

Логические элементы,  
их комбинации,  
логические операторы  
и выражения

# На уроке мы узнаем

- Что такое логические элементы (AND, OR, NOT) и как они работают.
- Как логические элементы используются в реальной жизни.
- Как создавать комбинации логических элементов для решения сложных задач.
- Что такое логические операторы и выражения в программировании.
- Как применять знания для анализа и моделирования логических ситуаций.

# Логические элементы вокруг нас

## Задумайтесь:

- Как автоматически открываются двери в супермаркетах?
- Почему работает сигнализация?

**Ответ:** Логические операции управляют этими процессами!

# Пример из жизни

Автоматические двери открываются, если:

- Датчик движения активирован (1)
- Датчик приближения активирован (1)

**Логический элемент: AND**

# Зачем это нужно знать?

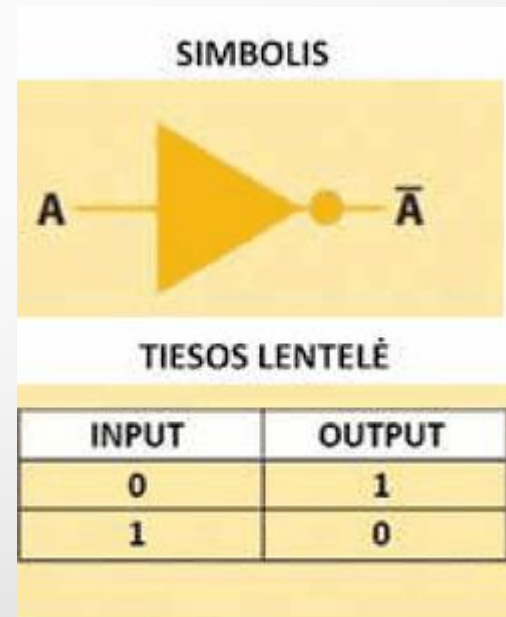
- Логические элементы – основа работы компьютеров и гаджетов.
- Используются в программировании, схемотехнике и автоматизации.
- **Вопрос классу:**  
Как еще можно использовать подобные логические процессы в жизни?

# NOT (НЕ)

## Логическое отрицание или инверсия

- Отрицание: инвертирует входное значение.
- Если вход = 1, результат = 0, и наоборот.

Обозначение:  $\neg$   $\sim$  !



# AND (И)

**Логическое умножение  
или конъюнкция**

- Результат = 1, только если оба входа = 1

**Обозначение:  $\wedge$  • &**

<https://github.com/agorbatniov/inf10>

SIMBOLIS



TIESOS LENTELE


INPUT		OUTPUT
A	B	A AND B
0	0	0
0	1	0
1	0	0
1	1	1

# OR (ИЛИ)

## Логическое сложение или дизъюнкция

- Результат = 1, если хотя бы один вход = 1

Обозначение:  $\vee$  + ||

SIMBOLIS		
A		A+B
B		
TIESOS LENTELE		
INPUT		OUTPUT
A	B	A OR B
0	0	0
0	1	1
1	0	1
1	1	1



## LOGINIAI VARTAI

## SIMBOLIS

## TIESOS LENTELĖ

### NE (NOT)

NE vartų išvedinys (OUTPUT) yra priešingas įvediniui (INPUT).  
Jei įvedinys yra **0**, išvedinys – **1**.  
O jei įvedinys yra **1**, išvedinys – **0**.



INPUT		OUTPUT
0		1
1		0

### IR (AND)

IR vartų išvedinys yra **1** tik tada, kai abu įvediniai yra **1**.



INPUT		OUTPUT
A	B	A AND B
0	0	0
0	1	0
1	0	0
1	1	1

### ARBA (OR)

ARBA vartų išvedinys yra **1**, jei vienas iš įvedinių yra **1**.

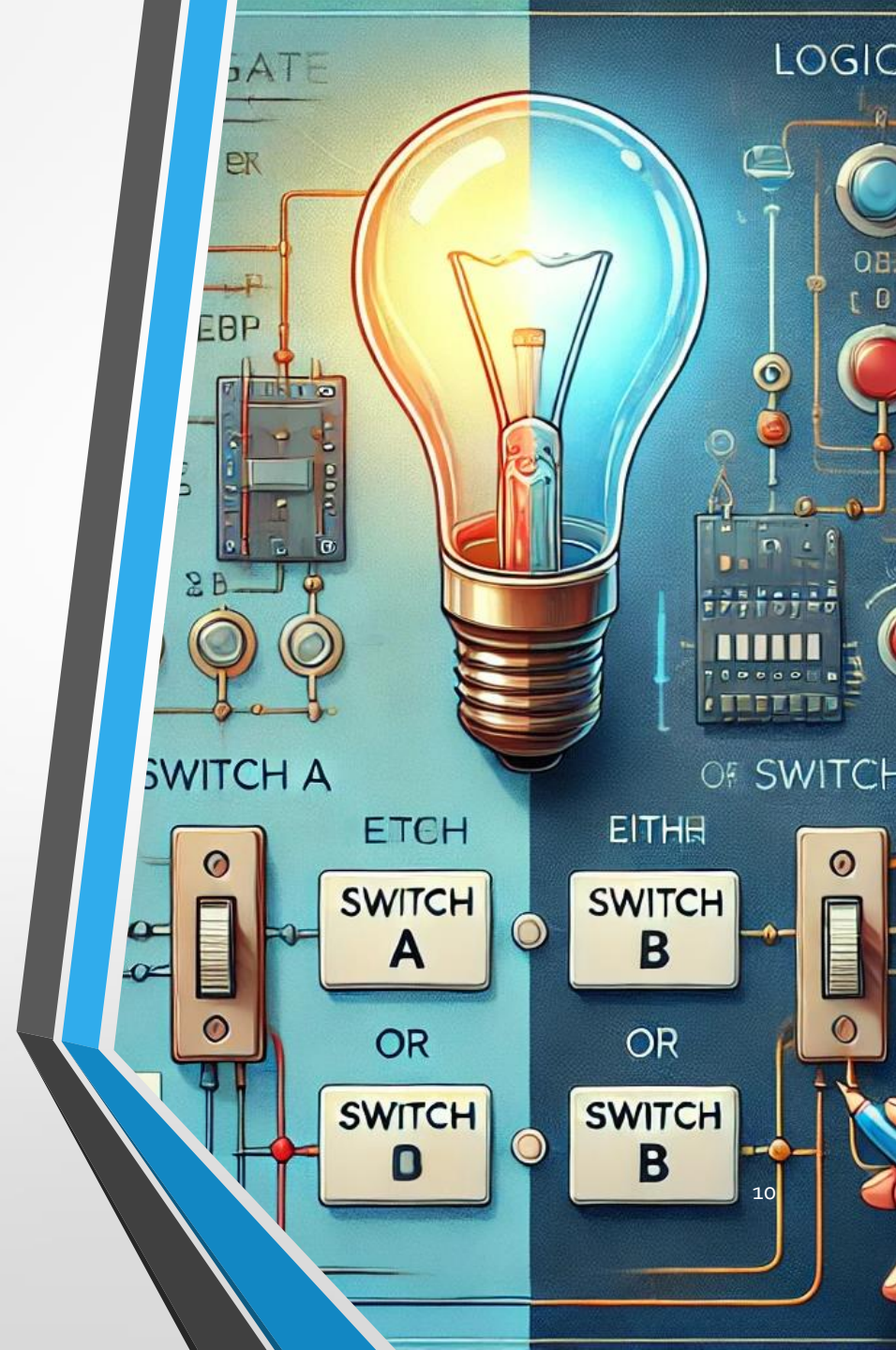


INPUT		OUTPUT
A	B	A OR B
0	0	0
0	1	1
1	0	1
1	1	1

# Вопрос к классу

Какой логический элемент вы бы использовали для управления светом, если он должен включаться при любом из двух условий?

<https://github.com/agorbatniov/inf10>

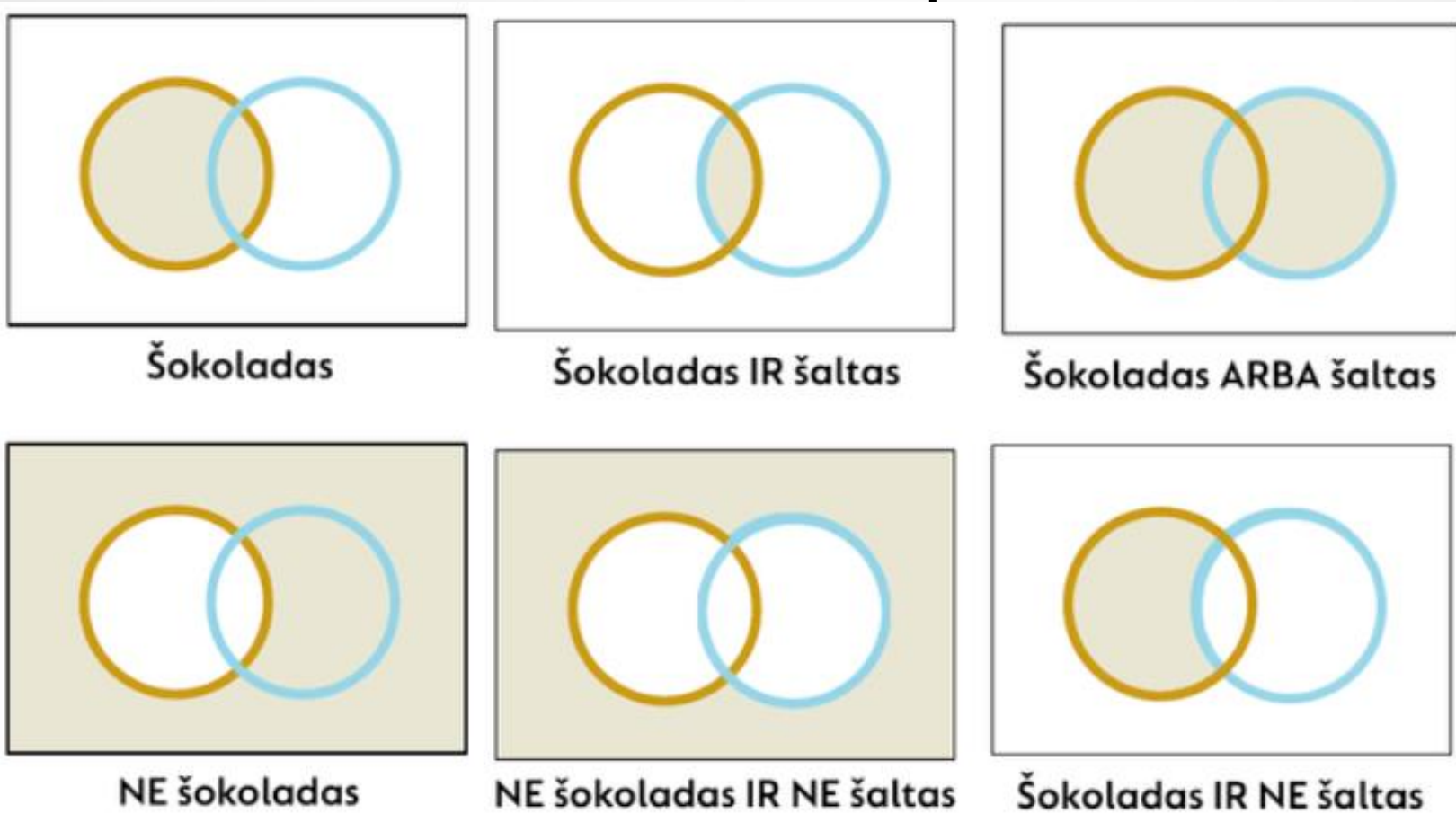


# Булева логика для логических операций

**Булева логика** — основа для понимания, как работают компьютеры и цифровые устройства.

**Булева логика** — это раздел математики и информатики, который изучает логические операции над булевыми переменными. Каждая переменная может принимать только два значения: **1** (истина) или **0** (ложь). Булева логика лежит в основе работы цифровых устройств и программного обеспечения

# Логический запрос



# Пример задачи

## Условие:

- Два датчика:
  - А (Движение)
  - В (Свет)
- Сигнал включается, если:
  - Есть движение и свет (**A AND B**).
  - Либо света нет, но движение есть (**A AND NOT B**).



# Решение через комбинацию

- Составим логическое выражение:

$(A \text{ AND } B) \text{ OR } (A \text{ AND NOT } B)$

- Таблица истинности:

A (Движение)	B (Свет)	NOT B	A AND B	A AND NOT B	(A AND B) OR (A AND NOT B)
0	0	1	0	0	0
0	1	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1

# Код для решения на Python

```
1  # Логическое выражение: (A AND B) OR (A AND NOT B)
2
3  # Входные данные: A - движение, B - свет
4  A = int(input("Движение (1 - есть, 0 - нет): "))
5  B = int(input("Свет (1 - есть, 0 - нет): "))
6
7  # Расчёт результата
8  result = (A and B) or (A and not B)
9
10 # Вывод результата
11 if result:
12     print("Сигнал включён")
13 else:
14     print("Сигнал выключен")
```

# Код для решения на C++

```
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      // Входные данные: A - движение, B - свет
6      int A, B;
7      cout << "Введите движение (1 - есть, 0 - нет): ";
8      cin >> A;
9      cout << "Введите свет (1 - есть, 0 - нет): ";
10     cin >> B;
11
12     // Расчёт результата
13     bool result = (A && B) || (A && !B);
14
15     // Вывод результата
16     if (result) {
17         cout << "Сигнал включён" << endl;
18     } else {
19         cout << "Сигнал выключен" << endl;
20     }
21
22     return 0;
23 }
```



# Что мы сегодня выучили

## 1. Основы логических элементов:

- NOT (НЕ), AND (И), OR (ИЛИ).
- Их символы и таблицы истинности.

## 2. Комбинации логических элементов:

- Как объединять логические элементы для решения сложных задач.
- Пример с движением и светом: логическое выражение и его реализация.

## 3. Практическое применение:

- Таблица истинности и схема для сложной логики.
- Программирование комбинаций на Python и C++.

# Рефлексия

- Почему логические элементы важны для компьютеров?
- Где вы можете применить эти знания в повседневной жизни?



Конец