Блок- схема

Блок-схема - это графический способ представления алгоритма с помощью геометрических фигур, называемых блоками и стрелок, показывающих последовательность выполнения действий

Каждый блок обозначает определенное действие, которое записывается внутри блока.

Обычно блок-схема рисуется сверху вниз или слева направо.

Название действия	Язык блок- схемы	Язык Pascal
Начало алгоритма	НАЧАЛО	BEGIN
Конец алгоритма	КОНЕЦ	END.
Присваивание	$X := \frac{3 + b^2}{a^2 + 1}$	X:=(3+b*b) / (a*a+1);
Вывод данных на экран	Вывод в строку A Вывод a, b Вывод ‘Text’	WRITE (A, ‘ ’); WRITELN (a, b); WRITELN (‘Text');

Название действия	Язык блок- схемы	Язык Pascal
Ввод данных через клавиатуру	Ввод A Ввод A,B	WRITE (‘Задайте значение A’); READLN(A); WRITELN (‘Задайте значение A и B ‘); READLN(A,B);

Примечание:

- Можно использовать оператор READ.
- Пояснение для вводимых значений можно оформлять по-разному.

Пример:

Задача.

Найти значение арифметического выражения:

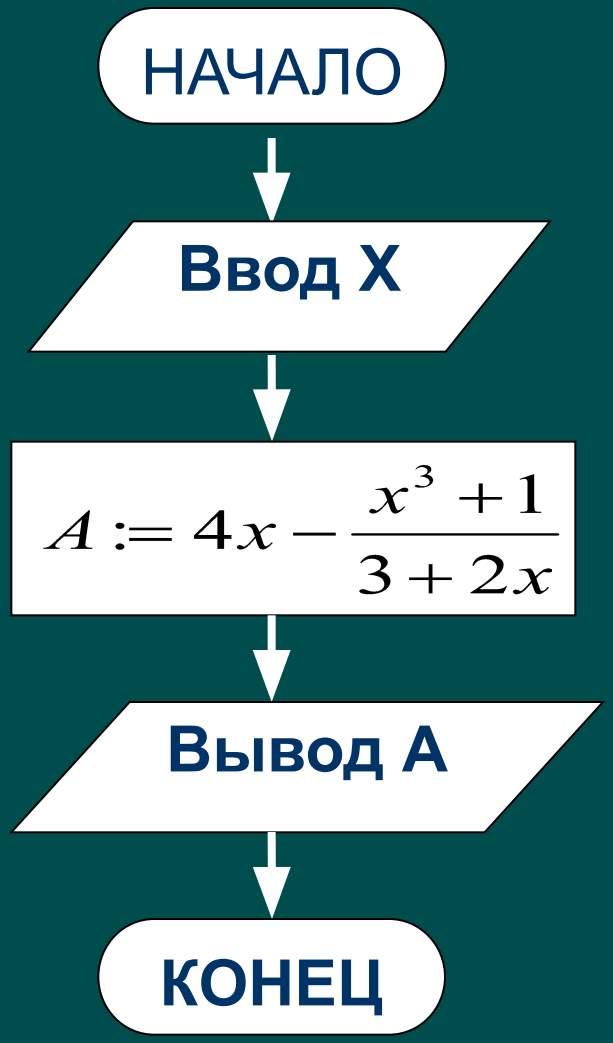
$$4x - \frac{x^3 + 1}{3 + 2x}$$

при x=-5,64 ; -4,9 ; -2,78; -0,6 ; 0,25; 2,83.

Ответ получить с тремя знаками после запятой.

Решение:

Блок-схема:



Программа:

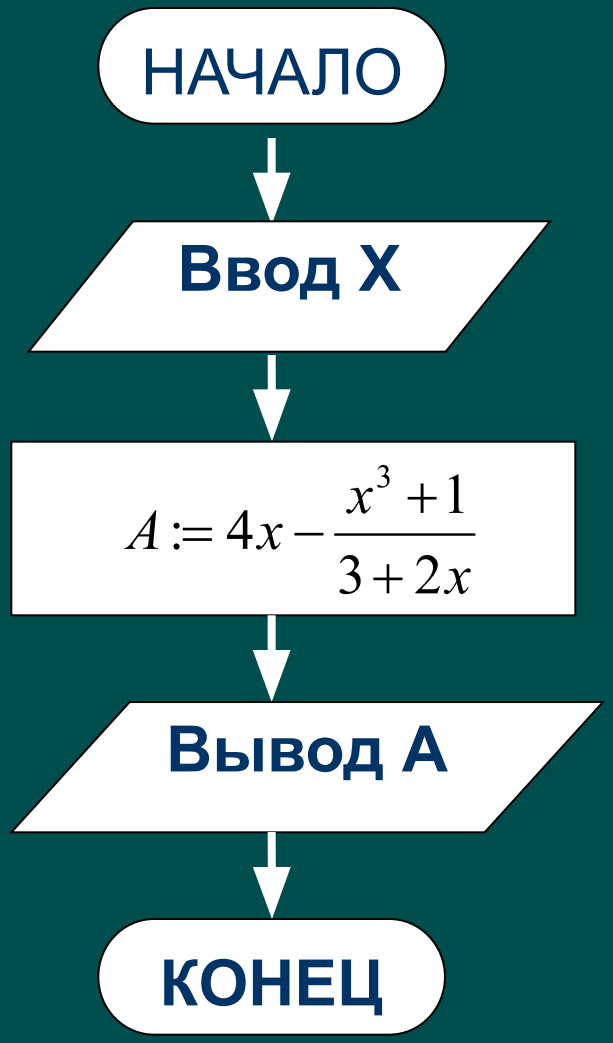
Трассировка

Трассировка – проверка правильности составления алгоритма или программы.

Оформляется в виде **таблицы (трассировочной)**, количество колонок которой зависит от количества переменных, количества условий и экран.

Решение:

Блок-схема:



Программа:

Трассировка :
в блок-схеме две переменные X, A и экран.

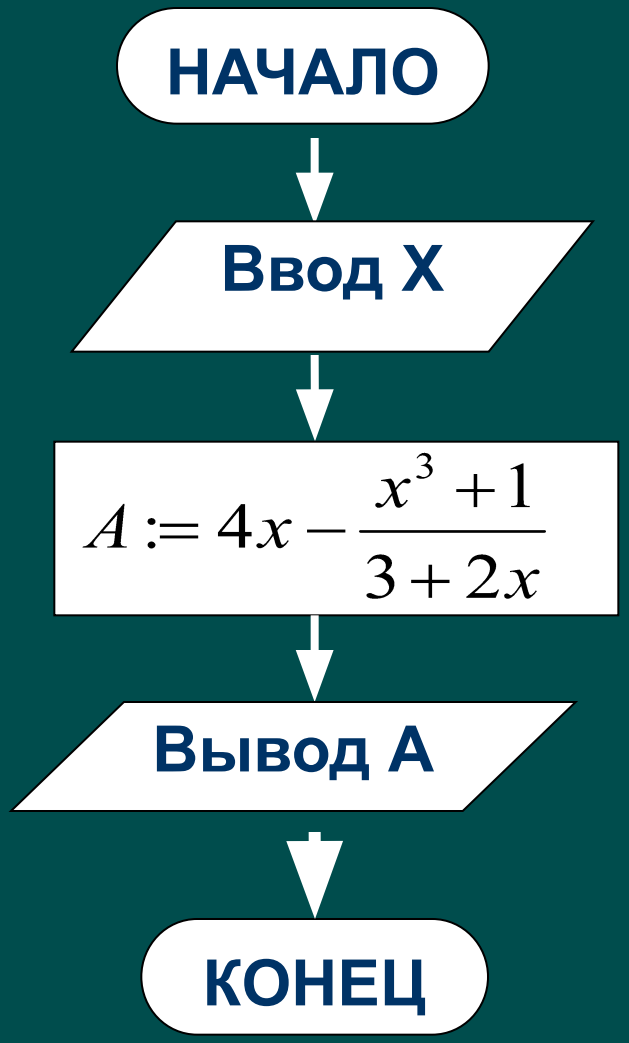
X	A	Экран

В качестве исходных данных нужно брать значения легко считаемые и, которые проверяют всевозможные ситуации.

Пишем программу!

Решение:

Блок-схема:

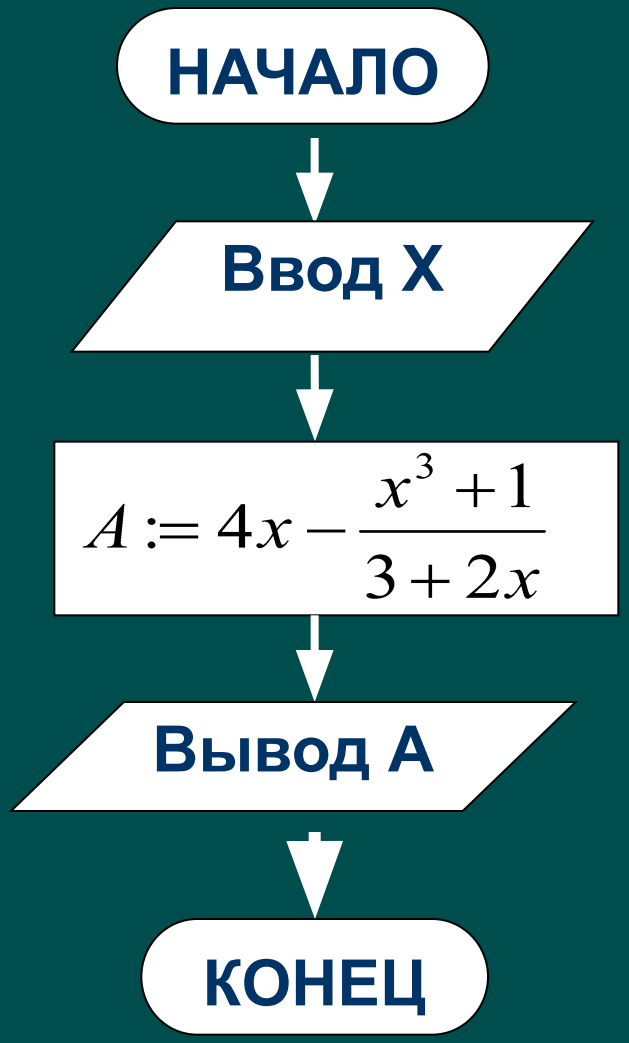


Программа:

```
PROGRAM  ARIFM;
```

Решение:

Блок-схема:

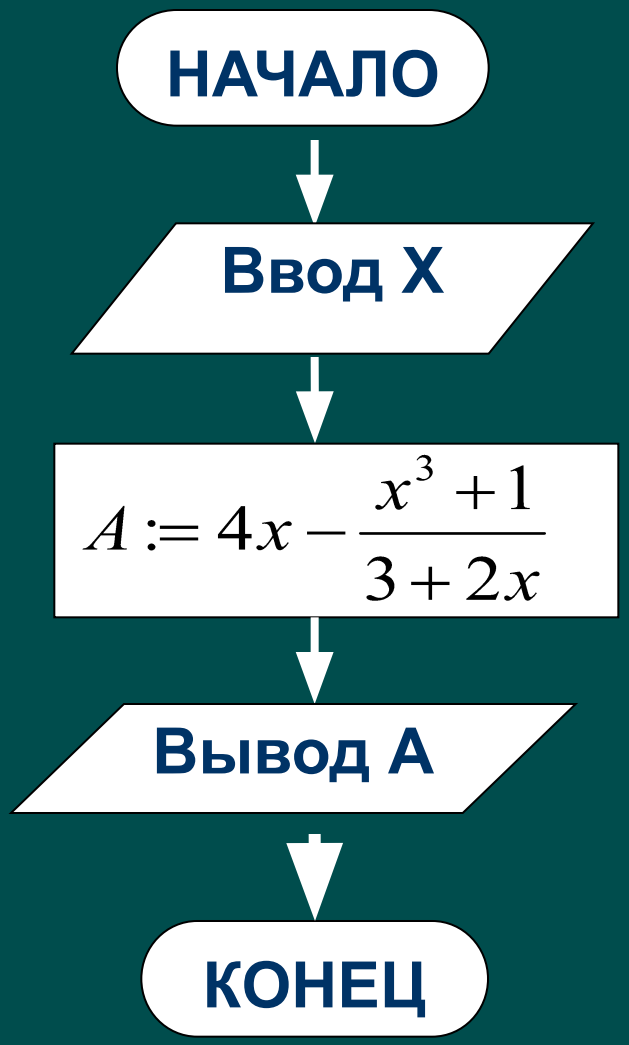


Программа:

```
PROGRAM  ARIFM;
  VAR
```

Решение:

Блок-схема:

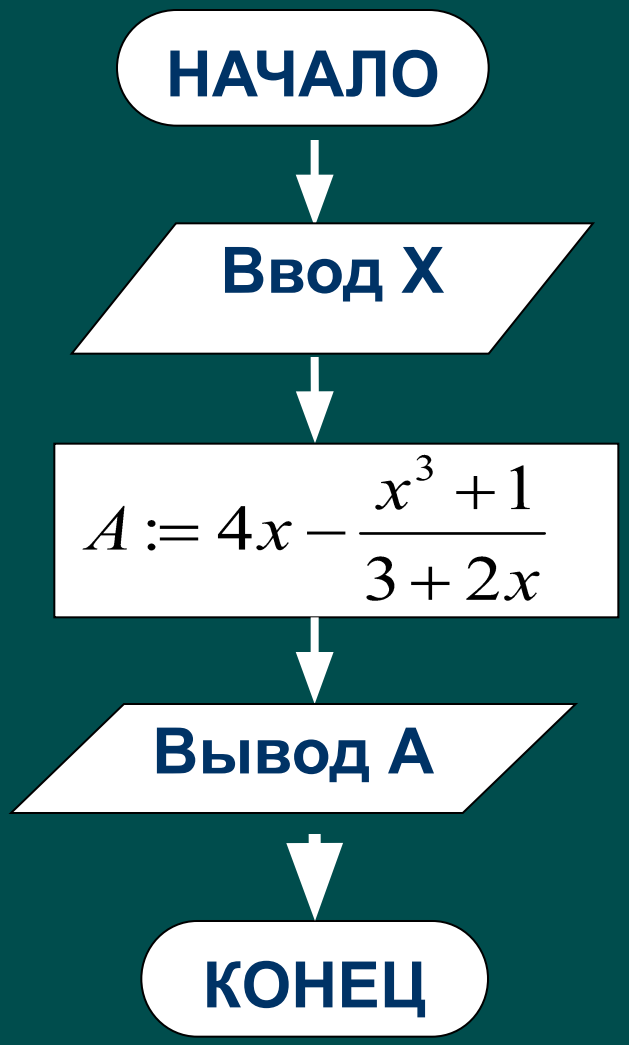


Программа:

```
PROGRAM  ARIFM;
  VAR  X,A:
```

Решение:

Блок-схема:

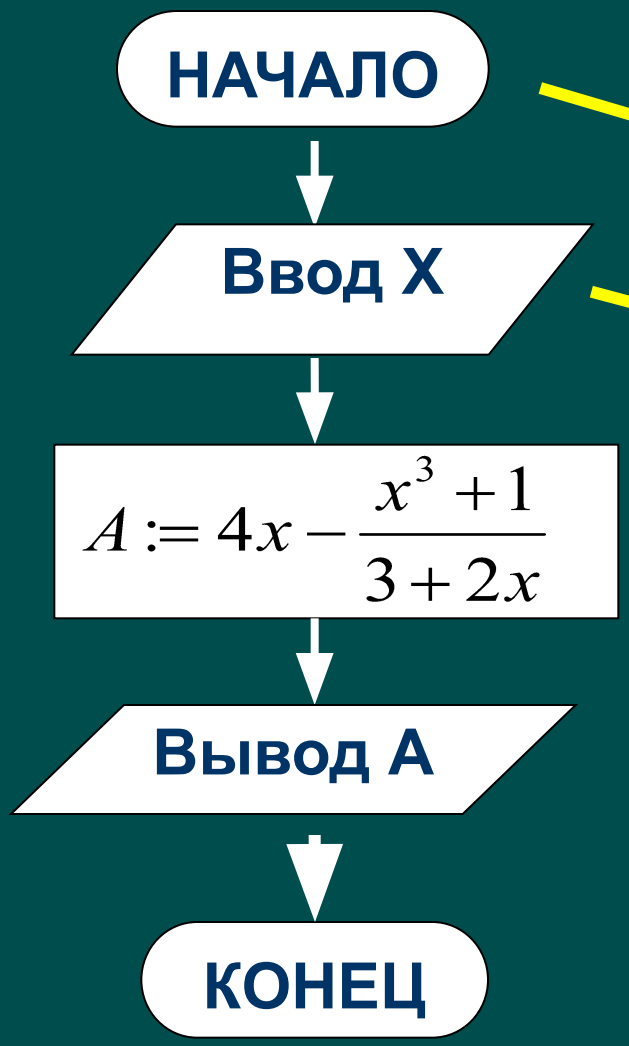


Программа:

```
PROGRAM  ARIFM;
  VAR  X,A:REAL;
```

Решение:

Блок-схема:

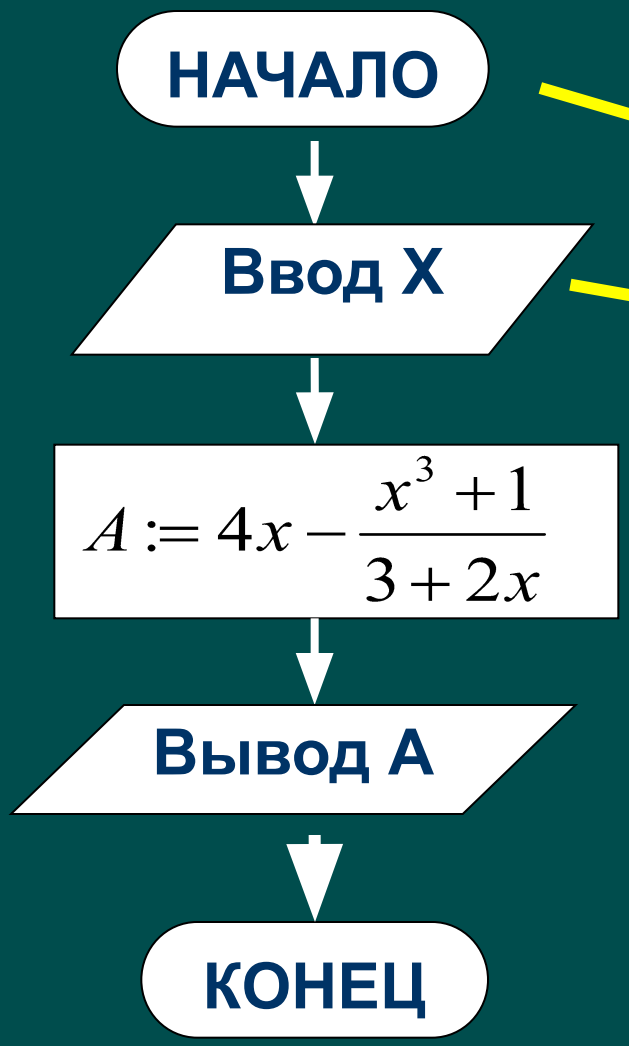


Программа:

```
PROGRAM  ARIFM;
  VAR  X,A:REAL;
BEGIN
```

Решение:

Блок-схема:

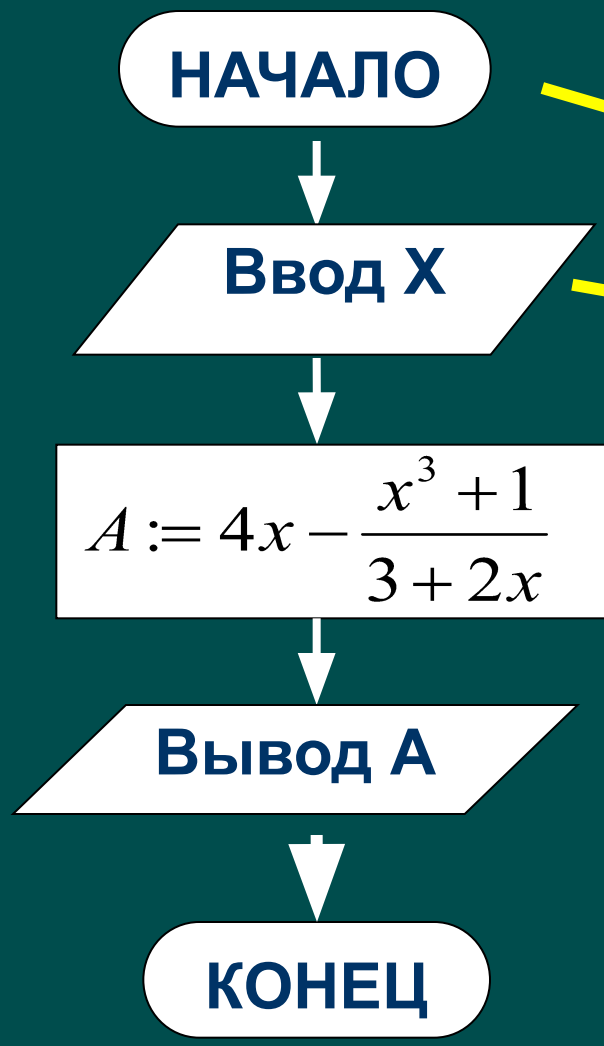


Программа:

```
PROGRAM  ARIFM;
   VAR  X,A:REAL;
BEGIN
   WRITE ('X=');
   READLN (X);
```

Решение:

Блок-схема:

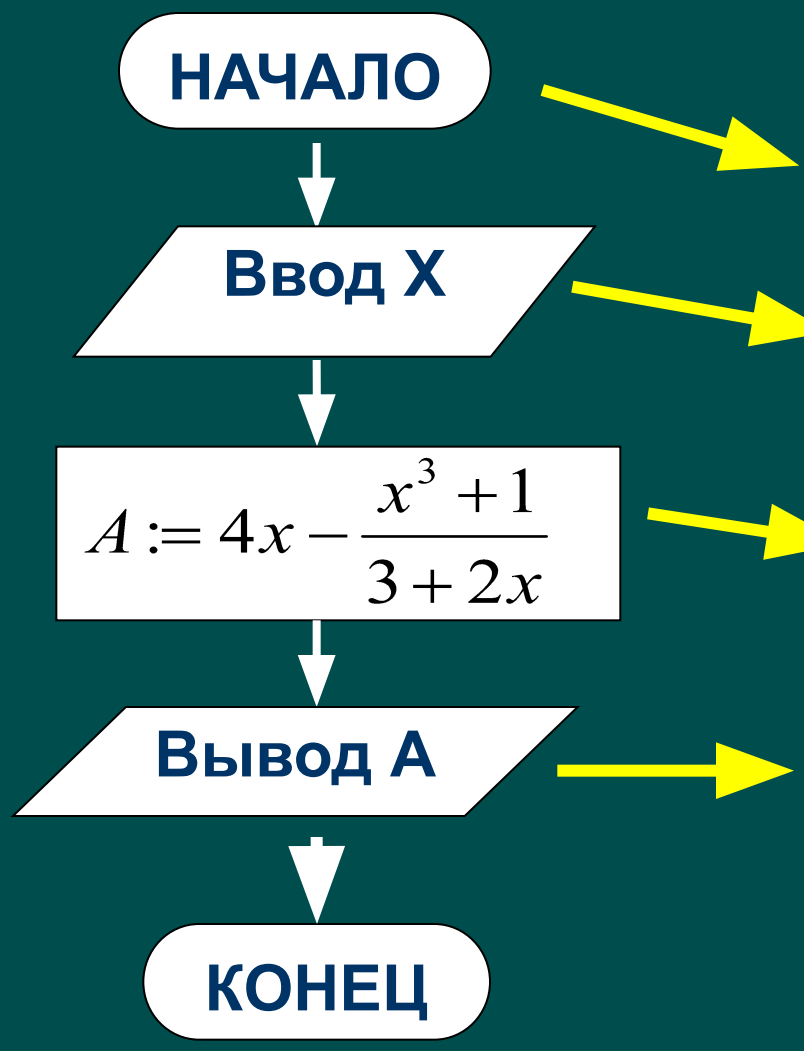


Программа:

```
PROGRAM  ARIFM;
   VAR  X,A:REAL;
BEGIN
   WRITE ('X=');
   READLN (X);
```

Решение:

Блок-схема:



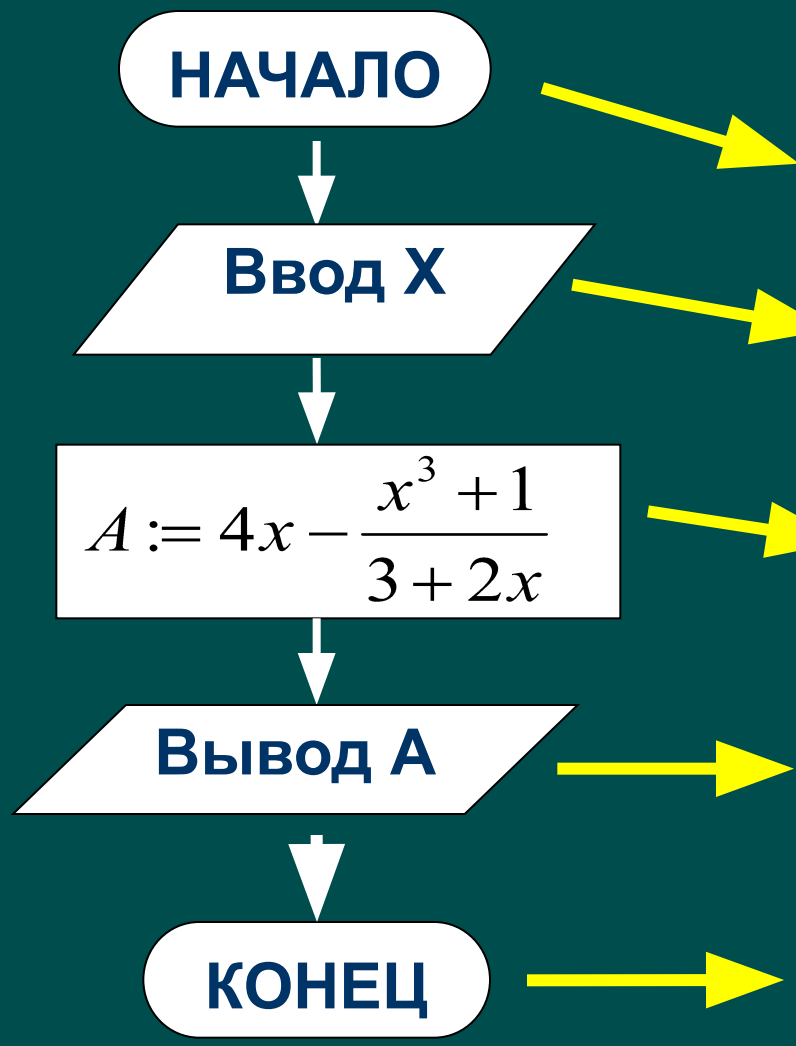
Программа:

```
PROGRAM  ARIFM;
   VAR  X,A:REAL;
BEGIN
   WRITE ('X=');
   READLN (X);

   A:=4*X - (X*X*X + 1)/(3+2*X);
```

Решение:

Блок-схема:



Программа:

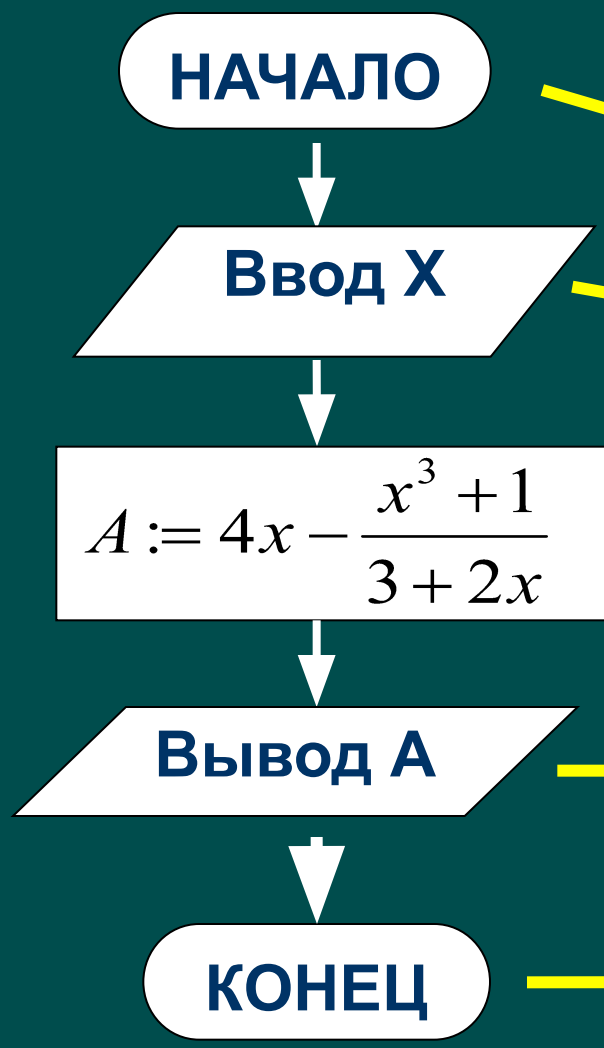
```
PROGRAM  ARIFM;
   VAR  X,A:REAL;
BEGIN
   WRITE ('X=');
   READLN (X);

   A:=4*X - (X*X*X + 1)/(3+2*X);

   WRITELN ( A : 5 :3);
```

Решение:

Блок-схема:



Программа:

```
PROGRAM  ARIFM;
   VAR  X,A:REAL;
BEGIN
   WRITE ('X=');
   READLN (X);

   A:=4*X - (X*X*X + 1)/(3+2*X);

   WRITELN ( A : 5 :3);

READLN;
END.
```

Совет:

В первый раз программу запускаете со значениями из трассировочной таблицы, чтобы проверить правильно ли вы ее написали. Если ответы совпадают, значит все в порядке, и можно находить все остальные значения (по задаче).