Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет среднего профессионального образования

**ОТЧЕТ**

**о Лабораторной работе № 1**

по теме: Программирование арифметических операций

по дисциплине: Системное программирование

Специальность:

09.02.07 Информационные системы и программирование

|  |  |
| --- | --- |
| Проверил:  Тропченко А.А. \_\_\_\_\_\_\_\_\_\_  Дата: «\_\_» \_\_\_\_\_\_\_ 20\_\_г.  Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Выполнил:  Студент группы Y2431  Головачев Д.А. |

Санкт-Петербург 2020/2021

Лабораторная работа 1-1 «Выполнение арифметических операций»

Цель занятия:

— углубить и закрепить знания по архитектуре МП платформы х64 и навыки по их программированию;

— приобрести практические навыки составления, налаживания и выполнения программ, написанных языком ассемблера для программирования арифметических операций под МП платформы 64 в среде masm64.

Вариант 6: 2d/с – сd

Текст программы:

Листинг программы 64-разрядной системы.

include \masm64\include64\masm64rt.inc

.data ; секция данных

const dq 2 ; создаие константы

d1 dq 7 ; операнд а1 размерностью 64 разряда

c1 dq 6

titl1 db "ЛР1. Выполнение арифметич. операций на masm64",0

strbuf dq ?,0

text db "Вывод результата 2d/с – сd через MessageBox:",0ah,"Результат: %d - целая часть",0ah,0ah,

"Автор: Головачев Д.А.",10,

9,"Сайт: http://blogs.kpi.kharkov.ua/v2/asm/",0

.code ; секция кода

entry\_point proc

mov rax,d1 ; внос числа в аккумулятор

mul const ; умножение на 2

div c1 ; деление на С

mov strbuf,rax ; сохранение промежуточного значения

mov rax,d1 ; внос числа в аккумулятор

mul c1 ; умножение на С

sub strbuf, rax; итоговая разность, результат в strbuf

invoke wsprintf, ADDR strbuf, ADDR text, strbuf ; функция преобразования

invoke MessageBox, 0, addr strbuf, addr titl1, MB\_ICONINFORMATION

invoke ExitProcess,0 ; ; завершение процесса и освобождение ресурсов

entry\_point endp ; завершение процедуры с именем

end ; конец

Листинг программы 32-разрядной системы

.686

.model flat, stdcall

option casemap:none

include \masm32\include\windows.inc

include \masm32\include\kernel32.inc

include \masm32\include\user32.inc

includelib \masm32\lib\user32.lib

includelib \masm32\lib\kernel32.lib

count PROTO arg\_c:DWORD, arg\_a:DWORD, arg\_e:DWORD

.data

\_a1 dd 7

\_b1 dd 2 ; константа

\_c1 dd 7

\_res1 dd ?

\_title db "Лабораторная работа No1. Арифметические операции",0

strbuf dw ?,0

\_text db "masm32. Вывод результата 2d/с – сd через MessageBox:",0ah,"Результат: %d - целая часть",0ah, 0ah,

"Автор: Головачев Д.А.",0

.code ; секция кода

count proc arg\_a:DWORD, arg\_b:DWORD, arg\_c:DWORD

mov eax,arg\_a ; внос числа в аккумулятор

mul arg\_b ; умножение на 2

div arg\_c ; деление на С

mov \_res1,eax ; сохранение помежуточого значения

mov eax,arg\_a ; внос числа в аккумулятор

mul arg\_c ; умножение на С

sub \_res1,eax ; итоговая разность, результат в \_res1

ret

count endp

start:

invoke count, \_a1, \_b1, \_c1

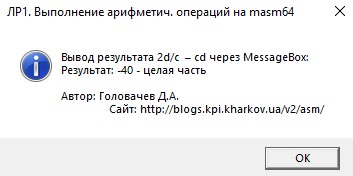
invoke wsprintf, ADDR strbuf, ADDR \_text, \_res1

invoke MessageBox, NULL, addr strbuf, addr \_title, MB\_ICONINFORMATION

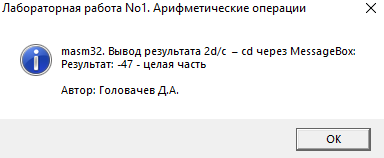
invoke ExitProcess, 0

end start ; конец программы

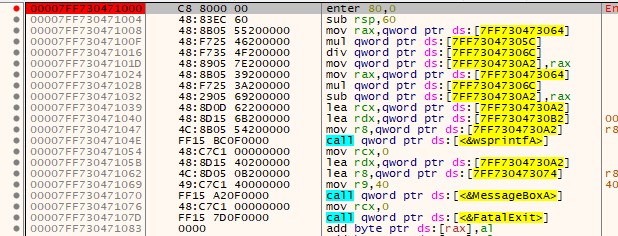
Результат выполнения программы и отладчиков:



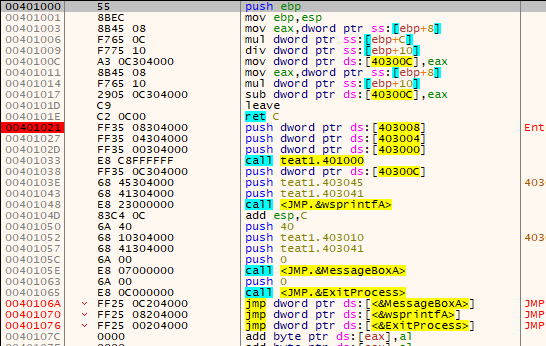
*Рисунок 1. Результат выполнения программы 64-разрядной системы*



*Рисунок 2. Результат выполнения программы 32-разрядной системы*



*Рисунок 3. Алгоритм выполнения 64-разрядной программы*



*Рисунок 3. Алгоритм выполнения 32-разрядной программы*

Особенности программы.

Вводятся 2 числа и 1 числовая константа, равная «2». После выполнения деления, промежуточное значение сохраняется где-то в памяти, потом выполняется умножение (с сохранением результата в аккумулятор). Последнее действие выполняется с сохранением результата в ячейку памяти, а не в аккумулятор, в ответе выводится тоже ячейка памяти.