

Арифметические и логические основы работы компьютера. Алгоритмы и способы их описания.

Автор:
*преподаватель информатики
Кизунова Анастасия Викторовна*

- Законспектируйте главные аспекты данной лекции.
- Выполните задания.

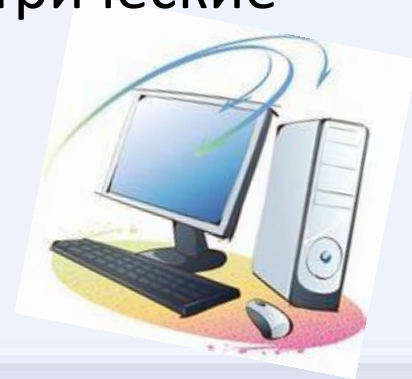
Компьютер и его функциональное устройство

Компьютер – это техническое средство преобразования информации, в основу работы которого заложены те же принципы обработки электрических сигналов, что и в любом электронном устройстве:

1. Входная информация, представленная различными физическими процессами, как электрической, так и неэлектрической природы (буквами, цифрами, звуковыми сигналами и т.д.), преобразуется в электрический сигнал;

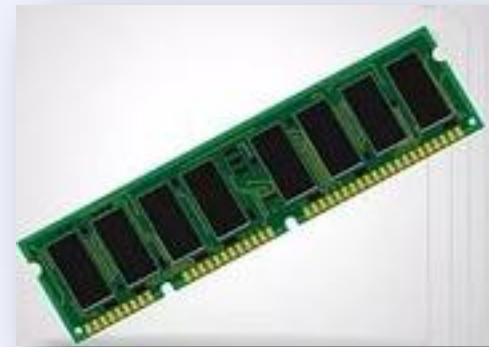
2. Сигналы обрабатываются в блоке обработки;

3. С помощью преобразователя выходных сигналов обработанные сигналы преобразуются в неэлектрические сигналы (изображения на экране).



С позиции функционального назначения компьютер – это система, состоящая из 4-х основных устройств, выполняющих определенные функции: запоминающего устройства или памяти, КОТОРАЯ разделяется на оперативную и постоянную, арифметико-логического устройства (АЛУ), устройства управления (УУ) и устройства ввода-вывода (УВВ).

Запоминающее устройство (память) предназначается для хранения информации и команд программы в ЭВМ. Информация, которая хранится в памяти, представляет собой закодированные с помощью 0 и 1 числа, символы, слова, команды, адреса и т.д.



Характеристики памяти :

- 1) емкость памяти – максимальное количество хранимой информации в байтах;
- 2) быстродействие памяти – время обращения к памяти, определяемое временем считывания или временем записи информации.

Виды памяти

Внутренняя

ОЗУ



ПЗУ



Внешняя

Диски



Флешки



Дискеты



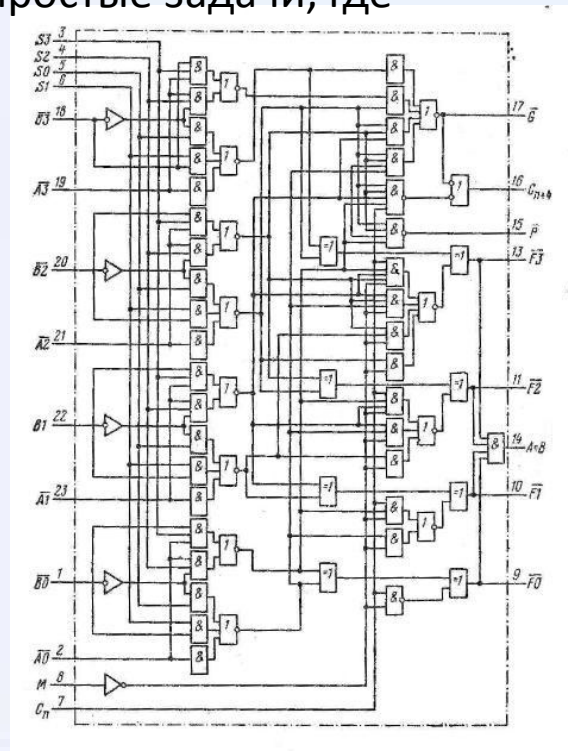
Магнитные ленты



Арифметико-логическое устройство (АЛУ). Производит арифметические и логические действия.

Следует отметить, что любую арифметическую операцию можно реализовать с использованием операции сложения.

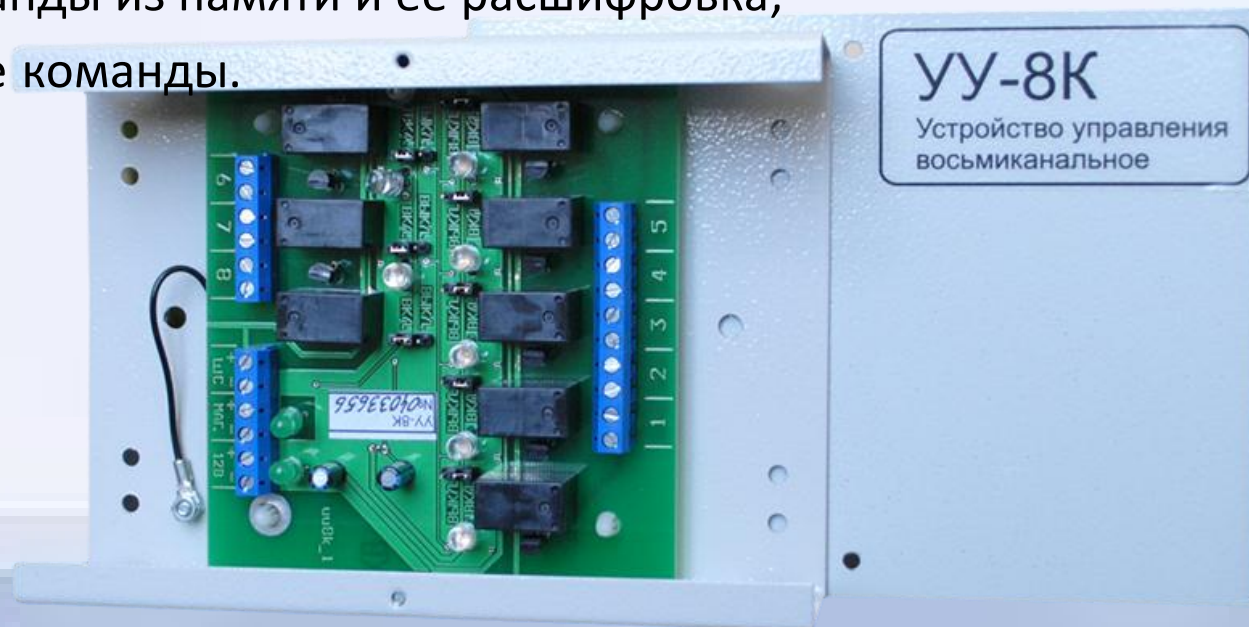
Сложная логическая задача раскладывается на более простые задачи, где достаточно анализировать только два уровня: ДА и НЕТ.



Устройство управления (УУ) управляет всем ходом вычислительного и логического процесса в компьютере, т.е. выполняет функции "регулирующего движения" информации. УУ читает команду, расшифровывает ее и подключает необходимые цепи для ее выполнения. Считывание следующей команды происходит автоматически.

Фактически УУ выполняет следующий цикл действий:

1. формирование адреса очередной команды;
2. чтение команды из памяти и ее расшифровка;
3. выполнение команды.



- В современных компьютерах функции УУ и АЛУ выполняет одно устройство, называемое центральным процессором.



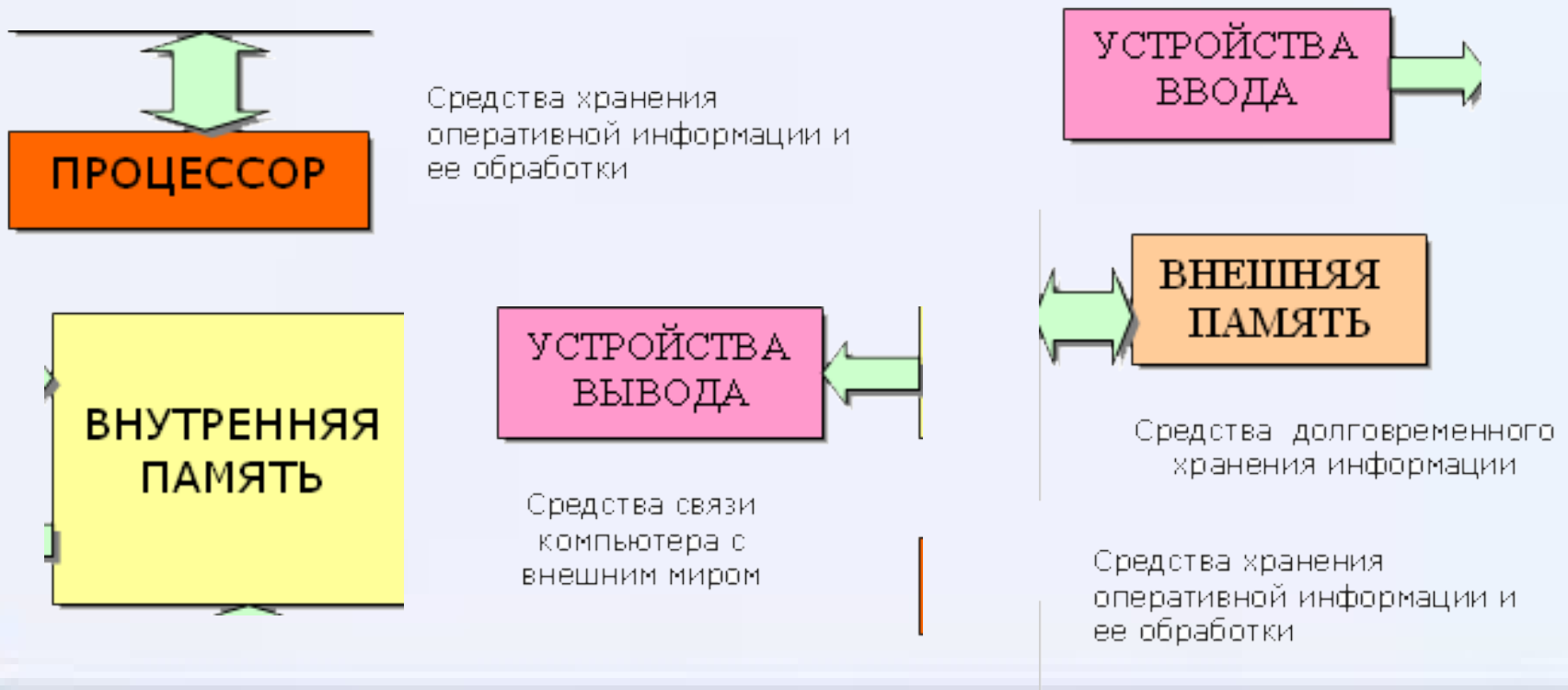
Устройства ввода и вывода - устройства взаимодействия компьютера с внешним миром: с пользователями или другими компьютерами.

Устройства ввода позволяют вводить информацию в компьютер для дальнейшего хранения и обработки.

Устройства вывода - получать информацию из компьютера.

Задание 1.

- Из данных блоков составьте
Функциональную схему компьютера



Решение



АРИФМЕТИЧЕСКИЕ ОСНОВЫ РАБОТЫ ЭВМ

Правила выполнения арифметических действий над двоичными числами задаются таблицами сложения, вычитания и умножения.

| Сложение | Вычитание | Умножение |
|-----------------|------------------|------------------|
| $0+0 = 0$ | $0 - 0 = 0$ | $0 \times 0 = 0$ |
| $0+1 = 1$ | $1 - 0 = 1$ | $0 \times 1 = 0$ |
| $1+0 = 1$ | $1 - 1 = 0$ | $1 \times 0 = 0$ |
| $1+1 = 10$ | $10 - 1 = 1$ | $1 \times 1 = 1$ |

В ВТ с целью упрощения реализации арифметических операций применяют специальные коды: *прямой, обратный, дополнительный*.

Прямой код

Прямой код складывается из знакового разряда (старшего) и собственно числа. Знаковый разряд имеет значение

0 – для положительных чисел;

1 – для отрицательных чисел.

Например: прямой код для чисел –4 и 5:

$$-4 \quad 4_{10} = 100_2 \quad 1_100$$

$$5 \quad 5_{10} = 101_2 \quad 0_101$$

Обратный код

Обратный код образуется из прямого кода заменой нулей - единицами, а единиц - нулями, кроме цифр знакового разряда. Для положительных чисел обратный код совпадает с прямым. Используется как промежуточное звено для получения дополнительного кода.

Например:

Прямой код 1_100 1_101

Обратный код 1_011 1_010

Дополнительный код

Дополнительный код образуется из обратного кода добавлением 1 к младшему разряду.

Например: найти дополнительный код -7_{10}

$$-7_{10} = 111_2$$

Прямой код 1_111

Обратный код 1_000

Дополнительный код :1_001 (1_000+1)

Правило сложения двоичных чисел:

При алгебраическом сложении двоичных чисел с использованием дополнительного кода положительные слагаемые представляют в прямом коде, а отрицательные – в дополнительном коде. Затем производят суммирование этих кодов, включая знаковые разряды, которые при этом рассматриваются как старшие разряды. При возникновении переноса из знакового разряда единицу переноса отбрасывают. В результате получают алгебраическую сумму в прямом коде, если эта сумма положительная, и в дополнительном коде, если сумма отрицательная.

Задание 2.

- Найдите дополнительный код для числа -12.

Задание 3.

Вычислите:

а) $1011_2 + 10001_2$;

б) $1100_2 - 1001_2$;

в) $110_2 * 11001_2$.

ЛОГИЧЕСКИЕ ОСНОВЫ РАБОТЫ ЭВМ

Алгебра логики

Для описания логики функционирования аппаратных и программных средств ЭВМ используется или, как ее часто называют, *булева алгебра* (по имени основоположника этого раздела математики – Дж. Буля).



Булева алгебра оперирует логическими переменными, которые могут принимать только два значения: *истина* или *ложь* (true или false), обозначаемые соответственно 1 и 0.

Логической функцией называется функция, которая может принимать только 2 значения – истина или ложь (1 или 0). Любая логическая функция может быть задана с помощью *таблицы истинности*. В левой ее части записываются возможные наборы аргументов, а в правой – соответствующие им значения функции.

Логическая операция ИНВЕРСИЯ (операция отрицания) – новое высказывание, которое ложно, когда высказывание истинно и истинно, когда само высказывание ложно.

Соответствует частице **НЕ**, обозначается: \bar{A} , $\neg A$

Таблица истинности

| A | $\neg A$ |
|---|----------|
| 0 | 1 |
| 1 | 0 |

Логическая операция **КОНЪЮНКЦИЯ**

Конъюнкция двух переменных истинна тогда и только тогда, когда оба высказывания истинны.

Соответствует союзу **И**, обозначается знаками $\&$, \wedge , $*$.

Таблица истинности

| A | B | A\wedgeB |
|----------|----------|------------------------------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Логическая операция ДИЗЬЮНКЦИЯ

Дизъюнкция двух переменных ложна тогда и только тогда, когда оба высказывания ложны.

Соответствует союзу **ИЛИ**, обозначается знаками \vee , $+$.

Таблица истинности

| A | B | $A \vee B$ |
|----------|----------|------------------------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Задание 4

Постройте таблицу истинности для логической формулы:

$$\bar{x} \vee (\bar{y} \wedge x) \wedge y$$

Задание 5

Для какого символьного выражения
ИСТИННО высказывание

\neg (Первая буква согласная) \wedge \neg (Вторая буква гласная)?

1) abcde 2) bcade 3) babas 4) cabab

Алгоритм

Алгоритм – система точных и понятных предписаний (команд, инструкций, директив) о содержании и последовательности выполнения конечного числа действий, необходимых для решения любой задачи данного типа.

В качестве **исполнителя** алгоритмов можно рассматривать человека, любые технические устройства, среди которых особое место занимает компьютер.

Система команд исполнителя (СКИ) – набор действий, которые может совершить исполнитель

Свойства алгоритма

- **Дискретность** (от лат. discretus – разделенный, прерывистый) указывает, что любой алгоритм должен состоять из конкретных действий, следующих в определенном порядке.
- **Детерминированность** (от лат. determinate – определенность, точность) указывает, что любое действие алгоритма должно быть строго и недвусмысленно определено в каждом случае.
- **Конечность** определяет, что каждое действие в отдельности и алгоритм в целом должны иметь возможность завершения.
- **Результативность** означает, при точном исполнении всех команд процесс решения задачи должен прекратиться за конечное число шагов и при этом должен быть получен определенный постановкой задачи результат (ответ).
- **Массовость**. Это свойство показывает, что один и тот же алгоритм можно использовать с разными исходными данными, т.е. применять при решении всего класса задач данного типа, отвечающих общей постановке задачи.

Типовые конструкции алгоритмов:

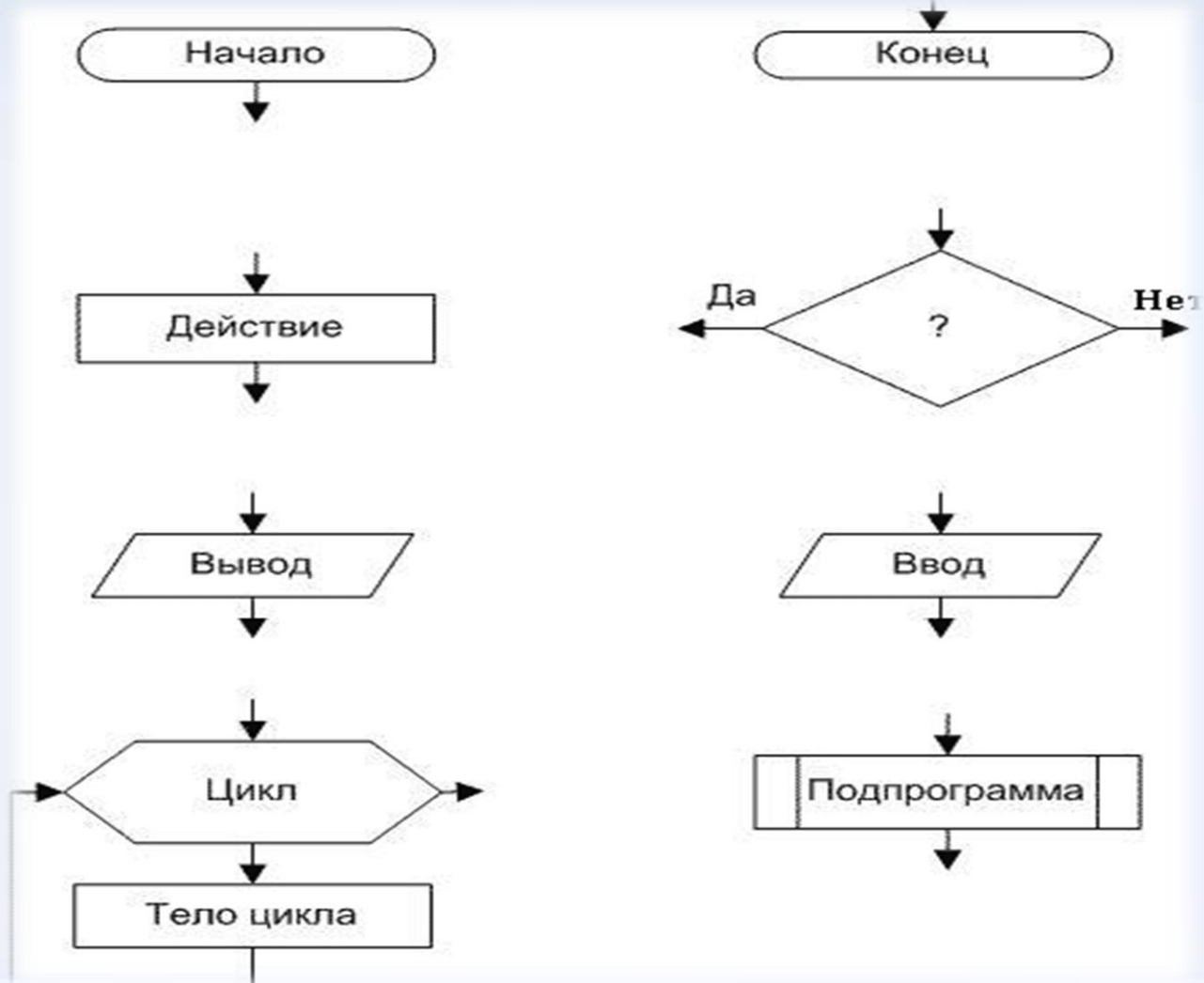
- Линейный.
- Циклический.
- Разветвляющийся.
- вспомогательный.

- **Линейный** (последовательный) алгоритм – описание действий, которые выполняются однократно в заданном порядке.
- **Циклический** – описание действий или группы действий, которые должны повторяться указанное число раз или пока не выполнено заданное условие. Совокупность повторяющихся действий – тело цикла.
- **Разветвляющийся** – алгоритм, в котором в зависимости от условия выполняется либо одна, либо другая последовательность действий. Условие – выражение, находящееся между словом «если» и словом «то» и принимающее значение «истина» (ветвь «да») или «ложь» (ветвь «нет»). Возможна полная и неполная форма ветвления.
- **Вспомогательный** – алгоритм, который можно использовать в других алгоритмах, указав только его имя. Вспомогательному алгоритму должно быть присвоено имя.

Способы описания алгоритмов.

- на естественном языке;
- на специальном (формальном) языке;
- с помощью формул, рисунков, таблиц;
- с помощью стандартных графических объектов (геометрических фигур) – блок-схемы.

Основные элементы блок-схемы

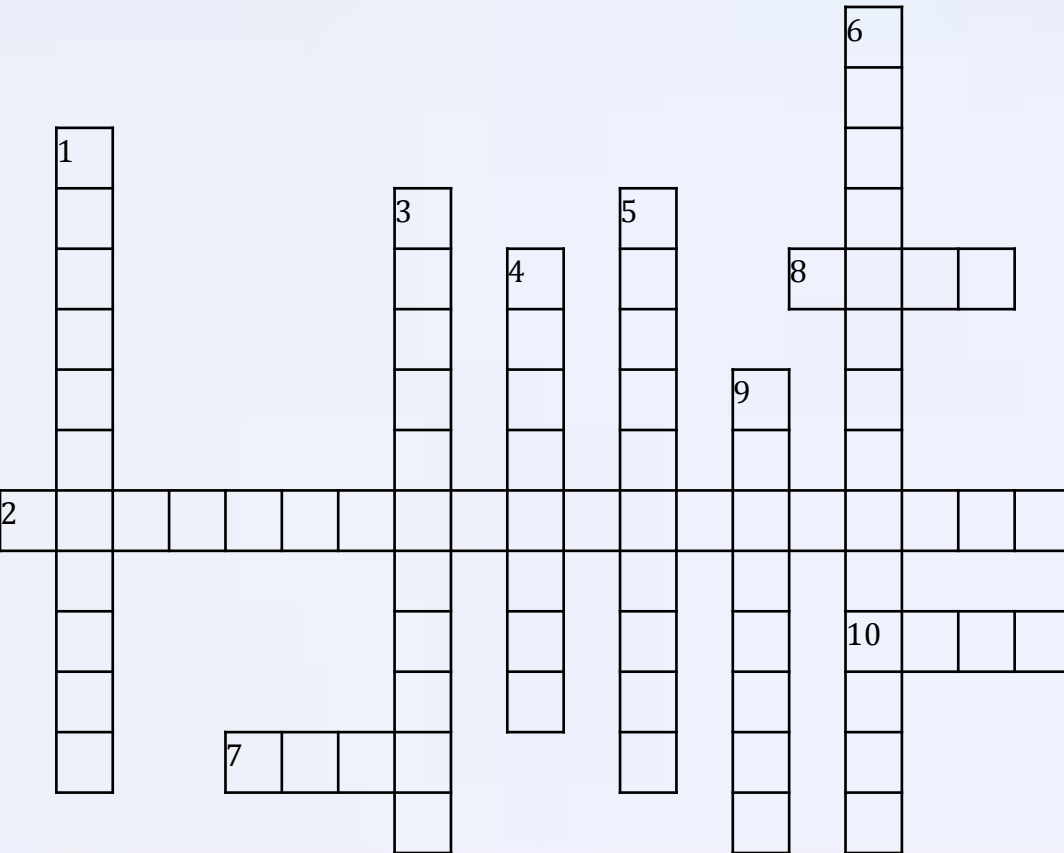


Задание 6

Составьте блок-схему для решения полного квадратного уравнения $ax^2+bx+c=0$ /

Задание 7

Разгадайте кроссворд



По горизонтали:

2. Свойство алгоритма, означающее однозначность действий.
7. Повторяющаяся последовательность действий.
8. Синоним слову алгоритм.
10. Фигура, в которой записывается условие в блок-схеме.

По вертикали:

1. Способ описания алгоритма.
3. Объект, умеющий выполнять определенный набор действий.
4. Строго определенная последовательность действий при решении задачи.
5. Свойство, показывающие, что алгоритм можно применять для решения класса задач.
6. Фигура ввода-вывода данных.
9. Алгоритм, действия в котором выполняются однократно в заданном порядке.