

Алгебра логики. Задание №15 (Отрезки)

Пример типового задания

Заданы два отрезка $P = [5, 30]$ и $Q = [14, 23]$, лежащие на числовой прямой. Также существует отрезок A . Он таков, что формула $((x \in P) \equiv (x \in Q)) \rightarrow \neg(x \in A)$ истинна, причем переменная x может принимать любые значения. Какую наибольшую длину может принимать отрезок A ? Определите и запишите в ответ целое число.

Решение задания

Способ решения руками, то есть аналитически:

- 1 Введем новые обозначения: $(x \in P) \equiv P$; $(x \in Q) \equiv Q$; $(x \in A) \equiv A$.
- 2 Получим выражение: $(P \equiv Q) \rightarrow \neg A$
- 3 Преобразуем наше выражение, представив импликацию по правилу:
 $A \rightarrow B = \neg A \vee B$
- 4 Получим выражение: $\neg(P \equiv Q) \vee \neg A$ или $A \rightarrow \neg(P \equiv Q)$

Для того чтобы выражение $A \rightarrow \neg(P \equiv Q)$ не было ложным, нам нужно исключить случай, когда $A = 1$, а $\neg(P \equiv Q) = 0$.

Выражение $\neg(P \equiv Q)$ истинно только тогда, когда $x \in [5; 14)$ и $x \in (23; 30]$, а когда $x \in [14; 23]$, или $x \in (-\infty; 5)$, или $x \in (30; \infty)$, то выражение ложно, следовательно, A может быть либо в промежутке $[5; 14)$, либо $(23; 30]$. Следовательно, наибольшая возможная длина промежутка равна $14 - 5 = 9$.

Ответ: 9

Способ решения с помощью программирования:

В начале объявим наши отрезки в виде списков: генерируем ряды чисел, а потом преобразуем их в списки P и Q .

Далее заполним список A , то есть наш отрезок, числами от 0 до 999.

После переберем значения x в диапазоне $[0; 999]$. С помощью условного оператора будем проверять, если при данном значении x функция ложна, то будем удалять его из нашего списка A .

Выводим полученный список и получаем наши отрезки. Обратите внимание, концы полученных отрезков должны совпадать с концами отрезков из условия задания.



Полный код Python:

```
P = list(range(5, 30 + 1))
Q = list(range(14, 23 + 1))
A = (list(range(1000)))
for x in range (1000):
    if (((x in P) == (x in Q)) <= (x not in A)) == False:
        A.remove(x)
print(A)
```

Универсальный шаблон для решения задач на отрезки с помощью программирования

1 Создадим списки, которые будут имитировать отрезки, описанные в условии задачи.

```
P = list(range(x, y + 1))
Q = list(range(a, b + 1))
```

2.1 Если в задаче просят найти наименьшую длину / наименьшее количество точек:

Создадим пустой массив A и циклом for переберём значения x, чтобы подобрать подходящие для A.

```
A = []
for x in range (1000)
    if F(x, A) == False:
        A.append(x)
print(A)
```

2.2 Если в задаче просят найти наибольшую длину / наибольшее количество точек:

Создадим массив A, который будет содержать в себе подходящий нам промежуток (или пару промежутков) и циклом for переберём значения x, чтобы подобрать подходящие для A.

```
A = list(range(1000))
for x in range (1000):
    if F(x, A) == False:
        A.remove(x)
```



3 Далее анализируем промежуток: если существует разрыв, то в промежутке значение с X сразу перейдет на значение Y (т. е. не будет промежуточных значений с интервалом в 0.1), тогда мы делим промежуток на два отрезка:

$$[x; X] \cup [Y; y],$$

где x, y — начало и конец промежутка соответственно.

```
print(A)
```

Заметки