

Основные алгоритмические конструкции и их описание средствами языков программирования.

Цели:

образовательные: развитие знаний по составлению алгоритмов с использованием различных структур

развивающая: развитие познавательного интереса, логического мышления, речи и внимания учащихся, формирование информационной культуры и потребности приобретения знаний;

воспитательная: привитие учащимся навыка самостоятельности в работе, воспитание трудолюбия, эстетического отношения к результатам своего труда.

Теоретический материал:

Алгоритм – это точная конечная система правил, определяющая содержание и порядок действий исполнителя над некоторыми объектами (исходными и промежуточными данными) для получения после конечного числа шагов искомого результата.

Алгоритм может быть описан одним из трех способов:

- словесным (пример в начале раздела);
- графическим (виде специальной блок-схемы);
- с помощью специальных языков программирования.

Блок-схема – распространенный тип схем, описывающий алгоритмы или процессы, изображая шаги в виде блоков различной формы, соединенных между собой стрелками.

1. **Линейный алгоритм** – это такой алгоритм, в котором все операции выполняются последовательно одна за другой.

2. **Алгоритмы разветвленной структуры** применяются, когда в зависимости от некоторого условия необходимо выполнить либо одно, либо другое действие.

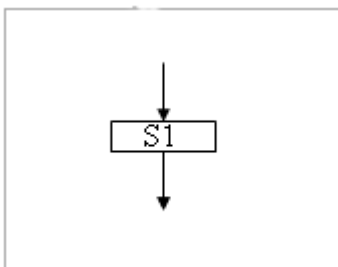
3. **Алгоритмы циклической структуры.**

Циклом называют повторение одних и тех же действий (шагов). Последовательность действий, которые повторяются в цикле, называют **телом цикла**.

Циклические алгоритмы подразделяют на алгоритмы с предусловием, постусловием и алгоритмы с конечным числом повторов. В алгоритмах с предусловием сначала выполняется проверка условия окончания цикла и затем, в зависимости от результата проверки, выполняется (или не выполняется) так называемое тело цикла.

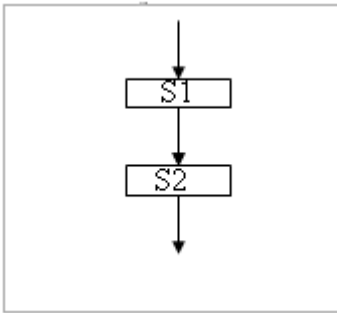
Наиболее понятно структуру алгоритма можно представить с помощью блок-схемы, в которой используются геометрические фигуры (блоки), соединенные между собой стрелками, указывающими последовательность выполнения действий. Приняты определенные стандарты графических изображений блоков. Например, команду обработки 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.

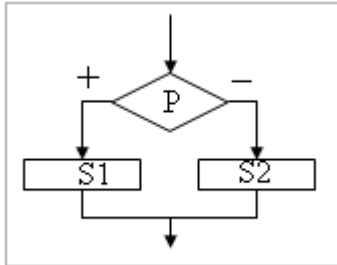


Данный блок имеет *один вход* и *один выход*. Из простых команд и проверки условий образуются составные команды, имеющие более сложную структуру и тоже *один вход* и *один выход*.

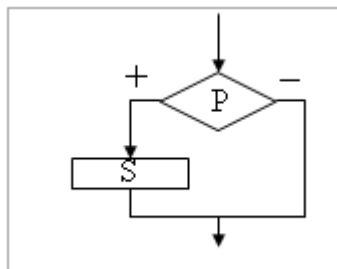
Структурный подход к разработке алгоритмов определяет использование только базовых алгоритмических структур (конструкций): следование, ветвление, повторение, которые должны быть оформлены стандартным образом.



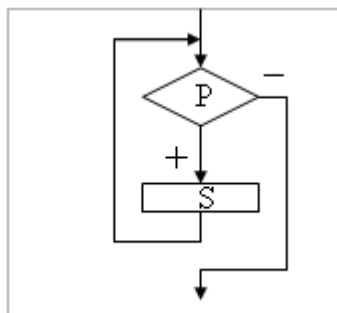
Рассмотрим основные структуры алгоритма. Команда *следования* состоит только из простых команд. На рисунке простые команды имеют условное обозначение $S1$ и $S2$. Из команд следования образуются линейные алгоритмы. Примером линейного алгоритма будет нахождение суммы двух чисел, введенных с клавиатуры.



Команда *ветвления* - это составная команда алгоритма, в которой в зависимости от условия P 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.

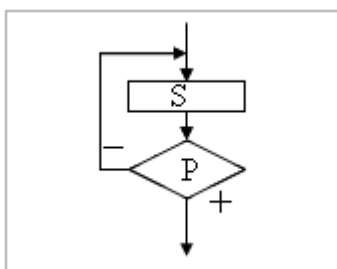


Команда ветвления может быть полной и неполной формы. Неполная форма команды ветвления используется тогда, когда необходимо выполнять действие S только в случае соблюдения условия P . Если условие P не соблюдается, то команда ветвления завершает свою работу без выполнения действия. Примером команды ветвления неполной формы будет уменьшение в два раза только четного числа.



Команда *повторения* - это составная команда алгоритма, в которой в зависимости от условия P 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.



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.

| Блок-схема | Псевдокоды |
|--|---|
|  <pre>graph TD; Start([начало]) --> Input[/ввод a, b/]; Input --> Process[g := √(a * b)]; Process --> Output[/вывод g/]; Output --> End([конец]);</pre> | <p>алг среднее геометрическое</p> <p>вещ a, b, g</p> <p>нач</p> <p>ввод a, b</p> <p>$g := (a * b) ^ (1/2)$</p> <p>вывод g</p> <p>кон</p> |

1. Построить линейный алгоритм вычисления значения Y по формуле $Y=(7X+4)(2X-2)$ при $X=3$. Составьте алгоритм самостоятельно, выделяя каждое действие как отдельный шаг.
2. В какой форме записываются алгоритмы?

Напишите вывод.

Литература

1. Информатика и ИКТ: учебник для начального и среднего профессионального образования. Цветкова Н.С., Великович Л.С.