МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Инженерная школа информационных технологий и робототехники Направление: Информатика и вычислительная техника

Отделение информационных технологий

Отчет по лабораторной работе №1 по дисциплине

«Организация ЭВМ»

«РАБОТА С ЗАГРУЗЧИКОМ x86»

Вариант 28

Выполнил: Студент группы 8В01

И.П. Тюрин

Проверил: Инженер ОИТ

##### Ю.В. Котова

Томск 2022

**Цель работы**

Изучить регистры и систему команд x86.

**Задание**

1. Разработать программу на языке ассемблер для x86 в виде загрузчика,

которая выведет фамилию и номер группы на экран на разных строчках.

2. Выполнить задание согласно своему варианту (2):

Ввести операцию +,-,\*,/. Вывести результат операции между числами 23

и 15. Формат вывода такой «23 op 15=Z», где op – операция, а Z

результат.

3. Найти соответствие команды ассемблера и её кода в своем \*.bin файле.

4. Написать отчёт о проделанной работе.

**Ход работы**

**Код программы**

[bits 16] ;ассемблер будет работать в 16битном режиме

[org 0x7c00] ;адрес старта программы

pre\_set:

mov ax, 0x0000 ;адрес сегментов

mov ds, ax ;запись указателя сегмента данных

mov ss, ax ; запись указателя сегмента стека

mov sp, pre\_set+1024+512 ;установка указателя стека

lab\_start:

mov AH, 0x0e

mov SI, MyName

lodsb

int 0x10

call show\_string ; Вывести Имя

call next\_line

mov AH, 0x0e

mov SI, MyGroup ; Вывести Группу

call show\_string

call input\_loop\_start

jmp $

input\_loop\_start: ; Вход в цикл

call next\_line

mov AH, 0x0e

mov SI, PreInputText ; Вывод информации о выводе

call show\_string

input\_loop: ; Основной цикл ввода

mov ah, 0x0

int 0x16

cmp al, 0x2b ; при нажатии на + переход на метку plus\_key

je plus\_key

cmp al, 0x2d ; при нажатии на - переход на метку minus\_key

je minus\_key

cmp al, 0x2a ; при нажатии на \* переход на метку mult\_key

je mult\_key

cmp al, 0x2f ; при нажатии на / переход на метку divis\_key

je divis\_key

cmp ah, 0x01

je end\_loop

jmp input\_loop

plus\_key: ; выполнение при нажатии на +

mov AH, 0x0e

mov al, 0x2b

int 0x10

call next\_line

mov AH, 0x0e

mov SI, PlusOp

call show\_string

jmp input\_loop\_start

minus\_key: ; выполнение при нажатии на -

mov AH, 0x0e

mov al, 0x2d

int 0x10

call next\_line

mov AH, 0x0e

mov SI, MinusOp

call show\_string

jmp input\_loop\_start

mult\_key: ; выполнение при нажатии на \*

mov AH, 0x0e

mov al, 0x2a

int 0x10

call next\_line

mov AH, 0x0e

mov SI, MultiOp

call show\_string

jmp input\_loop\_start

divis\_key: ; выполнение при нажатии на /

mov AH, 0x0e

mov al, 0x2f

int 0x10

call next\_line

mov AH, 0x0e

mov SI, DiviOp

call show\_string

jmp input\_loop\_start

end\_loop: ; Написать END перед выходом из цикла

mov AH, 0x0e

mov al, 0x45 ; E

int 0x10

mov al, 0x4e ; N

int 0x10

mov al, 0x44 ; D

int 0x10

call next\_line

call next\_line

ret

show\_string: ; Написать строку из ячейки памяти

cmp al, 0

je done

jne continue

continue:

lodsb

int 0x10

jmp show\_string

done:

ret

next\_line: ; Переход га новую строку

mov AH, 3

mov al, DH

add al, 1

mov AH, 0x02

mov DH, al

int 0x10

ret

MyName:

db "Tyurin Ilya", 0

MyGroup:

db "8V01", 0

PreInputText:

db "Input + , - , \* or / >", 0

PlusOp:

db "23 + 15 = 38", 0

MinusOp:

db "23 - 15 = 8", 0

MultiOp:

db "23 \* 15 = 345", 0

DiviOp:

db "23 / 15 = 1,5(3)", 0

times 510-($-$$) db 0 ;Вставляем нули до 510 байта

dw 0xaa55 ;байты для загрузчика

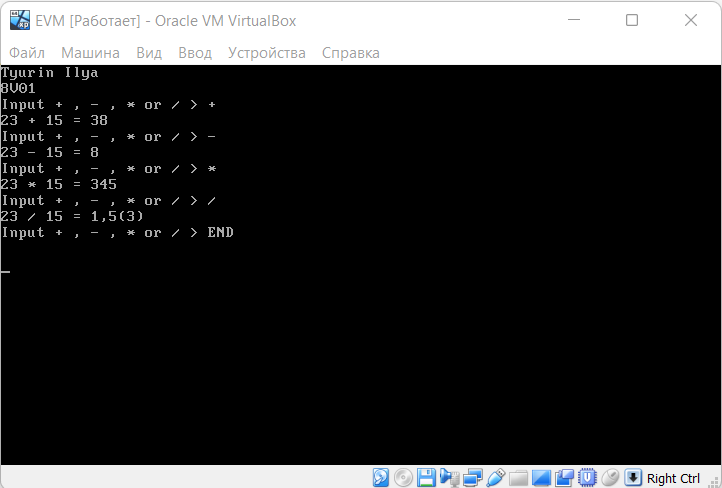


Рис. 1. Результат работы программы

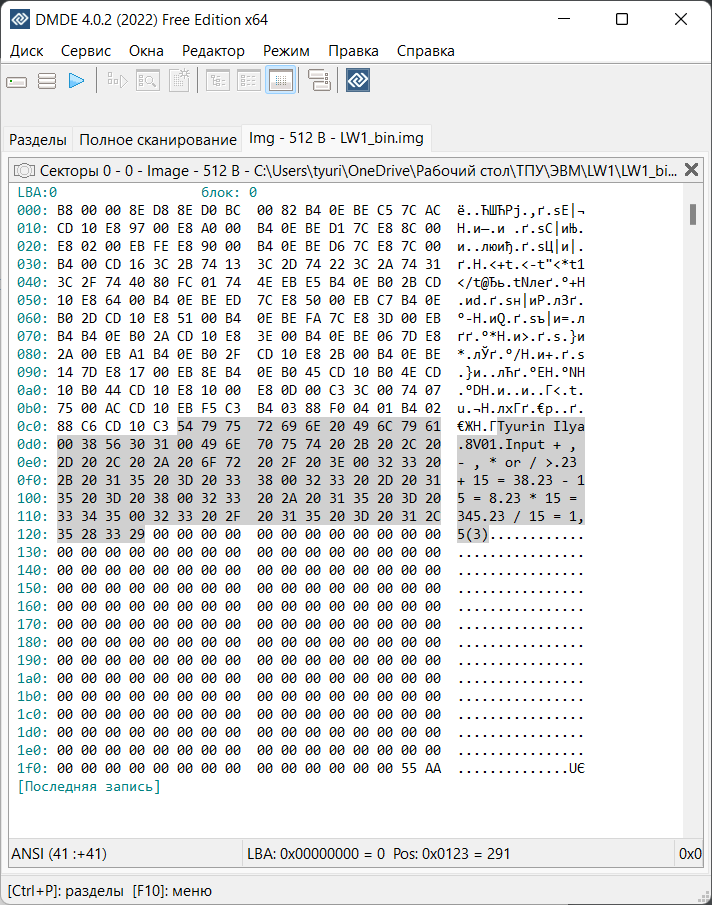


Рис. 2. Бинарный файл.

**Вывод**

В ходе лабораторной работы была создана программа на языке ассемблера для x86 в виде загрузчика, которая выводит на экране на разных строчках фамилию и номер группы. Также был сделан вывод результата операции между числами 23 и 15 в виде «23 op 15=Z», где op – операция, а Z результат.