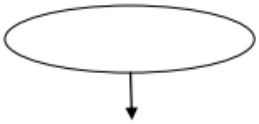
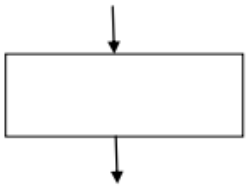
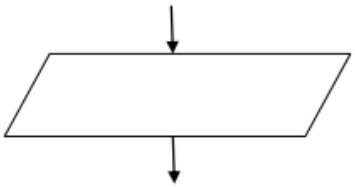
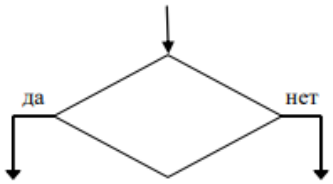
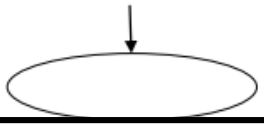
Методический материал по информатике. Блок-схемы

Блок-схема – это представление алгоритма в графической форме. Все команды и действия представлены геометрическими фигурами (блоками). Внутри каждой фигуры вписывается вся информация о тех действиях, которые нужно выполнить. Связи изображены в виде обычных линий со стрелками (при необходимости).

Для оформления блок-схем алгоритмов имеется ГОСТ 19.701-90. Он описывает порядок и правила создания их в графической форме, а также основные методы решения.

Графические элементы блок схем	Их значение
	Блок начала
	Блок действия
	Блок ввода-вывода
	Блок разветвления
	Блок конца

Линейный алгоритм - алгоритм, в котором действия выполняются последовательно и однократно.

Составление блок-схем.

Условие

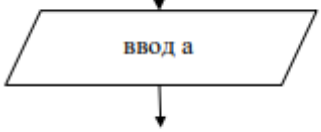
Составить блок-схему алгоритма нахождения периметра квадрата по его стороне.

1. Сначала вспомним формулу нахождения периметра квадрата по его стороне. Обозначим сторону квадрата буквой *a*.

P=a+a+a+a=4*a

2. Каждый алгоритм начинается с блока начала. Нарисуем его.

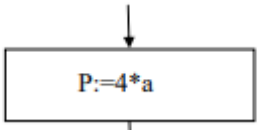
3. Далее происходит ввод переменной *a*. Изобразим блок ввода.



4. После мы находим периметр. Переменной *P* мы присваиваем *4*a*. Для этого нарисуем блок действия.

Примечание:

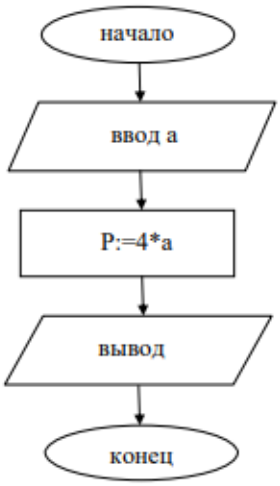
“:=” -знак присвоения.



5. Далее выводим значение переменной *P*. Изобразим блок вывода.

6. Нарисуем блок конца алгоритма.

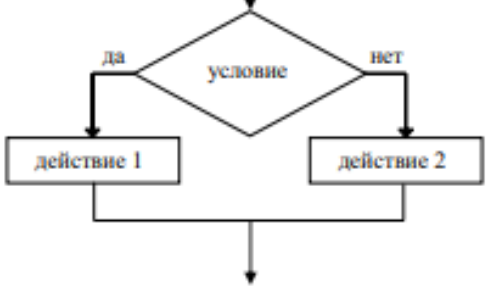
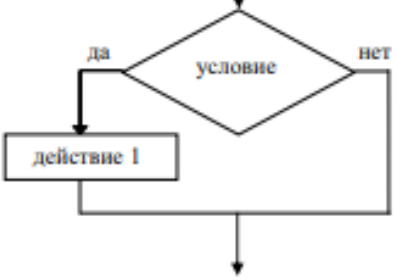
В итоге мы получили блок-схему алгоритма нахождения периметра квадрата.



§3.2 Разветвляющийся алгоритм.

Разветвляющийся алгоритм-алгоритм, в котором в зависимости от условия выполняется одно или другое действие.

Виды ветвления:

1. Полное	
2. Неполное	

Составление блок-схем.

Условие:

Составить блок-схему алгоритма, определяющего, является ли заданное число положительным или отрицательным.

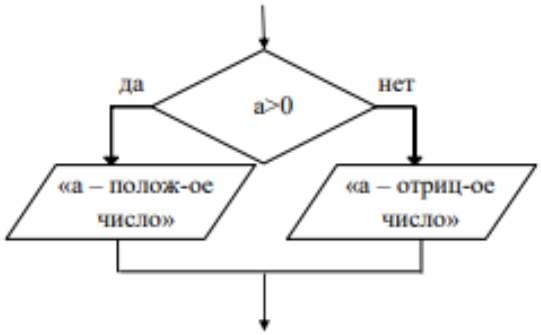
1. Вы уже знаете, что каждый алгоритм начинается с блока начала. Нарисуем его.

2. Далее происходит ввод переменной *a*. Изобразим блок ввода.

3. По условию задачи нам необходимо проверить, является ли заданное число положительным или отрицательным. Для этого мы используем блок разветвления с полным ветвлением.

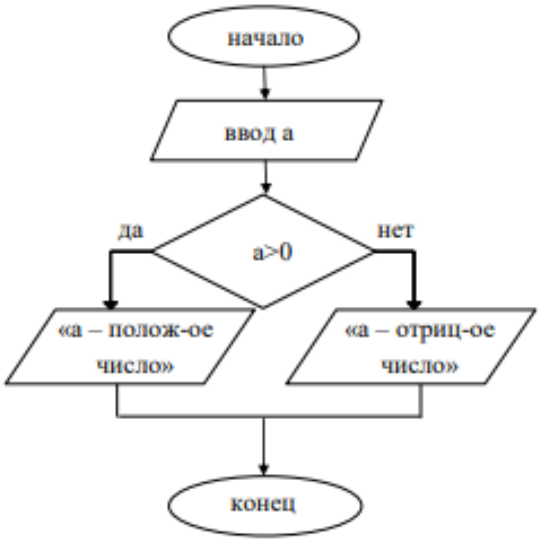
Зададим условие:
Если *a*>0, то выведем «а – положительное число».
Если *a* не удовлетворяет данному условию, то выведем «а – отрицательное число».

Примечание:
вместо блока действия, мы используем блок вывода, так как нам необходимо просто определить каким является число.



4. Нарисуем блок конца алгоритма.

Итог:

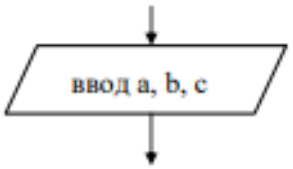


Условие:

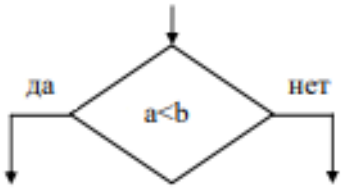
Составить блок-схему алгоритма нахождения меньшего числа из *a*, *b* и *c*. Вывести меньшее число.

1. Рисуем блок начала

2. Далее происходит ввод переменных *a*, *b*, *c*. Изобразим блок ввода.



3. Сначала сравним две переменные, к примеру ''*a*''и ''*b*''. Для этого используем блок выбора с полным разветвлением.



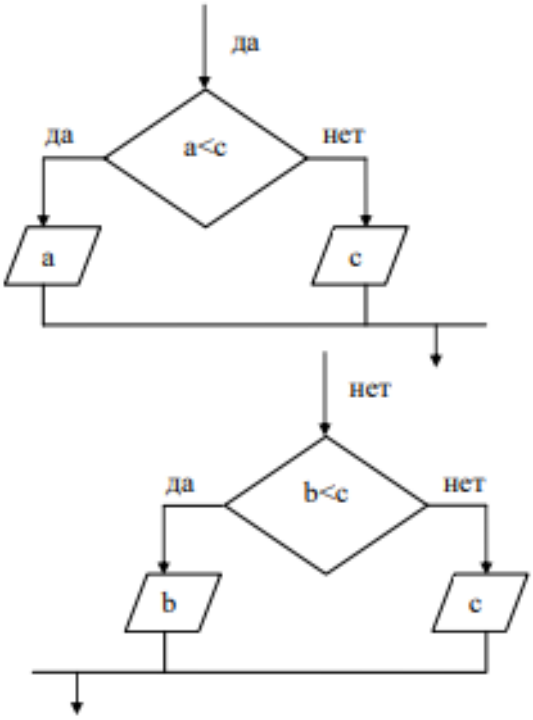
4. Далее используем еще два блока выбора, где сравним переменные "a" и "*c*", "*b*" и "*c*".

Ветка «да»:

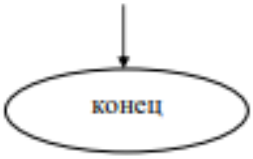
1. если *a* меньше, чем *b*, то сравним *a* и *c*. Если *a* также меньше, чем *c*, то *a* является наименьшим числом. Выведем его;
2. если *a* меньше, чем *b*, но *a* не меньше, чем *c*, то *c* – наименьшее число. Выведем его.

Ветка «нет»:

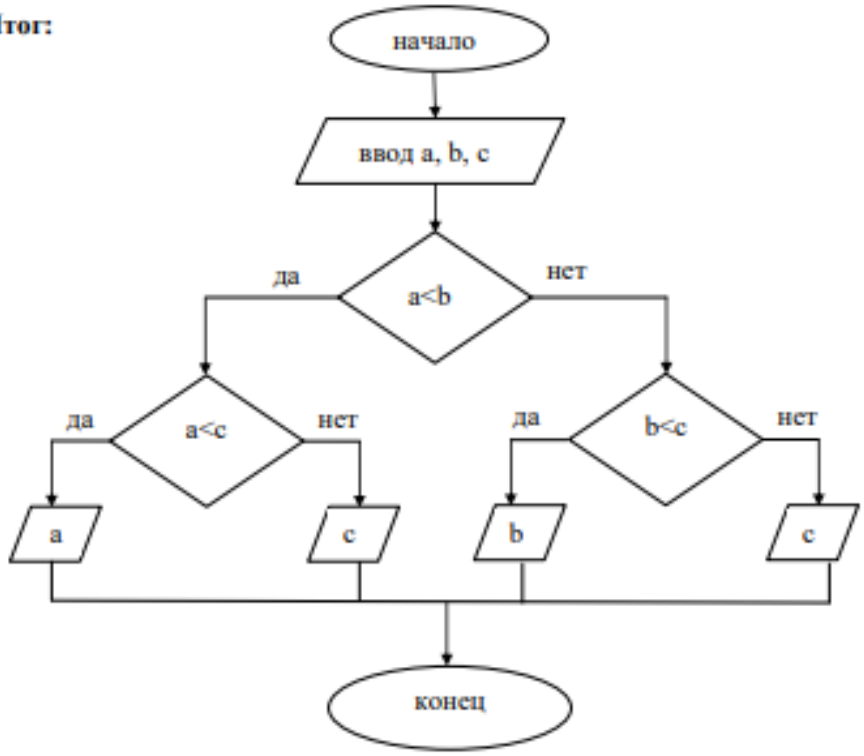
1. если *a* не меньше, чем *b*, то сравним *b* и *c*. Если *b* меньше, чем *c*, то *b* является наименьшим числом. Выведем *b*;
2. если *a* не меньше, чем *b*, но *b* не меньше, чем *c*, то *c* – наименьшее число. Выведем его.



5. Нарисуем блок конца.



Итог:



Задачи:

1. Составить блок-схему алгоритма нахождения частного *a* и *b*.
   Примечание: на ноль делить нельзя.
2. Составить блок-схему алгоритма нахождения большего числа из *a* и *b*. Вывести большее число.
3. Составить блок-схему алгоритма нахождения большего числа из *a*, *b* и *c*. Вывести большее число.
4. Составить блок-схему алгоритма нахождения меньшего числа из *a* и *b*. Вывести меньшее число.