**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В. И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра САПР**

ОТЧЕТ

**по курсовой работе № 1**

**по дисциплине Программирование**

Выполнил : Баринов Артур Андреевич Группа № 3351

Преподаватель: Рыжов Николай Геннадьевич

г. Санкт-Петербург 2023 г.

# Цель работы

Составить функцию для разложения заданного целого числа на простые множители.

# Задание

Составить функцию для разложения заданного целого числа на простые множители. Результатом функции должен быть массив, содержащий простые множители, и целое число - количество множителей.

# Задачи

Написать программу для разложения заданного целого числа на простые множители на языке С++.

# Требования

Результатом функции должен быть массив, содержащий простые множители, и целое число - количество множителей.

# int simplefactor(int n, int \*masfactor);

# 

# Схема алгоритма «Разложение заданного числа на простые множители»

# C:\Users\Comp\AppData\Local\Microsoft\Windows\INetCache\Content.Word\laba 5 osn.png

Рисунок 1

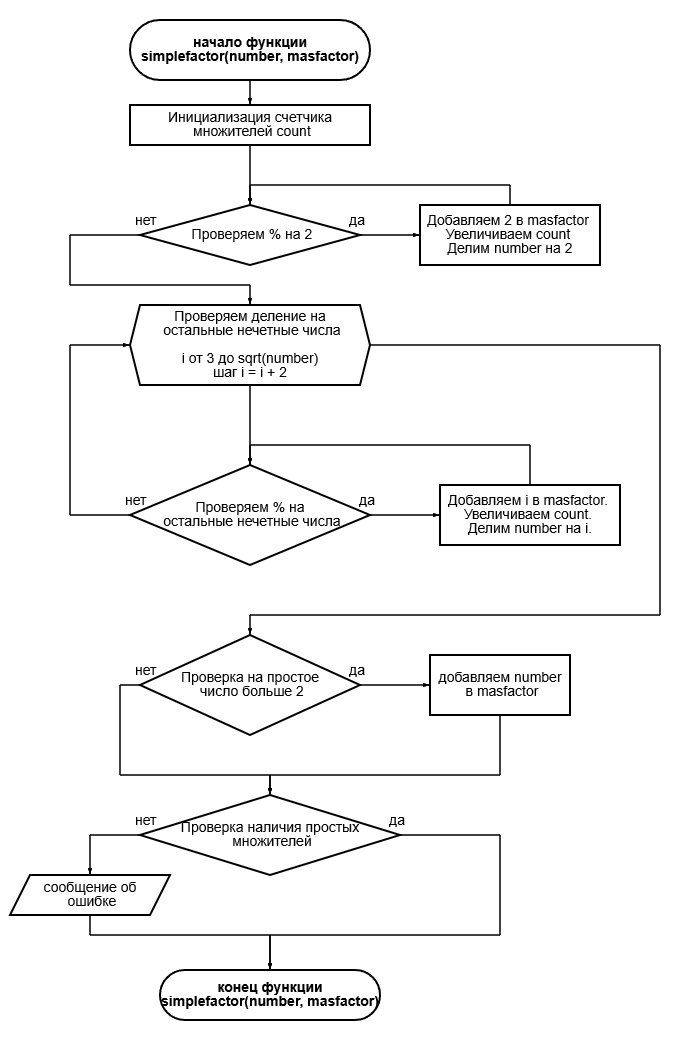


Рисунок 2 - функция

simplefactor (number, masfactor)

# Код программы «Разложение заданного числа на простые множители»

#include <iostream>

#include <cmath>

using namespace std;

// Функция для разложения числа на простые множители

int simplefactor(int n, int\* masfactor) {

int count = 0; // Инициализация счетчика множителей

// Проверяем деление на 2

while (n % 2 == 0) {

masfactor[count++] = 2; // Добавляем 2 в массив множителей и увеличиваем счетчик

n = n / 2;

}

// Проверяем деление на остальные нечетные числа

//sqrt(n) в цикле позволяет уменьшить количество итераций

for (int i = 3; i <= sqrt(n); i = i + 2) {

while (n % i == 0) {

masfactor[count++] = i; // Добавляем i в массив множителей и увеличиваем счетчик

n = n / i;

}

}

// Если n стало простым числом больше 2

if (n > 2) {

masfactor[count++] = n;

}

// Если не найдено ни одного простого множителя

if (count == 0) {

setlocale(LC\_ALL, "Russian");

cout << "Простых множителей заданного числа не обнаружено. " << "\n" << "Пожалуйста, повторите попытку ввода числа.";

exit(0);

}

return count; // Возвращаем количество простых множителей

}

int main() {

setlocale(LC\_ALL, "Russian"); // Установка локали для отображения сообщений на русском языке

int number;

cout << "Введите целое число для разложения на простые множители: ";

cin >> number; // Ввод числа пользователем

const int Maxsize = 1000;

// Массив для хранения простых множителей

int masfactor[Maxsize];

// Вызываем функцию разложения на множители

int count = simplefactor(number, masfactor);

// Выводим результат

cout << "Простые множители числа " << number << ": ";

for (int i = 0; i < count; ++i) {

cout << masfactor[i] << " ";

}

cout << "\nКоличество простых множителей: " << count << endl;

return 0;

}

**Описание программы «Разложение заданного числа на простые множители»**

Идентификатор: Lab5.exe

Необходимое ПО: Visual Studio 2022

Язык: С++

Программа позволяет пользователю ввести целое число и затем разлагает это число на простые множители. Результат разложения выводится на экран, включая простые множители и их количество.

**Описание схемы алгоритма:**

1.Ввод данных:

Программа запрашивает у пользователя ввод целого числа.

2.Инициализация переменных:

count: переменная для отслеживания количества простых множителей.

masfactor: массив для хранения простых множителей.

number: переменная для хранения введенного числа.

3. Вызываем функцию разложения на множители.

4.Проверка деления на 2:

Используя цикл while, проверяется деление введенного числа number на 2.

Если делится, добавляется 2 в массив masfactor, увеличивается count и обновляется значение number.

5.Проверка деления на нечетные .

Используя цикл for, проверяется деление number на нечетные числа от 3 до sqrt(number).

Если делится, добавляется делитель в массив masfactor, увеличивается count и обновляется значение number.

6.Проверка на простое число больше 2:

Если после цикла for осталось простое число больше 2, оно добавляется в массив masfactor.

7. Проверка наличия простых множителей:

Если count остался равным 0, выводится сообщение об ошибке и программа завершает выполнение.

8.Вывод результата:

Выводятся простые множители числа и их количество на экран.

9.Завершение программы:

Возвращается 0 в качестве кода успешного завершения программы.

Программа использует циклы и условные операторы для проверки деления числа на простые множители.

.

# Руководство пользователя программы

При запуске программа запрашивает пользователя ввести целое. После ввода, на экране отображаются простые множители заданного числа и их количество. Пример работы программы представлен на рисунке 3.

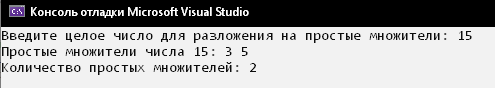


Рисунок 3

Если не найдено ни одного простого множителя, программа выведет данное сообщение: "Простых множителей заданного числа не обнаружено. Пожалуйста, повторите попытку ввода числа." Пример работы представлен на рисунке 4.

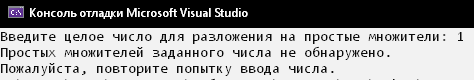


Рисунок 4

# Пути дальнейшего улучшения

Путём дальнейшего улучшения является введение дополнительного функционала, такого как: работа с множеством чисел.