РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ № 7

дисциплина: Архитектура компьютера

Студент: Семёнов Александр Дмитриевич

Группа: НКАбд-05-25

МОСКВА

2025 г.

Содержание

1. Цель работы.

Изучить команды условного и безусловного переходов, приобрести навыки написания программ с использованием переходов познакомиться с назначением и структурой файла листинга.

2. Задание.

Изучить команды условного и безусловного переходов, приобрести навыки написания программ с использованием переходов и познакомиться с назначением и структурой файла листинга.

3. Выполнение лабораторной работы.

3.1 Реализация переходов в NASM.

Я создание и переход в каталог для лабораторной №7.

```
adsemyonov@fedora:~$ mkdir -p ~/work/arch-pc/lab07
adsemyonov@fedora:~$ cd ~/work/arch-pc/lab07
adsemyonov@fedora:~/work/arch-pc/lab07$
```

Рис. 1. Создание каталога и переход в него.

Я создал файл **lab7-1.asm** и ввел в него текст программы из листинга 7.1.

```
adsemyonov@fedora:~/work/arch-pc/lab07$ touch lab7-1.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ gedit lab7-1.asm
```

Рис. 2. Создание и открытие файла.

Открыть | lab7-1.asm ~/work/arch-pc/lab07 | Сохранить

```
1 %include 'in_out.asm'
2 SECTION .data
3 msg1: DB 'Сообщение № 1',0
4 msg2: DB 'Сообщение № 2',0
5 msg3: DB 'Сообщение № 3',0
6 SECTION .text
7 GLOBAL _start
8 _start:
9 jmp _label2
10 _label1:
11 mov eax, msg1
12 call sprintLF
13 _label2:
14 mov eax, msg2
15 call sprintLF
16 _label3:
17 mov eax, msg3
18 call sprintLF
19 _end:
20 call quit
```

Рис. 3. Текст программы.

Потом я создал файл и запустил его. Файл **in_out.asm** я скопировал через **Midnight Commander.**

```
adsemyonov@fedora:~/work/arch-pc/lab07$ ls
in_out.asm  lab7-1.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
lab7-1.asm:3: warning: no operand for data declaration [-w+db-empty]
lab7-1.asm:4: warning: no operand for data declaration [-w+db-empty]
lab7-1.asm:5: warning: no operand for data declaration [-w+db-empty]
adsemyonov@fedora:~/work/arch-pc/lab07$ gedit lab7-1.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ gedit lab7-1.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1.o
ld: no input files
adsemyonov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
adsemyonov@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 2
Сообщение № 3
adsemyonov@fedora:~/work/arch-pc/lab07$
```

Рис. 4. Создание и запуск файла.

Открыть | *lab7-1.asm ~/work/arch-pc/lab07 | Сохранить

```
%include 'in_out.asm'
SECTION .data
msg1: DB 'Сообщение № 1',0
msg2: DB 'Сообщение № 2',0
msg3: DB 'Сообщение № 3',0
SECTION .text
GLOBAL _start
_start:
jmp _label2
_label1:
mov eax, msg1
call sprintLF
jmp _end
_label2:
mov eax, msg2
call sprintLF
jmp _label1
_label3:
mov eax, msg3
call sprintLF
_end:
call quit
```

Рис. 5. Измененный текст программы.

Далее я изменил текст программы в файле **lab7-1.asm** в соответсвие с листингом 7.2.

```
adsemyonov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
adsemyonov@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 2
Сообщение № 1
adsemyonov@fedora:~/work/arch-pc/lab07$
```

Рис. 6. Создание и запуск файла.

Потом я опять изменил текст программы и запустил ее.

Открыть | lab7-1.asm ~/work/arch-pc/lab07 | Сохранить

```
%include 'in_out.asm'
SECTION .data
msg1: DB 'Сообщение № 1',0
msg2: DB 'Сообщение № 2',0
msg3: DB 'Сообщение № 3',0
SECTION .text
GLOBAL _start
_start:
jmp _label3
_label1:
mov eax, msg1
call sprintLF
jmp _end
_label2:
mov eax, msg2
call sprintLF
jmp _label1
_label3:
mov eax, msg3
call sprintLF
jmp_label2
_end:
call quit
```

Рис. 7. Изменение программы во второй раз.

```
adsemyonov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
lab7-1.asm:21: warning: label alone on a line without a colon might be in error [-w+la
bel-orphan]
adsemyonov@fedora:~/work/arch-pc/lab07$ gedit lab7-1.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
adsemyonov@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 3
Сообщение № 2
Сообщение № 1
adsemyonov@fedora:~/work/arch-pc/lab07$
```

Рис. 8. Создание и запуск файла.

Я создал файл **lab7-2.asm.**

```
adsemyonov@fedora:~/work/arch-pc/lab07$ touch lab7-2.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ ls
in_out.asm  lab7-1  lab7-1.asm  lab7-1.o  lab7-2.asm
adsemyonov@fedora:~/work/arch-pc/lab07$
```

Рис. 9. Создание файла.

Потом я ввел текст программы из листинга 7.3 и проверил работу программы для разных значений B.

Открыть | lab7-2.asm ~/work/arch-pc/lab07 | Сохранить

```
%include 'in_out.asm'
section .data
msg1 db 'Введите B: ',0h
msg2 db "Наибольшее число: ",0h
A dd '20'
C dd '50'
section .bss
max resb 10
B resb 10
section .text
global _start
_start:
mov eax,msg1
call sprint
mov ecx,B
mov edx,10
call sread
mov eax,B
call atoi
mov [B],eax
mov ecx,[A]
mov [max],ecx
cmp ecx,[C]
jg check_B
mov ecx,[C]
mov [max],ecx
check_B:
mov eax,max
call atoi
mov [max],eax
mov ecx,[max]
cmp ecx,[B]
jg fin
mov ecx,[B]
mov [max],ecx
fin:
mov eax, msg2
call sprint
mov eax,[max]
call iprintLF
call quit
```

Рис. 10. Текст программы.

```
adsemyonov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
adsemyonov@fedora:~/work/arch-pc/lab07$ ./lab7-2
Введите B: 6
Наибольшее число: 50
adsemyonov@fedora:~/work/arch-pc/lab07$ ./lab7-2
Введите B: 29
Наибольшее число: 50
adsemyonov@fedora:~/work/arch-pc/lab07$ ./lab7-2
Введите B: 100
Наибольшее число: 100
adsemyonov@fedora:~/work/arch-pc/lab07$
```

Рис. 11. Создание и запуск файла.

3.2 Изучение структуры файла листинга.

Я создал файл листинга для программы из файла **lab7-2.asm** и открыл его с помощью текстового редактора **mcedit**.

```
adsemyonov@fedora:~/work/arch-pc/lab07$ nasm -f elf -l lab7-2.lst lab7-2.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ mcedit lab7-2.lst

adsemyonov@fedora:~/work/arch-pc/lab07$
```

Рис. 12. Создание файла листинга.

С помощью **ls -la** я сравнил время когда были созданы исполняемый файл и файл листинга. Как и ожидалось, исполняемый файл старый, потому новый не был создан из-за ошибки, а файл листинга создался только что.

```
adsemyonov@fedora:~/work/arch-pc/lab07$ ls -la lab7-2 lab7-2.lst
-rwxr-xr-x. 1 adsemyonov adsemyonov  5144 ноя 23 16:12 lab7-2
-rw-r--r--. 1 adsemyonov adsemyonov 13037 ноя 23 16:30 lab7-2.lst
```

Рис. 13. Просмотр.

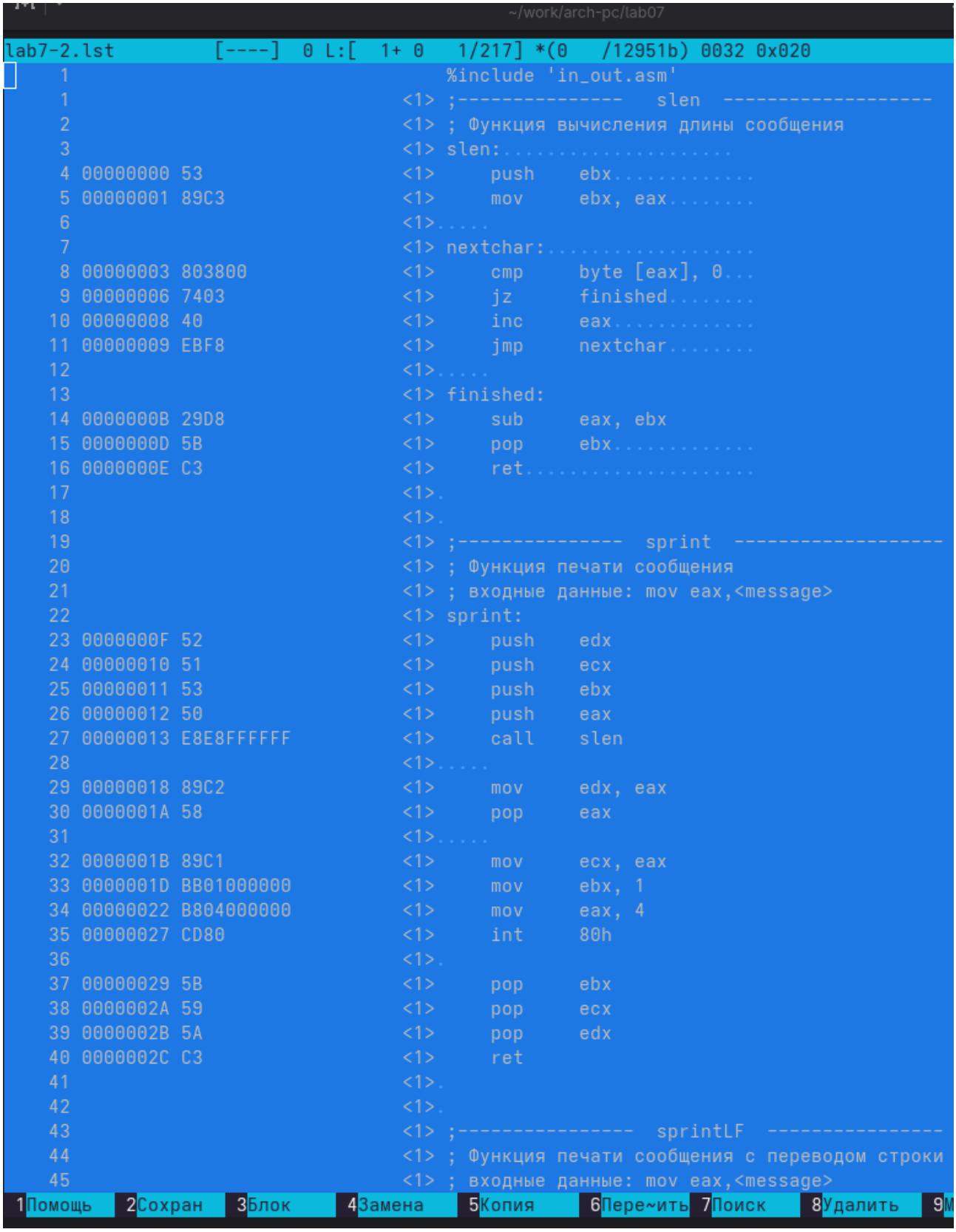

Рис. 13. Содержимое листинга.

Строка 8: cmp byte [eax], 0

Что делает: Сравнивает байт (символ) по адресу в регистре EAX с нулем.
Объяснение:

· byte - операция с одним байтом
· [eax] - обращение к памяти по адресу в регистре EAX
· 0 - нулевой байт (символ конца строки в C-style строках)
Назначение: Проверяет, не достигнут ли конец строки (нулевой терминатор)

Строка 27: call slen

Что делает: Вызывает функцию slen (string length)
Объяснение:

· call - команда вызова подпрограммы
· slen - имя функции вычисления длины строки
· Адрес FFFFFFEB - это смещение -21 в дополнительном коде (переход назад)
Назначение: Вычисляет длину строки для последующего вывода

Строка 35: int 80h

Что делает: Вызывает прерывание ядра Linux
Объяснение:

· int - команда вызова программного прерывания
· 80h - номер прерывания (128 в десятичной системе)
· Перед этим в регистрах устанавливаются параметры:
  · eax=4 - номер системного вызова (sys_write)
  · ebx=1 - файловый дескриптор (stdout)
  · ecx - указатель на строку
  · edx - длина строки
Назначение: Вывод строки на экран через системный вызов Linux

После этого я открыла файл **lab7-2.asm** и удалила один операнд.

Рис. 14. До удаления было: ***mov ecx,[A].***

Потом я выполнил трансляцию с получением файла листинга.

Рис. 15. Выполнение трансляции.

Использовал команду **ls,** чтобы проверить какие файлы у меня есть.

Рис. 16. Проверка файлов.

Рис. 17. Сообщение об ошибке.

4. Задания для самостоятельной работы.

Задание №1.

Я создал файл **lab7-3.asm** для выполнения первого задания.

Рис. 18. Создание файла.

Я написал программу по нахождению наименьшей из 3 целочисленных переменных a,b и c. Значения я взял из таблицы в соответствии с полученным мной 8 вариантом, и потом я запустил ее.

Открыть

~/work/arch-pc/lab07

```
%include 'in_out.asm'
section .data
msg1 db 'Наименьшее число: ',0h
A db '52',0
B db '33',0
C db '40',0
section .bss
min resb 10
section .text
global _start
_start:
mov eax,A
call atoi
mov [min],eax
mov eax,B
call atoi
mov ebx,eax
mov eax,[min]
cmp eax,ebx
jl check_C
mov [min],ebx
check_C:
mov eax,C
call atoi
mov ebx,eax
mov eax,[min]
cmp eax,ebx
jl fin
mov [min],ebx
fin:
mov eax,msg1
call sprint
mov eax,[min]
call iprintLF
call quit
```

Рис. 19. Текст программы.

```
adsemyonov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-3.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-3 l
adsemyonov@fedora:~/work/arch-pc/lab07$ ./lab7-3
Наименьшее число: 33
adsemyonov@fedora:~/work/arch-pc/lab07$
```

Рис. 20. Создание и запуск файла.

Задание №2.

Я создал файл **lab7-4.asm** для выполнения второго задания.

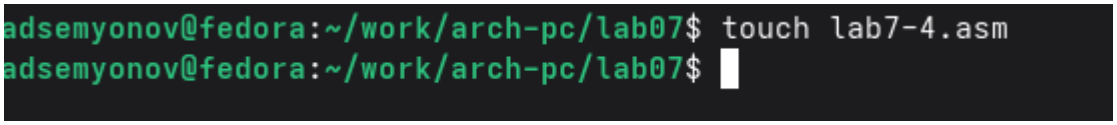

Рис. 21. Создание файла.

Я написал программу, вычисляющую значение заданной функции для введенных x и a в соответствии с 8 вариантом и запустил программу.

Открыть

lab7-4.asm
~/work/arch-pc/lab07

```
%include 'in_out.asm'
SECTION .data
msg.x: DB 'Введите x: ',0
msg.a: DB 'Введите a: ',0
msg.res: DB 'Результат: ',0
SECTION .bss
x: RESB 10
a: RESB 10
SECTION .text
GLOBAL _start
_start:
mov eax, msg.x
call sprint
mov ecx, x
mov ebx, 10
call sread
mov eax, x
call atoi
mov [x], eax
mov eax, msg.a
call sprint
mov ecx, a
mov ebx, 10
call sread
mov eax, a
call atoi
mov [a], eax
mov ebx, [a]
cmp ebx, 3
jl case1
mov eax, [x]
add eax, 1
jmp result
case1:
mov eax, [a]
mov ebx, 3
mul ebx
result:
mov ebx, eax
mov eax, msg.res
call sprint
mov eax, ebx
call iprintLF
call quit
```

Рис. 22. Текст программы.

```
adsemyonov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-4.asm
adsemyonov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-4 lab7-4.o
adsemyonov@fedora:~/work/arch-pc/lab07$ ./lab7-4
Введите x: 1
Введите a: 4
Результат: 2
adsemyonov@fedora:~/work/arch-pc/lab07$ ./lab7-4
Введите x: 1
Введите a: 2
Результат: 6
adsemyonov@fedora:~/work/arch-pc/lab07$
```

Рис. 23. Выполнение программы.

5. Выводы.

Я изучил команды условного и безусловного переходов. Я приобрел навыки написания программ с использованием переходов и познакомился с назначением и структурой файла листинга.

Список литературы.

1. https://esystem.rudn.ru/course/view.php?id=112