Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Ярославский государственный технический университет»

Кафедра «Информационные системы и технологии»

Отчет по лаб/работе защищен

с оценкой \_\_\_\_\_\_\_\_\_

Преподаватель к.т.н Ю.В. Царёв

«13» апреля 2022

**Программирование ветвлений и циклов**

Отчет о лабораторной работе по курсу «Машино-зависимые языки программирования»

ЯГТУ 09.03.04-007 ЛР

|  |  |
| --- | --- |
|  | Отчет выполнил  студент группы ЦПИ-11    Аристов Д.В.  «13» апреля 2022 |

2022

Цель работы: закрепление знаний по командам условного и безусловного переходов и циклов на примере программ на языке ассемблера, а также приобретение навыков написания программ с циклами.

Задание:

7. Подсчитать количество нулевых и ненулевых элементов в заданном векторе и определить, каких элементов в векторе больше:

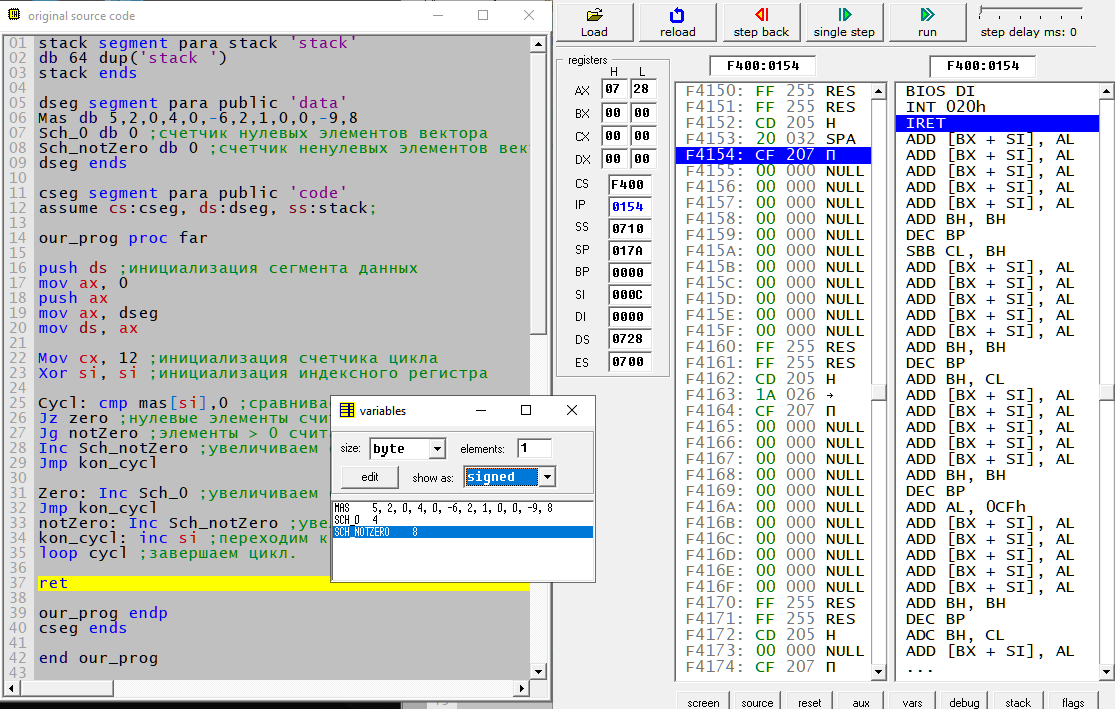
а) элементы вектора однобайтовые;

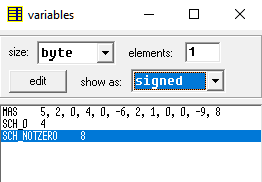
б) элементы вектора двухбайтовые.

X = 12 (количество элементов);

Схема выполнения: создаем массив данных (положительные, отрицательные числа и нули) и счетчик нулевых и ненулевых чисел. Инициализируем сегмент данных. Инициализируем счетчик цикла, поместив в регистр CX количество элементов массива. Проходим циклом по элементам массива, сравнивая их значение с нулем и увеличиваем соответствующий счетчик.

А)





stack segment para stack 'stack'

db 64 dup ('stack ')

stack ends

dseg segment para public 'data'

Mas db 5,2,0,4,0,-6,2,1,0,0,-9,8

Sch\_0 db 0 ;счетчик нулевых элементов вектора

Sch\_notZero db 0 ;счетчик ненулевых элементов вектора

dseg ends

cseg segment para public 'code'

assume cs: cseg, ds: dseg, ss: stack;

our\_prog proc far

push ds

mov ax, 0

push ax

mov ax, dseg

mov ds, ax

Mov cx, 12

Xor si, si

Cycl: cmp mas[si], 0

Jz zero

Jg notZero

Inc Sch\_notZero

Jmp kon\_cycl

Zero: Inc Sch\_0

Jmp kon\_cycl

notZero: Inc Sch\_notZero

kon\_cycl: inc si

loop cycl

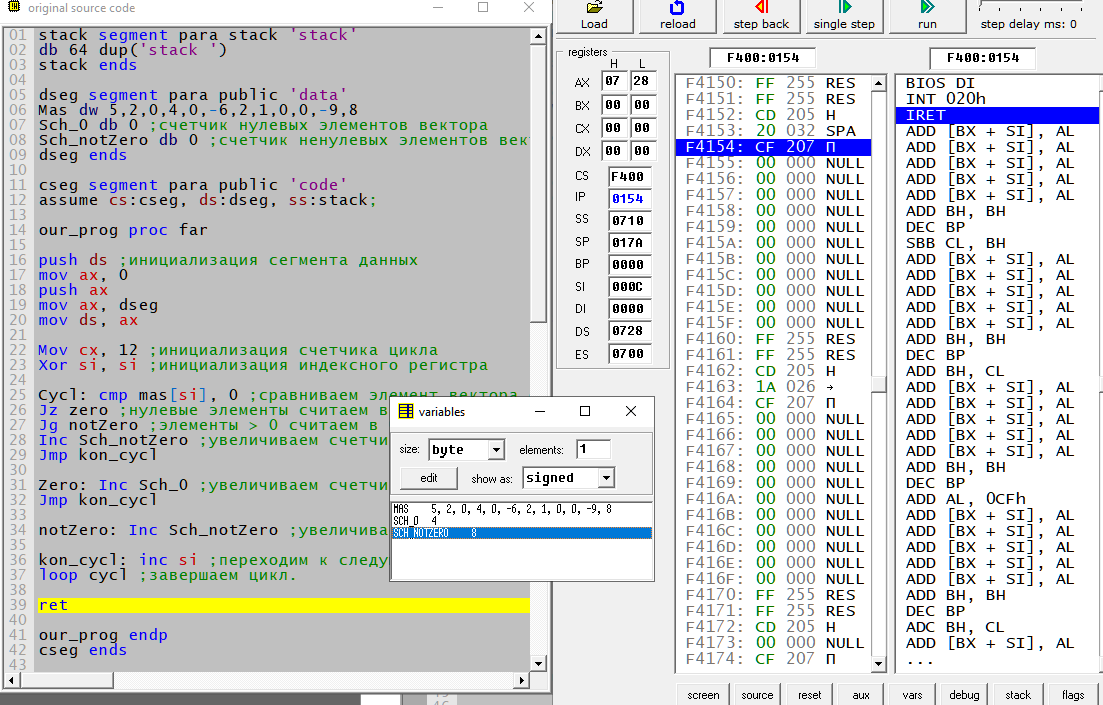
ret

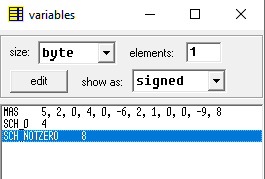
our\_prog endp

cseg ends

end our\_prog

Б)





stack segment para stack 'stack'

db 64 dup ('stack ')

stack ends

dseg segment para public 'data'

Mas dw 5,2,0,4,0,-6,2,1,0,0,-9,8

Sch\_0 db 0 ;счетчик нулевых элементов вектора

Sch\_notZero db 0 ;счетчик ненулевых элементов вектора

dseg ends

cseg segment para public 'code'

assume cs: cseg, ds: dseg, ss: stack;

our\_prog proc far

push ds

mov ax, 0

push ax

mov ax, dseg

mov ds, ax

Mov cx, 12

Xor si, si

Cycl: cmp mas[si], 0

Jz zero

Jg notZero

Inc Sch\_notZero

Jmp kon\_cycl

Zero: Inc Sch\_0

Jmp kon\_cycl

notZero: Inc Sch\_notZero

kon\_cycl: inc si

loop cycl

ret

our\_prog endp

cseg ends

end our\_prog

Вывод: с помощью данной лабораторной работы я закрепил знания по командам условного и безусловного переходов и циклов на примере программ на языке ассемблера, а также приобрёл навыки написания программ с циклами.