**Тема: Примеры построения алгоритмов и их реализация на компьютере.**

**Основные алгоритмические конструкции и их описание средствами языков программирования.**

**Задание: выполните конспект в тетради, выполните задания (схемы рисуются в тетради при помощи линейки и простого карандаша, подписи ручкой или сделайте схемы на ПК, пришлите фото), ответьте на контрольные вопросы (в тетради или на ПК), пришлите фото и файлы работ.**

**Цели:**

* Научиться составлять алгоритмы с использованием различных структур

**Оборудование**: ПК

**Теоретический материал:**

АЛГОРИТМ - это последовательность команд, ведущих к какой-либо цели.

Это строго определенная процедура, гарантирующая получение результата за конечное число шагов. Это правило, указывающее действия, в результате цепочки которых происходит переход от исходных данных к искомому результату. Указанная цепочка действий называется алгоритмическим процессом, а каждое отдельное действие - его шагом. Пример: площадь прямоугольника S=a · b.

Виды алгоритмов: вычислительные, диалоговые, графические, обработки

данных, управления объектами и процессами и др.

Свойства алгоритмов - однозначность (и определенность), результативность (и выполнимость), правильность (и понятность), массовость или универсальность (т.е. применимость для целого класса задач, к различным наборам исходных данных).

Способы записи алгоритмов:

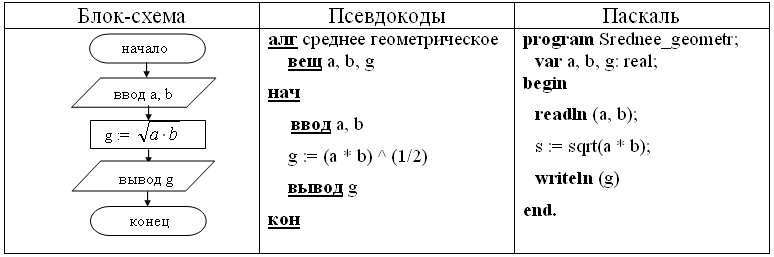
В виде блок-схем,в виде программ, в виде текстовых описаний (рецепты, например, рецепты приготовления пищи, лекарств и др.).

Наиболее понятно структуру алгоритма можно представить с помощью блок-схемы, в которой используются геометрические фигуры (блоки), соединенные между собой стрелками, указывающими последовательность выполнения действий. Приняты определенные стандарты графических изображений блоков. Например, команду обработки информации помещают в блок, имеющий вид прямоугольника, проверку условий - в ромб, команды ввода или вывода - в параллелограмм, а овалом обозначают начало и конец алгоритма.  
     Структурной элементарной единицей алгоритма является простая команда, обозначающая один элементарный шаг переработки или отображения информации. Простая команда на языке схем изображается в виде функционального блока.

|  |  |
| --- | --- |
| f63 | Данный блок имеет *один вход* и *один выход*. Из простых команд и проверки условий образуются составные команды, имеющие более сложную структуру и тоже *один вход и один выход*.       Структурный подход к разработке алгоритмов определяет использование только базовых алгоритмических структур (конструкций): следование, ветвление, повторение, которые должны быть оформлены стандартным образом. |
| f64 | Рассмотрим основные структуры алгоритма.      Команда *следования* состоит только из простых команд. На рисунке простые команды имеют условное обозначение *S1* и *S2*. Из команд следования образуются линейные алгоритмы. Примером линейного алгоритма будет нахождение суммы двух чисел, введенных с клавиатуры. |
| f65 | Команда *ветвления* - это составная команда алгоритма, в которой в зависимости от условия Р выполняется или одно *S1*, или другое *S2* действие. Из команд следования и команд ветвления составляются разветвляющиеся алгоритмы (алгоритмы ветвления). Примером разветвляющегося алгоритма будет нахождение большего из двух чисел, введенных с клавиатуры. |
| f66 | Команда ветвления может быть полной и неполной формы. Неполная форма команды ветвления используется тогда, когда необходимо выполнять действие *S* только в случае соблюдения условия *P*. Если условие*P* не соблюдается, то команда ветвления завершает свою работу без выполнения действия. Примером команды ветвления неполной формы будет уменьшение в два раза только четного числа. |
| f67 | Команда *повторения* - это составная команда алгоритма, в которой в зависимости от условия *Р* возможно многократное выполнение действия *S*. Из команд следования и команд повторения составляются циклические алгоритмы (алгоритмы повторения). На рисунке представлена команда повторения с предусловием. Называется она так потому, что вначале проверяется условие, а уже затем выполняется действие. Причем действие выполняется, пока условие соблюдается. Пример циклического алгоритма может быть следующий. Пока с клавиатуры вводятся положительные числа, алгоритм выполняет нахождение их суммы.       Команда повторения с предусловием не является единственно возможной. Разновидностью команды повторения с предусловием является команда повторения с параметром. Она используется тогда, когда известно количество повторений действия. В блок-схеме команды повторения с параметром условие записывается не в ромбе, а в шестиугольнике. Примером циклического алгоритма с параметром будет нахождение суммы первых 20 натуральных чисел. |
| f68 | В команде повторения с постусловием вначале выполняется действие *S* и лишь затем, проверяется условие *P*. Причем действие повторяется до тех пор, пока условие не соблюдается. Примером команды повторения с постусловием будет уменьшение положительного числа до тех пор, пока оно неотрицательное. Как только число становится отрицательным, команда повторения заканчивает свою работу.      С помощью соединения только этих элементарных конструкций (последовательно или вложением) можно "собрать" алгоритм любой степени сложности. |

***Линейный алгоритм***

Приведем пример записи алгоритма в виде блок-схемы, псевдокодов и на языке Паскаль. Ручное тестирование и подбор системы тестов выполняются аналогично предыдущему заданию.

******

**Задания.**

**Задание 1.**Составить блок -схему к задаче.

Лена попросила Сашу задумать двузначное число и, если задуманное число четное, то разделить его на 2 и назвать результат. Если задуманное число нечетное, то просто назвать число.

**Задание 2.** Построить линейный алгоритм вычисления значения У по формуле

У=(7Х+4)(2Х-2) при Х=3.

Составьте алгоритм самостоятельно, выделяя каждое действие как отдельный шаг.

**Задание 3. Решение квадратного уравнения**

Квадратное уравнение имеет вид ax2+ bx + c = 0, где а ≠ 0. Требуется найти корни этого уравнения.Cоставьте алгоритм решения квадратного уравнения в словесной форме

**Контрольные вопросы:**

1. Что такое алгоритм?
2. Какие способы записи алгоритмов вы знаете?
3. Какие свойства алгоритмов Вам известны?
4. Составьте алгоритм приготовления любого блюда?
5. Постройте блок-схему на составленный алгоритм?
6. В какой форме записываются алгоритмы?