**Урок информатики 8 класс**

**Тема: «Логические элементы и логические схемы»**

Тип урока: комбинированный

## Цели

- продолжить развитие навыков логического анализа;
- выработать умение корректно и логически безупречно формулировать определения понятий;
- продолжить решение логических задач с целью тренировки ума и его развития;
- научить выполнять вычисления по логическим схемам и строить логические выражения.

## Задачи

**Развивающие:**

- формировать развитие алгоритмического мышления;
- развивать мировоззрение (способствовать формированию взглядов на окружающий мир);
- продолжать способствовать развитию ИКТ-компетентности;
- уметь извлекать пользу из образовательного опыта, получать и обрабатывать информацию, использовать информационные технологии.

**Воспитательные:**

- продолжить формирование познавательного интереса к предмету информатика;
- воспитывать личностные качества: активность, самостоятельность, аккуратность в работе.

## План урока

1. Организационный момент (1 мин.).
2. Постановка цели и формулировка задач урока (2 мин.).
3. Проверка домашнего задания (игра «Верю — не верю» и «Найди ошибку») (10 мин.).
4. Изучение нового материала (12 мин.).
5. Закрепление изученного материала (работа в парах, «детектор ошибок») (10 мин.).
6. Рефлексия (2 мин.).
7. Домашнее задание (2 мин.).
8. Итог урока (1 мин.).

# Ход урока

## I. Организационный момент (1 мин.)

Здравствуйте, ребята, садитесь.

## II. Постановка цели и формулировка задач урока (2 мин.)

Сегодня мы завершим повторение формальной логики, но проверять знания будем не обычным опросом, а в форме игры. Затем познакомимся с тем, как из «кирпичиков» — логических элементов И, ИЛИ, НЕ — собираются схемы, по которым работает процессор компьютера, и научимся сами читать и строить такие схемы.

## III. Проверка домашнего задания (10 мин.)

*За каждое верное решение или аргументированный ответ — 1 балл.*

### Игра «Верю — не верю»

Учитель читает утверждение, класс сигнализирует: верите — поднимаете руку, не верите — не поднимаете. Двое учеников объясняют свой выбор у доски.

1. Высказывание — это предложение, которое может быть только истинным. (Не верим: оно может быть и истинным, и ложным.)
2. Основоположник алгебры логики — английский математик Джордж Буль. (Верим.)
3. Конъюнкция истинна, если хотя бы одно из исходных выражений истинно. (Не верим: это дизъюнкция; конъюнкция истинна, когда истинны оба.)
4. Инверсия меняет значение выражения на противоположное. (Верим.)
5. Импликация ложна только тогда, когда первое выражение истинно, а второе ложно. (Верим.)
6. В сложном логическом выражении первой выполняется дизъюнкция, а потом инверсия. (Не верим: порядок обратный — сначала инверсия, затем конъюнкция, дизъюнкция, импликация, эквивалентность.)
7. Эквивалентность истинна, когда оба выражения принимают одинаковые значения. (Верим.)
8. Рассуждение — это логическая операция, а не цепочка высказываний. (Не верим: рассуждение — цепочка высказываний или утверждений, связанных друг с другом.)

### Игра «Найди ошибку»

На слайде (или на доске) — таблица истинности для $F = A \wedge B$, в которой намеренно перепутаны две строки. Учащиеся находят и исправляют ошибку, объясняя, каким свойством конъюнкции они пользовались.

Затем аналогично проверяется таблица для $F = A \vee B$: ученики сравнивают её с конъюнкцией и формулируют правило «И — редко истина, ИЛИ — редко ложь».

Проверка наличия домашней работы в тетрадях проходит параллельно: учитель обходит класс, пока класс работает с таблицами.

*Если учащийся не получил ни одного балла — он не ответил ни на один вопрос. Делаем вывод, что домашнее задание не выполнено и тема не усвоена; таких учащихся необходимо контролировать.*

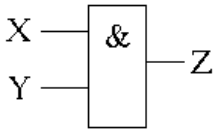
## IV. Изучение нового материала (12 мин.)

Как известно, любая информация при обработке на компьютере представляется в двоичной форме, то есть кодируется последовательностью 0 и 1. Поэтому упрощённо компьютер можно представить как устройство, обрабатывающее двоичные сигналы. Такую обработку выполняют логические элементы, из которых собираются логические схемы.

Логический элемент — это электронное устройство, реализующее одну из логических функций. Реализация любых логических операций над двоичными сигналами основана на элементах трёх типов: И, ИЛИ, НЕ.

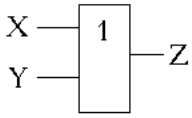
*Чтобы новые понятия «легли» легче, каждый элемент вводим через бытовую аналогию:*

- Элемент И — как охранный замок с двумя ключами: напряжение на выходе появится, только если оба замка (оба входа) открыты. Выполняет логическое умножение



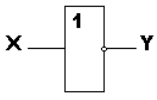

| X | Y | Z |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 1 | 1 |

- Элемент ИЛИ — как светильник с двумя выключателями в разных концах коридора: свет загорится, если нажат хотя бы один выключатель. Выполняет логическое сложение




| X | Y | Z |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 1 | 1 |

- Элемент НЕ — как отрицание: на выходе всегда сигнал, противоположный входному. Один вход и один выход.




| X | Y |
|---|---|
| 0 | 1 |
| 1 | 0 |

Далее разбираем условные обозначения элементов и их таблицы истинности (отсутствие сигнала — «0», наличие — «1»).

Обработка любой информации на компьютере сводится к выполнению процессором арифметических и логических операций. Для этого в состав процессора входит арифметико-логическое устройство (АЛУ), построенное на логических элементах. Важнейшие его узлы — регистры и сумматор.

- Регистр — электронный узел для хранения многоразрядного двоичного кода: кода команды или числа. Упрощённо регистр — совокупность ячеек, в каждой из которых записан один разряд: 0 или 1. Такая ячейка — триггер — является логической схемой из элементов И, ИЛИ, НЕ. Под воздействием входных сигналов триггер переходит в одно из двух устойчивых состояний и выдаёт 0 или 1. Для хранения одного байта нужно 8 триггеров.
- Сумматор — электронная схема для суммирования двоичных кодов. При сложении двух единиц по правилам двоичной арифметики получается 10 и происходит перенос 1 в старший разряд. Простейший одноразрядный сумматор строится из двух элементов И, одного элемента ИЛИ и одного элемента НЕ; у него три входа (два слагаемых и перенос из предыдущего разряда) и два выхода (сумма и перенос в следующий разряд). Многоразрядный сумматор собирается из одноразрядных.

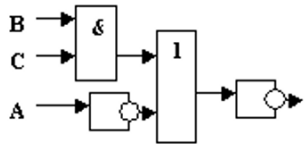
Вывод: логические элементы — строительные «кирпичики», из которых конструированием логических схем строится «здание» любого современного компьютера.

## V. Закрепление изученного материала (10 мин.)

### Карточка 1. «Детектор ошибок»

Каждая пара получает схему для выражения $F = (A \vee B) \wedge \neg C$, в которой элемент НЕ подключён не к тому входу (типичная ошибка).

Задание:
1) найдите ошибку и объясните, почему схема неверна;
2) перечертите правильную схему;
3) вычислите F при A = 1, B = 0, C = 1.
Разбор вместе с классом: сначала порядок операций — скобки, затем инверсия, конъюнкция; затем проверка вычисления: $1 \vee 0 = 1$; $1 \wedge (\neg 1) = 1 \wedge 0 = 0$.




### Карточка 2. Схема по выражению

Постройте логическую схему для выражения $F = \neg(A \wedge B) \vee C$ и вычислите её значение при A = 1, B = 1, C = 0.

Ход рассуждений: две переменные — A и B и C; три операции: 1-я — ∧, 2-я — ¬, 3-я — ∨; схему строим слева направо в порядке выполнения операций. Вычисление: $1 \wedge 1 = 1$; $\neg 1 = 0$; $0 \vee 0 = 0$.

Один из учеников воспроизводит решение на доске, класс проверяет себя.

## VI. Рефлексия (2 мин.)

Ребята, вернёмся к началу урока. Ответьте одним предложением: «Сегодня самым полезным для меня было…», а «самым трудным — …». (Учащиеся отвечают по одному с места.)

## VII. Домашнее задание (2 мин.)

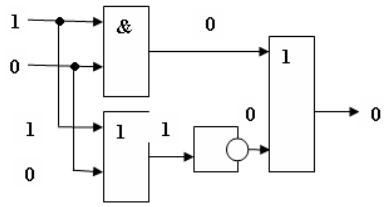
**1. Построить логические схемы по формулам и вычислить значения:**

F = A ∧ ¬B ∨ C, если A = 1, B = 1, C = 0;

F = ¬(A ∨ B) ∧ C, если A = 0, B = 1, C = 1;

F = (A ∨ C) ∧ ¬(B ∧ C), если A = 1, B = 0, C = 1.

**2. Составить логические выражения по схемам**




*3*. По желанию: придумать своё выражение с тремя переменными, построить для него схему и обменяться с соседом по парте на следующем уроке.*

## VIII. Итог урока (1 мин.)

Всем спасибо за урок, до свидания!

## Приложение. Ответы для учителя

*Игра «Найди ошибку»: в таблице F = A ∧ B перепутаны строки A = 0, B = 1 и A = 1, B = 0 — в обеих должно стоять 0, а не 1.*

*Карточка 1: F = (1 ∨ 0) ∧ (¬1) = 1 ∧ 0 = 0. Карточка 2: F = ¬(1 ∧ 1) ∨ 0 = ¬1 ∨ 0 = 0 ∨ 0 = 0.*

*Домашнее задание:*

1) F = 1 ∧ 0 ∨ 0 = 0 ∨ 0 = 0;

2) F = ¬(0 ∨ 1) ∧ 1 = ¬1 ∧ 1 = 0 ∧ 1 = 0;

3) F = (1 ∨ 1) ∧ ¬(0 ∧ 1) = 1 ∧ ¬0 = 1 ∧ 1 = 1.