Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики

и радиоэлектроники»

Специальность «Программная инженерия»

Кафедра инженерной психологии и эргономики

Учебная дисциплина «Основы алгоритмизации и программирования»

ОТЧЕТ

по лабораторной работе №8

«Функции»

Вариант 13

Подготовил: Джалилова Н.Р.

Проверил: Усенко Ф.В.

Минск 2024

**Цель работы:** сформировать навыки и умения обработки структурированных типов данных, организованных в виде функций.

Задание1: Дано натуральное n-значное число P. Проверьте, является ли данное число палиндромом (перевертышем).

Задание 2:Используя перегрузку методов, создайте программу согласно варианту. а) для преобразования десятичной дроби в обыкновенную; б) для преобразования обыкновенной дроби в десятичную.

Задание 1

#include <iostream>

using namespace std;

// Функция для проверки, является ли число палиндромом

bool isPalindrome(int number) {

int original = number; // Сохраняем исходное число

int reversed = 0;

// Переворачиваем число

while (number > 0) {

int digit = number % 10; // Берем последнюю цифру

reversed = reversed \* 10 + digit; // Добавляем цифру в перевернутое число

number /= 10; // Убираем последнюю цифру

}

// Сравниваем исходное и перевернутое числа

return original == reversed;

}

// Основная функция

int main() {

setlocale(LC\_ALL, "Russian");

int number;

/\*while (number <= 0) {

cout << "Введите натуральное число больше 0." << endl;

cin >> number;

}\*/

// Ввод числа и проверка ввода

while (true) {

printf("Введите натуральное число: ");

// Попытка считывания числа

if (scanf\_s("%d", &number) == 1) {

// Если ввод правильный, проверяем, не является ли это целым числом

if (number <= 0) {

// Если это целое число, выводим ошибку

printf("Ошибка ввода! Пожалуйста, введите натуральное число. \n");

}

else {

// Ввод успешен, выходим из цикла

break;

}

}

else {

// Если ввод некорректный

printf("Ошибка ввода! Пожалуйста, введите натуральное число. \n");

while (getchar() != '\n'); // Очистка буфера ввода

}

}

// Проверка на палиндром

if (isPalindrome(number)) {

cout << "Число " << number << " является палиндромом." << endl;

}

else {

cout << "Число " << number << " не является палиндромом." << endl;

}

return 0;

}

Задание 2.

#include <iostream>

#include <cmath> // Для использования функции gcd

using namespace std;

// Функция для вычисления наибольшего общего делителя (НОД)

int gcd(int a, int b) {

while (b != 0) {

int temp = b;

b = a % b;

a = temp;

}

return a;

}

// Перегруженная функция 1: Преобразование десятичной дроби в обыкновенную

void convert(double decimal) {

// Константа для точности

const int precision = 1000000;

// Представляем десятичную дробь как n / precision

int numerator = static\_cast<int>(decimal \* precision + 0.5); // Числитель

int denominator = precision; // Знаменатель

// Сокращаем дробь, используя НОД

int divisor = gcd(numerator, denominator);

numerator /= divisor;

denominator /= divisor;

// Выводим результат

cout << "Обыкновенная дробь: " << numerator << "/" << denominator << endl;

}

// Перегруженная функция 2: Преобразование обыкновенной дроби в десятичную

void convert(int numerator, int denominator) {

if (denominator == 0) {

cout << "Ошибка: деление на ноль!" << endl;

return;

}

double decimal = static\_cast<double>(numerator) / denominator;

cout << "Десятичная дробь: " << decimal << endl;

}

// Основная функция

int main() {

setlocale(LC\_ALL, "Russian");

setlocale(LC\_NUMERIC, "C"); // Устанавливаем локаль, которая использует точку для десятичных чисел

// Вариант а) Преобразование десятичной дроби в обыкновенную

double decimalNumber;

while (true) {

printf("Введите число с плавающей точкой: ");

// Попытка считывания числа

if (scanf\_s("%lf", &decimalNumber) == 1) {

// Если ввод правильный, проверяем, не является ли это целым числом

if (decimalNumber == static\_cast<int>(decimalNumber)) {

// Если это целое число, выводим ошибку

printf("Ошибка ввода! Пожалуйста, введите число с плавающей точкой, например, 3.14.\n");

}

else {

// Ввод успешен, выходим из цикла

break;

}

}

else {

// Если ввод некорректный

printf("Ошибка ввода! Пожалуйста, введите число с плавающей точкой.\n");

while (getchar() != '\n'); // Очистка буфера ввода

}

}

convert(decimalNumber);

// Вариант б) Преобразование обыкновенной дроби в десятичную

int numerator, denominator;

while (true) {

printf("Введите целое число (числитель дроби): ");

// Попытка считать целое число

if (scanf\_s("%d", &numerator) == 1) {

// Если введено целое число, проверим, не осталось ли в буфере символов

char ch;

if (scanf\_s("%c", &ch, 1) == 1 && ch != '\n') {

// Если есть дополнительные символы после числа, это не целое число

printf("Ошибка ввода! Пожалуйста, введите целое число.\n");

while (getchar() != '\n'); // Очистка буфера ввода

}

else {

// Ввод корректен, выходим из цикла

break;

}

}

else {

// Если ввод некорректный

printf("Ошибка ввода! Пожалуйста, введите целое число.\n");

while (getchar() != '\n'); // Очистка буфера ввода

}

}

while (true) {

printf("Введите целое число (знаменатель дроби): ");

// Попытка считать целое число

if (scanf\_s("%d", &denominator) == 1) {

// Если введено целое число, проверим, не осталось ли в буфере символов

char ch;

if (scanf\_s("%c", &ch, 1) == 1 && ch != '\n') {

// Если есть дополнительные символы после числа, это не целое число

printf("Ошибка ввода! Пожалуйста, введите целое число.\n");

while (getchar() != '\n'); // Очистка буфера ввода

}

else {

// Ввод корректен, выходим из цикла

break;

}

}

else {

// Если ввод некорректный

printf("Ошибка ввода! Пожалуйста, введите целое число.\n");

while (getchar() != '\n'); // Очистка буфера ввода

}

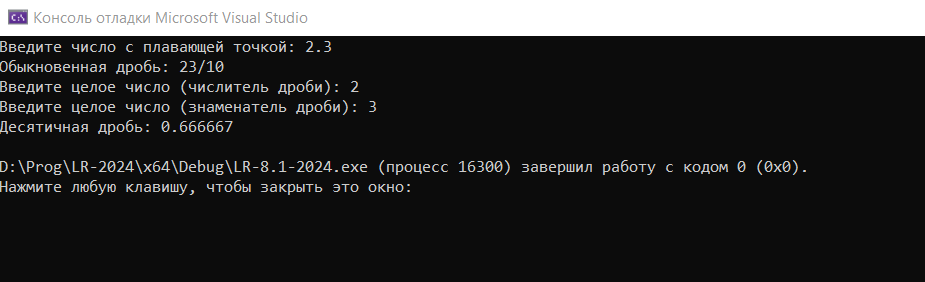
}

convert(numerator, denominator);

return 0;

}

Результат работы программ представлен на рисунках 1 и 2.

Рисунок 1

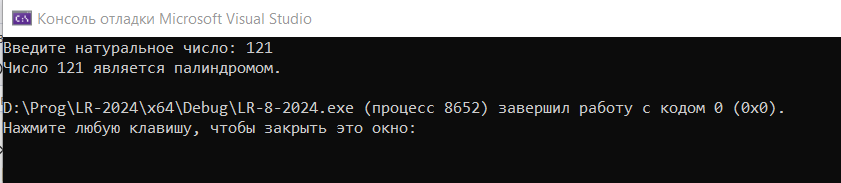


Рисунок 2

Блок-схемы работы программ представлены на рисунках для задания 1

(1, 2) для задания 2 (3, 4, 5, 6).

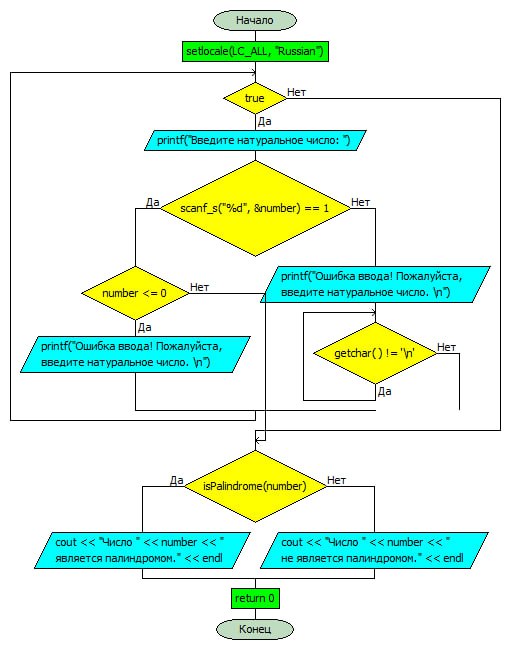


Рисунок 1 — main()

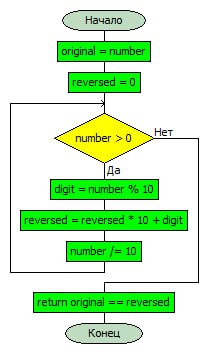


Рисунок 2 — функция isPalindrome(int number)

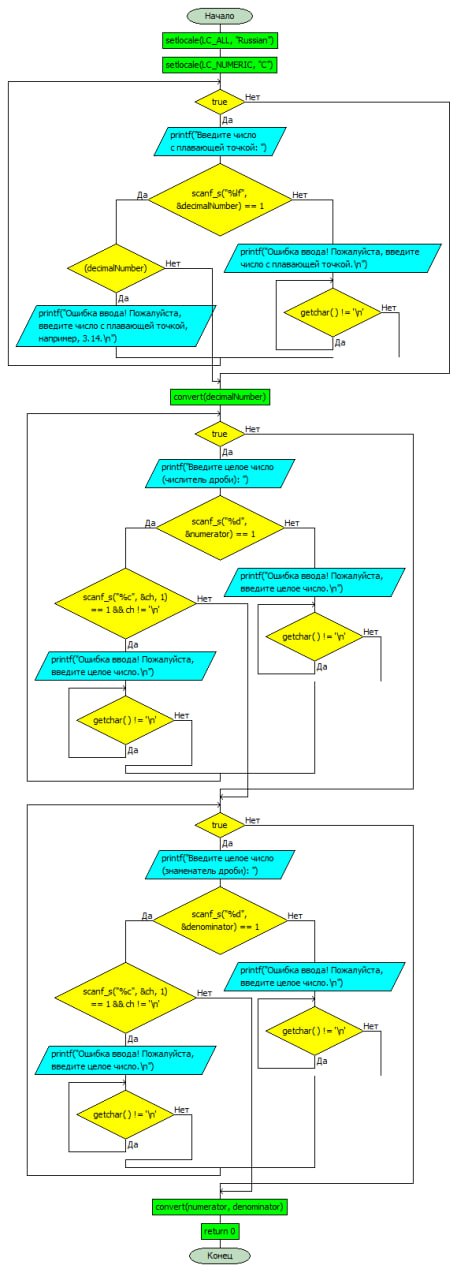


Рисунок 3 - функция main()

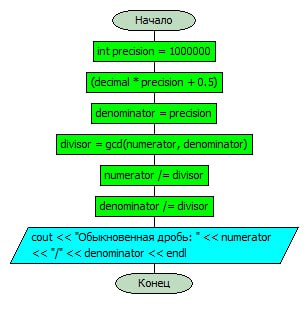


Рисунок 4 - void convert(double decimal)

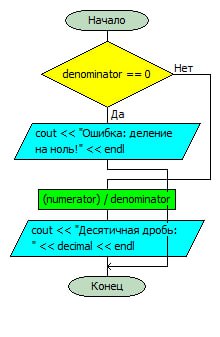


Рисунок 5 - void convert(int numerator, int denominator)

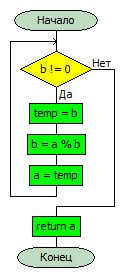


Рисунок 6 - int gcd(int a, int b)

**Вывод:** в ходе выполнения работы была достигнута цель данной лабораторной работы: сформировать навыки и умения обработки структурированных типов данных, организованных в виде функций.