

Здесь будет титульник, листай ниже

СОДЕРЖАНИЕ

1 ПОСТАНОВКА ЗАДАЧИ.....	5
1.1 Описание входных данных.....	5
1.2 Описание выходных данных.....	5
2 МЕТОД РЕШЕНИЯ.....	6
3 ОПИСАНИЕ АЛГОРИТМОВ.....	7
3.1 Алгоритм функции main.....	7
3.2 Алгоритм метода set_n класса MyClass.....	7
3.3 Алгоритм метода count_digits класса MyClass.....	8
3.4 Алгоритм метода get_count класса MyClass.....	8
3.5 Алгоритм метода get_n класса MyClass.....	9
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	10
5 КОД ПРОГРАММЫ.....	13
5.1 Файл main.cpp.....	13
5.2 Файл MyClass.cpp.....	13
5.3 Файл MyClass.h.....	14
6 ТЕСТИРОВАНИЕ.....	15
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	16

1 ПОСТАНОВКА ЗАДАЧИ

Создать объект, который обрабатывает переменную целого типа максимальной длины.

У объекта есть закрытое свойство n целого типа максимальной длины.

Объект обладает следующей функциональностью:

- задает значение свойства n ;
- вычисляет количество цифр значения свойства n ;
- возвращает количество цифр значения свойства n .

Написать программу, которая:

1. Создает объект.
2. Вводит значение переменной целого типа.
3. Определяет значение свойства n по значению переменной целого типа.
4. Вычисляет количество цифр свойства n .
5. Выводит значение свойства n .
6. Выводит количество цифр значения свойства n .

1.1 Описание входных данных

Первая строка:

Целое число в десятичном формате.

1.2 Описание выходных данных

Первая строка:

n = «значение свойства n »

Вторая строка:

N = «количество цифр свойства n »

2 МЕТОД РЕШЕНИЯ

Для решения задачи используется:

- объект obj класса MyClass;
- функция main для основная функция;
- объект стандартного вывода cout.

Класс MyClass:

- свойства/поля:
 - поле число:
 - наименование — n;
 - тип — long long long int;
 - модификатор доступа — private;
 - поле количество цифр:
 - наименование — count;
 - тип — int;
 - модификатор доступа — private;
- функционал:
 - метод set_n — получает значение n;
 - метод count_digits — считает количество цифр n;
 - метод get_count — возвращает count;
 - метод get_n — возвращает n.

3 ОПИСАНИЕ АЛГОРИТМОВ

Согласно этапам разработки, после определения необходимого инструментария в разделе «Метод», составляются подробные описания алгоритмов для методов классов и функций.

3.1 Алгоритм функции `main`

Функционал: считает количество цифр.

Параметры: нет.

Возвращаемое значение: целое число.

Алгоритм функции представлен в таблице 1.

Таблица 1 – Алгоритм функции `main`

№	Предикат	Действия	№ перехода
1		Создание объекта <code>obj</code>	2
2		Инициализация <code>n</code>	3
3		Ввод <code>n</code>	4
4		Обращение ко всем методам	Ø

3.2 Алгоритм метода `set_n` класса `MyClass`

Функционал: получает значение `n`.

Параметры: нет.

Возвращаемое значение: целое число.

Алгоритм метода представлен в таблице 2.

Таблица 2 – Алгоритм метода *set_n* класса *MyClass*

№	Предикат	Действия	№ перехода
1		Присваивание значению n1 значение n	Ø

3.3 Алгоритм метода *count_digits* класса *MyClass*

Функционал: считает количество цифр.

Параметры: нет.

Возвращаемое значение: целое число.

Алгоритм метода представлен в таблице 3.

Таблица 3 – Алгоритм метода *count_digits* класса *MyClass*

№	Предикат	Действия	№ перехода
1		присваивание значению n2 значение n	2
2	n2 == 0	count равно 1	Ø
			3
3		считает количество цифр	Ø

3.4 Алгоритм метода *get_count* класса *MyClass*

Функционал: выводит значение count.

Параметры: нет.

Возвращаемое значение: целое число.

Алгоритм метода представлен в таблице 4.

Таблица 4 – Алгоритм метода *get_count* класса *MyClass*

№	Предикат	Действия	№ перехода
1		Возвращает значение count	Ø

3.5 Алгоритм метода `get_n` класса `MyClass`

Функционал: выводит значение `n`.

Параметры: нет.

Возвращаемое значение: целое число.

Алгоритм метода представлен в таблице 5.

Таблица 5 – Алгоритм метода `get_n` класса `MyClass`

№	Предикат	Действия	№ перехода
1		Возвращает значение <code>n</code>	Ø

4 БЛОК-СХЕМЫ АЛГОРИТМОВ

Представим описание алгоритмов в графическом виде на рисунках 1-3.

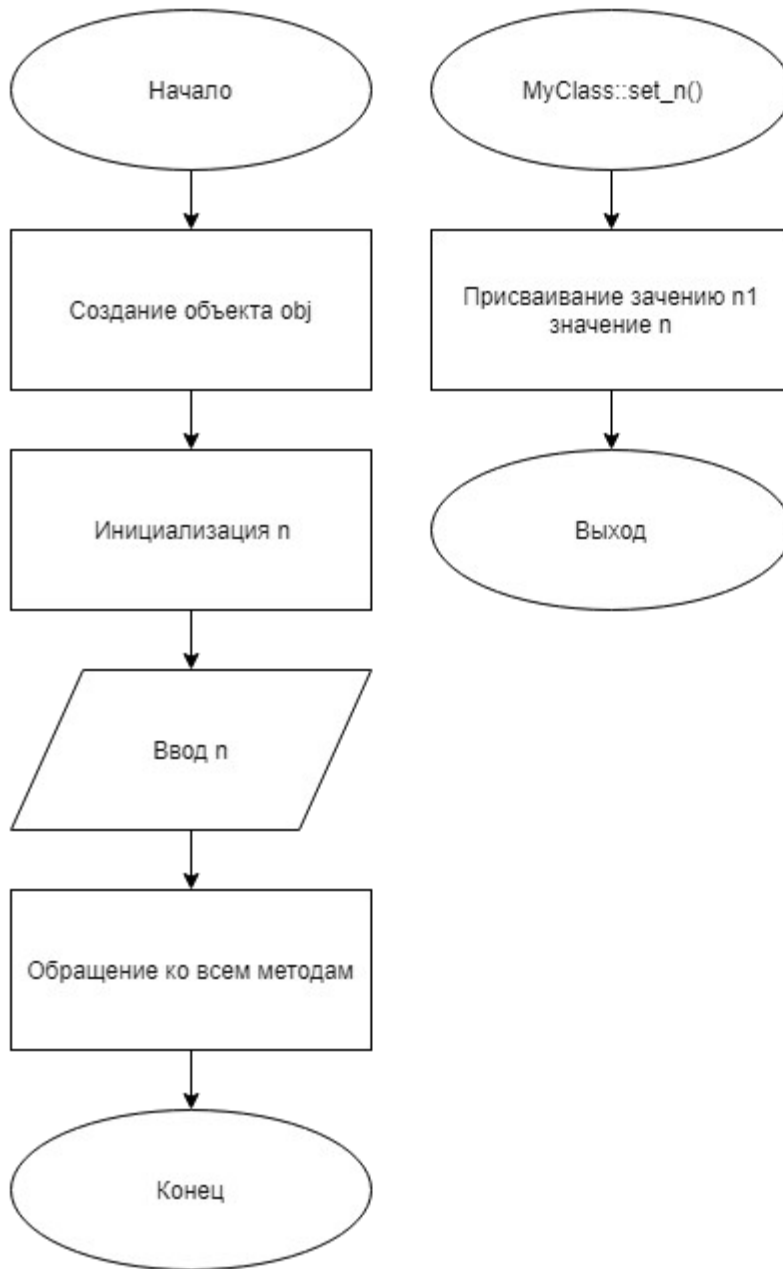


Рисунок 1 – Блок-схема алгоритма

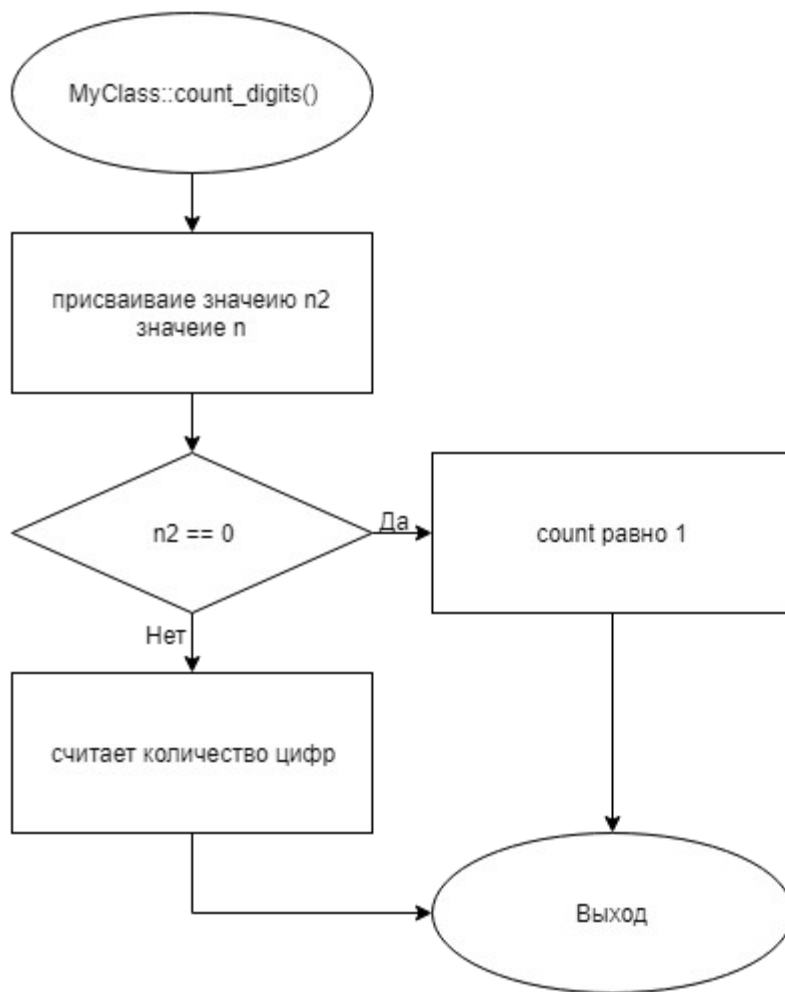


Рисунок 2 – Блок-схема алгоритма

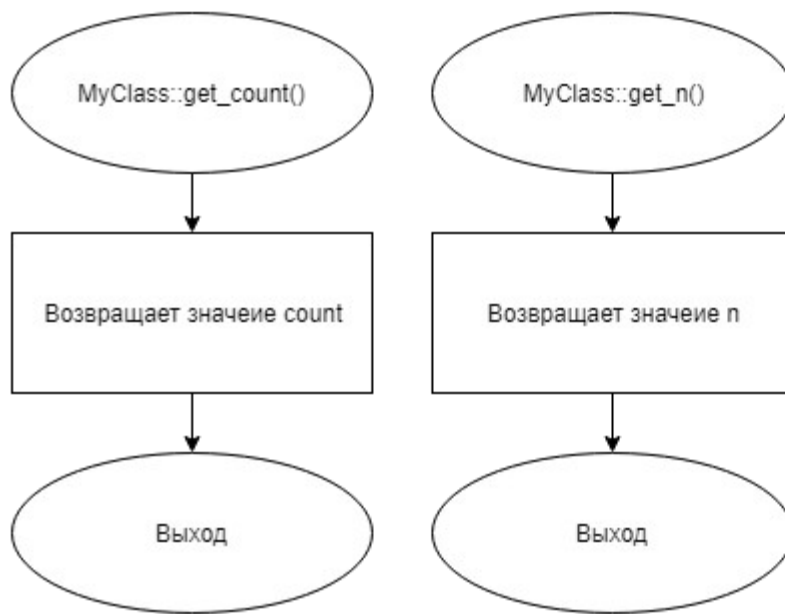


Рисунок 3 – Блок-схема алгоритма

5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

5.1 Файл main.cpp

Листинг 1 – main.cpp

```
#include <stdlib.h>
#include <stdio.h>
#include "MyClass.h"
#include <iostream>
using namespace std;
int main()
{
    MyClass obj;
    long long int n;
    cin >> n;
    obj.set_n(n);
    obj.count_digits();
    cout << "n = " << obj.get_n() << endl;
    cout << "N = " << obj.get_count();

}
```

5.2 Файл MyClass.cpp

Листинг 2 – MyClass.cpp

```
#include "MyClass.h"
#include <iostream>
using namespace std;

void MyClass::set_n(long long int n1)
{
    n = n1;
};

void MyClass::count_digits()
{
    long long int n2 = n;
    if (n2 == 0)
```

```

        {
            count = 1;
        }
        for(; n2 != 0; count++)
        {
            n2 /= 10;
        }
    }
    int MyClass::get_count()
    {
        return count;
    }
    long long int MyClass::get_n()
    {
        return n;
    }
}

```

5.3 Файл MyClass.h

Листинг 3 – MyClass.h

```

#ifndef __MYCLASS__H
#define __MYCLASS__H
class MyClass
{
    long long int n;
    int count;
public:
    void set_n(long long int n1);
    void count_digits();
    int get_count();
    long long int get_n();
};

#endif

```

6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 6.

Таблица 6 – Результат тестирования программы

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
12	n = 12 N = 2	n = 12 N = 2
0	n = 0 N = 1	n = 0 N = 1

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 19 Единая система программной документации.
2. Методическое пособие студента для выполнения практических заданий, контрольных и курсовых работ по дисциплине «Объектно-ориентированное программирование» [Электронный ресурс] – URL: https://mirea.aco-avvora.ru/student/files/methodichescoe_posobie_dlya_laboratornyh_rabot_3.pdf (дата обращения 05.05.2021).
3. Приложение к методическому пособию студента по выполнению заданий в рамках курса «Объектно-ориентированное программирование» [Электронный ресурс]. URL: https://mirea.aco-avvora.ru/student/files/Prilozheniye_k_methodichke.pdf (дата обращения 05.05.2021).
4. Шилдт Г. С++: базовый курс. 3-е изд. Пер. с англ.. — М.: Вильямс, 2019. — 624 с.
5. Видео лекции по курсу «Объектно-ориентированное программирование» [Электронный ресурс]. АСО «Аврора».
6. Антик М.И. Дискретная математика [Электронный ресурс]: Учебное пособие /Антик М.И., Казанцева Л.В. — М.: МИРЭА — Российский технологический университет, 2018 — 1 электрон. опт. диск (CD-ROM).