

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ЭЛЕКТРОТЕХНИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА «ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
АВТОМАТИЗИРОВАННЫХ СИСТЕМ»

ОТЧЁТ ПО ПРАКТИЧЕСКИМ ЗАДАЧАМ

Дисциплина: «Основы алгоритмизации и программирования»

Выполнил:
Студент группы ИВТ-21-26
Безух Владимир Сергеевич

Проверил:
Доцент кафедры ИТАС
Полякова Ольга Андреевна

Пермь,
Ноябрь 2021

Постановка задач

1. Подсчитать сумму цифр введённого числа.
2. Развернуть введённое число.
3. Вывести равномерную квадратную матрицу *.

Анализ задач

1. Для решения поставленной задачи необходимо: а) найти остаток от деления числа на 10 (это будет правостоящая цифра числа); б) результат, полученный в шаге «а», добавить в сумматор; в) нацело поделить введенное число на 10 и присвоить полученное значение (для перехода к следующему разряду); г) повторять до тех пор, пока введенное число не станет равным нулю.
2. Для нахождения цифры числа достаточно применять алгоритм, схожий с алгоритмом из задачи №1. При этом, результирующее число, которое мы ищем, будет сумматором с шагом на каждой итерации $t = (t * 10) + (N \% 10)$.
3. Указать размер курсора невозможно программными средствами языка C++. Этот параметр зависит исключительно от среды выполнения (в частности окна консольного интерфейса). При этом, в зависимости от среды, могут существовать манипуляторы для настройки визуальной составляющей консоли. Мною не было найдено доступного способа указать размер курсора в среде «Visual Studio 19».

Описание переменных

1.

unsigned int digit_summa = 0; — сумматор для результата.

unsigned int N; — введённое число.

2.

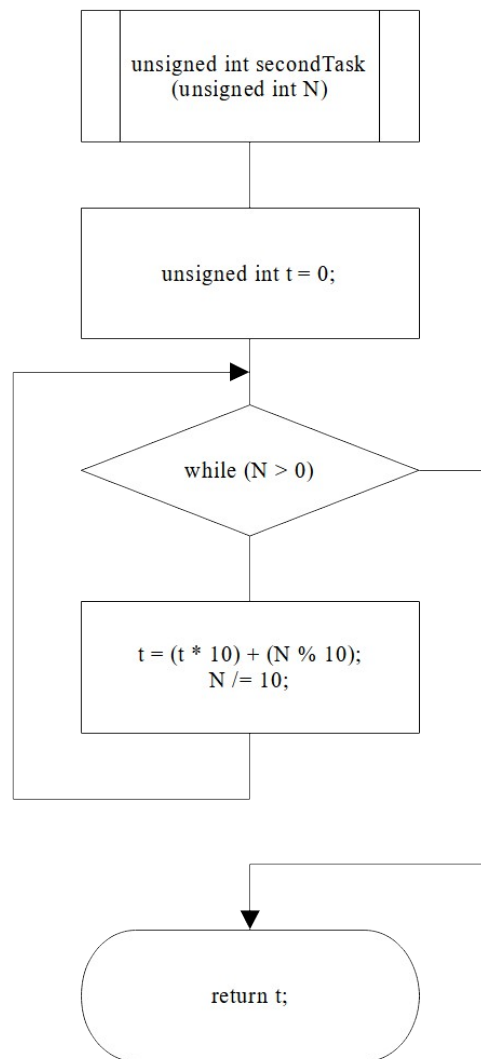
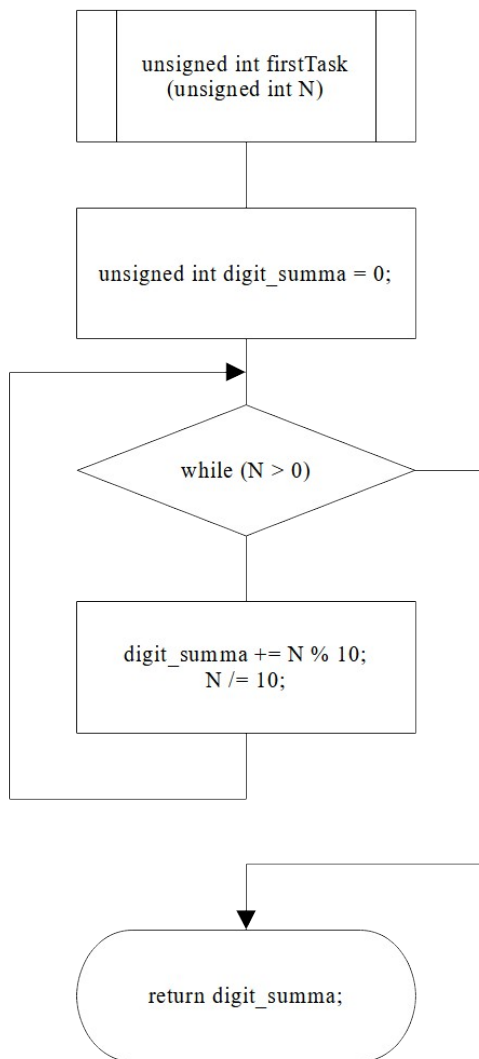
unsigned int t = 0; — сумматор для результата.

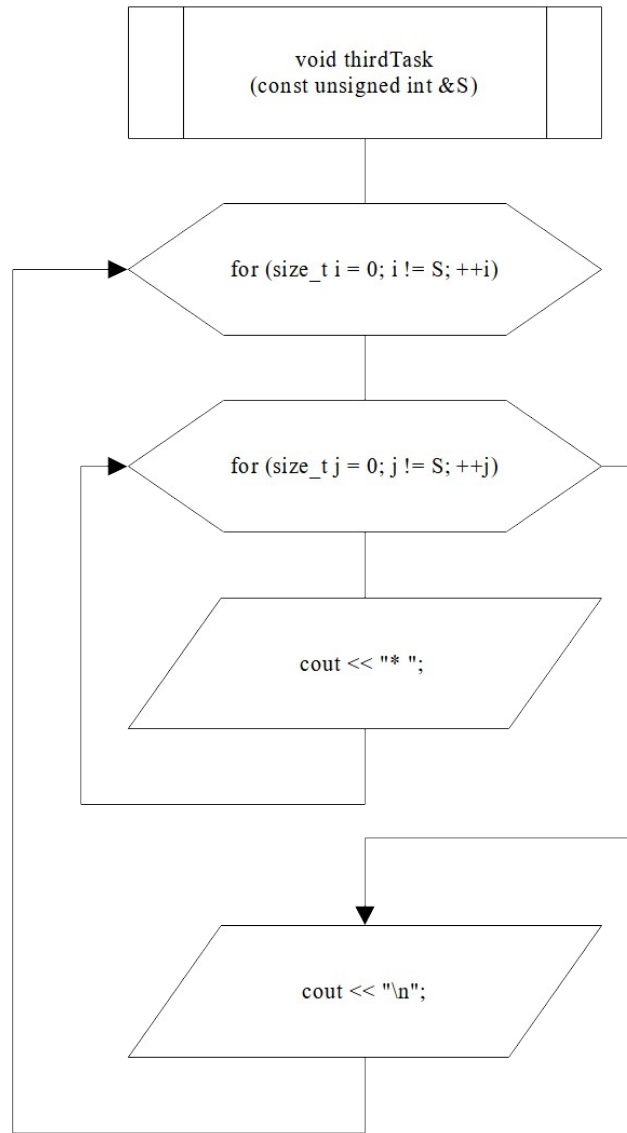
unsigned int N; — введённое число.

3.

unsigned int S; — введённый размер матрицы.

Блок-схема





Исходный код

```
#include <iostream>

using namespace std;

unsigned int firstTask(unsigned int N)
{
    unsigned int digit_summa = 0;

    while (N > 0)
    {
        digit_summa += N % 10;
        N /= 10;
    }

    return digit_summa;
}

unsigned int secondTask(unsigned int N)
{
    unsigned int t = 0;

    while (N > 0)
    {
        t = (t * 10) + (N % 10);
        N /= 10;
    }

    return t;
}

void thirdTask(const unsigned int &S)
{
    for (size_t i = 0; i != S; ++i)
    {
        for (size_t j = 0; j != S; ++j)
            cout << "*" << " ";

        cout << "\n";
    }
}

int main()
{
    setlocale(LC_ALL, "Russian");

    unsigned int N;

    cin >> N; const unsigned int DIGIT_SUMMA_N = firstTask(N);
    cout << DIGIT_SUMMA_N << "\n";

    cin >> N; const unsigned int REVERSE_N = secondTask(N);
    cout << REVERSE_N << "\n";

    cin >> N; thirdTask(N);
}
```


Скриншоты консольного интерфейса программы

[illegible]

Анализ результатов

1. Алгоритм корректен только для натуральных целых чисел.
2. Алгоритм корректен только для натуральных целых чисел.
3. С помощью добавленного после звёздочки пробела, можно симитировать равномерную матрицу.