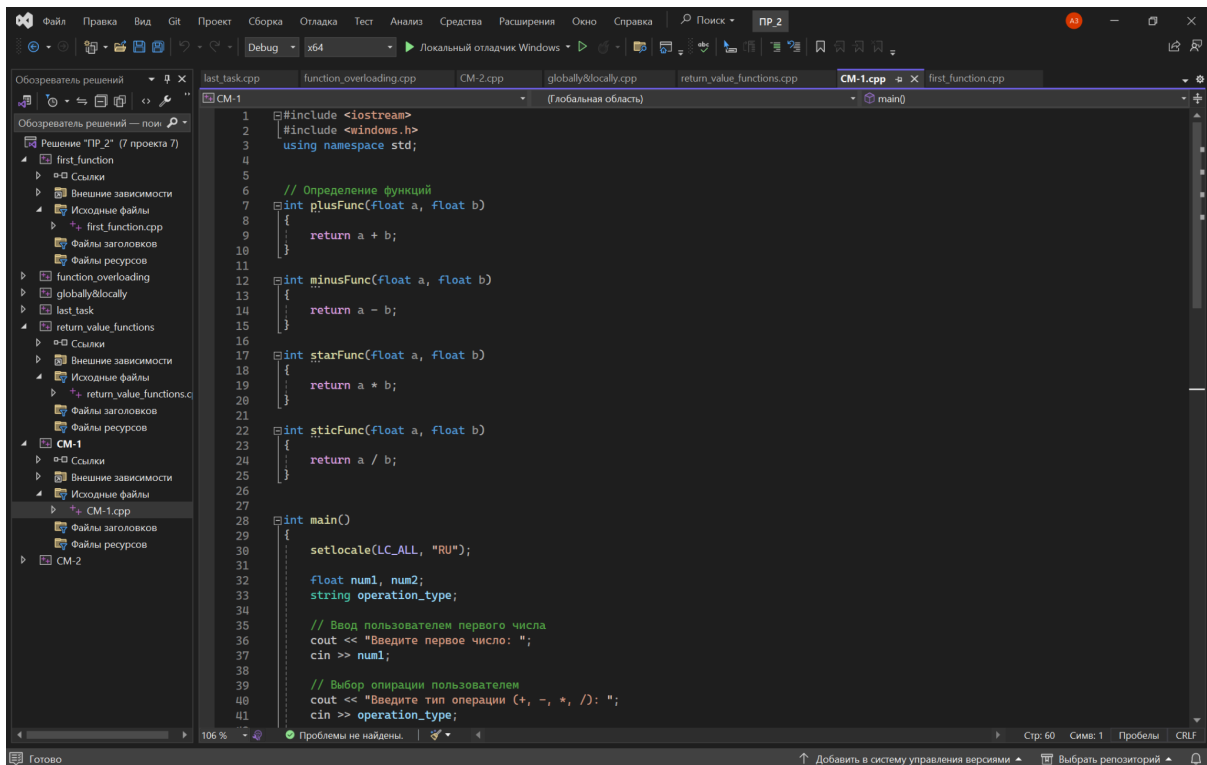
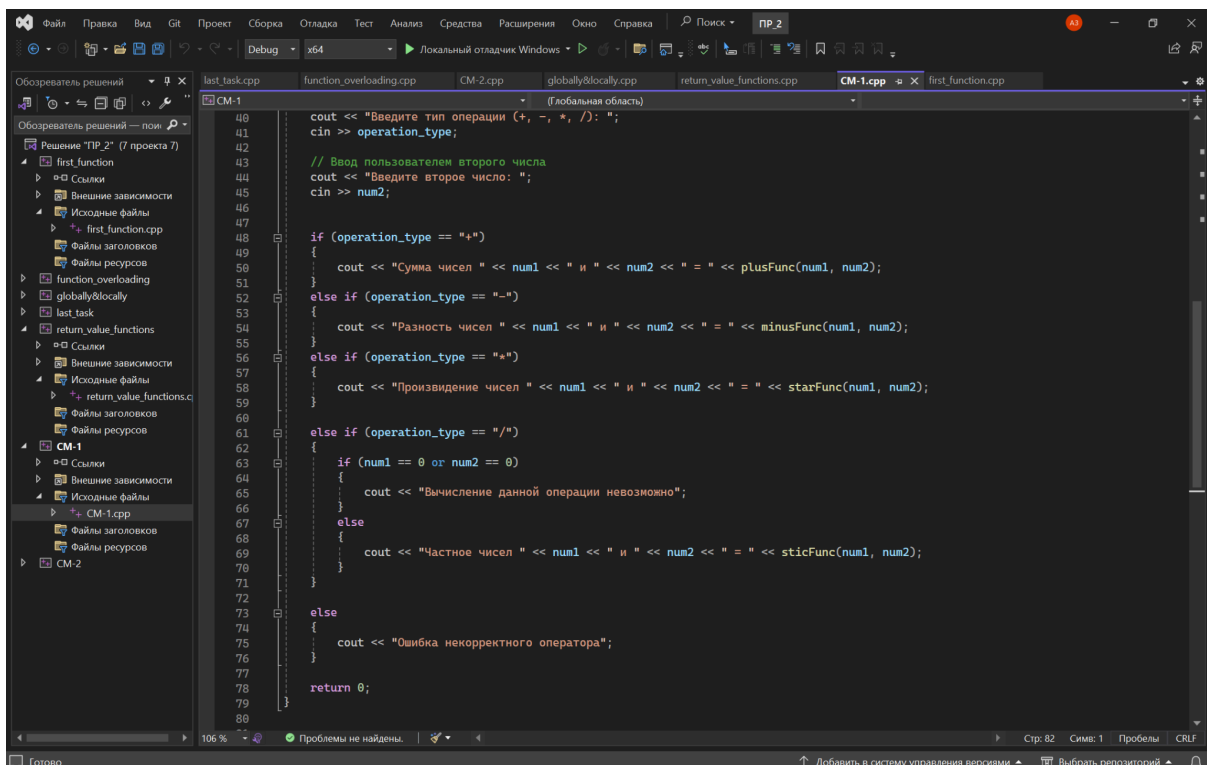


Самостоятельные работы

Первая самостоятельная программа



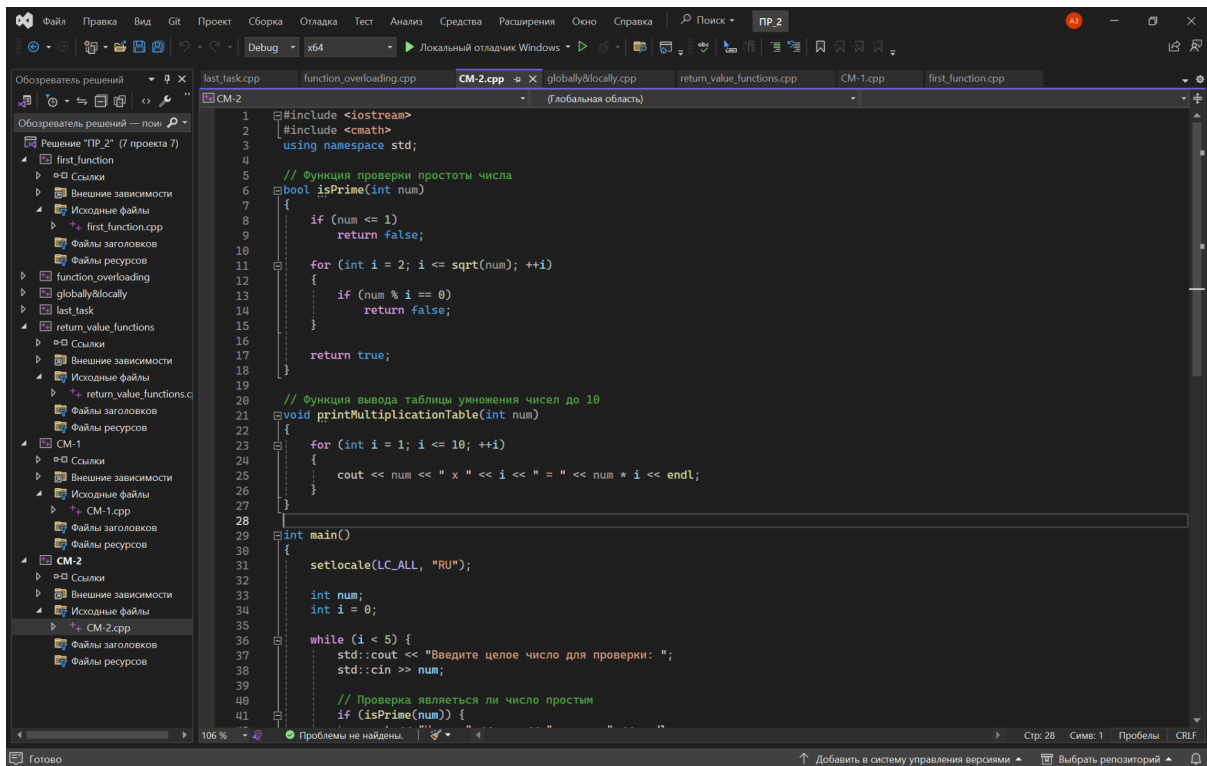
```
1 #include <iostream>
2 #include <windows.h>
3 using namespace std;
4
5 // Определение функций
6 int plusFunc(float a, float b)
7 {
8     return a + b;
9 }
10
11 int minusFunc(float a, float b)
12 {
13     return a - b;
14 }
15
16 int starFunc(float a, float b)
17 {
18     return a * b;
19 }
20
21 int sticFunc(float a, float b)
22 {
23     return a / b;
24 }
25
26
27 int main()
28 {
29     setlocale(LC_ALL, "RU");
30
31     float num1, num2;
32     string operation_type;
33
34     // Ввод пользователем первого числа
35     cout << "Введите первое число: ";
36     cin >> num1;
37
38     // Выбор операции пользователем
39     cout << "Введите тип операции (+, -, *, /): ";
40     cin >> operation_type;
```



```
40
41
42     cout << "Введите тип операции (+, -, *, /): ";
43     cin >> operation_type;
44
45     // Ввод пользователем второго числа
46     cout << "Введите второе число: ";
47     cin >> num2;
48
49     if (operation_type == "+")
50     {
51         cout << "Сумма чисел " << num1 << " и " << num2 << " = " << plusFunc(num1, num2);
52     }
53     else if (operation_type == "-")
54     {
55         cout << "Разность чисел " << num1 << " и " << num2 << " = " << minusFunc(num1, num2);
56     }
57     else if (operation_type == "*")
58     {
59         cout << "Произведение чисел " << num1 << " и " << num2 << " = " << starFunc(num1, num2);
60     }
61     else if (operation_type == "/")
62     {
63         if (num1 == 0 or num2 == 0)
64         {
65             cout << "Вычисление данной операции невозможно";
66         }
67         else
68         {
69             cout << "Частное чисел " << num1 << " и " << num2 << " = " << sticFunc(num1, num2);
70         }
71     }
72     else
73     {
74         cout << "Ошибка некорректного оператора";
75     }
76
77     return 0;
78 }
79
80
```

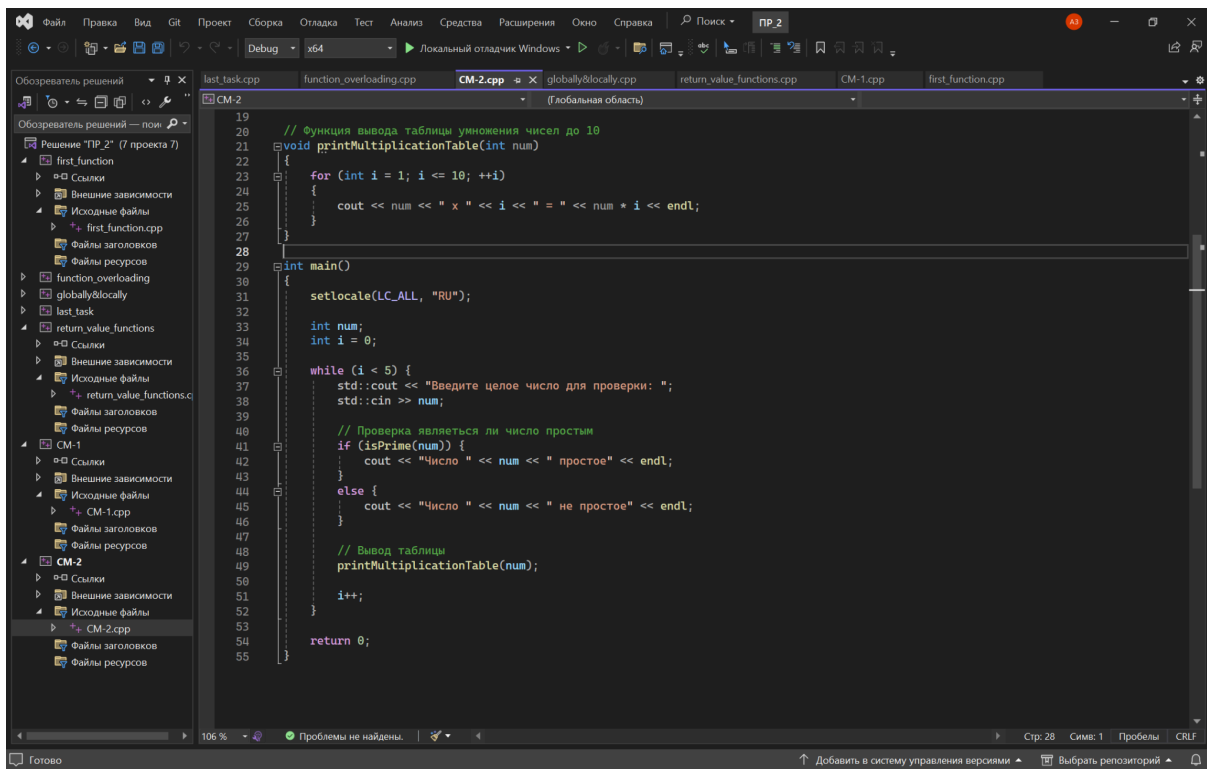
```
Консоль отладки Microsoft \ x
Введите первое число: 55
Введите тип операции (+, -, *, /): *
Введите второе число: 11
Произведение чисел 55 и 11 = 605
C:\Users\Trenj\source\repos\ПР_2\х64\Debug\CM-1.exe (процесс 11200) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:
```

Вторая самостоятельная программа



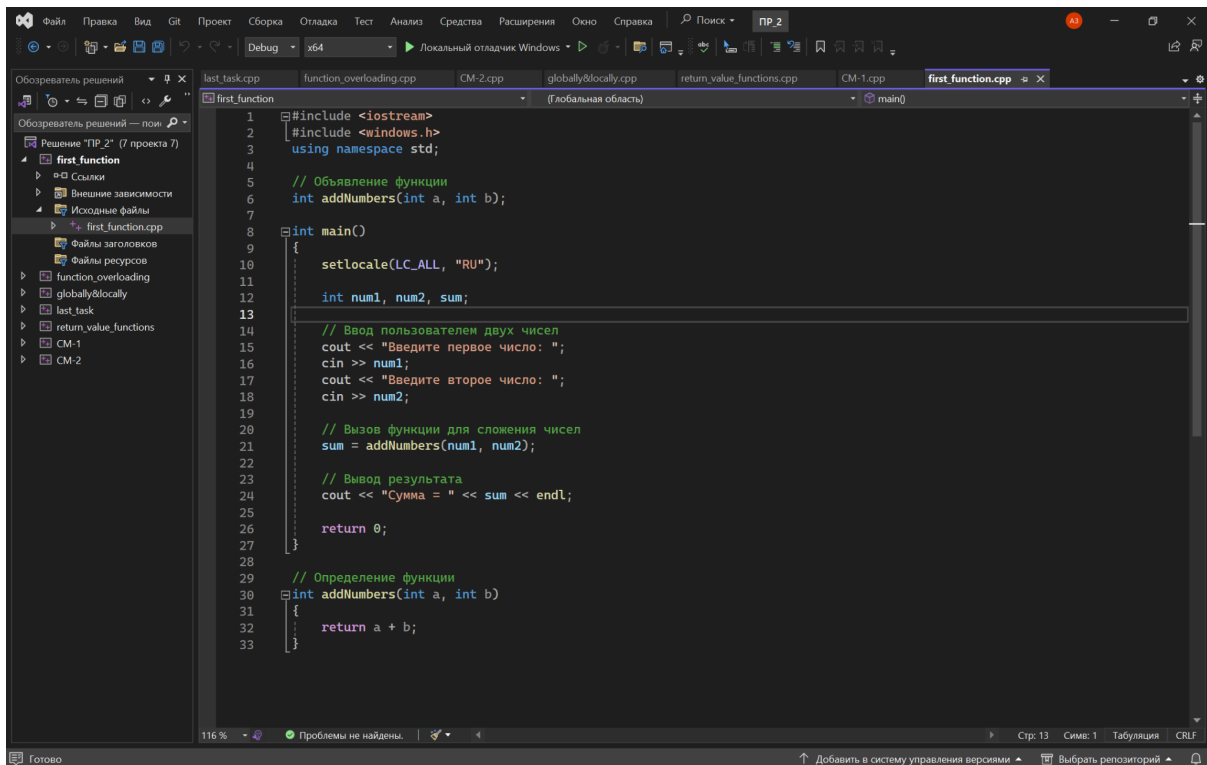
The screenshot shows the Visual Studio Code editor with a C++ project named "ПР_2". The file explorer on the left shows the project structure, including files like "first_function.cpp", "function_overloading.cpp", "globally&locally.cpp", "return_value_functions.cpp", "CM-1.cpp", and "CM-2.cpp". The main editor window displays the code for "CM-2.cpp".

```
1 #include <iostream>
2 #include <cmath>
3 using namespace std;
4
5 // Функция проверки простоты числа
6 bool isPrime(int num)
7 {
8     if (num <= 1)
9         return false;
10
11     for (int i = 2; i <= sqrt(num); ++i)
12     {
13         if (num % i == 0)
14             return false;
15     }
16
17     return true;
18 }
19
20 // Функция вывода таблицы умножения чисел до 10
21 void printMultiplicationTable(int num)
22 {
23     for (int i = 1; i <= 10; ++i)
24     {
25         cout << num << " x " << i << " = " << num * i << endl;
26     }
27 }
28
29 int main()
30 {
31     setlocale(LC_ALL, "RU");
32
33     int num;
34     int i = 0;
35
36     while (i < 5) {
37         std::cout << "Введите целое число для проверки: ";
38         std::cin >> num;
39
40         // Проверка является ли число простым
41         if (isPrime(num)) {
```

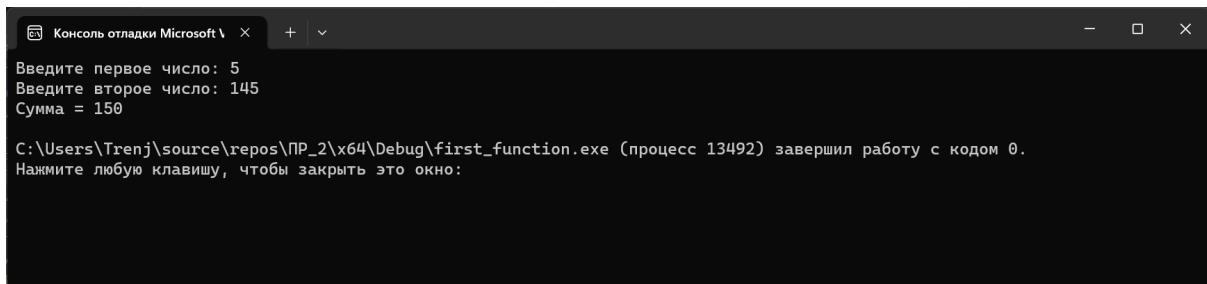


Реализация данных в файле программ

Первая программа реализации функции



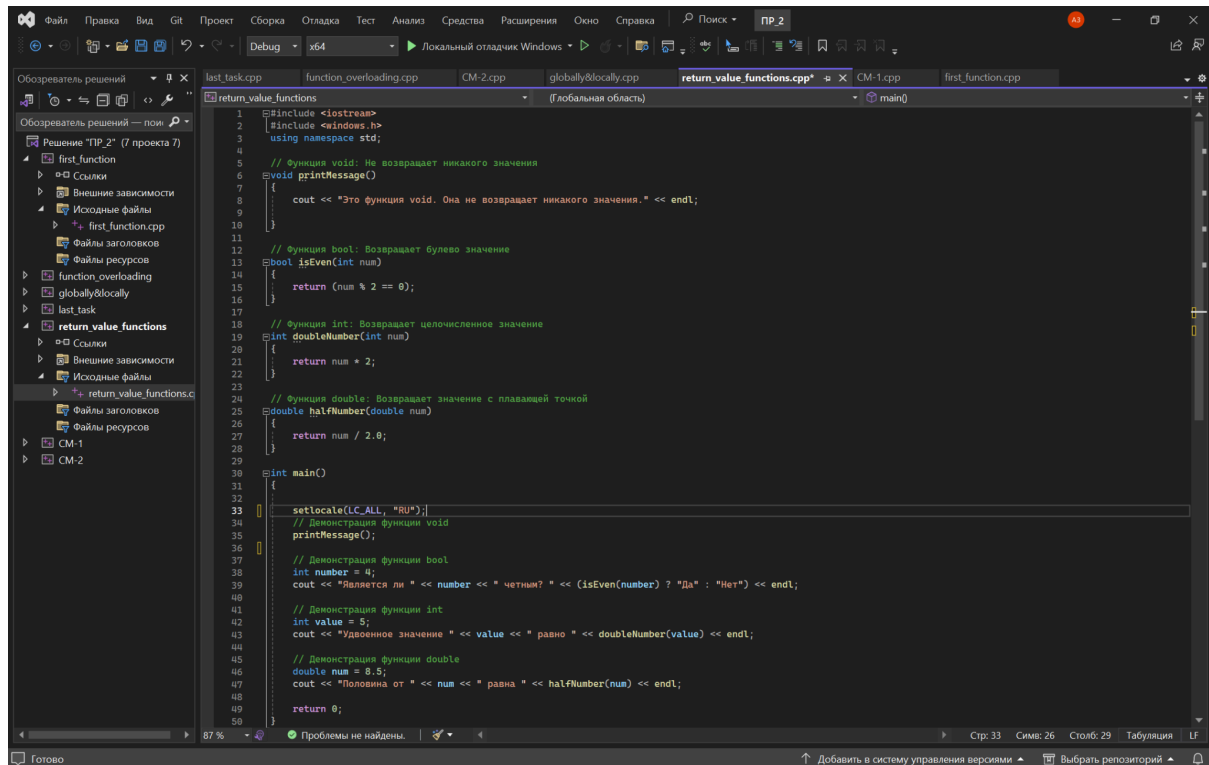
```
1 #include <iostream>
2 #include <windows.h>
3 using namespace std;
4
5 // Объявление функции
6 int addNumbers(int a, int b);
7
8 int main()
9 {
10     setlocale(LC_ALL, "RU");
11
12     int num1, num2, sum;
13
14     // Ввод пользователем двух чисел
15     cout << "Введите первое число: ";
16     cin >> num1;
17     cout << "Введите второе число: ";
18     cin >> num2;
19
20     // Вызов функции для сложения чисел
21     sum = addNumbers(num1, num2);
22
23     // Вывод результата
24     cout << "Сумма = " << sum << endl;
25
26     return 0;
27
28     // Определение функции
29     int addNumbers(int a, int b)
30     {
31         return a + b;
32     }
33 }
```



```
Консоль отладки Microsoft \
Введите первое число: 5
Введите второе число: 145
Сумма = 150

C:\Users\Trenj\source\repos\PP_2\x64\Debug\first_function.exe (процесс 13492) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:
```

Программа “Параметры Функций и Типы Возвращаемых Значений”

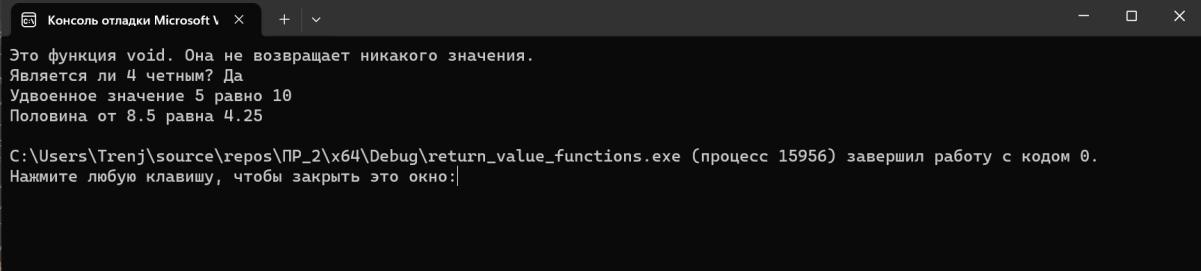


The screenshot shows the Visual Studio Code editor with a C++ project named "PP_2". The active file is "return_value_functions.cpp". The code defines several functions: a void function, a bool function, an int function, and a double function. The main function calls these functions and prints their results. The debug console shows the output of the program.

```
1 #include <iostream>
2 #include <windows.h>
3 using namespace std;
4
5 // Функция void: Не возвращает никакого значения
6 void printMessage()
7 {
8     cout << "Это функция void. Она не возвращает никакого значения." << endl;
9 }
10
11
12 // Функция bool: Возвращает булево значение
13 bool isEven(int num)
14 {
15     return (num % 2 == 0);
16 }
17
18 // Функция int: Возвращает целочисленное значение
19 int doubleNumber(int num)
20 {
21     return num * 2;
22 }
23
24 // Функция double: Возвращает значение с плавающей точкой
25 double halfNumber(double num)
26 {
27     return num / 2.0;
28 }
29
30 int main()
31 {
32     setlocale(LC_ALL, "RU");
33     // Демонстрация функции void
34     printMessage();
35
36     // Демонстрация функции bool
37     int number = 4;
38     cout << "Является ли " << number << " четным? " << (isEven(number) ? "Да" : "Нет") << endl;
39
40     // Демонстрация функции int
41     int value = 5;
42     cout << "Удвоенное значение " << value << " равно " << doubleNumber(value) << endl;
43
44     // Демонстрация функции double
45     double num = 8.5;
46     cout << "Половина от " << num << " равна " << halfNumber(num) << endl;
47
48     return 0;
49 }
```

Готово

Добавить в систему управления версиями | Выбрать репозиторий

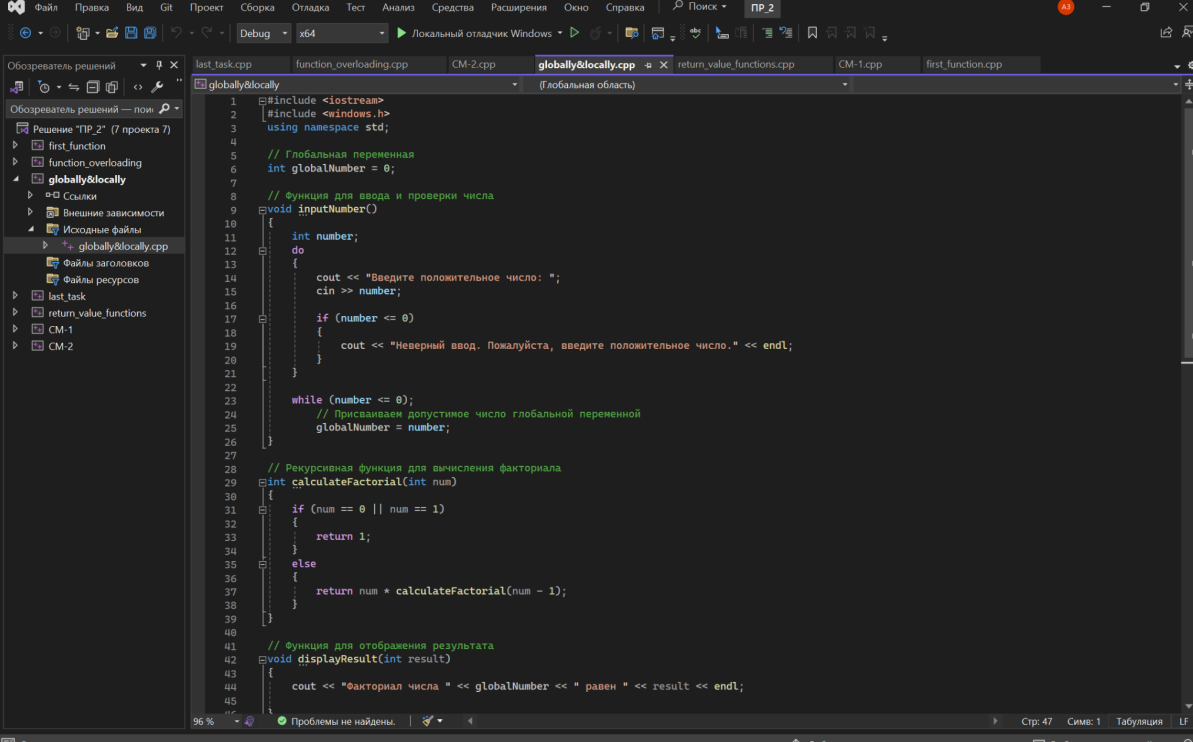


Консоль отладки Microsoft Visual Studio

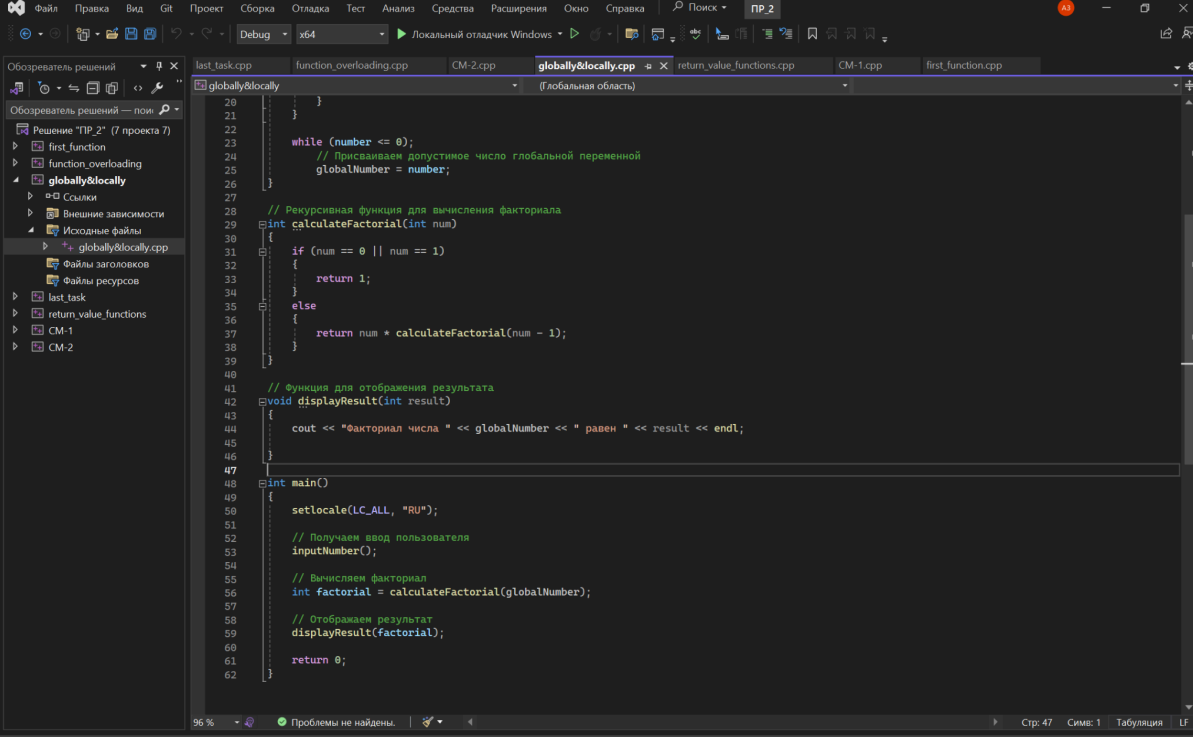
Это функция void. Она не возвращает никакого значения.
Является ли 4 четным? Да
Удвоенное значение 5 равно 10
Половина от 8.5 равна 4.25

C:\Users\Trenj\source\repos\PP_2\x64\Debug\return_value_functions.exe (процесс 15956) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:

Программа “Используя глобальные и локальные переменные, рекурсию, проверьте ввод пользователя”



```
1 #include <iostream>
2 #include <windows.h>
3 using namespace std;
4
5 // Глобальная переменная
6 int globalNumber = 0;
7
8 // Функция для ввода и проверки числа
9 void inputNumber()
10 {
11     int number;
12     do
13     {
14         cout << "Введите положительное число: ";
15         cin >> number;
16
17         if (number <= 0)
18         {
19             cout << "Неверный ввод. Пожалуйста, введите положительное число." << endl;
20         }
21     }
22     while (number <= 0);
23
24     // Присваиваем допустимое число глобальной переменной
25     globalNumber = number;
26 }
27
28 // Рекурсивная функция для вычисления факториала
29 int calculateFactorial(int num)
30 {
31     if (num == 0 || num == 1)
32     {
33         return 1;
34     }
35     else
36     {
37         return num * calculateFactorial(num - 1);
38     }
39 }
40
41 // Функция для отображения результата
42 void displayResult(int result)
43 {
44     cout << "Факториал числа " << globalNumber << " равен " << result << endl;
45 }
46
```

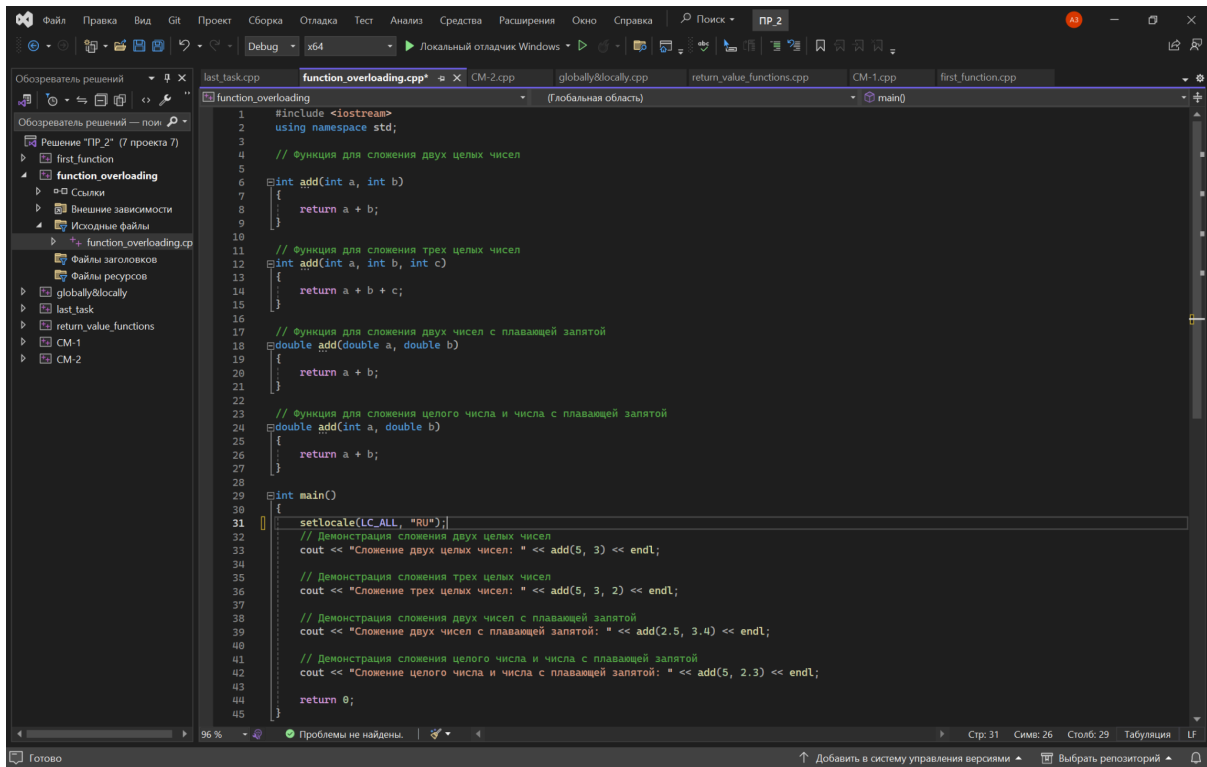


```
20 }
21 }
22
23 while (number <= 0);
24 // Присваиваем допустимое число глобальной переменной
25 globalNumber = number;
26
27 // Рекурсивная функция для вычисления факториала
28 int calculateFactorial(int num)
29 {
30     if (num == 0 || num == 1)
31     {
32         return 1;
33     }
34     else
35     {
36         return num * calculateFactorial(num - 1);
37     }
38 }
39
40 // Функция для отображения результата
41 void displayResult(int result)
42 {
43     cout << "Факториал числа " << globalNumber << " равен " << result << endl;
44 }
45
46
47 int main()
48 {
49     setlocale(LC_ALL, "RU");
50
51     // Получаем ввод пользователя
52     inputNumber();
53
54     // Вычисляем факториал
55     int factorial = calculateFactorial(globalNumber);
56
57     // Отображаем результат
58     displayResult(factorial);
59
60     return 0;
61 }
62
```

```
Консоль отладки Microsoft \ x
+
-
Введите положительное число: 15
Факториал числа 15 равен 2004310016

C:\Users\Trenj\source\repos\ПР_2\х64\Debug\globally&locally.exe (процесс 13848) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:
```

Программа “Перегрузка Функций”

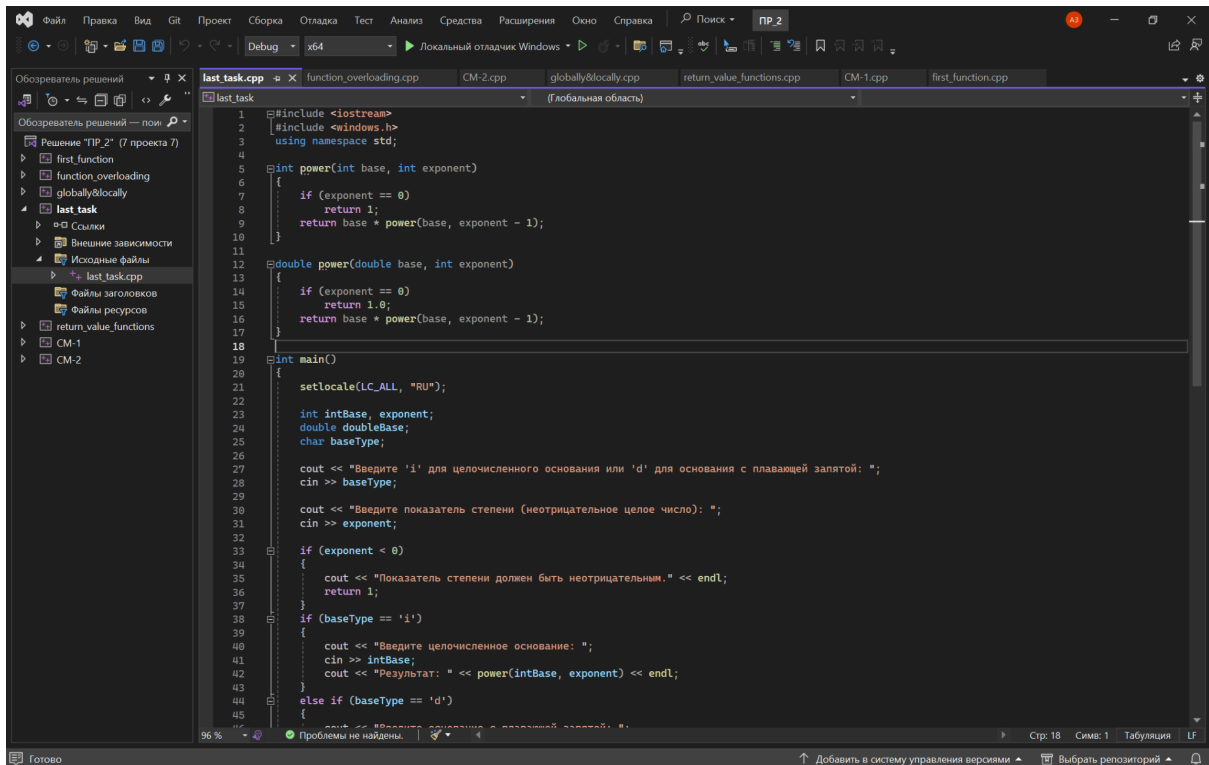


```
function_overloading.cpp (Глобальная область)
1 #include <iostream>
2 using namespace std;
3
4 // Функция для сложения двух целых чисел
5
6 int add(int a, int b)
7 {
8     return a + b;
9 }
10
11 // Функция для сложения трех целых чисел
12 int add(int a, int b, int c)
13 {
14     return a + b + c;
15 }
16
17 // Функция для сложения двух чисел с плавающей запятой
18 double add(double a, double b)
19 {
20     return a + b;
21 }
22
23 // Функция для сложения целого числа и числа с плавающей запятой
24 double add(int a, double b)
25 {
26     return a + b;
27 }
28
29 int main()
30 {
31     setlocale(LC_ALL, "RU");
32     // Демонстрация сложения двух целых чисел
33     cout << "Сложение двух целых чисел: " << add(5, 3) << endl;
34
35     // Демонстрация сложения трех целых чисел
36     cout << "Сложение трех целых чисел: " << add(5, 3, 2) << endl;
37
38     // Демонстрация сложения двух чисел с плавающей запятой
39     cout << "Сложение двух чисел с плавающей запятой: " << add(2.5, 3.4) << endl;
40
41     // Демонстрация сложения целого числа и числа с плавающей запятой
42     cout << "Сложение целого числа и числа с плавающей запятой: " << add(5, 2.3) << endl;
43
44     return 0;
45 }
```

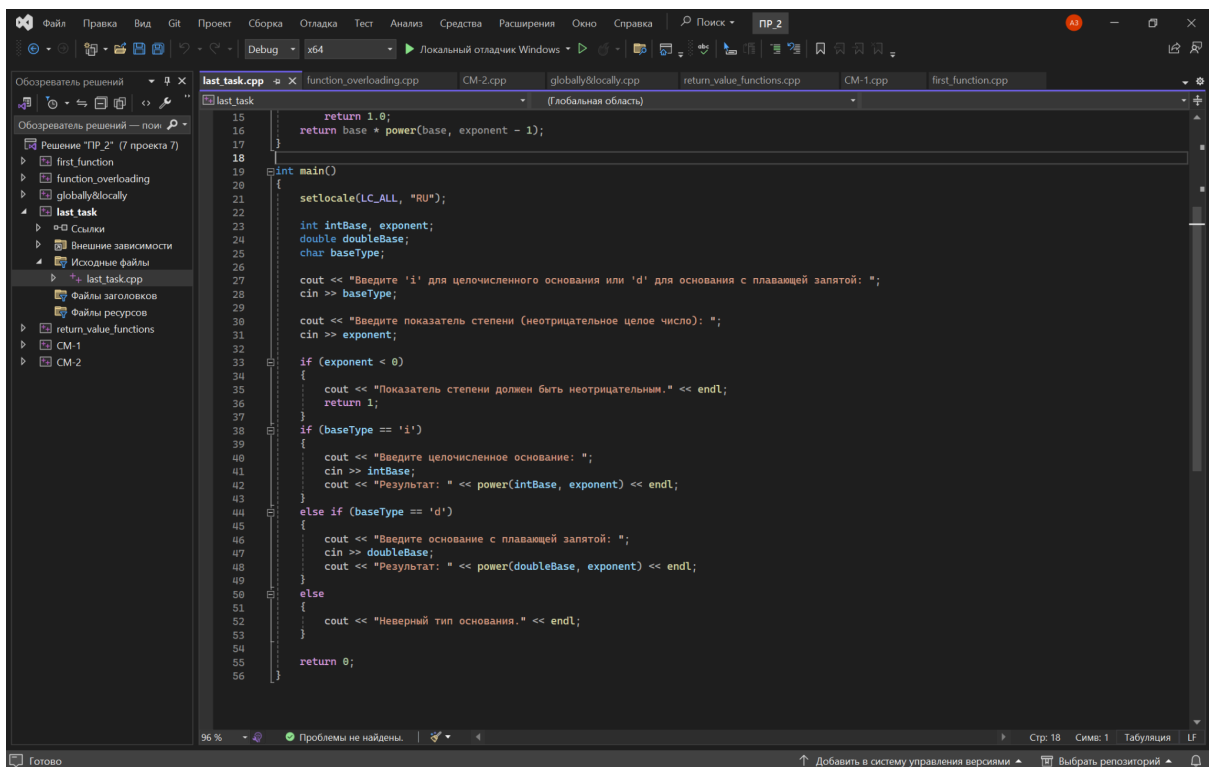
```
Консоль отладки Microsoft \ x
+
-
Сложение двух целых чисел: 8
Сложение трех целых чисел: 10
Сложение двух чисел с плавающей запятой: 5.9
Сложение целого числа и числа с плавающей запятой: 7.3

C:\Users\Trenj\source\repos\ПР_2\х64\Debug\function_overloading.exe (процесс 2324) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:
```

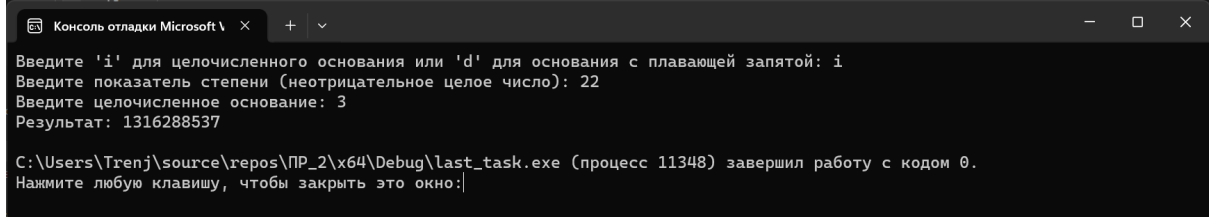
Последняя программа



```
1 #include <iostream>
2 #include <windows.h>
3 using namespace std;
4
5 int power(int base, int exponent)
6 {
7     if (exponent == 0)
8         return 1;
9     return base * power(base, exponent - 1);
10 }
11
12 double power(double base, int exponent)
13 {
14     if (exponent == 0)
15         return 1.0;
16     return base * power(base, exponent - 1);
17 }
18
19 int main()
20 {
21     setlocale(LC_ALL, "RU");
22
23     int intBase, exponent;
24     double doubleBase;
25     char baseType;
26
27     cout << "Введите 'i' для целочисленного основания или 'd' для основания с плавающей запятой: ";
28     cin >> baseType;
29
30     cout << "Введите показатель степени (неотрицательное целое число): ";
31     cin >> exponent;
32
33     if (exponent < 0)
34     {
35         cout << "Показатель степени должен быть неотрицательным." << endl;
36         return 1;
37     }
38     if (baseType == 'i')
39     {
40         cout << "Введите целочисленное основание: ";
41         cin >> intBase;
42         cout << "Результат: " << power(intBase, exponent) << endl;
43     }
44     else if (baseType == 'd')
45     {
46         cout << "Введите основание с плавающей запятой: ";
47         cin >> doubleBase;
48         cout << "Результат: " << power(doubleBase, exponent) << endl;
49     }
50     else
51     {
52         cout << "Неверный тип основания." << endl;
53     }
54     return 0;
55 }
```



```
15     return 1.0;
16     return base * power(base, exponent - 1);
17 }
18
19 int main()
20 {
21     setlocale(LC_ALL, "RU");
22
23     int intBase, exponent;
24     double doubleBase;
25     char baseType;
26
27     cout << "Введите 'i' для целочисленного основания или 'd' для основания с плавающей запятой: ";
28     cin >> baseType;
29
30     cout << "Введите показатель степени (неотрицательное целое число): ";
31     cin >> exponent;
32
33     if (exponent < 0)
34     {
35         cout << "Показатель степени должен быть неотрицательным." << endl;
36         return 1;
37     }
38     if (baseType == 'i')
39     {
40         cout << "Введите целочисленное основание: ";
41         cin >> intBase;
42         cout << "Результат: " << power(intBase, exponent) << endl;
43     }
44     else if (baseType == 'd')
45     {
46         cout << "Введите основание с плавающей запятой: ";
47         cin >> doubleBase;
48         cout << "Результат: " << power(doubleBase, exponent) << endl;
49     }
50     else
51     {
52         cout << "Неверный тип основания." << endl;
53     }
54     return 0;
55 }
```



```
Консоль отладки Microsoft \
Введите 'i' для целочисленного основания или 'd' для основания с плавающей запятой: i
Введите показатель степени (неотрицательное целое число): 22
Введите целочисленное основание: 3
Результат: 1316288537

C:\Users\Trenj\source\repos\PP_2\Debug\last_task.exe (процесс 11348) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:
```