

ПОНЯТИЕ АЛГОРИТМА. СВОЙСТВА АЛГОРИТМА

Исполнитель алгоритма

Исполнитель алгоритма – это субъект или устройство, способные правильно интерпретировать описание алгоритма и выполнить содержащийся в нём перечень действий.



Исполнитель алгоритма

Неформальный исполнитель

- понимает смысл алгоритма, может его корректировать и изменять, а также отказаться выполнять
- одну и ту же команду выполняет каждый раз по-разному
- неформальный исполнитель сам отвечает за свои действия
- в роли неформального исполнителя чаще всего выступает человек

Формальный исполнитель

- не размышляет над выполняемыми командами, а строго следует пошаговым инструкциям алгоритма
- одну и ту же команду всегда выполняет одинаково
- за действия формального исполнителя отвечает управляющий им объект
- в роли формального исполнителя чаще всего выступает техническое устройство

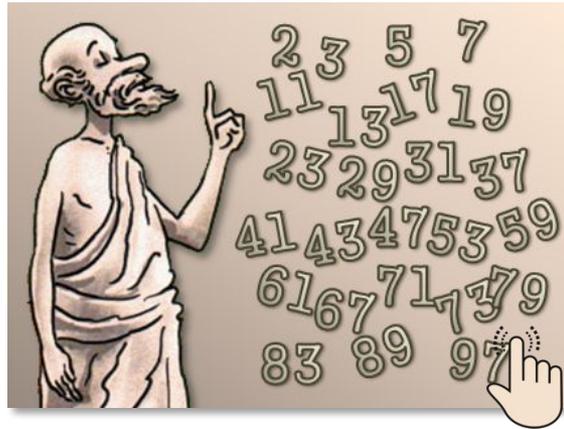
Понятие алгоритма

Алгоритм – точная система предписаний, определяющая содержание и порядок действий исполнителя над некоторыми объектами (исходными и промежуточными данными) для получения искомого результата за конечное число шагов.

ПРИМЕРЫ АЛГОРИТМОВ



Закреть
входную дверь
ключом



Нахождение n
первых простых
чисел
(метод Эратосфена)



Построение
перпендикуляра
к прямой

Пример 1



Алгоритм

«Закреть входную дверь ключом»

1. Вставить ключ в замочную скважину.
2. Повернуть ключ два раза на 180 градусов против часовой стрелки.
3. Вынуть ключ из замочной скважины.

Исполнитель: человек

Объекты алгоритма: ключ, дверь



Пример 2

Алгоритм «Нахождение всех простых чисел не больше заданного числа n по методу Эратосфена»

1. Выписать подряд все целые числа от 2 до n ($2, 3, 4, \dots, n$).
2. Присвоить переменной p значение 2 (2 – первое простое число).
3. Зачеркнуть в списке числа, кратные p : $2p, 3p, 4p, \dots$
4. Найти первое незачёркнутое число в списке, большее чем p , и присвоить p соответствующее значение.
5. Повторять шаги 3 и 4, пока возможно (пока $p^2 \leq n$).
6. Незачёркнутые числа и есть все простые числа от 2 до n .

Простые числа от 2 до 100



2 3 5 7
11 13 17 19
23 29 31 37
41 43 47 53 59
61 67 71 73 79
83 89 97

Выполнить



Пример 2

Алгоритм «Нахождение всех простых чисел не больше заданного числа n по методу Эратосфена»

1. Выписать подряд все целые числа от 2 до n ($2, 3, 4, \dots, n$).
2. Присвоить переменной p значение 2 (2 – первое простое число).
3. Удалить из списка числа, кратные p : $2p, 3p, 4p, \dots$
4. Найти первое число в списке, большее чем p , и присвоить p соответствующее значение.
5. Повторять шаги 3 и 4, пока возможно (пока $p^2 \leq n$).
6. Оставшиеся числа и есть все простые числа от 2 до n .

Простые числа от 2 до 100

	2	3		5		7			
11		13				17		19	
		23						29	
31						37			
41		43				47			
		53						59	
61						67			
71		73						79	
		83						89	
						97			

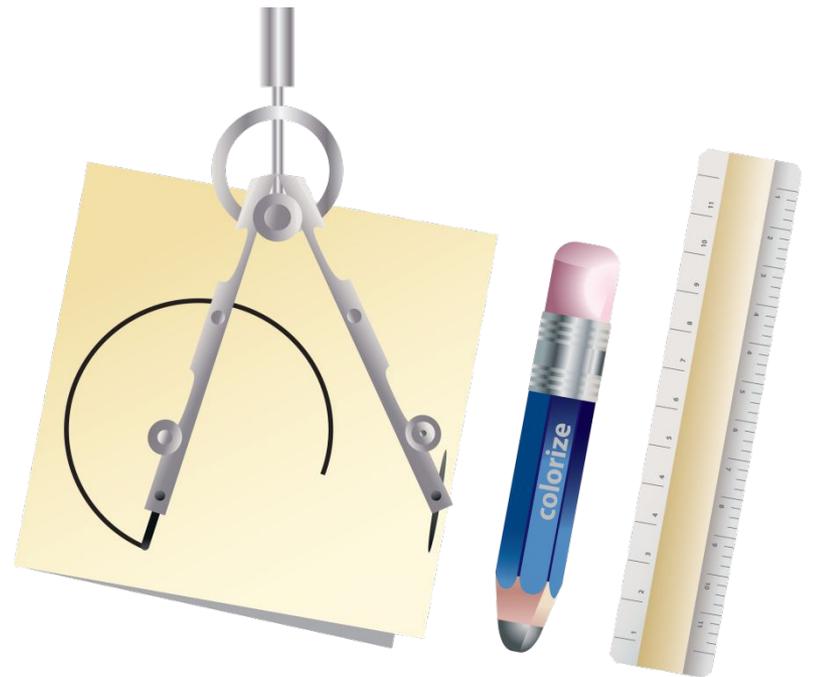
$p = 97$



Пример 3

Алгоритм «Построение перпендикуляра к прямой, проходящей через заданную точку O , лежащую на прямой с помощью циркуля и линейки»

1. Провести окружность с центром в точке O и радиусом 1 см.
2. Обозначить точки пересечения окружности с прямой: левую - A , правую - B .
3. Провести окружность с центром в точке A и радиусом равным AB .
4. Провести окружность с центром в точке B и радиусом равным AB .
5. Обозначить точки пересечения окружностей: верхнюю - C , нижнюю - D .
6. Провести прямую CD .



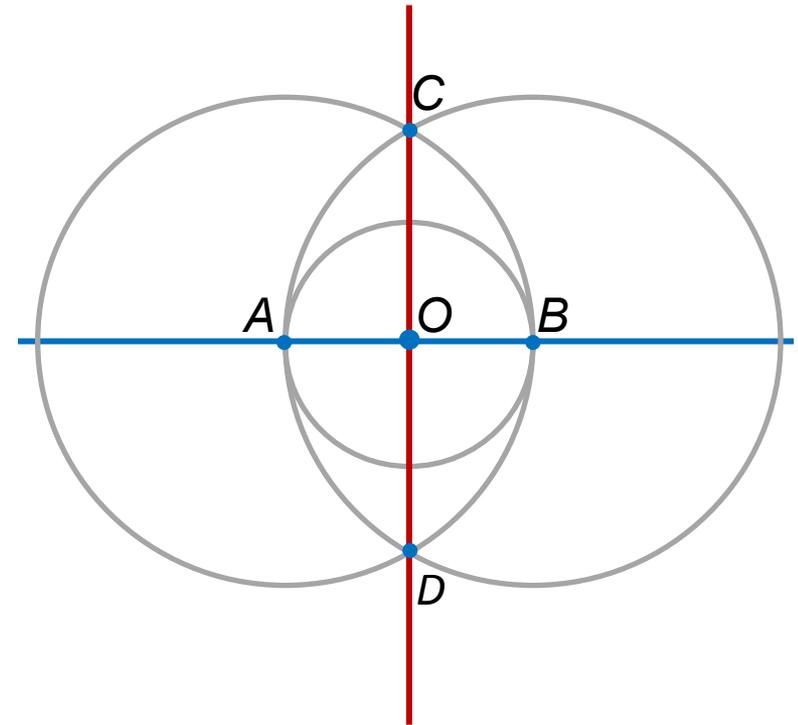
Выполнить



Пример 3

Алгоритм «Построение перпендикуляра к прямой, проходящей через заданную точку O , лежащую на прямой с помощью циркуля и линейки»

1. Провести окружность с центром в точке O и радиусом 1 см.
2. Обозначить точки пересечения окружности с прямой: левую - A , правую - B .
3. Провести окружность с центром в точке A и радиусом равным AB .
4. Провести окружность с центром в точке B и радиусом равным AB .
5. Обозначить точки пересечения окружностей: верхнюю - C , нижнюю - D .
6. Провести прямую CD .



Свойства алгоритма

Алгоритм – конечная **система правил**, сформулированных на **языке исполнителя**, которая определяет последовательность перехода от допустимых **исходных данных к конечному результату** и обладает свойствами дискретности, детерминированности, понятности, результативности, конечности и массовости.

Понятность

Дискретность

Результативность

Детерминированность

Массовость

Можно ли кулинарный рецепт считать алгоритмом?

Шампиньоны в сметане



Chef-daw

Способы записи алгоритмов

словесная запись
алгоритма
на естественном языке

запись алгоритма с
помощью
формул, рисунков,
таблиц

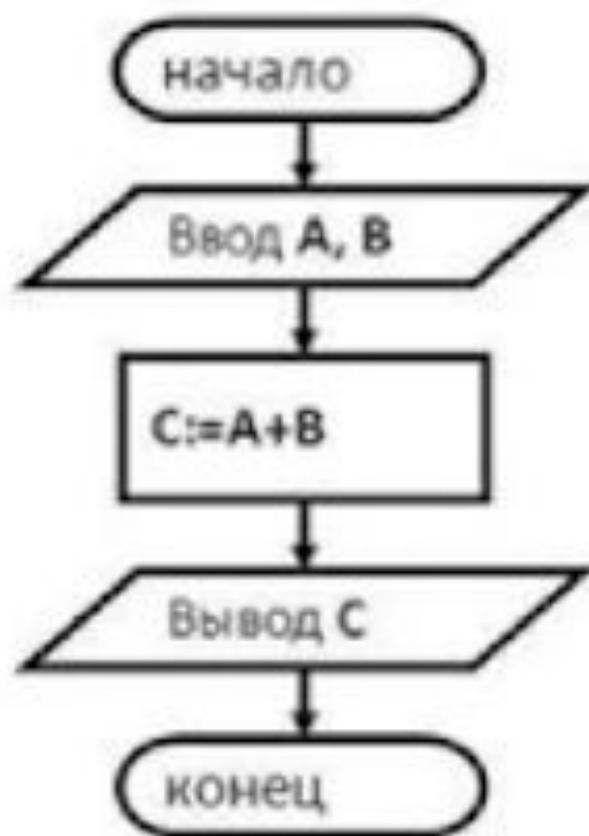
запись алгоритма на
языке
программирования

с помощью блок-схемы –
стандартных
графических объектов
(геометрических фигур)



Программа в Паскале

Блок-схема



```
Program summa;  
Var  
    a,b,sum : integer;  
  
Begin  
    readln(a);  
    readln(b);  
    sum := a + b;  
    writeln(sum)  
  
end.
```

Блок-схема

СИМВОЛ	ФУНКЦИЯ
	Пуск/остановка. Начало, конец, прерывание процесса обработки данных или выполнения программы
	Ввод/вывод. Преобразование данных в форму, пригодную для обработки (ввод) или отображения результатов (вывод)
	Процесс. Выполнение операций или группы операций, в результате которых изменяется значение, форма представления или расположение данных
	Решение. Выбор направления выполнения алгоритма или программы в зависимости от некоторых переменных условий
	Модификация. Выполнение операций, меняющих команды или группу команд, изменяющих программу
	Предопределённый процесс. Использование ранее созданных и отдельно описанных алгоритмов или программ

Понятие сложности алгоритма

Сложность алгоритма – количество элементарных шагов (действий) в вычислительном процессе этого алгоритма.

Лучшим среди них считается **алгоритм**, имеющий **наименьшую сложность**.

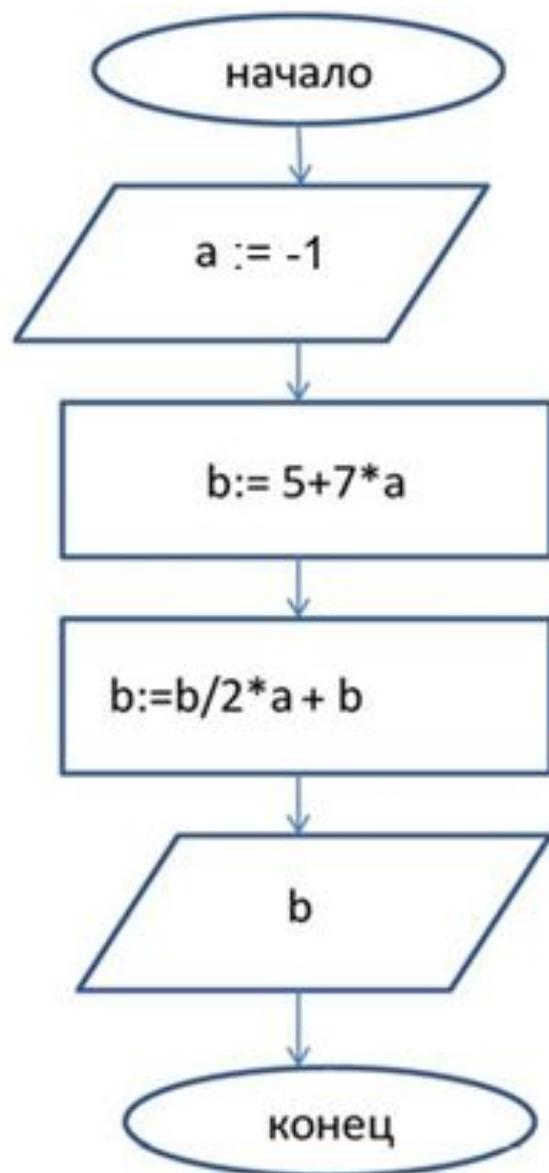
Эффективность

количество элементарных операций...

количество памяти...

Записать результат алгоритма (*подробно записать, чему будет равна переменная на каждом шаге*).

Составить программу на Паскале.



Временная сложность

Сложность алгоритма выражают в виде функции от объёма входных данных.

Задание. Оцените сложность алгоритмов:

«Найти книгу с секретом»

В италийской библиотеке в одном из 1000 томов проверены все вкладышки, но не найдено нужной сейфа. Надо найти в n количестве книг, т.е. $O(n) = 1000$.



«Поиск в телефонной книге»

В сейфе оказался клочок страницы с фамилией и первой цифрой номера телефона. Надо найти страницу с нужной фамилией в телефонном справочнике, в котором 1000 страниц.

В данном алгоритме, забудет $O(n^2)$ портировки имен по такому образцу, в книге объёмом в 1000 страниц половина нужного деления (открыл книгу примерно в середине, мы уменьшаем размер «оставшейся проблемы» вдвое).



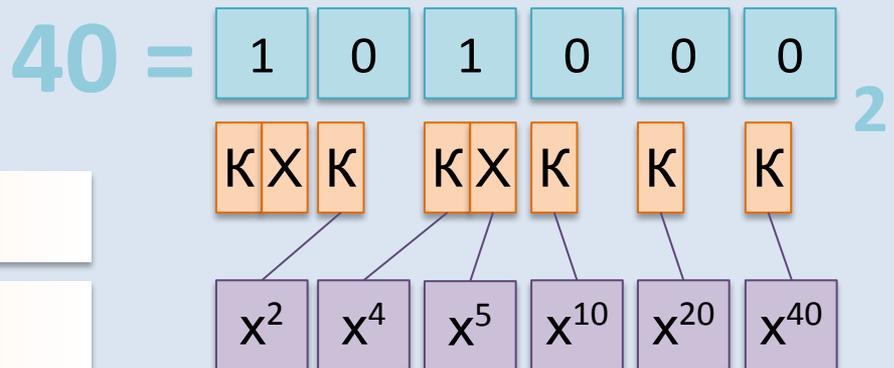
Пример 4

Алгоритм «Возведение числа в натуральную степень (x^n)»

1. Запишем n в двоичной системе счисления.
2. Заменяем каждую 1 парой букв KX , а каждый 0 – буквой K .
3. Вычеркнем крайнюю левую пару KX .
4. Полученная строка, читаемая слева направо, даёт правило быстрого вычисления x^n , если букву K рассматривать как операцию возведения результата в квадрат, а букву X – как операцию умножения результата на x . Вначале результат равен x .

Задание. Найти x^{40}

K	возведение результата в Квадрат
X	умножение результата на X



Алгоритм – конечная система правил, сформулированных на языке исполнителя, которая определяет последовательность перехода от допустимых исходных данных к конечному результату и обладает свойствами дискретности, детерминированности, понятности, результативности, конечности и массовости.

Исполнитель алгоритма – субъект или устройство, способные правильно интерпретировать описание алгоритма и выполнить содержащийся в нём перечень действий.

Один и тот же алгоритм может быть записан разными способами: на естественном языке, псевдокодом, с помощью блок-схем, на языке программирования и т. д.

Теория алгоритмов предоставляет аппарат анализа различных алгоритмов решения одной и той же задачи, на основе которого можно выбрать самый эффективный

Алгоритм состоит из команд. **Команда** – отдельная инструкция в описании алгоритма. **Шаг алгоритма** – отдельное действие, которое исполнитель выполняет по команде. Вычислительным процессом, порождённым алгоритмом, называется последовательность шагов алгоритма, пройденных при его исполнении.

Сложность алгоритма – количество элементарных шагов (действий) в вычислительном процессе этого алгоритма. Наряду со сложностью важной характеристикой алгоритма является эффективность. Эффективность оценивается количеством элементарных операций, которые необходимо выполнить для решения задачи, а также количеством памяти, требующейся для выполнения алгоритма.

Вопросы и задания

Задание 1. Автомат получает на вход трёхзначное число. По этому числу строится новое число по следующим правилам:

1. Складываются первая и вторая, а также вторая и третья цифры исходного числа.
2. Полученные два числа записываются друг за другом в порядке убывания (без разделителей).

Укажите *наименьшее* число, в результате обработки которого автомат выдаст число *1711*.

Решение:

1. Единственный способ разбить запись *1711* на два числа – это *17* и *11*.
2. Чтобы число было меньше, надо чтобы сумма первой и второй цифр была наименьшей, в данном случае *11*.
3. Сумма значений двух последних цифр равна *17*. Не трудно заметить, что $17 = 8 + 9 = 9 + 8$. Других вариантов нет.
4. Тогда $11 = 2 + 9 = 3 + 8$. Выбираем пару, которая даст *меньшее* число.

Вопросы и задания

Задание 2. Подсчитайте сложность алгоритма сложения двух натуральных чисел «столбиком» при условии, что одно из них состоит из n , а второе – из m десятичных цифр.

Решение:

Сложение двух чисел столбиком в случае, если одно из них состоит из n , а другое – из m цифр требует не более $\max(n, m)$ сложений и не более $\max(n, m)$ запоминаний (в случае перехода через десяток).

Т.е. данный алгоритм имеет сложность порядка $O(n+m)$.

Выражение показывает только порядок величины – постоянные факторы в нем не учитываются.

Вопросы и задания

Задание 1. Есть двое песочных часов: на 3 и на 8 минут. Для приготовления эликсира бессмертия его надо варить ровно 7 минут. Как это сделать? Придумайте систему команд исполнителя Колдун. Запишите с их помощью план действий исполнителя по приготовлению эликсира.

Графический способ решения:



Домашнее задание

§ 5.1. , стр. 64 - 76

-
- Можете ли вы назвать тему урока?
 - Вам было легко или были трудности?
 - Как бы вы оценили свою работу?

