|  |  |
| --- | --- |
| https://lh7-us.googleusercontent.com/QuBaagabzZYLr7U3ZbYOtMusd5FbWWxIvMUTN8jrLHzSLMg534z9gXRTIG1Us4i_lOwmWlaBxKedNt-SQ26dm4WmyqwjGDmEO6z8GE3QrZosqvHM88J2EFeVf1u0GzyCZQlhWmp1Zeo85tKo4LJVXQ | МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ (НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)» |

Институт № 3 «Системы управления, информатика и электроэнергетика»

Кафедра 304 «Вычислительные машины, системы и сети»

Отчёт по лабораторной работе №2  
по учебной дисциплине «Введение в профессиональную деятельность»  
на тему «Работа с типами данных в Си»

Выполнил  
студент группы М3О-119БВ-24  
Нарзиев А.Т.

Принял

ст. преподаватель каф. 304, Ведьманов И.С.

Москва  
2024

**Содержание**

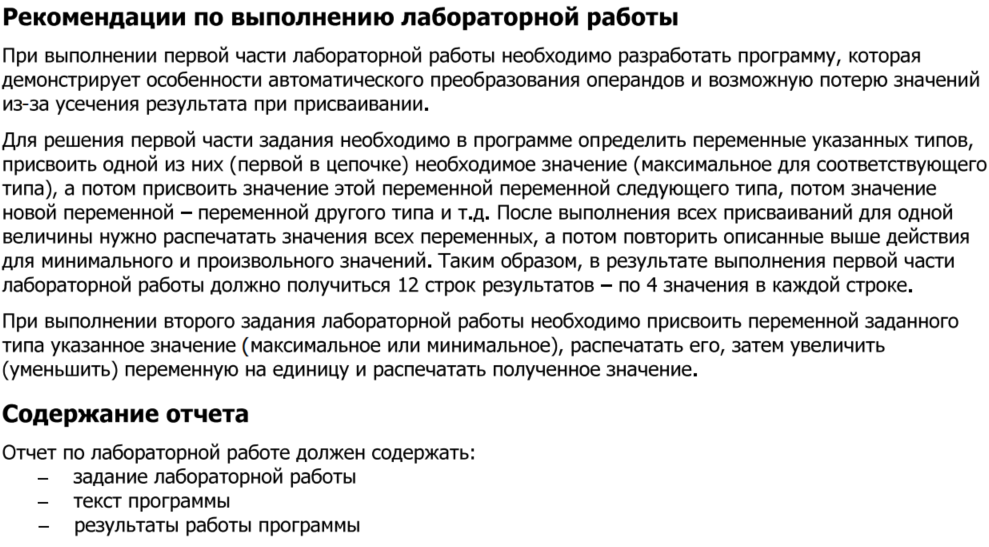
Задание 3

Исходный код 4

Результаты работы программы 7

Вывод по работе 8

**Задание**



**Исходный код**

#include <iostream>

#include <limits> // для лимитов типов

using namespace std;

int main() {

    // Примеры значений для знаковых типов

    signed char sch = -100;

    signed short int ssh = -30000;

    signed int si = -2000000000;

    signed long sl = -10000000000;

    // Примеры преобразования знаковых типов

    signed short int sch\_to\_ssi = sch; // signed char -> signed short int

    signed int sch\_to\_si = sch; // signed char -> signed int

    signed long sch\_to\_sl = sch; // signed char -> signed long

    signed int ssh\_to\_si = ssh; // signed short int -> signed int

    signed long ssh\_to\_sl = ssh; // signed short int -> signed long

    signed long si\_to\_sl = si; // signed int -> signed long

    signed short int si\_to\_ssh = si; // signed int -> signed short int (может вызвать переполнение)

    signed char si\_to\_sch = si; // signed int -> signed char (может вызвать переполнение)

    signed long sl\_to\_si = sl; // signed long -> signed int (может вызвать переполнение)

    signed short int sl\_to\_ssh = sl; // signed long -> signed short int (может вызвать переполнение)

    signed char sl\_to\_sch = sl; // signed long -> signed char (может вызвать переполнение)

    // Для беззнаковых типов

    unsigned char uch = 200;

    unsigned short int ush = 60000;

    unsigned int ui = 3000000000;

    unsigned long ul = 10000000000;

    // Примеры преобразования беззнаковых типов

    unsigned short int uch\_to\_ush = uch; // unsigned char -> unsigned short int

    unsigned int uch\_to\_ui = uch; // unsigned char -> unsigned int

    unsigned long uch\_to\_ul = uch; // unsigned char -> unsigned long

    unsigned short int ush\_to\_ui = ush; // unsigned short int -> unsigned int

    unsigned long ush\_to\_ul = ush; // unsigned short int -> unsigned long

    unsigned long ui\_to\_ul = ui; // unsigned int -> unsigned long

    unsigned int ul\_to\_ui = ul; // unsigned long -> unsigned int (может вызвать переполнение)

    unsigned short int ul\_to\_ush = ul; // unsigned long -> unsigned short int (может вызвать переполнение)

    unsigned char ul\_to\_uch = ul; // unsigned long -> unsigned char (может вызвать переполнение)

    // Вывод результатов

    cout << "SIGNED TYPES:" << endl;

    cout << "signed char to signed short int: " << (int)sch\_to\_ssi << endl;

    cout << "signed char to signed int: " << sch\_to\_si << endl;

    cout << "signed char to signed long: " << sch\_to\_sl << endl;

    cout << "signed short int to signed int: " << ssh\_to\_si << endl;

    cout << "signed short int to signed long: " << ssh\_to\_sl << endl;

    cout << "signed int to signed long: " << si\_to\_sl << endl;

    cout << "signed int to signed short int: " << si\_to\_ssh << " (potential overflow)" << endl;

    cout << "signed int to signed char: " << (int)si\_to\_sch << " (potential overflow)" << endl;

    cout << "signed long to signed int: " << sl\_to\_si << " (potential overflow)" << endl;

    cout << "signed long to signed short int: " << sl\_to\_ssh << " (potential overflow)" << endl;

    cout << "signed long to signed char: " << (int)sl\_to\_sch << " (potential overflow)" << endl;

    cout << "\nUNSIGNED TYPES:" << endl;

    cout << "unsigned char to unsigned short int: " << (int)uch\_to\_ush << endl;

    cout << "unsigned char to unsigned int: " << uch\_to\_ui << endl;

    cout << "unsigned char to unsigned long: " << uch\_to\_ul << endl;

    cout << "unsigned short int to unsigned int: " << ush\_to\_ui << endl;

    cout << "unsigned short int to unsigned long: " << ush\_to\_ul << endl;

    cout << "unsigned int to unsigned long: " << ui\_to\_ul << endl;

    cout << "unsigned long to unsigned int: " << ul\_to\_ui << " (potential overflow)" << endl;

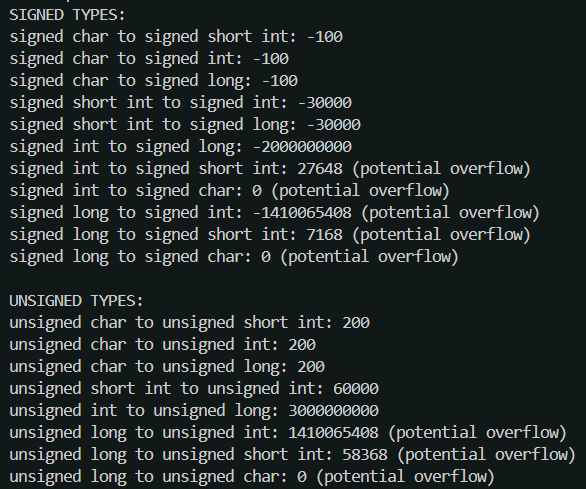
    cout << "unsigned long to unsigned short int: " << ul\_to\_ush << " (potential overflow)" << endl;

    cout << "unsigned long to unsigned char: " << (int)ul\_to\_uch << " (potential overflow)" << endl;

    return 0;

}

**Результаты работы программы**



**Вывод по работе**

Разработка программы завершена на том основании, что:

1. Результаты получены
2. Считаю тему освоенной