Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение высшего образования

**«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)**

Факультет систем управления (ФСУ)

Кафедра автоматизированных систем управления (АСУ)

**Основы языка Ассемблер. Команды пересылки данных.**

Отчет по лабораторной работе №2 по дисциплине

«Вычислительная техника»

Студент гр. 431-3

\_\_\_\_\_\_\_\_\_ Гурулёв А.В

«10» октября 2022 г.

Руководитель

\_\_\_\_\_\_\_\_\_\_\_ Алфёров С.М.

«\_\_\_» \_\_\_\_\_\_\_\_\_ 2022 г.

Томск 2022

**Оглавление**

[**ВВЕДЕНИЕ** 3](#_Toc116308340)

[**1 РЕЗУЛЬТАТ РАБОТЫ ПРОГРАММЫ** 4](#_Toc116308341)

[**ВЫВОД** 4](#_Toc116308342)

[**Приложение A** 5](#_Toc116308343)

**ВВЕДЕНИЕ**

Цель работы – Освоить команды пересылки данных mov, xchg и средства формирования операндов с помощью byte ptr, word ptr и смещения [const], +const.const], +const.const.

Задание:

* 7) Даны три переменные A B и C (int) (int) (long int). Переставить байты в переменных по следующей схеме:

Начальная нумерация байт A = 11 22 | B = 33 44 | C = 55 66 77 88

После перестановки A = 33 77 | B = 44 88 | C = 11 22 66 55

## **1** **РЕЗУЛЬТАТ РАБОТЫ ПРОГРАММЫ**

На рисунке 1.1 представлен результат работы программы.

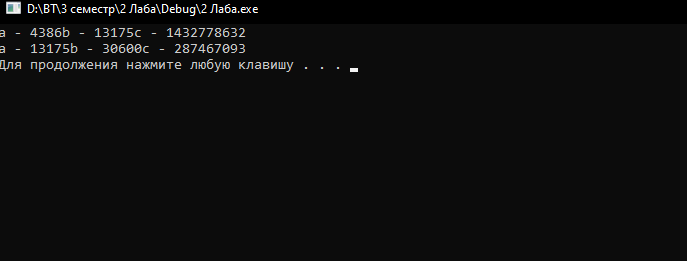


Рисунок 1.1 - Результат работы программы

## **ВЫВОД**

Я освоил команды пересылки данных mov, xchg и средства формирования операндов с помощью byte ptr, word ptr и смещения [const], +const.const], +const.const.

## **Приложение A**

**Листинг кода**

#include <iostream>

using namespace std;

//7 Вариант

int main()

{

//Числа только положительные 2 байт

short int a = 0x1122;

short int b = 0x3377;

int c = 0x55667788;

cout << a << ' ' << b << ' ' << c << endl;

\_asm

{

mov AH, byte ptr b[1];

xchg AH, byte ptr a[1];

xchg AH, byte ptr c[3];

xchg AH, byte ptr c[0];

xchg AH, byte ptr b[0];

mov byte ptr b[1], AH;

mov AH, byte ptr c[1];

xchg AH, byte ptr a[0];

xchg AH, byte ptr c[2];

mov byte ptr c[1], AH;

}

cout << a << ' ' << b << ' ' << c << endl;

system("pause");

return 0;

}