

1. Цель работы

Целью работы является изучение основных управляющих структур программирования и функций.

2. Задание

Вычислить и вывести на экран в виде таблицы значение функции F на интервале $[X_{нач}, X_{конеч}]$ с шагом dx . Вид функции F определяется индивидуальным вариантом. Коэффициенты a , b , c являются действительными числами. Значения a , b , c , $X_{нач}$, $X_{конеч}$, dx вводятся с клавиатуры.

3. Описание созданных функций

Для реализации задания были использованы следующие функции:

Имя: Check ();

Назначение: ввод и проверка вводимых значений.

Входные данные:

- a – переменная типа double;
- b – переменная типа double;
- c – переменная типа double.

Выходные данные: нет

Побочный эффект: отсутствует.

Прототип: Check ();

Алгоритм:

- псевдокод
Вывод информативного сообщения в консоль
Отдельный цикл с проверкой введенных значений для каждой переменной

Имя: output();

Назначение: принимает значения из функции Check(); для расчёта, ввод диапазона значений функции, вывод таблицы.

Входные данные:

- a – переменная типа double;
- b – переменная типа double;
- c – переменная типа double;
- Startx – начало диапазона;

- Finishx – конец диапазона;
- Dx – шаг;
- F – переменная принимающая вычисленного значения.

a	b	c	Startx	Finishx	Dx	F
0.25	5.6	3	5	10	1.3	(5;2)
						(6.3;2.325)
						(7.6;2.65)
						(8.9;2.975)

Побочный эффект: отсутствует.

Прототип: output();

Алгоритм:

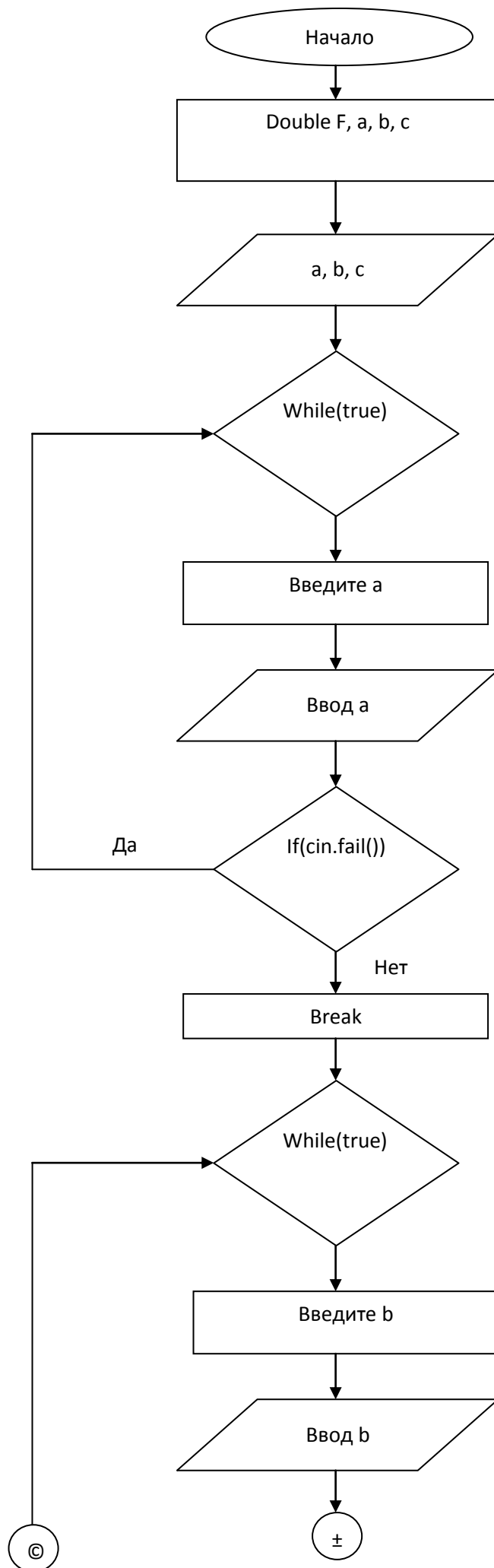
- псевдокод

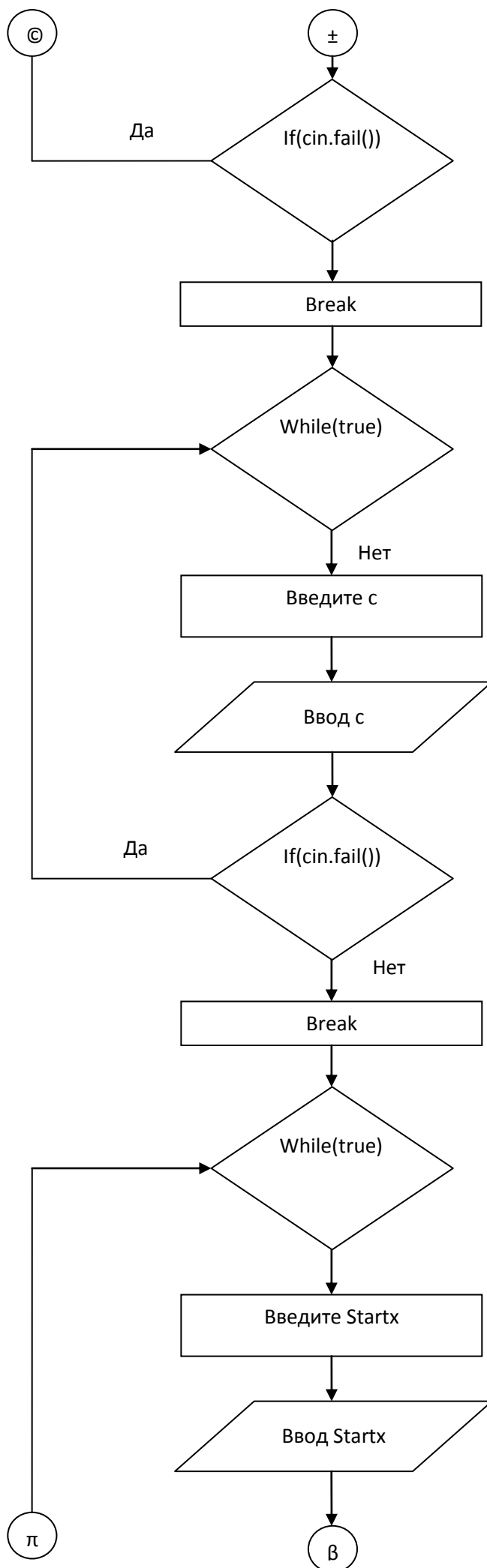
Для ввода каждого значения отдельный цикл с проверкой введённого значения

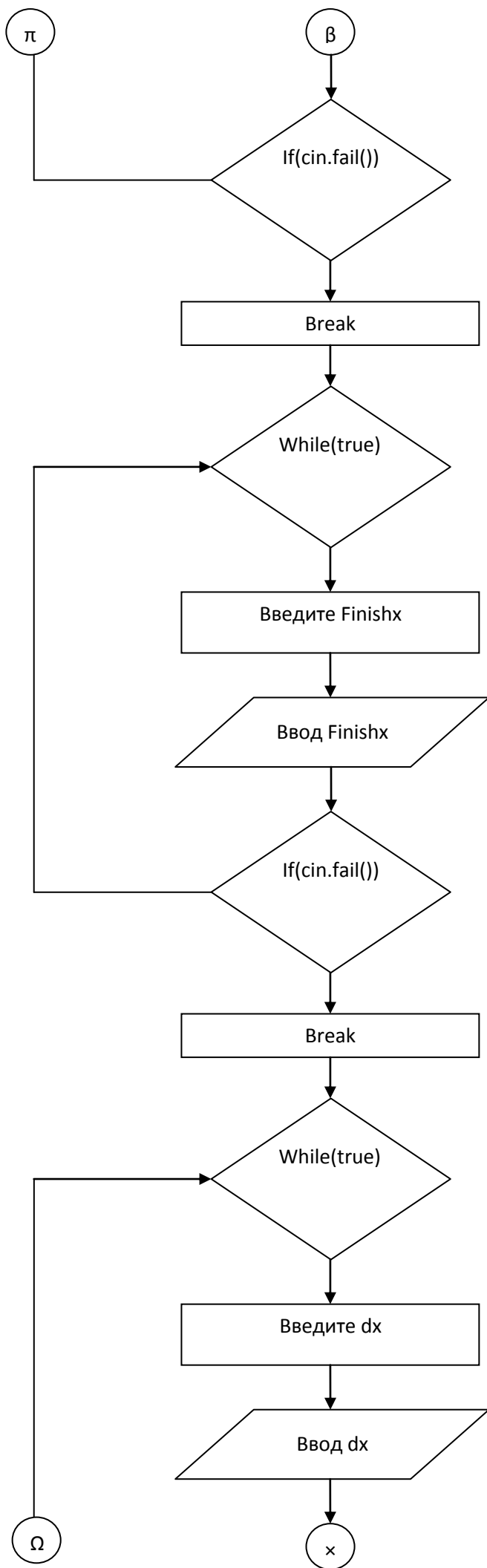
Очистка консоли

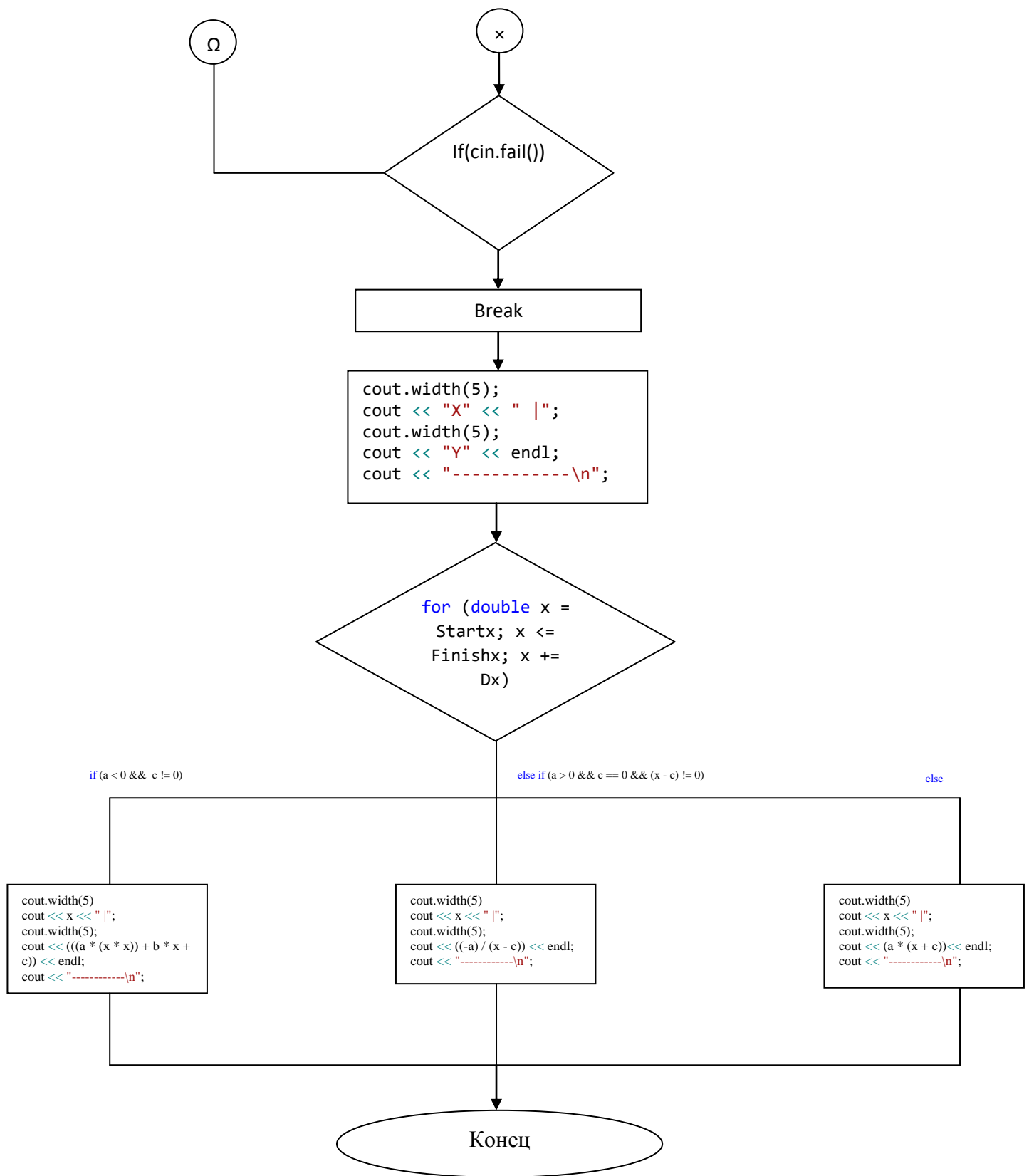
Выделение фиксированного места для вывода таблицы

Цикл для расчёта значений по определённой формуле









4. Листинг программы

```
1. #include <iostream>
2. #include <conio.h>

3. using namespace std; // использование имен пространства std

4. double F, a, b, c; // объявление глобальных переменных

5. void Check() // функция ввода и проверки вводимых значений
6. {
7.     cout << "\t\t***Программа по вычислению кусочной функции***\n\n"; //вывод
        сообщения в консоль
8.     cout << "Функция задана формулами:\n"
9.     << "F = (a * (x*x)) + b при a < 0 и b > 0 \n"
10.    << "F = (-a) / (x - c) при a > 0 и c = 0 \n"
11.    << "F = a * (x + c) при всех остальных случаях\n\n"; //вывод сообщения в консоль

12. while (true) // начало цикла
13. {
14.     cout << "Введите a: "; //вывод сообщения в консоль
15.     cin >> a; // ввод значения

16. if (cin.fail()) // начало проверки (если ввод не удался то )
17. {
18.     cin.clear(); // очистка потока
19.     cin.ignore(32767, '\n'); //игнорирование значений
20.     cout << "Введите значение a заново" << endl; // вывод сообщения в консоль
21. }
22. else // второе условие проверки (если первое не выполнится)
23. break; // оператор прерывания цикла
24. }

25. while (true)// начало цикла
26. {
27.     cout << "Введите b: "; //вывод сообщения в консоль
28.     cin >> b; // ввод значения

29. if (cin.fail()) // начало проверки (если ввод не удался то )
30. {
31.     cin.clear(); // очистка потока
32.     cin.ignore(32767, '\n'); //игнорирование значений
33.     cout << "Введите значение a заново" << endl; // вывод сообщения в консоль
34. }
35. else // второе условие проверки (если первое не выполнится)
36. break; // оператор прерывания цикла
37. }

38. while (true) // начало цикла
39. {
40.     cout << "Введите c: "; //вывод сообщения в консоль
41.     cin >> c; // ввод значения

42. if (cin.fail()) // начало проверки (если ввод не удался то )
43. {
44.     cin.clear(); // очистка потока
45.     cin.ignore(32767, '\n'); //игнорирование значений
46.     cout << "Введите значение a заново" << endl; // вывод сообщения в консоль
47. }
```

```

48. else // второе условие проверки (если первое не выполнится)
49. break; // оператор прерывания цикла
50. }
51. }

52. double output ( )
53. {
54. double Startx, Finishx, Dx; // объявление локальных переменных

55. while (true) // начало цикла
56. {
57. cout << "Введите начало интервала\n"; //вывод сообщения в консоль
58. cin >> Startx; // ввод значения

59. if (cin.fail()) // начало проверки (если ввод не удался то )
60. {
61. cin.clear(); // очистка потока
62. cin.ignore(32767, '\n'); //игнорирование значений
63. cout << "Введено не верное значение\n" << endl; // вывод сообщения в консоль
64. }
65. else // второе условие проверки (если первое не выполнится)
66. break; // оператор прерывания цикла
67. }
68. while (true) // начало цикла
69. {
70. cout << "Введите конец интервала\n"; //вывод сообщения в консоль
71. cin >> Finishx; // ввод значения

72. if (cin.fail()) // начало проверки (если ввод не удался то )
73. {
74. cin.clear(); // очистка потока
75. cin.ignore(32767, '\n'); //игнорирование значений
76. cout << "Введено не верное значение\n" << endl; // вывод сообщения в консоль
77. }
78. else // второе условие проверки (если первое не выполнится)
79. break; // оператор прерывания цикла
80. }
81. while (true) // начало цикла
82. {
83. cout << "Введите шаг\n"; //вывод сообщения в консоль
84. cin >> Dx; // ввод значения

85. if (cin.fail()) // начало проверки (если ввод не удался то )
86. {
87. cin.clear(); // очистка потока
88. cin.ignore(32767, '\n'); //игнорирование значений
89. cout << "Введено не верное значение\n" << endl; // вывод сообщения в консоль
90. }
91. else // второе условие проверки (если первое не выполнится)
92. break; // оператор прерывания цикла
93. }

94. system("cls"); // очистка консоли

95. cout.width(5); // задаём фиксированный размер до
96. cout << "X" << " |"; // вывод сообщения в консоль
97. cout.width(5); // задаём фиксированный размер после
98. cout << "Y" << endl; // вывод сообщения в консоль
99. cout << "-----\n"; // вывод сообщения в консоль

100. for (double x = Startx; x <= Finishx; x += Dx) // начало цикла с
    предусловием
101. {

```



```

102.     if (a < 0 && c != 0) // первая итерация проверки
103.     {
104.         cout.width(5); // задаём фиксированный размер до
105.         cout << x << " |"; // вывод сообщения в консоль
106.         cout.width(5); // задаём фиксированный размер после
107.         cout << (((a * (x * x)) + b * x + c)) << endl; // вывод сообщения в консоль
108.         cout << "-----\n"; // вывод сообщения в консоль
109.     }
110.     else if (a > 0 && c == 0 && (x - c) != 0) // вторая итерация проверки
111.     {
112.         cout.width(5); // задаём фиксированный размер до
113.         cout << x << " |"; // вывод сообщения в консоль
114.         cout.width(5); // задаём фиксированный размер после
115.         cout << ((-a) / (x - c)) << endl; // вывод сообщения в консоль
116.         cout << "-----\n"; // вывод сообщения в консоль
117.     }
118.     else // третья итерация проверки
119.     {
120.         cout.width(5); // задаём фиксированный размер до
121.         cout << x << " |"; // вывод сообщения в консоль
122.         cout.width(5); // задаём фиксированный размер после
123.         cout << (a * (x + c)) << endl; // вывод сообщения в консоль
124.         cout << "-----\n"; // вывод сообщения в консоль
125.     }
126. }

127.     system ("pause"); // сообщение консоли о паузе
128.     return 0; // возвращаем значение 0
129. }

130. int main() //основная функция
131. {
132.     setlocale(LC_ALL, "Russian"); // подключение русского языка

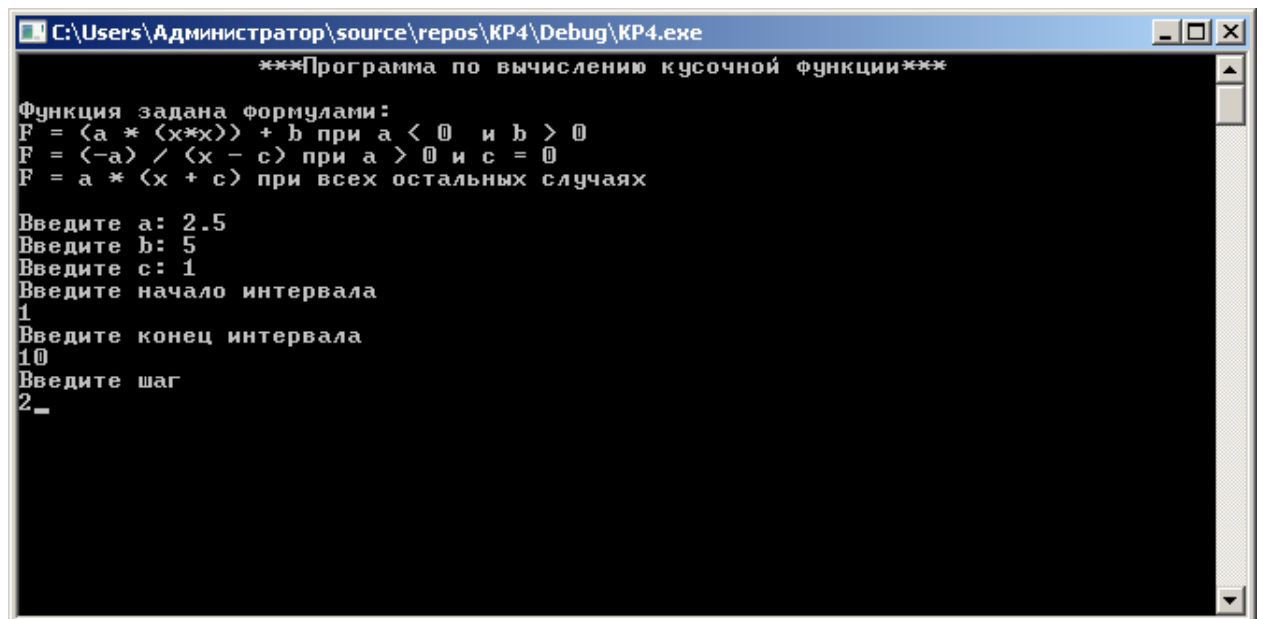
133.     Check(); // подключени функции
134.     output(); // подключение функции

135.     return 0; // возврат значени 0
136. }

```

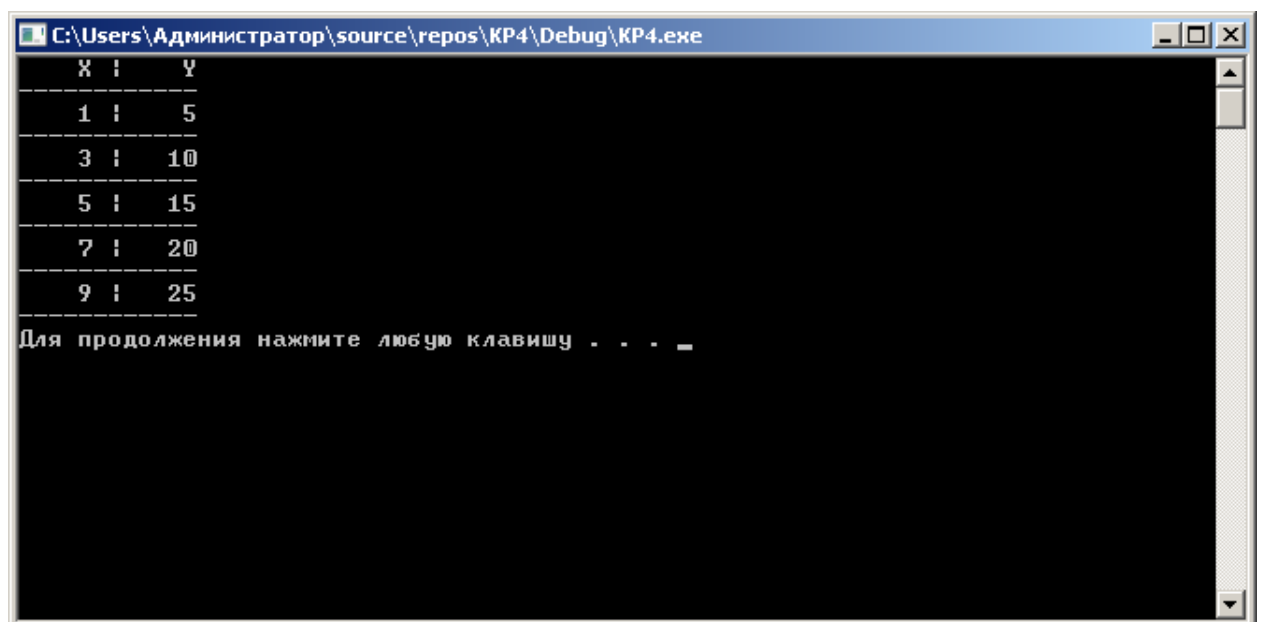
5. Пример выполнения программы

Ниже приведён пример выполнения программы



```
C:\Users\Администратор\source\repos\KP4\Debug\KP4.exe
***Программа по вычислению кусочной функции***
Функция задана формулами:
F = <a * (x*x)> + b при a < 0 и b > 0
F = <-a> / <x - c> при a > 0 и c = 0
F = a * <x + c> при всех остальных случаях
Введите a: 2.5
Введите b: 5
Введите c: 1
Введите начало интервала
1
Введите конец интервала
10
Введите шаг
2_
```

Рис 1. Пример выполнения программы



```
C:\Users\Администратор\source\repos\KP4\Debug\KP4.exe
X | Y
---|---
1 | 5
3 | 10
5 | 15
7 | 20
9 | 25
Для продолжения нажмите любую клавишу . . . _
```

Рис 2. Пример выполнения программы