**Министерство науки и высшего образования Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**УНИВЕРСИТЕТ ИТМО»**

**ФАКУЛЬТЕТ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ**

**ОТЧЕТ**

**по Лабораторной работе № 1**

**«Исследование работы ЭВМ при выполнении линейных программ»**

Специальность 09.02.07«Информационные системы и программирование»

Дисциплина «Архитектура аппаратных средств»

|  |  |
| --- | --- |
| Преподаватель:  Ошурок Э.Э.  Ассистент:  Теричев М.С.  «\_\_» \_\_\_\_\_\_ 2020 г.  Оценка \_\_\_\_\_\_\_\_\_ | Выполнил:  студент группы Y2337  Князев А. А. |

Санкт-Петербург

2019/2020

# ЗАДАНИЕ

Цель работы – исследование порядка выполнения арифметических команд, команд пересылки данных и организации процедур.

Задание: написать процедуру подсчета отрицательных числовых значений в числовом массиве. Процедуру реализовать любым способом.

Таблица 1. Текст исходной программы

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарии |
|  |  | org 0x100 |  |
| 0100 | B90500 | mov cx, 5 | Размер массива |
| 0103 | 6A00 | Push 0000 | Заталкиваем в стек |
| 0105 | E682E01 | Push 012E |  |
| 0108 | E80100 | Call 010C |  |
| 010B | C3 | ret |  |
|  |  | proc: |  |
| 010С | 55 | push bp |  |
| 010D | 89E5 | mov bp,sp | Теперь bp указывает на вершину стека |
| 010F | 8B5E04 | mov bx,[bp+4] |  |
| 0112 | 8B5606 | mov dx,[bp+6] |  |
|  |  | while: |  |
| 0115 | 8B07 | mov ax, [bx] | В ах перемещаем элемент из массива |
| 0117 | 783C302 | add bx,2 | Сдвигаемся на один элемент в массиве |
| 011A | 83F800 | cmp ax,0 | Сравниваем c 0 |
| 011D | 7C05 | jl min |  |
| 011F | 49 | dec cx |  |
| 0120 | 75F3 | jnz while |  |
| 011E | 7406 | jz zer |  |
|  |  | if: | +1 к счетчику, если число отрицательное |
| 0124 | 42 | Inc dx |
| 0127 | 49 | dec cx |
| 128 | 75ED | jnz while |
| 012A | 7400 | jz zer |
|  |  | ic: | Записываем в ах кол-во отрицательных чисел |
| 012C | 89D0 | mov ax, dx |
| 012E | 5D | pop bp | Возвращаем стек в изначальное состояние |
| 012F | C3 | ret |  |
|  |  | Array dw 1,-2,3,-4,5 |  |

Таблица 2. Трассировка

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Адрес | Команда | AX | BX | CX | DX | IP | OF | DF | IF | SF | ZF | AF | PF | CF | Память | Стек | Аргументы команды |
| 0100 | mov | 0000 | 0000 | 0005 | 0000 | 0103 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | CX,0005 |
| 0103 | push | 0000 | 0000 | 0005 | 0000 | 0105 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | 0000 |
| 0105 | push | 0000 | 0000 | 0005 | 0000 | 0108 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 012E | 012E |
| 0108 | call | 0000 | 0000 | 0005 | 0000 | 010C | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 010B | 010C |
| 010C | push | 0000 | 0000 | 0005 | 0000 | 010D | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | BP |
| 010D | mov | 0000 | 0000 | 0005 | 0000 | 010F | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | BP,SP |
| 010F | mov | 0000 | 012E | 0005 | 0000 | 0112 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | BP+04 | 0000 | BX,[BP+04] |
| 0112 | mov | 0000 | 012E | 0005 | 0000 | 0115 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | BP+06 | 0000 | DX,[BP+06] |
| 0115 | mov | 0001 | 012E | 0005 | 0000 | 0117 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 012E | 0000 | AX,[BX] |
| 0117 | add | 0001 | 0130 | 0005 | 0000 | 011A | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |  | 0000 | BX,0002 |
| 011A | cmp | 0001 | 0130 | 0005 | 0000 | 011D | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | AX,0000 |
| 011D | jl | 0001 | 0130 | 0005 | 0000 | 011F | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | 0124 |
| 011F | dec | 0001 | 0130 | 0004 | 0000 | 0120 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | CX |
| 0120 | jnz | 0001 | 0130 | 0004 | 0000 | 0115 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | 0115 |
| 0115 | mov | FFFE | 0130 | 0004 | 0000 | 0117 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0130 | 0000 | AX,[BX] |
| 0117 | add | FFFE | 0132 | 0004 | 0000 | 011A | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | BX,0002 |
| 011A | cmp | FFFE | 0132 | 0004 | 0000 | 011D | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |  | 0000 | AX,0000 |
| 011D | jl | FFFE | 0132 | 0004 | 0000 | 0124 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |  | 0000 | 0124 |
| 0124 | inc | FFFE | 0132 | 0004 | 0001 | 0125 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | DX |
| 0125 | dec | FFFE | 0132 | 0003 | 0001 | 0126 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |  | 0000 | CX |
| 0126 | jnz | FFFE | 0132 | 0003 | 0001 | 0115 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |  | 0000 | 0115 |
| 0115 | mov | 0003 | 0132 | 0003 | 0001 | 0117 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0132 | 0000 | AX,[BX] |
| 0117 | add | 0003 | 0134 | 0003 | 0001 | 011A | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | BX,0002 |
| 011A | cmp | 0003 | 0134 | 0003 | 0001 | 011D | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |  | 0000 | AX,0000 |
| 011D | jl | 0003 | 0134 | 0003 | 0001 | 011F | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |  | 0000 | 0124 |
| 011F | dec | 0003 | 0134 | 0002 | 0001 | 0120 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | CX |
| 0120 | jnz | 0003 | 0134 | 0002 | 0001 | 0115 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | 0115 |
| 0115 | mov | FFFC | 0134 | 0002 | 0001 | 0117 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0134 | 0000 | AX,[BX] |
| 0117 | add | FFFC | 0136 | 0002 | 0001 | 011A | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |  | 0000 | BX,0002 |
| 011A | cmp | FFFC | 0136 | 0002 | 0001 | 011D | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |  | 0000 | AX,0000 |
| 011D | jl | FFFC | 0136 | 0002 | 0001 | 0124 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |  | 0000 | 0124 |
| 0124 | inc | FFFC | 0136 | 0002 | 0002 | 0125 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | DX |
| 0125 | dec | FFFC | 0136 | 0001 | 0002 | 0126 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | CX |
| 0126 | jnz | FFFC | 0136 | 0001 | 0002 | 0115 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | 0115 |
| 0115 | mov | 0005 | 0136 | 0001 | 0002 | 0117 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0136 | 0000 | AX,[BX] |
| 0117 | add | 0005 | 0138 | 0001 | 0002 | 011A | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |  | 0000 | BX,0002 |
| 011A | cmp | 0005 | 0138 | 0001 | 0002 | 011D | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |  | 0000 | AX,0000 |
| 011D | jl | 0005 | 0138 | 0001 | 0002 | 011F | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |  | 0000 | 0124 |
| 011F | dec | 0005 | 0138 | 0000 | 0002 | 0120 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |  | 0000 | CX |
| 0120 | jnz | 0005 | 0138 | 0000 | 0002 | 0122 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |  | 0000 | 0115 |
| 0122 | jz | 0005 | 0138 | 0000 | 0002 | 012A | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |  | 0000 | 012A |
| 012A | mov | 0002 | 0138 | 0000 | 0002 | 012C | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |  | 0000 | AX,DX |
| 012C | pop | 0002 | 0138 | 0000 | 0002 | 0000 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |  | 010B | BP |
| 012D | ret |  |  |  |  |  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |  |  |

3. Листинг программы:

org 0x100

mov cx,5

push 0

push Array

call func

ret

func:

push bp

mov bp,sp

mov bx,[bp+4]

mov dx,[bp+6]

while:

mov ax,[bx]

add bx,2

cmp ax,0

jl min

dec cx

jnz while

jz zer

min:

inc dx

dec cx

jnz while

jz zer

zer:

mov ax,dx

pop bp

ret

Array dw 1,-2,3,-4,5