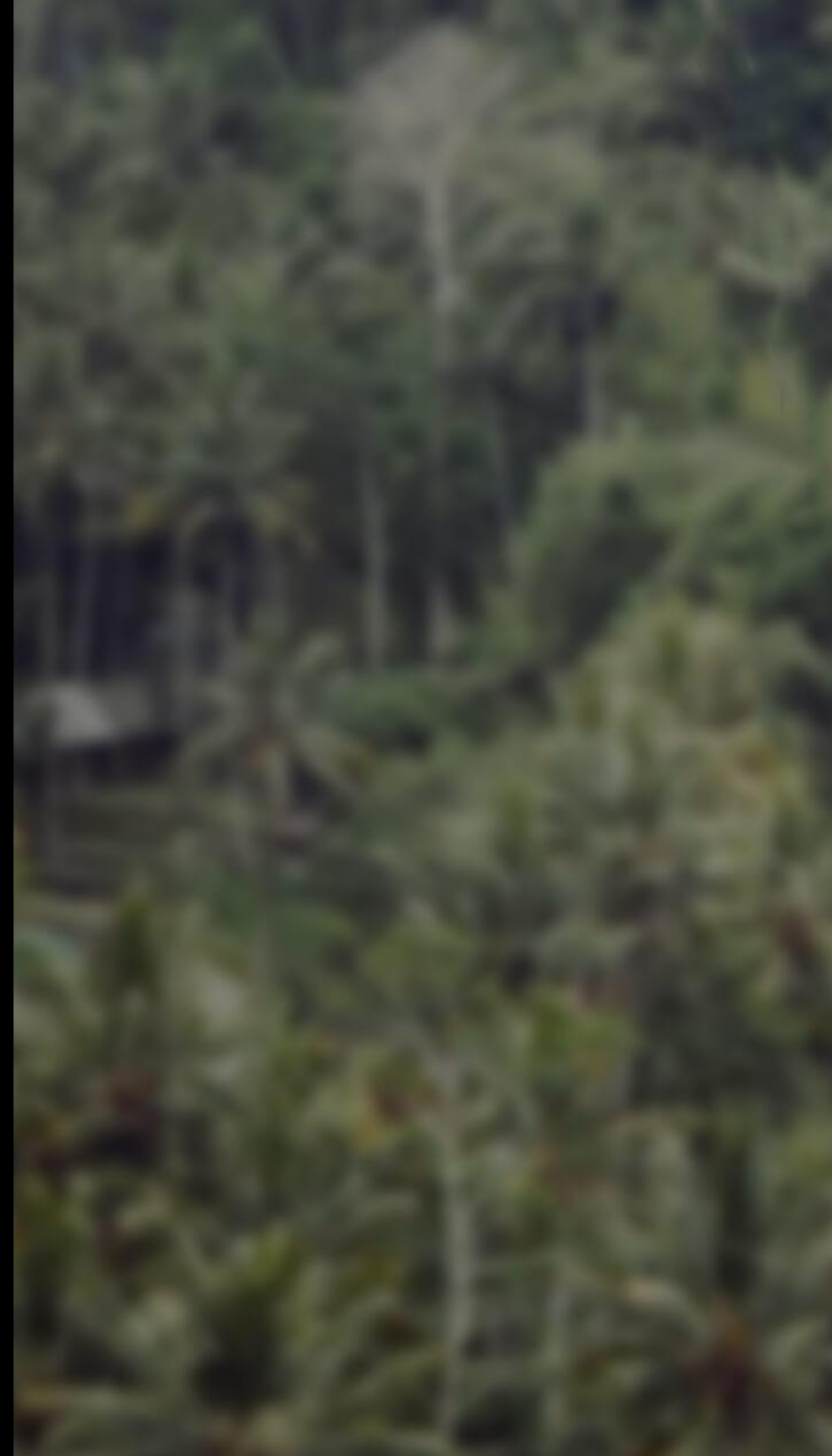
- Задание 26. Часть 5

**ВЕРОНИКА  
ПАВЛОВНА**

Геймификация  
подготовки к ЕГЭ  
по информатике



pavlovna\_ege



## Задание 26.7

Проводится вычислительный эксперимент для определения необходимого количества самокатов на разных парковках города в начальный момент времени. Всего есть  $M$  парковок с номерами от 1 до  $M$ . Поступило всего  $N$  заявок на аренду самокатов. В каждой заявке указано время начала аренды в минутах от начала суток, продолжительность аренды, а также номера парковок старта и финиша. Будем считать, что заряда самоката хватает на весь день и самокат может быть арендован со следующей минуты после окончания предыдущей аренды.

Определите, начиная с какого момента (в минутах от начала эксперимента) было арендовано максимальное количество самокатов, и номер парковки, на которой нужно установить максимальное количество самокатов.

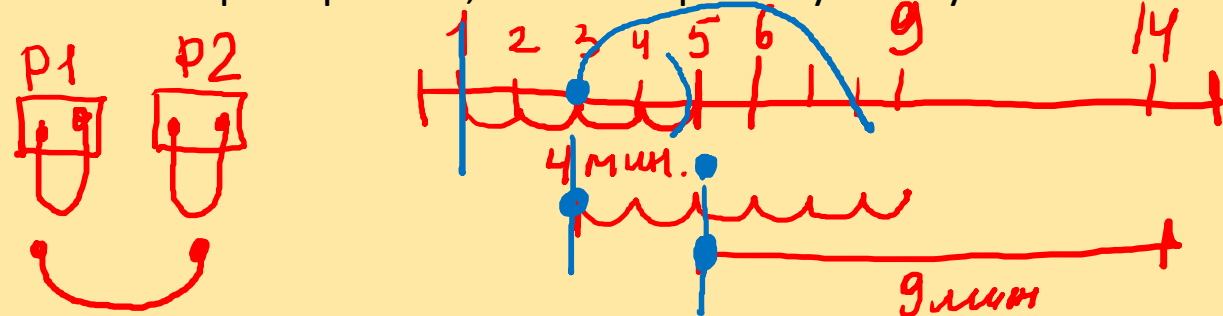
**Входные данные** представлены в файле **26-123.txt** следующим образом. Первая строка входного файла содержит два натуральных числа, записанных через пробел:  $M$  ( $1 \leq M \leq 100$ ) – количество парковок, и  $N$  ( $1 \leq N \leq 10^6$ ) – количество заявок. Каждая из  $N$  последующих строк описывает содержит четыре целых числа: время начала аренды в минутах от начала суток, длительность аренды в минутах, номер парковки старта и номер парковки финиша.

В ответе запишите два числа: сначала время (в минутах от начала эксперимента), когда впервые началась аренда максимального количества самокатов, затем – номер парковки, на которой нужно установить максимальное количество самокатов.

## Пример входного файла:

$M = \begin{bmatrix} 2 & 3 \\ 1 & 4 \\ 3 & 6 \\ 5 & 9 \end{bmatrix} = N$

При таких исходных данных нужно три самоката: два в начале размещаются на парковке 1 и один – на парковке 2. Одновременно в аренде находятся максимум два самоката (с 3-й по 8-ю минуту включительно). Ответ: 3 1.



unit  $\uparrow$   $\uparrow$   $\uparrow$

```
File Edit Format Run Options Window Help
with open('test.txt') as f:
    P, N = map(int, f.readline().split())
    s = [list(map(int, i.split())) for i in f]
time = [0 for i in range(2*24*60+1)]
kP = [0 for i in range(P+1)]
for i in s:
    startTime, pr, startP, endP = i
    kP[startP] += 1
    for j in range(startTime, startTime+pr):
        time[j] += 1
    #print(time)
print(time.index(max(time)), kP.index(max(kP)))
```

## Задание 26.8

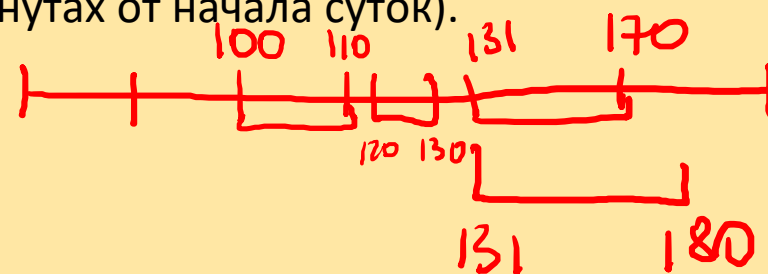
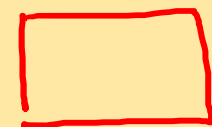
(ЕГЭ-2023) Входной файл содержит сведения о заявках на проведение занятий в конференц-зале. В каждой заявке указаны время начала и время окончания мероприятия (в минутах от начала суток). Если время начала одного мероприятия меньше времени окончания другого, то провести можно только одно из них. Если время окончания одного мероприятия совпадает с временем начала другого, то провести можно оба. Определите максимальное количество мероприятий, которое можно провести в конференц-зале и самое позднее время окончания последнего мероприятия.

**Входные данные** представлены в файле `26-128.txt` следующим образом. Первая строка входного файла содержит натуральное число  $N$  ( $1 \leq N \leq 1000$ ) – количество заявок на проведение мероприятий. Следующие  $N$  строк содержат пары чисел, обозначающих время начала и время окончания мероприятий. Каждое из чисел натуральное, не превосходящее 1440.

Запишите в ответе два числа: максимальное количество мероприятий, которое можно провести в конференц-зале и самое позднее время окончания последнего мероприятия (в минутах от начала суток).

**Пример входного файла:**

5



- 1 10 150
- 2 100 110
- 3 131 170
- 4 131 180
- 5 120 130

При таких исходных данных можно провести максимум три мероприятия, например, по заявкам 2, 3 и 5. Конференц-зал освободится самое позднее на 180-й минуте, если состоятся мероприятия по заявкам 2, 4, 5. Ответ: 3 180.

```
File Edit Format Run Options Window Help
with open('26-128.txt') as f:
    N = int(f.readline())
    s = [list(map(int, i.split())) for i in f]
for i in range(N):
    start, end = s[i]
    s[i].insert(0, end-start)
s = sorted(s)
k = 0
time = [0 for i in range(1441)]
for i in range(N):
    d, start, end = s[i]
    if sum(time[start+1:end+1]) == 0:
        k += 1
        pm = [start, end]

        for j in range(start, end+1):
            time[j] = 1
    else:
        if end > pm[1] and start >= pm[0]:
            pm = [start, end]
print(k, pm[1])
```