

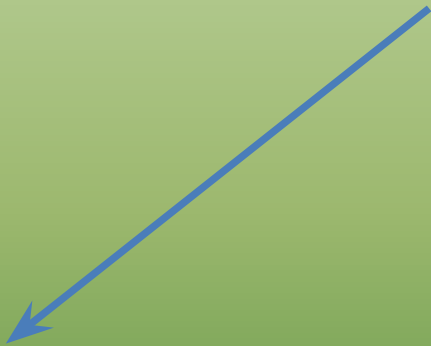
# Типы алгоритмов и формы их записи.



# *Алгоритм*

это система точных и понятных предписаний (команд) о содержании и последовательности выполнения конечного числа действий, необходимых для решения любой задачи данного типа, определяющих действия исполнителя.

# Типы алгоритмов



линейные



ветвящиеся



циклические

# Формы записи алгоритмов



табличная



словесная

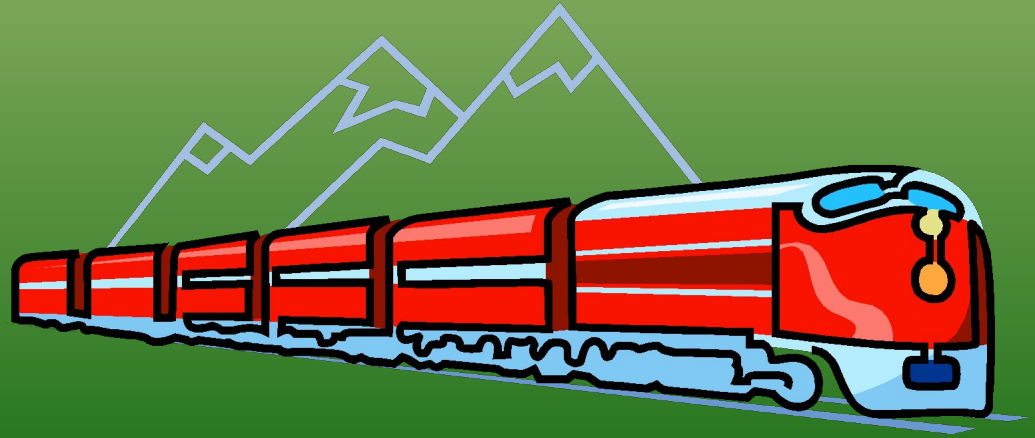


графическая  
(блок-схема)

# *Линейные алгоритмы.*

# Линейные алгоритмы

**Линейным** называется алгоритм, в котором команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий.



# Задача 1. Вычислить сумму двух чисел.

На примере решения этой задачи рассмотрим все три формы записи алгоритма.

*Решение любой задачи начинается с I этапа*

## Постановка задачи:

Дано:  $a$  – первое число;  
 $b$  – второе число.

Найти:  $c$  – сумму чисел  $a$  и  $b$ .

## *II этап*

## Математическая формализация

Связь:  $c = a + b$ .

### III этап

## Составление алгоритма

### Табличная форма



- Применяется только для линейных вычислительных алгоритмов.

<b>a</b>	<b>b</b>	<b>c=a+b</b>
1	3	4
-2	-6	-8
5	-9	-4



# Словесная форма



- Применима для всех типов вычислительных алгоритмов.

Выберем русский язык для записи алгоритма в этой форме и запишем последовательность команд, выполнение которых позволит при заданных значениях двух чисел найти их сумму.

1. Начало алгоритма.
2. Прочеть (ввести) значение первого числа -  $a$ .
3. Прочеть (ввести) значение второго числа -  $b$ .
4. Сложить значения  $a$  и  $b$ .
5. Записать полученный в предыдущей команде результат как значение  $c$ .
6. Конец алгоритма.

Если использовать *оператор присваивания* ( $:=$ ), то словесная форма представления этого же алгоритма станет более компактной:

1. Начало алгоритма.
2. Прочеть (ввести) значение первого числа -  $a$ .
3. Прочеть (ввести) значение второго числа -  $b$ .
4.  $c := a + b$ .
5. Вывести полученный результат -  $c$ .
6. Конец алгоритма.

# Графическая форма (блок-схема)

- ! Применима для всех типов вычислительных алгоритмов.

Основана на замене (кодировании) типичных алгоритмических команд определенными геометрическими фигурами. Например,



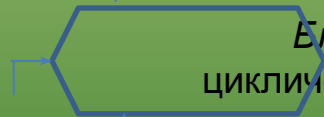
Блок начала и конца алгоритма. В овале пишут слова «начало» и «конец».



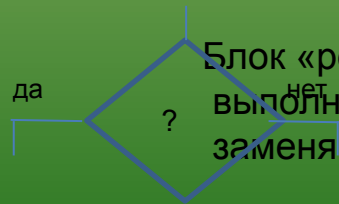
Блок ввода и вывода информации. В блоке ввода перечисляют имена данных, подлежащих вводу в алгоритм, в блоке вывода – выводу из алгоритма.



«Процесс» решения задачи. В прямоугольнике блока записывают действия

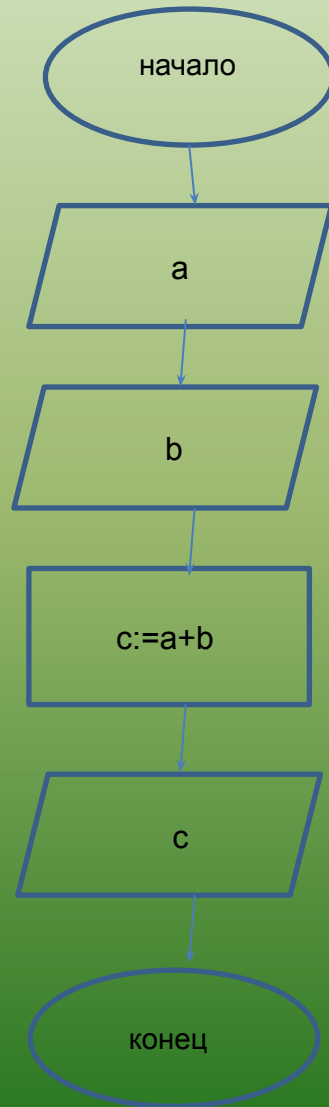


Блок изменения параметров. Используется, например, в блок-схемах циклических алгоритмов со счетчиком.



Блок «решение». Проверяет выполнение какого-либо условия: выход **да** при выполнении условия, выход **нет** – при его невыполнении (**да** и **нет** иногда заменяют на **0** и **1** или **+** и **-** соответственно).

Тогда *графическая форма* (блок-схема) записи алгоритма вычисления суммы двух чисел выглядит следующим образом:



## Задача 2.

Вычислить время просмотра фильма.

Дано:  $t_1$  – начальное время просмотра;  
 $t_2$  – конечное время просмотра.

Найти:  $t$  – общее время просмотра фильма.

Связь:  $t := t_2 - t_1$ .

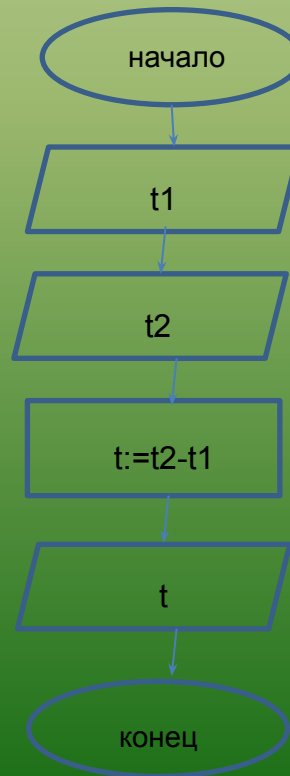
## Табличная форма

<b><i>t1</i></b>	<b><i>t2</i></b>	<b><i>t:=t2-t1</i></b>
13	15	2
12	16	4
20	21	1

## Словесная форма

1. Начало алгоритма.
2. Ввести начальное время просмотра –  $t1$ .
3. Ввести конечное время просмотра –  $t2$ .
4.  $t := t2 - t1$ .
5. Вывести полученный результат -  $t$ .
6. Конец алгоритма.

## Графическая форма (блок-схема)



# Ветвящиеся алгоритмы.



В рассмотренных до сих пор алгоритмах все команды выполнялись последовательно одна за другой в том порядке, в каком они были записаны. Однако таким образом может быть построен алгоритм для решения далеко не всякой задачи. В практике известны задачи, дальнейший ход решения которых зависит от выполнения каких либо условий.

# Алгоритм с ветвлением

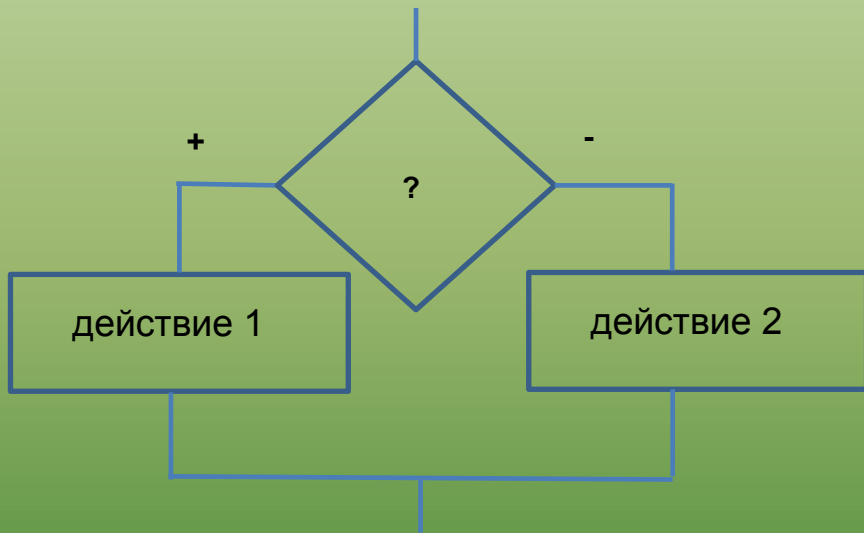
Алгоритм, в котором необходимо сделать определенный выбор в зависимости от некоторого условия называется **ветвящимся** (разветвляющимся).



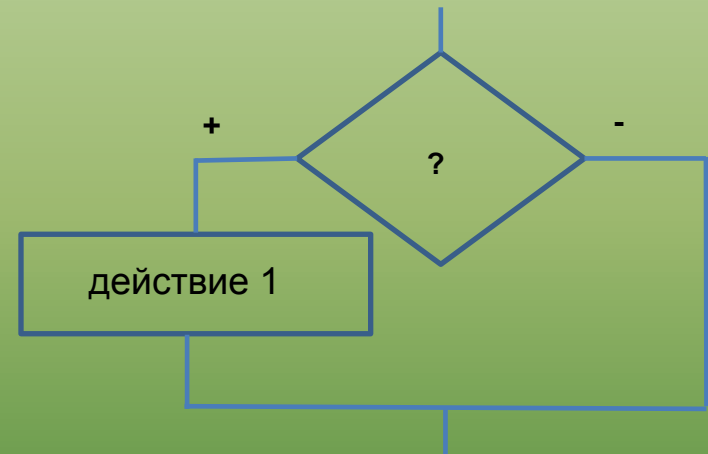


# Ветвление

*полное*



*неполное*



# Операции сравнения

**A < B**      A меньше B

**A <= B**      A меньше или равно B

**A = B**      A равно B

**A > B**      A больше B

**A >= B**      A больше или равно B

**A <> B**      A не равно B

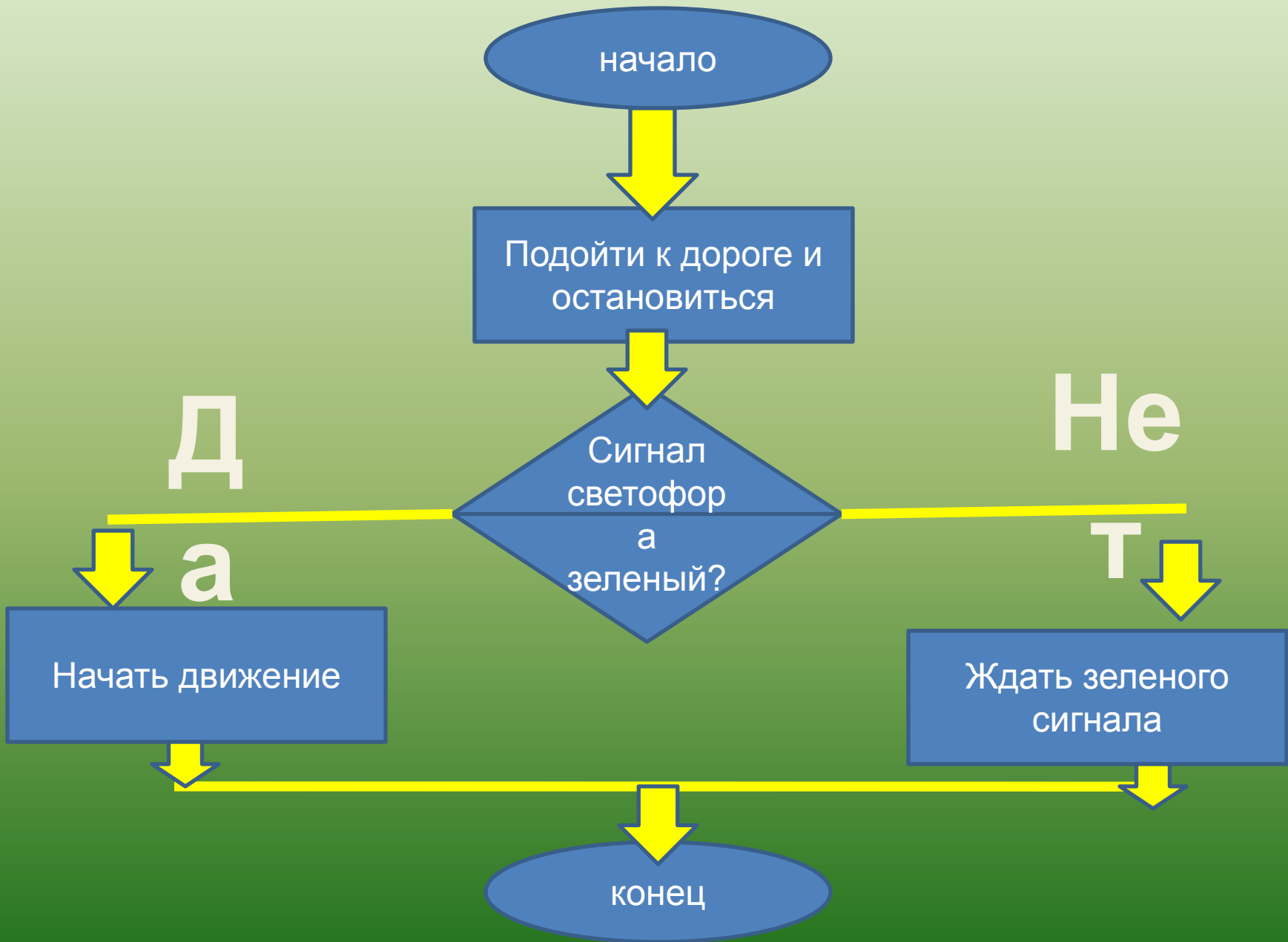


# Задание



- Составьте блок-схему алгоритма перехода дороги, в зависимости от сигнала светофора.





**Задача 3.** Вычислить значение величины  $c$ , определяемое по формулам:  $c=a+b$ , если  $a \leq b$  и  $c=a-b$ , если  $a > b$ .  
*Решение задачи оформим в соответствии с 1, 2 и 3 ЭРЗ на компьютере.*

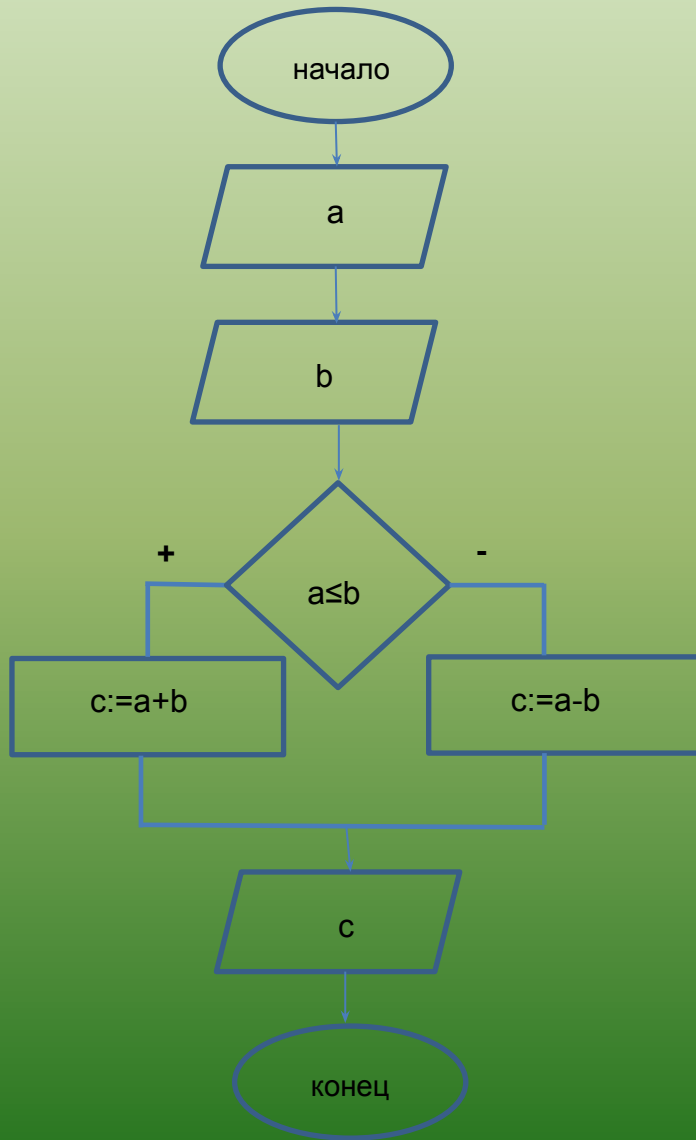
*Дано:*      $a$  – первое число;  
               $b$  – второе число.

*Найти:*      $c$  – значение величины.

*Связь:*      $c=a+b$ , если  $a \leq b$  и  
               $c=a-b$ , если  $a > b$

$$\text{или } c = \begin{cases} a+b, & \text{если } a \leq b \\ a-b, & \text{если } a > b \end{cases}$$

Алгоритм решения данной задачи составим в графической форме. Он организуется при помощи полной формы ветвления.



Протестируем алгоритм и оформим результат в виде таблицы:

Тест	a	b	Проверка условия	Результат проверки условия (+ или -)	Действия	c
1.	5	2	$5 \leq 2$	-	$c = 5 - 2$	3
2.	4	4	$4 \leq 4$	+	$c = 4 + 4$	8
3.	3	-3	$3 \leq -3$	-	$c = 3 - (-3)$	6
4.	0	0	$0 \leq 0$	+	$c = 0 + 0$	0
5.	11	7	?			
6.	-9	-3	?			

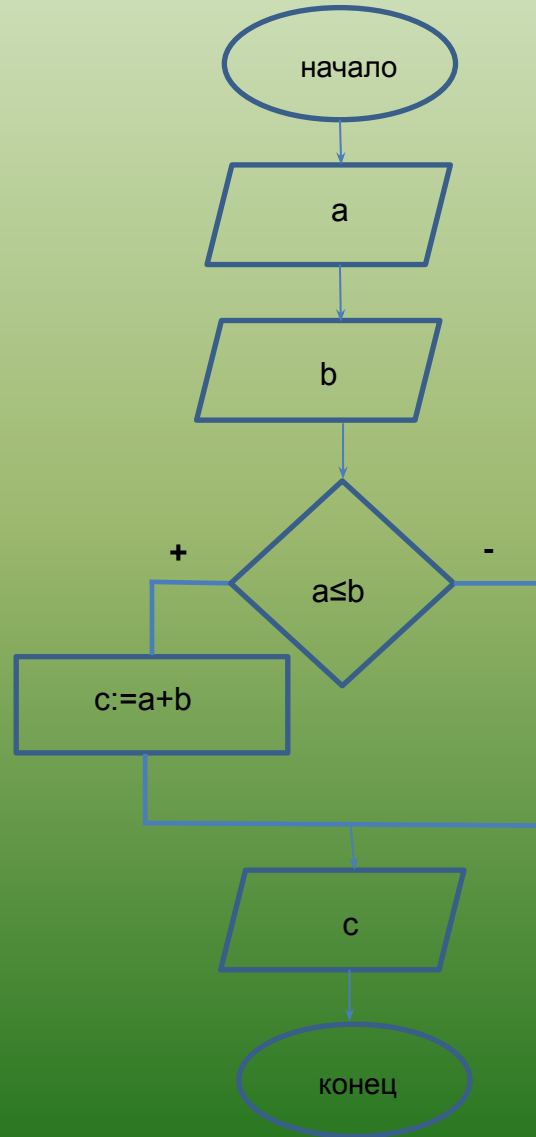
**Задача 4.** Вычислить значение величины  $c$ , определяемое по формулам:  $c=a+b$ , если  $a \leq b$ .

*Дано:*  $a$  – первое число;  
 $b$  – второе число.

*Найти:*  $c$  – значение величины.

*Связь:*  $c=a+b$ , если  $a \leq b$

Алгоритм решения данной задачи организуется при помощи неполной формы ветвления.





Самостоятельная работа.

**Задача 5.** Определите значение переменной  $c$  после выполнения следующего фрагмента алгоритма:

1. Начало алгоритма.
2.  $a := 40;$
3.  $b := 10;$
4.  $b := a - 2 * b;$
5. Если  $a < 2 * b$ , То  $c := a$ , Иначе  $c := b;$
6. Вывод  $c;$
7. Конец алгоритма.

**Задача 6.** Определите значение переменной  $c$  после выполнения следующего фрагмента алгоритма:

1. Начало алгоритма.
2.  $a := -5;$
3.  $b := 14;$
4.  $b := b + a * 2;$
5. Если  $a < b$ , То  $c := a + b$ , Иначе  $c := b - a;$
6. Вывод  $c;$
7. Конец алгоритма.