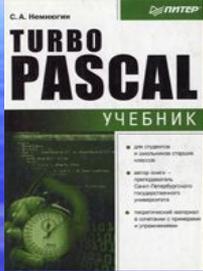
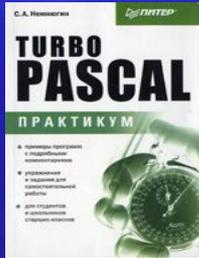


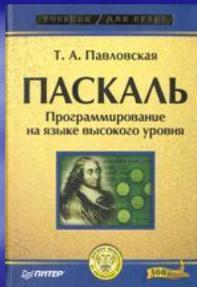
Рекомендуемая литература



TURBO PASCAL. Учебник
Немнюгин С.А.



TURBO PASCAL. Практикум
Немнюгин С.А.



ПАСКАЛЬ Программирование на языке высокого
уровня.

Павловская Т.А.

Учебно-методические пособия:

1. Иванов А.Ф., Потапова О.Н., Салихова Г.Л.
Учебное пособие. Основы алгоритмического языка Pascal.
2. Салихова Г.Л., Миндиярова О.Г.
Методические указания по выполнению контрольных и лабораторных работ по дисциплине «Информатика».
Алгоритмический язык PASCAL.

**Потапова О.Н., Салихова Г.Л., Садриева Л.М.,
Миндиярова О.Г., Мохова О.М., Фахрутдинова
Р.Р., Ханова И.А.**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ
ПО ВЫПОЛНЕНИЮ КУРСОВОЙ РАБОТЫ ПО
ИНФОРМАТИКЕ**

**сдать работы до конца апреля в печатном и
электронном виде**

Алгоритмизация вычислительных процессов

Основные этапы решения задач на компьютере

- 1. Постановка задачи.*
- 2. Математическое или информационное моделирование.*
- 3. Алгоритмизация задачи.*
- 4. Программирование.*
- 5. Ввод программы и исходных данных в ЭВМ.*
- 6. Тестирование и отладка программы.*
- 7. Исполнение отлаженной программы и анализ результатов.*

Алгоритм и его свойства

Способы записи алгоритмов

- **Алгоритм** – это подробное описание последовательности арифметических и логических действий, расположенных в строгом логическом порядке и позволяющих решить конкретную задачу.
- **Алгоритмизация** – составление пошагового описания процесса решения задачи.

Свойства алгоритма

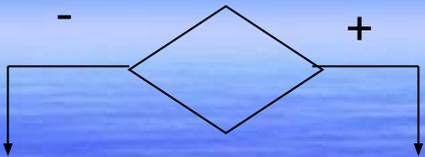
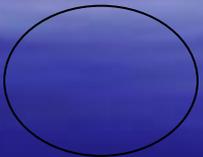
- **Определенность** (детерминированность, точность) – единственность толкования правил выполнения действий и порядка их выполнения;
- **Конечность** – обязательность завершения каждого из действий алгоритма и алгоритма в целом;
- **Результативность** – обязательность получения через определенное число шагов определенных результатов или сообщения о невозможности решения;
- **Массовость** – возможность применения одного и того алгоритма для решения однотипных задач с различными исходными данными;
- **Дискретность** – расчленение вычислительного процесса на отдельные этапы, элементарные операции.

Графический способ записи алгоритмов

Блок-схема – это способ описания алгоритмов, где действия представлены в виде блоков(геометрических фигур), а последовательность в виде стрелок между ними.

Элементы блок-схемы

<i>Название блока</i>	<i>Обозначение</i>	<i>Выполняемая функция</i>
Начало и конец вычислительного процесса		Вход и выход в отдельно описанных алгоритмах
Блок ввода и вывода		Задание численных значений исходных данных и фиксация результатов расчета
Вычислительный блок		Запись расчетных формул: слева записывается вычисляемая переменная, далее знак присваивания ($:=$), далее расчетная формула.

<i>Название блока</i>	<i>Обозначение</i>	<i>Выполняемая функция</i>
Логический блок		<p>Проверка условия и выбор направления хода вычислительного процесса. Имеет 2 выхода: +(да) при выполнении условия, -(нет) – в противном случае.</p>
Циклический блок (модификация)		Начало цикла
Соединитель		Указание связи между прерванными линиями связи

Основные структуры алгоритмов

Основные структуры алгоритмов – ЭТО ограниченный набор блоков и стандартных способов их соединения для выполнения типичных последовательностей действий.

К основным структурам алгоритмов относятся:

- *Следование*
- *Ветвление*
- *Повторение*

Алгоритмы линейной структуры

- Алгоритм линейной структуры
(следование)- алгоритм, в котором все действия выполняются последовательно друг за другом.

Рассмотрим несколько примеров:

Задача 1

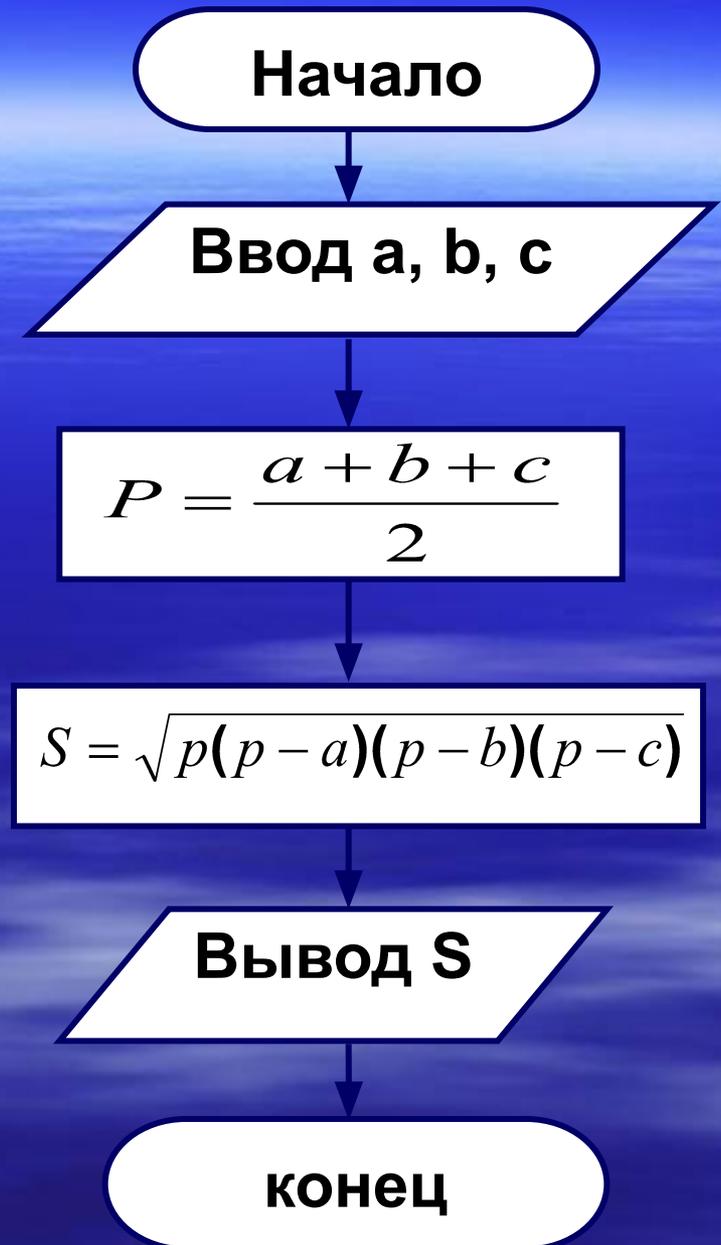
Определить площадь
треугольника по формуле
Герона, если известны
длины его сторон a, b, c .

Площадь треугольника:

$$S = \sqrt{p \cdot (p - a) \cdot (p - b) \cdot (p - c)}$$

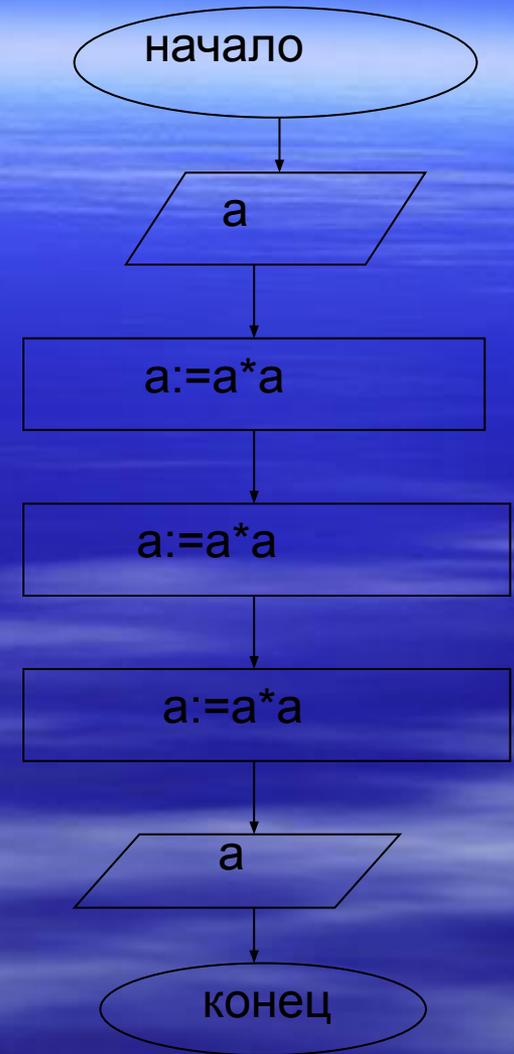
Полупериметр треугольника

$$p = \frac{a + b + c}{2}$$



Задача 2:

Дано действительное число a . Не пользуясь никакими операциями, кроме умножения получить a^8 за три операции.



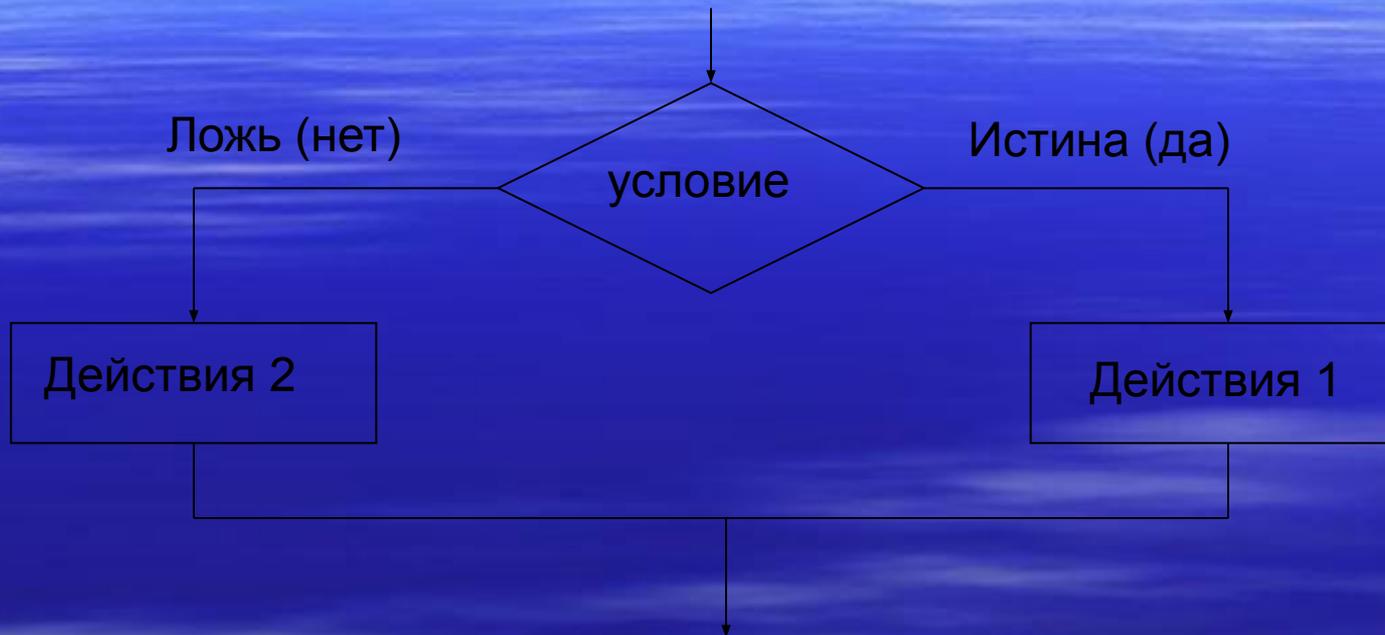
Алгоритмы разветвляющейся структуры

- Ветвление(развилка) – такая схема, в которой предусмотрено разветвление указанной последовательности действий на несколько направлений в зависимости от итога проверки заданного условия.

В схемах такой структуры используется логический блок.

- Различают полное (раздвоение) и неполное (ответвление) ветвления.

Полное ветвление



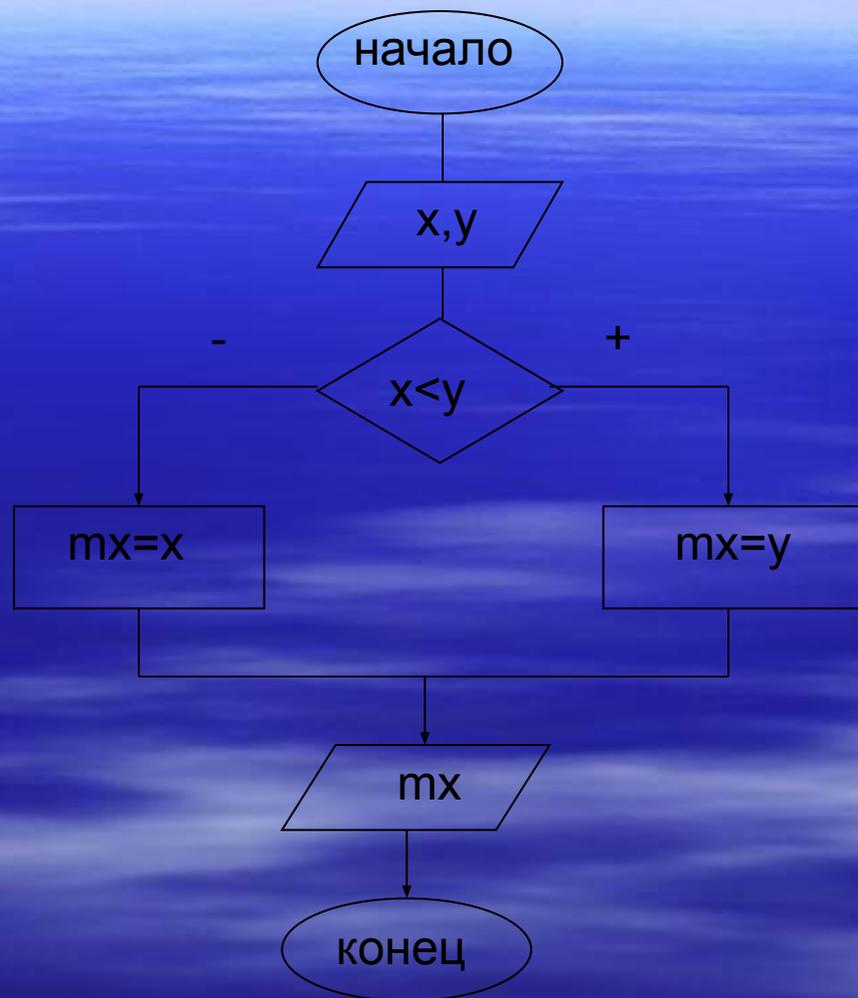
Неполное ветвление



Задачи на разветвляющуюся структуру

Задача 3

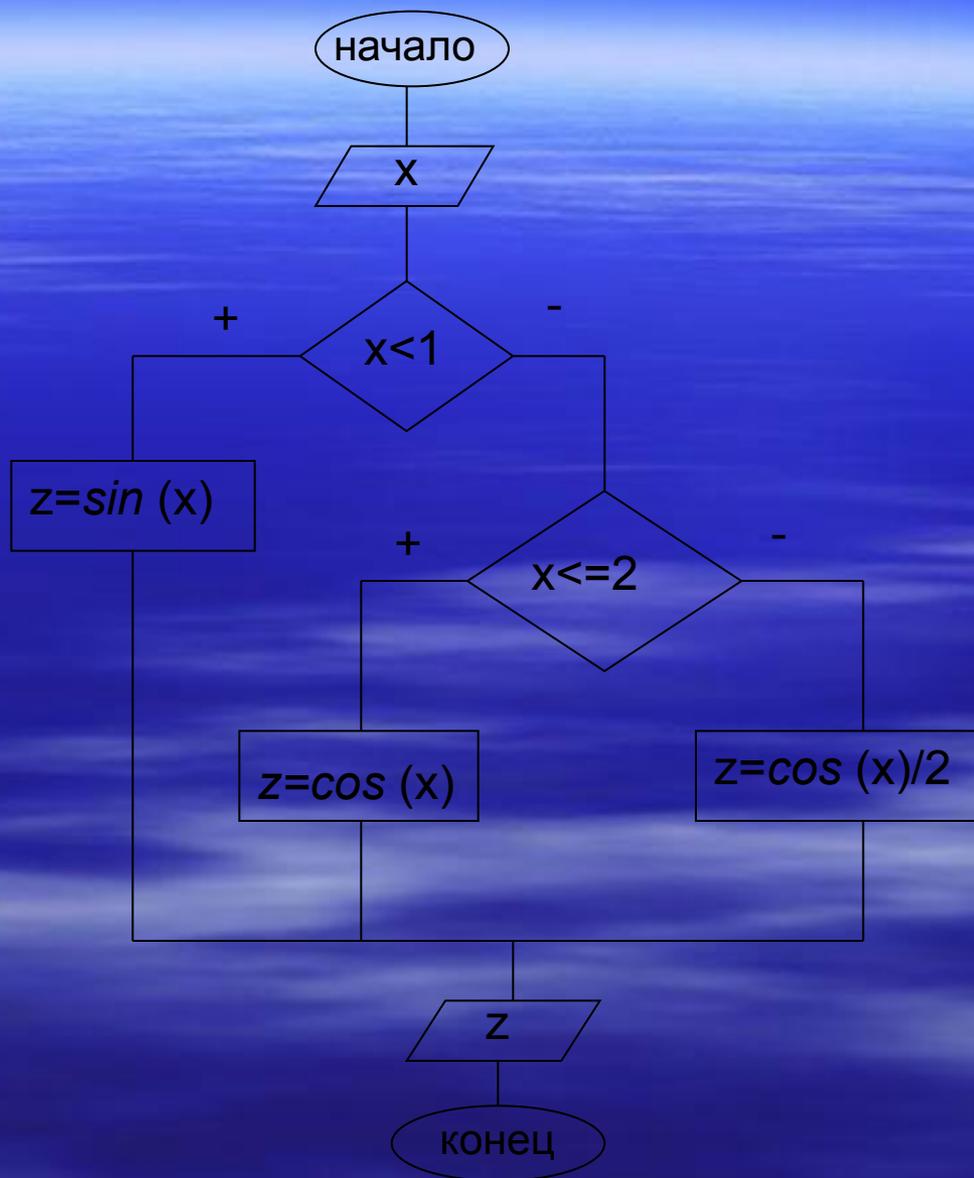
Даны два числа x, y . Найти максимальное из них.



Задача 4.

Вычислить значение функции Z по значению x .

$$Z = \begin{cases} \sin x, & \text{если } x < 1; \\ \cos x, & \text{если } 1 \leq x \leq 2 \\ \frac{\cos x}{2}, & \text{если } x > 2 \end{cases}$$

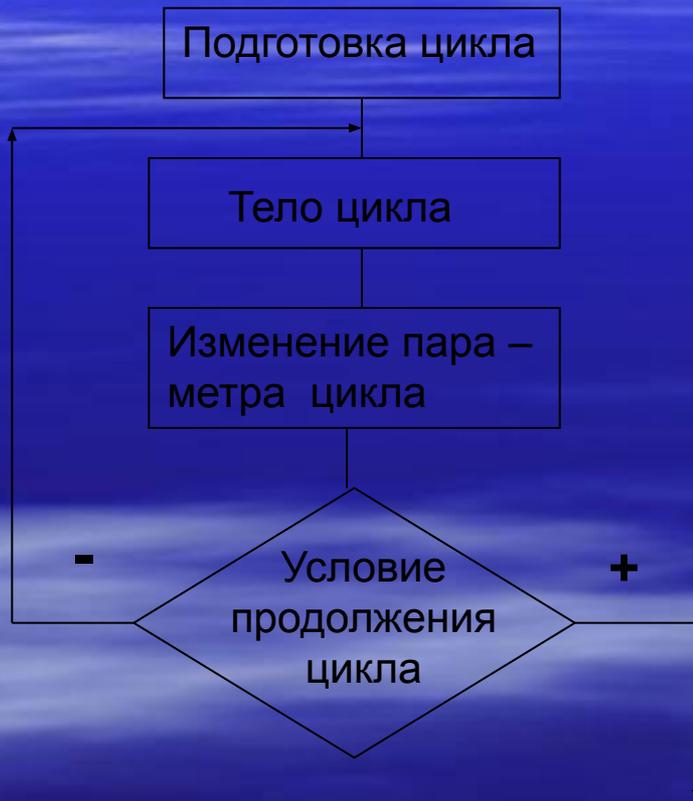


Алгоритмы циклической структуры

- Алгоритмы циклической структуры (повторение) – это алгоритмы, отдельные действия в которых многократно повторяются.
- **Цикл** – совокупность действий алгоритма, связанная с повторением.
- **Тело цикла** – многократно повторяющиеся действия алгоритма.
- **Параметр цикла** – величина, с изменением которой связано многократное выполнение цикла.
Различают циклические алгоритмы:
 - С заранее неизвестным числом повторений – цикл «ДО» и цикл «ПОКА»;
 - С заранее известным числом повторений – цикл «ДЛЯ».

Циклическая структура «ДО»

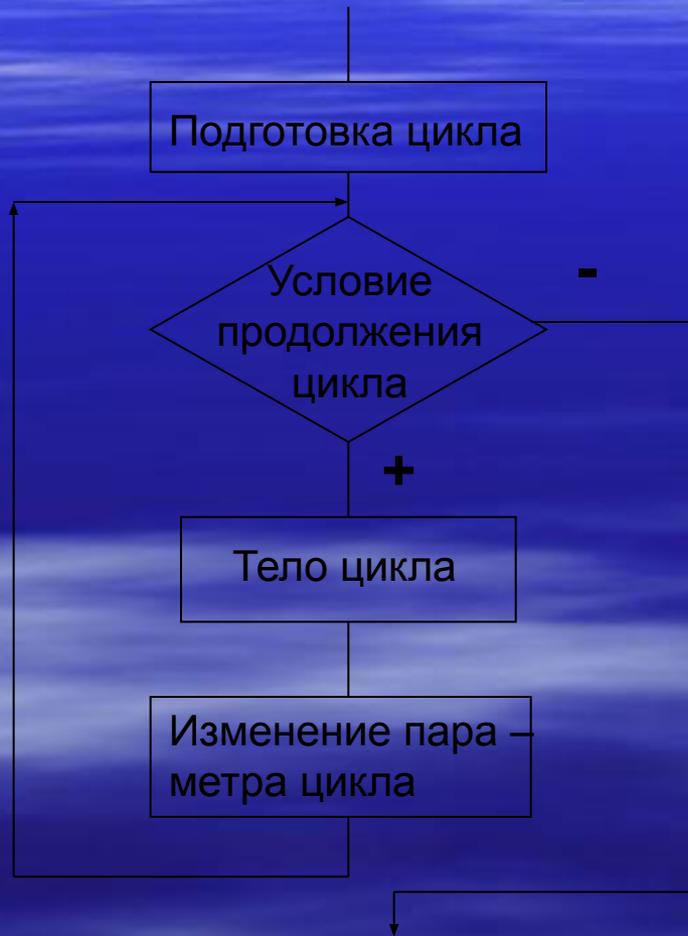
Схему этого цикла можно представить в следующем виде:



- Повторять до тех пор, пока не будет выполнено условие.
- Особенность этого цикла состоит в том, что он выполняется хотя бы один раз, так как первая проверка условия происходит после того, как тело цикла выполнено.

Циклическая структура «ПОКА»

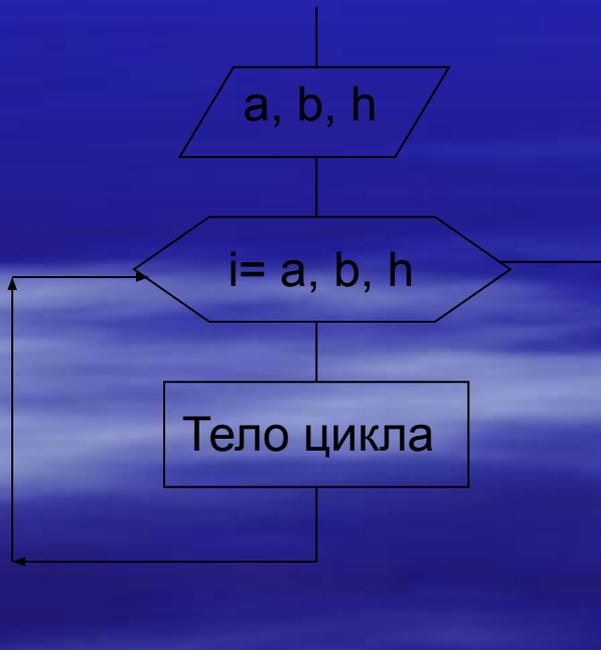
Схему этого цикла можно представить в следующем виде:



- Повторять до тех пор, пока выполняется условие.
- Цикл «ПОКА» отличается от цикла «ДО» тем, что здесь проверка условия проводится до выполнения тела цикла.
- Если при первой проверке условие выхода из цикла выполняется, то тело цикла не выполнится ни разу.

Циклическая структура «ДЛЯ»

- Для изображения алгоритмов этой циклической структуры используется блок «модификация». Схему этого цикла можно представить в следующем виде:



- В блоке «модификация» объединяются несколько блоков: подготовка цикла, проверка окончания, изменение параметра цикла.

В блоке модификации записывается:

- параметр цикла – i ;
- знак равенства (присваивание);
- начальное значение параметра цикла – a ;
- конечное значение параметра цикла – b ;
- шаг изменения параметра цикла – h

Если шаг изменения равен единице, то в блоке модификация шаг можно не указывать.