«Отчет по П. 3.2 Простейшие классы, одиночное наследование» Никита Преображенский \*\*\*. Вариант 5.

Составить описание класса для представления комплексных чисел. Обеспечить выполнение операций сложения и вычитания комплексных чисел. Написать программу, демонстрирующую все разработанные элементы класса. Создать дочерний класс, выполняющий операцию умножения комплексных чисел.

1. Постановка задачи:

Функции класса Complex:

public:

Complex(); // Конструктор без параметров

void Input(); // Функция ввода значений

void Sum(); // Функция суммы

void Div(); // Функция разности

Функции класса Son:

public:

Complex(); // Конструктор без параметров

void Input(); // Функция ввода значений

void Sum(); // Функция суммы

void Div(); // Функция разности

Son(); // Конструктор без параметров

void Mult(); // Функция умножения

Входные данные + ограничения:

А) Переменная а. Тип int. Мнимая часть числа.

Исключения: строка, буква, вещественное число.

Б) Переменная b. Тип int. Действительная часть числа.

Исключения: строка, буква, вещественное число.

В) Переменная v. Тип int. Пункт меню.

Исключения: строка, буква, вещественное число, отрицательные числа.

Промежуточные данные + ограничения:

А) Переменная t. Тип int. Временная мнимая часть первого числа.

Исключения: строка, буква, вещественное число.

Б) Переменная d. Тип int. Временная мнимая часть второго числа.

Исключения: строка, буква, вещественное число.

В) Переменная x. Тип int. Временная действительная часть первого числа.

Исключения: строка, буква, вещественное число.

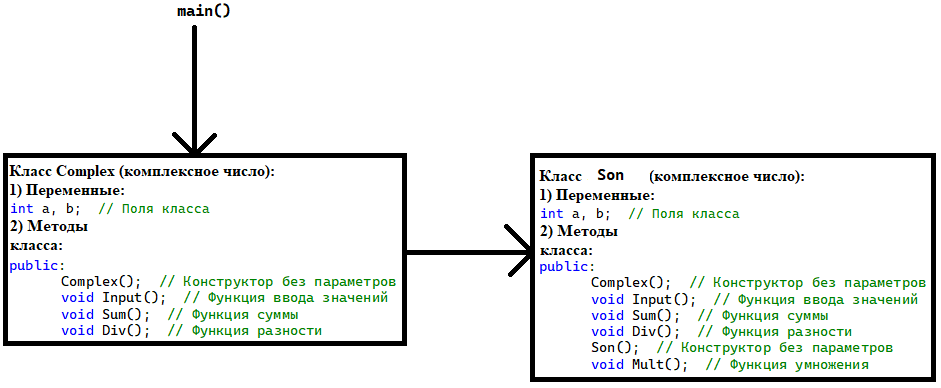
Г) Переменная z. Тип int. Временная действительная часть второго числа.

Исключения: строка, буква, вещественное число.

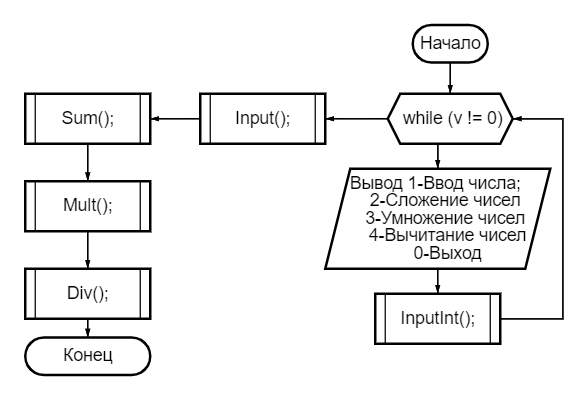
Среда разработки: Visual Studio 2022.

2. Разработка программы:

2.1 Описание классов, полей и методов классов.



2.2Разработка схемы алгоритма.



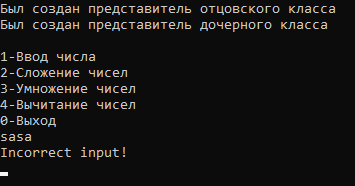
2.3 Описание пользовательского интерфейса.

«Взаимодействие пользователя с программой осуществляется с помощью диалога».

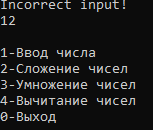
Примеры диалогов:

1. Выбор пункта меню:

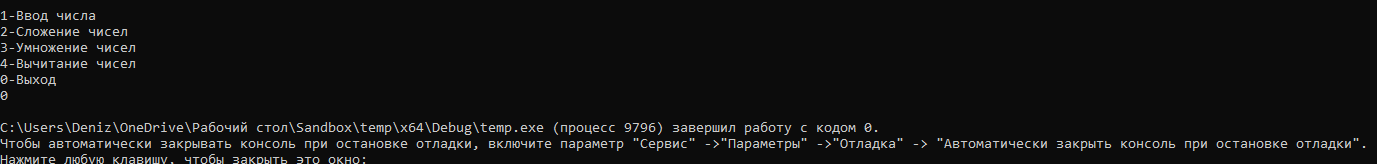
Вводимое значение – “sasa”



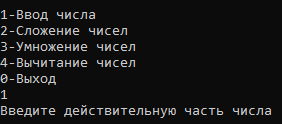
Вводимое значение – 12



Вводимое значение – 0



Вводимое значение – 1, 2, 3 , 4, 0

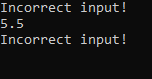


1. Ввод мнимой части числа:

Вводимое значение – “xxx”



Вводимое значение – 5.5

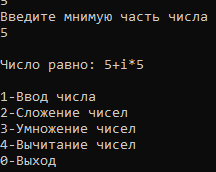


Вводимое значение – 5



1. Ввод действительной части числа:

Вводимое значение – 5



3. Реализация и тестирование программы

**3.1. Описание разработанной программы**

Для отцовского класса у нас определены методы для установления значений числа, расчёта суммы, разности комплексных чисел.

Для дочерного класса у нас определены всё те же методы, но добавлен метод произведения чисел.

В функции main мы создаём представителя дочерного класса. Затем с помощью меню поочерёдно вызываем функции вывода класса.

**3.2. Тестирование программы**

|  |  |  |
| --- | --- | --- |
| Входные данные | Результат программы | Смысл |
| “dsads” | Incorrect input! | Проверка на строку |
| 5 | Incorrect input! | Проверка на корректность ввода |
| 2 | Введите действительную часть первого числа | Проверка на корректность работы программы |
| “dsads” | Incorrect input! | Проверка на строку |
| 5.5 | Incorrect input! | Проверка на вещественные числа |
| 4 | Введите действительную часть первого числа  4  Введите мнимую часть первого числа | Проверка на корректность работы программы |
| “dsads” | Incorrect input! | Проверка на дробное число |
| 5.5 | Incorrect input! | Проверка на корректность работы программы |
| 4 | Число равно: 2+i\*4 | Проверка на корректность |
| 1  2  4 | Число равно: 2+i\*4  1-Ввод числа  2-Сложение чисел  3-Умножение чисел  4-Вычитание чисел  0-Выход | Проверка на корректность работы программы |

Возможный диалог:

Выбор пункта меню – ввод мнимой части первого числа – ввод мнимой части второго числа – ввод действительной части первого числа – ввод действительной части второго числа – выбор меню.

Код:

#include <iostream> // Подключение стандартной библиотеки

void InputInt(int\*); // Прототип функции ввода значений

using namespace std; // Использование стандартного пространства имён

class Complex // Описание класса комплексных чисел

{

public:

Complex() // Конструктор без параметров

{

cout << "Был создан представитель отцовского класса" << endl;

}

int a, b; // Поля класса

void Input(); // Функция ввода значений

void Sum(); // Функция суммы

void Div(); // Функция разности

};

void Complex::Input() // Описание функции ввода

{

cout << "Введите действительную часть числа";

cout << endl;

InputInt(&a);

cout << "Введите мнимую часть числа"; cout << endl;

InputInt(&b);

cout << endl; cout << "Число равно: ";

cout << a << "+i\*" << b; cout << endl;

}

void Complex::Sum() // Описание функции суммы

{

int t = 0, d = 0, x = 0, z = 0;

cout << "Введите действительную часть первого числа";

cout << endl;

InputInt(&t);

cout << "Введите мнимую часть первого числа"; cout << endl;

InputInt(&d);

cout << endl;

cout << "Введите действительную часть второго числа";

cout << endl;

InputInt(&x);

cout << "Введите мнимую часть второго числа"; cout << endl;

InputInt(&z);

a = (t + x);

b = (d + z);

cout << "Сумма равна";

cout << endl;

cout << a << "+i\*" << b; cout << endl;

}

class Son : public Complex // Описание дочерного класса

{

public:

Son() // Конструктор без параметров

{

cout << "Был создан представитель дочерного класса" << endl;

}

void Mult() // Функция умножения

{

int t = 0, d = 0, x = 0, z = 0;

cout << "Введите действительную часть первого числа";

cout << endl;

InputInt(&t);

cout << "Введите мнимую часть первого числа"; cout << endl;

InputInt(&d);

cout << endl;

cout << "Введите действительную часть второго числа";

cout << endl;

InputInt(&x);

cout << "Введите мнимую часть второго числа"; cout << endl;

InputInt(&z);

a = (t \* x - d \* z);

b = (t \* z + d \* x);

cout << "При умножении число равно"; cout << endl;

cout << a << "+i\*" << b; cout << endl;

}

};

void Complex::Div() // Описание функции разности

{

int t = 0, d = 0, x = 0, z = 0;

cout << "Введите действительную часть первого числа";

cout << endl;

InputInt(&t);

cout << "Введите мнимую часть первого числа"; cout << endl;

InputInt(&d);

cout << endl;

cout << "Введите действительную часть второго числа";

cout << endl;

InputInt(&x);

cout << "Введите мнимую часть второго числа"; cout << endl;

InputInt(&z);

a = (t - x);

b = (d - z);

cout << "При вычитании число равно"; cout << endl;

cout << a << "+i\*" << b; cout << endl;

}

int main(void)

{

setlocale(LC\_ALL, "Russian"); // Подключение русского языка в консоли

Son c; // Представитель сына

int v = 1; // Ключ для меню

while (v != 0) // Меню

{

cout << endl;

cout << "1-Ввод числа"; cout << endl;

cout << "2-Сложение чисел"; cout << endl;

cout << "3-Умножение чисел"; cout << endl;

cout << "4-Вычитание чисел"; cout << endl;

cout << "0-Выход"; cout << endl;

InputInt(&v);

if (v == 1)

{

c.Input();

}

if (v == 2)

{

c.Sum();

}

if (v == 3)

{

c.Mult();

}

if (v == 4)

{

c.Div();

}

}

return 0;

}

void InputInt(int\* tmp) // Описание функции ввода чисел

{

while (!(cin >> \*tmp) || (cin.peek() != '\n'))

{

cin.clear();

while (cin.get() != '\n');

cout << "Incorrect input!" << endl;

}

}