**16.11.2021 Учебная группа 1ТО, 3-я пара**

**Преподаватель Иванова Наталия Викторовна**

**ОДП.03 Информатика и ИКТ**

**Тема:** Этапы решения задачи на компьютере. Исполнитель алгоритмов. Система команд исполнителя. Компьютер как исполнитель алгоритмов.

**Цель:**

***Образовательная:*** знакомство с исполнителем алгоритма, повторение свойств алгоритмов, получение представления о этапах решения задач; объяснение видов алгоритмов и базовых алгоритмических структур.

***развивающая:*** развитие аналитического критического мышления;

***Воспитательная:*** воспитание таких базовых качеств личности, как коммуникативность, самостоятельность, толерантность, ответственность за собственный выбор и результаты своей деятельности

**Задачи занятия:** дать понятие алгоритма, рассказать о свойствах, дать классификацию алгоритмов; познакомить с формой записи алгоритмов – блок-схема.

**Мотивация:** Каждый раз, когда мы сталкиваемся с решением поставленной перед нами мы проходим одни и те же этапы, которые в итоге должны привести нас к конкретно поставленной цели. Как быстро прийти к решению конкретной задачи мы сегодня познакомимся на занятии

**Задание студентам:**

1. Познакомиться с видеоуроком <https://youtu.be/se7nzN76bgw>
2. Познакомиться с презентацией <https://drive.google.com/file/d/1bt1r5AGo-RspxK4arAUkNUuJth6PXil2/view?usp=sharing>
3. Законспектировать лекцию. Ответить на вопросы

Материал соответствует пар.12. Учебник Семакин И.Г., Хеннер Е.К., Шейна Т.Ю. Информатика, учеб, 10 кл. базовый уровень/ И.Г. Семакин, Е.К. Хеннер, Т.Ю. Шейна изд.-М.: Бином. Лаборатория знаний, 2015.-264 с.

Фотографию с выполненным заданием прислать на электронный адрес **atata17@yandex.ru** в срок **до 08.00 17.11.2021** **г.**

**Лекция 23**

**План:**

1. Этапы решения задачи на компьютере
2. Понятие алгоритма
3. Данные и величины
4. Вопросы и задания

**Вопрос 1.** image **Этапы решения задачи на компьютере**

image**Работа по решению любой задачи с использованием компьютера делится на следующие этапы:**

**1.** Постановка задачи.

**2.** Формализация задачи.

**3.** Построение алгоритма.

**4.** Составление программы на языке программирования.

**5.** Отладка и тестирование программы.

**6.** Проведение расчетов и анализ полученных результатов.

Часто эту последовательность называют ***технологической цепочкой решения задачи на компьютере***. Непосредственно к программированию в этом списке относятся пункты **3, 4, 5**.

На **этапе постановки задачи** должно быть четко определено, ***что дано и что требуется найти***. Здесь очень важно определить полный набор исходных данных, необходимый для решения задачи.

**Второй этап** — формализация задачи. Здесь чаще всего задача переводится на язык математических формул, уравнений, отношений. Если решение задачи требует математического описания какого-то реального объекта, явления или процесса, то формализация равносильна получению соответствующей ***математической модели***.

**Третий этап** — построение алгоритма. Опытные программисты часто сразу пишут программы на языках программирования, не прибегая к каким-либо специальным способам описания алгоритмов (блок-схемам, псевдокодам). Однако в учебных целях полезно использовать эти средства, а затем переводить полученный алгоритм на язык программирования.

**Первые три этапа** — это работа без компьютера. Дальше следует собственно программирование на определенном языке в определенной системе программирования. Последний (шестой) этап — это уже использование разработанной программы в практических целях. Выполнение учебных заданий на программирование обычно заканчивается пятым этапом, т. е. доказательством правильности составленной программы.

**Таким образом, программист должен обладать следующими знаниями и навыками:**  
• уметь строить алгоритмы;   
• знать языки программирования;   
• уметь работать в соответствующей системе программирования.

Основой программистской грамотности является развитое алгоритмическое мышление.

Вопрос 2. Понятие алгоритма



image**Понятие алгоритма**

Одним из фундаментальных понятий в информатике является понятие алгоритма. Сам термин «алгоритм» пришел из математики. Это слово происходит от «Algorithmi» — латинского написания имени Мухамеда аль-Хорезми (787-850 гг.), выдающегося математика средневекового Востока. В XII веке был осуществлен латинский перевод его математического трактата, из которого европейцы узнали о десятичной позиционной системе счисления и правилах арифметики многозначных чисел. Именно эти правила в то время называли алгоритмами. Сложение, вычитание, умножение «столбиком», деление «уголком» многозначных чисел — вот первые алгоритмы в математике. Правила алгебраических преобразований, вычисление корней уравнений также можно отнести к математическим алгоритмам.

В наше время понятие алгоритма трактуется шире.

image**Алгоритм — это последовательность команд управления каким-либо исполнителем.**

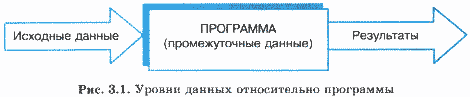
В школьном курсе информатики с понятием алгоритма, с методами построения алгоритмов ученики впервые знакомятся на примерах учебных исполнителей: Робота, Черепашки, Чертежника и др. В учебнике для 9 класса описан графический исполнитель — ГРИС. Эти исполнители ничего не вычисляют. Они создают рисунки на экране, перемещаются в лабиринтах, перетаскивают предметы с места на место. Таких исполнителей принято называть ***исполнителями, работающими в обстановке***.

В разделе информатики под названием «Программирование» изучаются методы программного управления работой компьютера. Следовательно, в качестве исполнителя выступает компьютер. Он работает с величинами — различными информационными объектами: числами, символами, кодами и пр. Поэтому алгоритмы, предназначенные для управления компьютером, принято называть ***алгоритмами работы с величинами***.

Вопрос 3 imageДанные и величины

image**Совокупность величин, с которыми работает компьютер, принято называть данными.**

По отношению к программе данные делятся на исходные, результаты (окончательные данные) и промежуточные данные, которые получаются в процессе вычислений (рис. 3.1).



Например, при решении квадратного уравнения: ах2 + bх + с = 0 исходными данными являются коэффициенты а, b, с; результатами — корни уравнения х1, х2; промежуточными данными — дискриминант уравнения: D = b2 - 4ас.

Для успешного освоения программирования необходимо усвоить следующее правило: ***всякая величина занимает свое определенное место в памяти компьютера***. Иногда говорят — ячейку памяти. Хотя термин «ячейка», с точки зрения архитектуры современных компьютеров, несколько устарел, однако в учебных целях его удобно использовать.

У всякой величины имеются три основных свойства: **имя, значение и тип**. На уровне команд процессора величина идентифицируется адресом ячейки памяти, в которой она хранится. В алгоритмах и языках программирования величины делятся на константы и переменные. Константа — неизменная величина, и в алгоритме она представляется собственным значением, например: 15, 34.7, 'k', true. Переменные величины могут изменять свои значения в ходе выполнения программы и представляются символическими именами — идентификаторами, например: X, S2, cod15. Любая константа или переменная занимают ячейку памяти, а значение этих величин определяется двоичным кодом в этой ячейке.

Теперь о типах величин — **типах данных**. С понятием типа данных вы уже встречались, изучая в курсе информатики основной школы электронные таблицы и базы данных. Это понятие является фундаментальным для программирования.

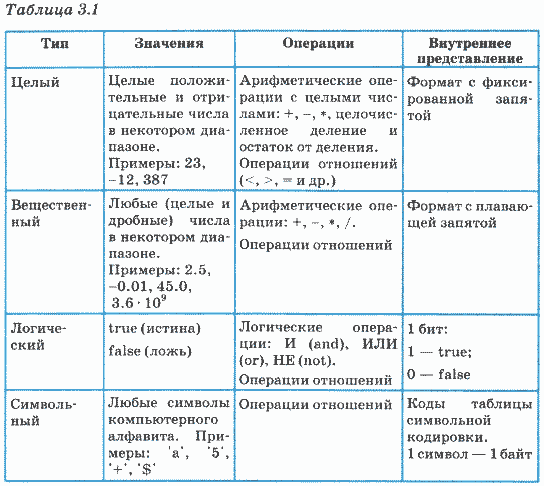
В каждом языке программирования существует своя концепция типов данных, своя система типов. Однако в любой язык входит минимально необходимый набор основных типов данных, к которому относятся целый, вещественный, логический и символьный типы. С типом величины связаны три ее свойства: множество допустимых значений, множество допустимых операций, форма внутреннего представления. В таблице 3.1 представлены эти свойства основных типов данных.

Типы констант определяются по контексту (т. е. по форме записи в тексте), а типы переменных устанавливаются в описаниях переменных.

Есть еще один вариант классификации данных: классификация по структуре. Данные делятся на **простые и структурированные**. Для простых величин (их еще называют скалярными) справедливо утверждение: одна величина — одно значение. Для структурированных: одна величина — множество значений.

К структурированным величинам относятся массивы, строки, множества и др.

image**Компьютер — исполнитель алгоритмов.**

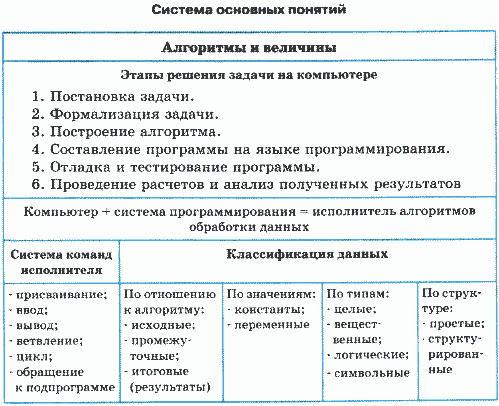


Как известно, всякий алгоритм (программа) составляется для конкретного исполнителя, в рамках его системы команд. О каком же исполнителе идет речь в теме «Программирование обработки информации»? Ответ очевиден: исполнителем является компьютер. Точнее говоря, исполнителем является комплекс: компьютер + система программирования (СП). Программист составляет программу на том языке, на который ориентирована СП. Схематически это изображено на рис. 3.2, где входным языком исполнителя является язык программирования Паскаль.



Независимо от того, на каком языке программирования будет написана программа, **алгоритм решения любой задачи на компьютере может быть составлен из команд:**  
• присваивания;   
• ввода;   
• вывода;   
• обращения к вспомогательному алгоритму (подпрограмме);   
• цикла;   
• ветвления.

Для описания алгоритмов в дальнейшем мы будем использовать блок-схемы и учебный Алгоритмический язык, применяемый в школьном курсе информатики.



Вопросы и задания

1. Перечислите и охарактеризуйте этапы решения задач на компьютере.

2. Дайте определение алгоритма.

3. Что такое «система команд исполнителя алгоритмов» (СКИ)?

4. Какими возможностями обладает компьютер как исполнитель алгоритмов?

5. Назовите команды, входящие в СКИ компьютера, из которых составляется любая программа обработки данных.

6. Перечислите различные варианты классификации данных.

7. Придумайте пример задачи, решаемой на компьютере, и назовите для нее исходные, промежуточные и итоговые данные.