

Министерство науки высшего образования Российской Федерации  
федеральное государственное бюджетное образовательное

учреждение высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
МОРДОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
им. Н. П. ОГАРЁВА»

(ФГБОУ ВО МГУ им. Н.П. Огарева)

Факультет довузовской подготовки и  
среднего профессионального образования

Выпускающая предметная цикловая комиссия общепрофессиональных и  
специальных (информационно-коммуникационных) дисциплин

## ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ 16

«Условный и безусловный переход»

по дисциплине: «Системное программирование»

Автор лабораторной работы \_\_\_\_\_ А.А. Пивкин  
(подпись) (дата)

Направление подготовки 09.02.03 Программирование в компьютерных  
системах

Преподаватель \_\_\_\_\_ В.В. Никулин  
(подпись) (дата)

Саранск

2023

## Выполнение задания:

1. Перед выполнением задания, изучил теоретический материал в тексте задания (Рисунок 1).

**Учебный курс. Часть 16. Условные и безусловные переходы**

Наконец-то мы добрались и до переходов! В этой части научимся программировать условные и безусловные переходы. Вообще, трудно представить себе программу без проверки условий и переходов. С их помощью в программе реализуются различные управляющие конструкции, ветвления и даже циклы.

**Безусловные переходы**

Безусловный переход – это переход, который выполняется всегда. Безусловный переход осуществляется с помощью команды **JMP**. У этой команды один операнд, который может быть непосредственным адресом (меткой), регистром или ячейкой памяти, содержащей адрес. Существуют также "дальние" переходы – между сегментами, однако здесь мы их рассматривать не будем. Примеры безусловных переходов:

```
jmp metka ;Переход на метку
jmp bx ;Переход по адресу в BX
jmp word[bx] ;Переход по адресу, содержащемуся в памяти по адресу в BX
```

**Условные переходы**

Условный переход осуществляется, если выполняется определённое условие, заданное флагами процессора (кроме одной команды, которая проверяет CX на равенство нулю). Как вы помните, состояние флагов изменяется после выполнения арифметических, логических и некоторых других команд. Если условие не выполняется, то управление переходит к следующей команде.

Существует много команд для различных условных переходов. Также для некоторых команд есть синонимы (например, **JZ** и **JE** – это одно и то же). Для наглядности все команды условных переходов приведены в таблице:

Команда	Переход, если	Условие перехода
---------	---------------	------------------

Рисунок 1 – Изучение материала

2. После изучения материала приступил к рассматриванию примеров программ, запустил программу в Turbo Debugger и посмотрел её исполнение по шагам (Рисунки 2-3).

```
1 use16
2 org 100h
3
4 mov al, [x]
5 add al, [y]
6 jo error
7 mov ah, 09h
8 mov dx, ok_msg
9 int 21h
10 exit:
11 mov ah, 09h
12 mov dx, pak
13 int 21h
14
15 mov ah, 08h
16 int 21h
17
18 mov ax, 4C00h
19 int 21h
20 error:
21 mov ah, 09h
22 mov dx, err_msg
23 int 21h
24 jmp exit
25 x db -89
26 y db -55
27 err_msg db 'Error: overflow detected.', 13, 10, '$'
28 ok_msg db 'OK', 13, 10, '$'
29 pak db 'Press any key...$'
30
```

Рисунок 2 – Код программы

```

C:\WINDOWS\system32\cmd.exe
E File Edit View Run Breakpoints Data Options Window
[ ] CPU Pentium Pro
cs:0100 B8FF00 mov ax,00FF
cs:0103 40 inc ax
cs:0104 90 nop
cs:0105 89C3 mov bx,ax
cs:0107 B8004C mov ax,4C00
cs:010A CD21 int 21
cs:010C 7105 jno 0113
cs:010E 83C404 add sp,0004
cs:0111 FF7616 push word ptr [bp+16]
cs:0114 90 nop
cs:0115 0E push cs
cs:0116 E81BA3 call A434
cs:0119 C7040000 mov word ptr [si],0000
cs:011D C744020000 mov word ptr [si+02],0000
cs:0122 C744060000 mov word ptr [si+06],0000
ds:0000 CD 20 FB 9F 00 9A F0 FE = JA bE1
ds:0008 1D F0 32 0D E5 10 0F 07 =E28x**
ds:0010 51 0E 56 01 57 04 34 0E QJUM*4f
ds:0018 01 01 01 00 02 04 FF FF @@@ @*
ds:0020 FF FF FF FF FF FF FF FF

```

Рисунок 3 – Запуск программы в Turbo Debugger

3.Выполнил задание, данное в тексте лабораторной работы (Рисунки 4-6)

```

1 use16
2 org 100h
3 jmp start
4 menu db '1 - Print hello',13,10
5      db '2 - Print go away',13,10
6      db '0 - Exit',13,10,'$'
7 select db 13,10,'Select>$'
8 hello db 13,10,'Hello!',13,10,13,10,'$'
9 go_away db 13,10,'Go away!',13,10,13,10,'$'
10 start:
11 mov ah,09h
12 mov dx,menu
13 int 21h
14
15 select_loop:
16 mov ah,09h
17 mov dx,select
18 int 21h
19
20 mov ah,01h
21 int 21h
22
23 cmp al,'1'
24 je c1
25 cmp al,'2'
26 je c2
27 cmp al,'0'
28 je exit
29 jmp select_loop
30 c1:
31 mov ah,09h
32 mov dx,hello
33 int 21h
34 jmp start
35 c2:
36 mov ah,09h
37 mov dx,go_away
38 int 21h
39 jmp start
40 exit:
41 mov ax,4C00h
42 int 21h

```

Рисунок 4 – Код программы

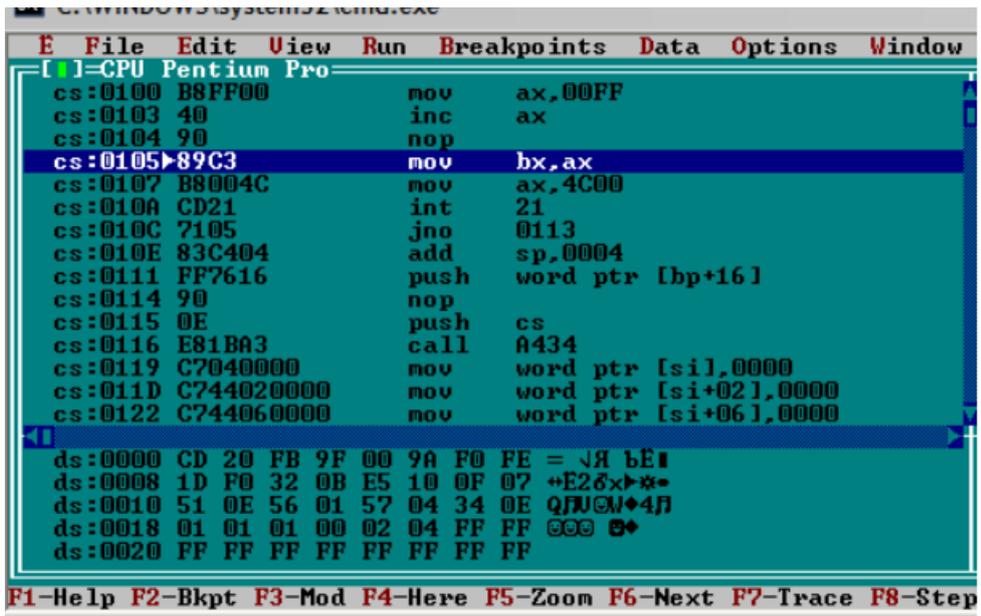


Рисунок 5 – Запуск программы в Turbo Debugger



Рисунок 6 – Запуск программы