**Министерство науки и высшего образования Российской Федерации**

Федеральное государственное бюджетное образовательное учреждение   
высшего образования

**«Вятский Государственный Университет»**

Институт математики и информационных систем

Факультет автоматики и вычислительной техники

Кафедра РЭС

**Отчёт по лабораторной работе №2:**

**ПРИНЦИПЫ**

**ВЫПОЛНЕНИЯ КОМАНД ВЕТВЛЕНИЯ, ОРГАНИЗАЦИЯ**

**ЦИКЛОВ И ПОДПРОГРАММ**

Дисциплина:

**«Цифровые устройства и микропроцессоры»**

Вариант 4

Выполнил: студент ИНБс-3301-01-00 Пелых А.В.

(должность) (группа) (ФИО) (подпись)

Проверил: доцент РЭС Земцов М.А.

(должность) (кафедра) (ФИО) (подпись)

Киров 2025

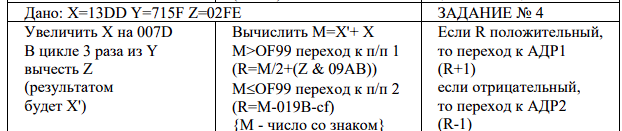
Изучение принципов выполнения команд ветвления, организации циклов и подпрограмм микропроцессоров с архитектурой x86. 

Рисунок 1 – Вариант задания

**Решение:**

.686

.model flat,stdcall

.stack 100h

ExitProcess PROTO STDCALL :DWORD

.data

X dd 013DDh

Y dd 0715Fh

Z dd 002FEh

X1 dd ?

M dd ?

R dd ?

.code

Start:

; X + 007Dh

mov eax, X

add eax, 007Dh

mov X, eax

; 3 раза (Y-Z)=X'

mov eax, Y

mov ecx, 3

SubLoop:

sub eax, Z

dec ecx

jnz SubLoop

mov X1, eax

; M = X' + X

mov eax, X1

add eax, X

mov M, eax

; Если M > 0F99h => Proc1, иначе Proc2

cmp eax, 00F99h

jg Proc1

jmp Proc2

Proc1:

; R = M/2 + (Z & 09ABh)

mov eax, M

shr eax, 1

mov ebx, Z

and ebx, 009ABh

add eax, ebx

mov R, eax

Call CheckSign

ret

Proc2:

; R = M - 019Bh - cf

mov eax, M

sub eax, 019Bh

sbb eax, 0 ; Вычитаем флаг cf

mov R, eax

Call CheckSign

ret

CheckSign:

; Если R положительный, то R = R + 1, иначе R = R - 1

mov eax, R

test eax, 08000h

jz Adr1

Adr2:

dec eax ; Если отрицательный R => R = R - 1

Call Done

Adr1:

inc eax ; Если положительный R => R = R + 1

Done:

mov R, eax

invoke ExitProcess, 0

End Start

Вывод: в ходе выполнения лабораторной работы ознакомился с циклами и процедурами в ассемблере, а также выполнил задание, содержащее ветвление на процедуры.

Я изучил следующие операции:

Loop: цикл,

Call: вызов подпрограммы,

ret: выход из подпрограммы,

jmp: прыжок к процедуре.