Министерство науки и высшего образования РФ

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №4

по дисциплине «ЭВМ и переферийные устройства»

на тему «Работа процессора с подпрограммами»

Выполнили: студенты группы 22ВВП1

Беляев Д. И.

Демин М. С.

Сергунов М. Р.

Приняли:

Никишин К. И.

Патунин Д. В.

Пенза 2025

**Цель работы**

Изучение особенностей выполнения команд работы с подпрограммами и передачи параметров в подпрограммы

**Задание**

Задание1: Составить программу, выполняющую обработку массива, аналогично заданию предыдущей работы, но оформленной в виде "ближней" процедуры, параметры в которую передаются через регистры.

Задание2: Составить программу, выполняющую обработку массива, аналогично заданию предыдущей работы, но оформленной в виде "ближней" процедуры, параметры в которую передаются через стек.

**Ход работы**

Задание 1

stseg segment para stack

dw 16 dup(?)

stseg ends

dseg segment para

x dw 0, 1, 0, 2, 3, 0, 4, 5 ; массив из 8 чисел

y dw ? ; результат – количество нулей

dseg ends

cseg segment para

assume cs:cseg, ds:dseg, ss:stseg

count\_zeros\_reg proc near

; Входные параметры:

; BX - адрес массива

; CX - количество элементов

; Выход:

; AX - количество нулевых элементов

xor ax, ax ; Обнуляем счетчик нулей

loop\_start:

mov dx, [bx] ; Загружаем текущий элемент массива

cmp dx, 0 ; Сравниваем с 0

jne next\_element ; Если не 0, переходим дальше

inc ax ; Если 0, увеличиваем счетчик

next\_element:

add bx, 2 ; Переход к следующему элементу

loop loop\_start ; Повторяем цикл

ret

count\_zeros\_reg endp

lab41 proc far

push ds

xor ax, ax

push ax

mov ax, dseg

mov ds, ax

mov bx, offset x ; Адрес массива

mov cx, 8 ; Количество элементов

call count\_zeros\_reg ; Вызов процедуры

mov y, ax ; Сохранение результата

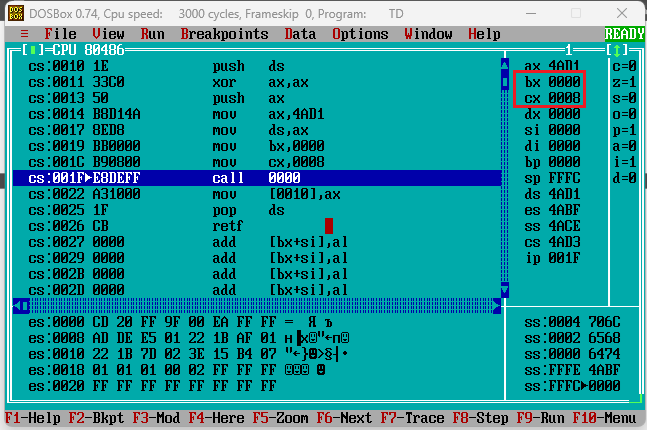
pop ds

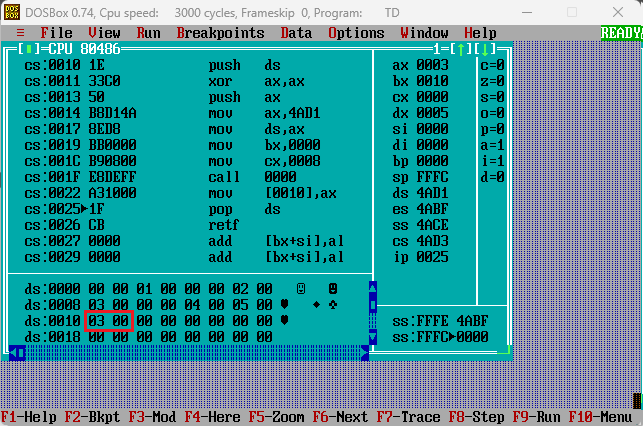
ret

lab41 endp

cseg ends

end lab41





Задание 2

stseg segment para stack

dw 16 dup(?)

stseg ends

dseg segment para

x dw 0, 1, 0, 2, 3, 0, 4, 5 ; массив из 8 чисел

y dw ? ; результат – количество нулей

dseg ends

cseg segment para

assume cs:cseg, ds:dseg, ss:stseg

count\_zeros\_stack proc near

; Входные параметры (через стек):

; [BP+4] - адрес массива

; [BP+6] - количество элементов

; Выход:

; AX - количество нулевых элементов

push bp

mov bp, sp ; Устанавливаем базовый указатель стека

mov bx, [bp+4] ; Загружаем адрес массива

mov cx, [bp+6] ; Загружаем количество элементов

xor ax, ax ; Обнуляем счетчик нулей

loop\_start:

mov dx, [bx] ; Загружаем текущий элемент массива

cmp dx, 0 ; Сравниваем с 0

jne next\_element ; Если не 0, переходим дальше

inc ax ; Если 0, увеличиваем счетчик

next\_element:

add bx, 2 ; Переход к следующему элементу

loop loop\_start ; Повторяем цикл

pop bp

ret 4 ; Убираем параметры из стека (2 параметра по 2 байта)

count\_zeros\_stack endp

lab5 proc far

push ds

mov ax, dseg

mov ds, ax

push 8 ; Количество элементов

push offset x ; Адрес массива

call count\_zeros\_stack ; Вызов процедуры

mov y, ax ; Сохранение результата

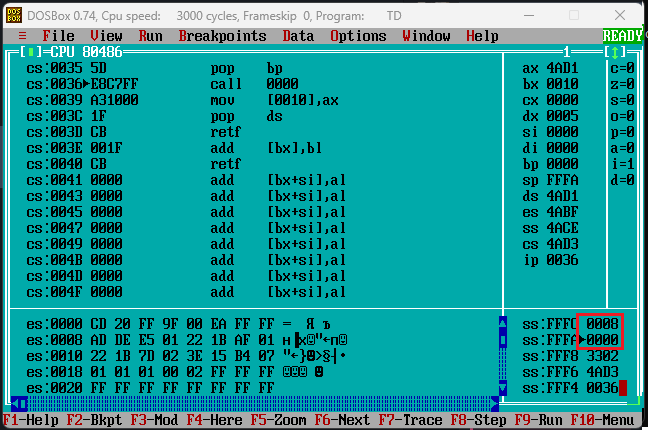
pop ds

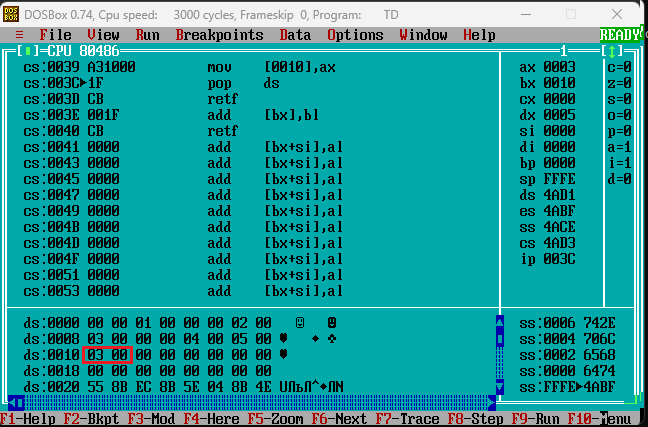
ret

lab5 endp

cseg ends

end lab5





**Вывод**

Изучили особенности выполнения команд работы с подпрограммами и передачу параметров в подпрограммы