

# Региональный этап Всероссийской олимпиады школьников по информатике

Казань, 20-22 января 2024 г.

## Задача «Посадка в самолет»



## Постановка задачи

- Имеется  $n$  рядов в самолете по 6 мест в каждом.
- $m$  – количество пассажиров, которые придут на стойку регистрации.
- Задана изначальная рассадка в самолете после онлайн-регистрации.

	1	2	3	4	5	6
1	✗					
2						
3					✗	
4	✗					
5						
6			✗	✗		

## Решение. Подзадача 1 (15 баллов)

 $m = 0$ 

- перебор всех рядов, проверка симметрии

```
for i in range(n):
    for j in range(3):
        if p[i][j] != p[i][5 - j]:
            print('Impossible')
```

## Решение. Подзадача 2 (16 баллов)

Изначально в самолете все места свободны

- $m$  – не отрицательное
- $m$  – не больше всех мест в самолете ( $6n$ )
- $m$  – четное

```
if m > 6 * n or m < 0 or m % 2 == 1:  
    print('Impossible')
```

## Решение. Подзадача 2 (16 баллов)

Изначально в самолете все места свободны

- размещаем симметрично новых пассажиров на свободные места

```
for i in range(n):  
    for j in range(3):  
        if p[i][j] == 0 and m > 0:  
            m -= 2  
            p[i][j] = 1  
            p[i][5 - j] = 1
```

## Решение. Подзадача 3 (17 баллов)

 $m = 1$ 

- занято нечетное количество мест
- перебор всех рядов, проверка симметрии

```
found = False
for i in range(n):
    for j in range(3):
        if p[i][j] != p[i][5 - j]:
            if found:
                print('Impossible ')
            else:
                p[i][j] = 1
                p[i][5 - j] = 1
                found = True
if not found:
    print('Impossible ')
```

## Решение. Подзадача 4 (18 баллов)

Изначально в самолете занято ровно одно место

- $m$  – не отрицательное
- $m + 1$  – не больше всех мест в самолете ( $6n$ )
- $m$  – нечетное

```
if m + 1 > 6 * n or m < 0 or m % 2 != 1:  
    print('Impossible')
```

## Решение. Подзадача 4 (18 баллов)

Изначально в самолет занято ровно одно место

- дополняем ряд с занятым местом

```
for i in range(n):
    for j in range(3):
        if p[i][j] != p[i][5 - j]:
            m -= 1
            p[i][j] = 1
            p[i][5 - j] = 1
```

## Решение. Подзадача 4 (18 баллов)

Изначально в самолет занято ровно одно место

- размещаем симметрично новых пассажиров

```
for i in range(n):
    for j in range(3):
        if p[i][j] == 0 and m > 0:
            m -= 2
            p[i][j] = 1
            p[i][5 - j] = 1
```

## Решение. Подзадача 5 (34 баллов)

- дополняем ряды с занятыми местами
- проверяем сколько для этого нужно пассажиров

```
total = 0
for i in range(n):
    for j in range(3):
        if p[i][j] != p[i][5 - j]:
            m += 1
            p[i][j] = 1
            p[i][5 - j] = 1
    total += sum(p[i])

if total + m > 6 * n or m < 0 or m % 2 == 1:
    print('Impossible')
```

## Решение. Подзадача 5 (34 баллов)

- размещаем симметрично новых пассажиров на свободные места

```
for i in range(n):
    for j in range(3):
        if p[i][j] == 0 and m > 0:
            m -= 2
            p[i][j] = 1
            p[i][5 - j] = 1
```

# Вопросы?