

УВАЖАЕМЫЕ СТУДЕНТЫ!
ВАМ НЕОБХОДИМО ВЫПОЛНИТЬ СЛЕДУЮЩЕЕ:

1. Ознакомиться с теорией и законспектировать лекцию не меньше трех листов, составить и ответить на вопросы.
2. Предоставит отчет конспекта лекции прислать в виде скриншота в течении трех дней .
3. Отправить преподавателю на почту v.vika2014@mail.ru и указать свою Ф.И.О, группу, и название дисциплины

Тема: Работа с линейными алгоритмами.

Линейная структура

Любой алгоритм составляется из ряда базовых структур. Простейшей базовой структурой является следование — структура с линейными характеристиками. Из этого можно сформулировать определение.

Линейный алгоритм — это алгоритм, образуемый командами, которые выполняются однократно и именно в той последовательности, в которой записаны. Линейная структура, по сути, проста. Записать её можно как в текстовой, так и в графической форме.

Представим, что у нас стоит задача пропылесосить ковёр в комнате. В текстовой форме алгоритм будет следующим: — принести пылесос к месту уборки; — включить; — пропылесосить; — выключить; — унести пылесос.

И каждый раз, когда нам надо будет пылесосить, мы будем выполнять один и тот же алгоритм.

И каждый раз, когда нам надо будет пылесосить, мы будем выполнять один и тот же алгоритм.

Теперь поговорим про графическую форму представления.

Блок-схема

Для изображения алгоритма графически используют блок-схемы. Они представляют собой геометрические фигуры (блоки), соединённые стрелками. Стрелки показывают связь между этапами и последовательность их выполнения. Каждый блок сопровождается надписью.

Рассмотрим фигуры, которые используются при визуализации типичной линейной последовательности.

Блок начала-конца:



Блок ввода-вывода данных (отображает список вводимых и выводимых переменных):



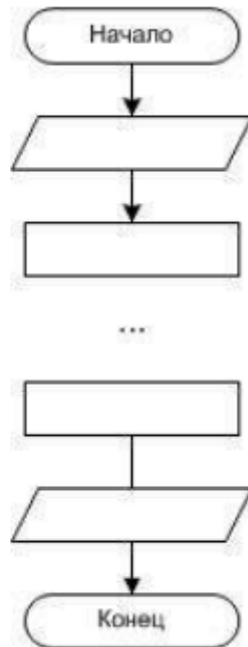
Арифметический блок (отображает арифметическую операцию/группу операций):



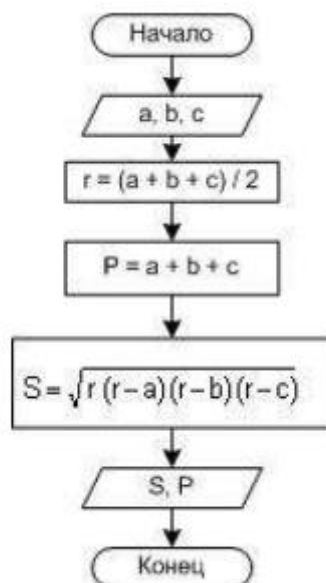
Условный блок (позволяет описать условие). Алгоритмы с таким блоком используются при графической визуализации алгоритмов с ветвлением:



Условного блока нет в классическом линейном алгоритме, так как в нём, как уже было сказано ранее, все операции выполняются последовательно, то есть одна за другой. В линейном алгоритме размещение блоков выглядит следующим образом:



А вот, как решается задача по нахождению площади треугольника по формуле Герона. Здесь a , b , c – это длины сторон, S – площадь треугольника, P – периметр.



Следует обратить внимание, что запись « $=$ » — это не математическое равенство, а **операция присваивания**. В результате этой операции переменная, стоящая слева от оператора, получает значение, которое указано

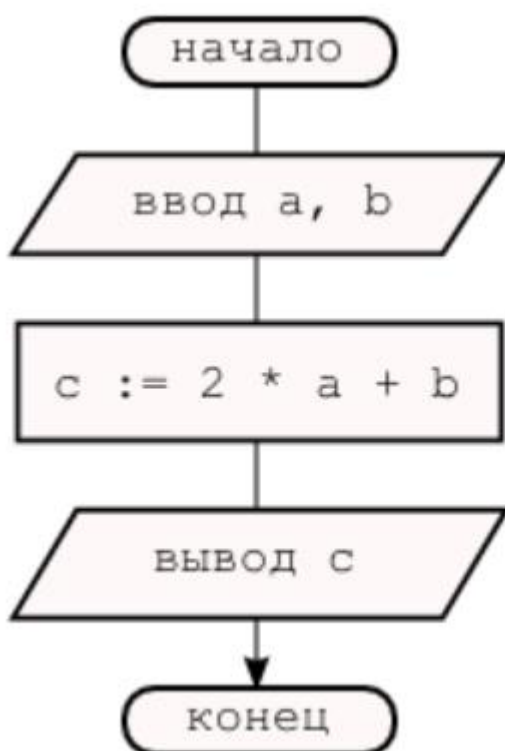
справа. Значение не обязательно должно быть сразу определено ($a = 3$) — оно может вычисляться посредством выражения ($a = b + z$), где $b = 1$, $a z = 2$.

Примеры линейных алгоритмов

Если рассмотреть примеры решения на языке Pascal (именно этот язык до сих пор используется для изучения основ алгоритмизации и программирования), то можно увидеть следующую картину:

| Программа на языке Pascal | Элементы программы |
|------------------------------------|----------------------------|
| <code>program line;</code> | 1. Заголовок |
| <code>var a, b, c: integer;</code> | 2. Объявление переменных |
| <code>begin</code> | 3. Начало блока операторов |
| <code> readln(a, b);</code> | 4. Ввод исходных данных |
| <code> c := 2 * a + b;</code> | 5. Вычисление по формуле |
| <code> writeln('c=', c);</code> | 6. Вывод результата |
| <code>end.</code> | 7. Конец блока операторов |

И, соответственно, блок-схема программы линейной структуры будет выглядеть следующим образом:



Как составить программу линейной структуры?

Порядок следующий: — определите, что именно относится к исходным данными, а также каков типы/класс этих данных, выберите имена переменных; — определите, каков тип данных будет у искомого результата, выберите название переменных (переменной); — определите, какие математические формулы связывают результат и исходные данные; — если требуется наличие промежуточных данных, определите класс/типы этих данных и выберите имена; — опишите все используемые переменные; — запишите окончательный алгоритм. Он должен включать в себя ввод данных, вычисления, вывод результатов.