

Отчёт по лабораторной работе

Простейший вариант

Тимур Ринатович Каримов

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	16
	Список литературы	17

Список иллюстраций

4.1	Создание файла lab7-1.asm	8
4.2	Текст в файле lab7-1.asm	9
4.3	Запуск программы lab7-1	9
4.4	Изменение текста программы	10
4.5	Проверка работы программы	10
4.6	Изменение текста	11
4.7	Запуск программы	11
4.8	Текст программы для сравнения чисел	12
4.9	Программа для сравнения чисел	12
4.10	Файл листинга lab7-2.lst	12
4.11	Объяснения первой строки	13
4.12	Объяснения второй строки	13
4.13	Объяснения третьей строки	13
4.14	Создание файла без одного операнда	13
4.15	Файл листинга без одного операнда	14
4.16	Текст программы для нахождения меньшего числа	14
4.17	Результат работы программы	14
4.18	Текст программы для вычисления выражения	15
4.19	Проверка работы программы	15

Список таблиц

3.1	Описание некоторых каталогов файловой системы GNU Linux . . .	7
-----	---	---

1 Цель работы

Изучение команд условного и безусловного переходов. Приобретение навыков написания программ с использованием переходов. Знакомство с назначением и структурой файла листинга.

2 Задание

1.Реализация переходов в NASM. 2.Изучение структуры файлы листинга

3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы. Например, в табл. 3.1 приведено краткое описание стандартных каталогов Unix.

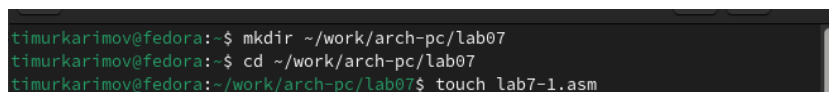
Таблица 3.1: Описание некоторых каталогов файловой системы GNU Linux	
Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую систему
/bin	Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям
/etc	Общесистемные конфигурационные файлы и файлы конфигурации установленных программ
/home	Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя
/media	Точки монтирования для сменных носителей
/root	Домашняя директория пользователя root
/tmp	Временные файлы
/usr	Вторичная иерархия для данных пользователя

Более подробно про Unix см. в [1–4].

4 Выполнение лабораторной работы

□1. Создание каталога и файла(рис. 4.1).

- Создан каталог lab7.
- Внутри каталога создан файл lab7-1.asm.

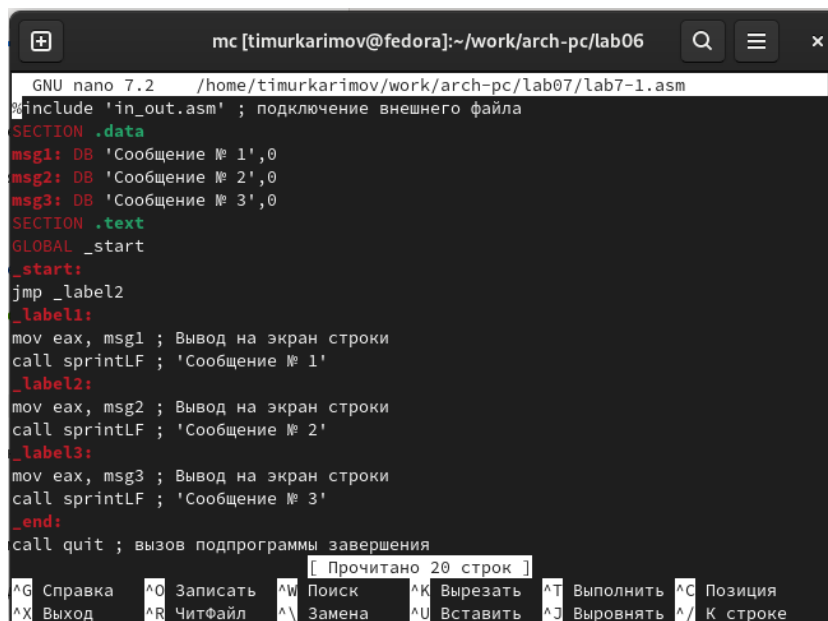


```
timurkarimov@fedora: ~$ mkdir ~/work/arch-pc/lab07
timurkarimov@fedora: ~$ cd ~/work/arch-pc/lab07
timurkarimov@fedora: ~/work/arch-pc/lab07$ touch lab7-1.asm
```

Рис. 4.1: Создание файла lab7-1.asm

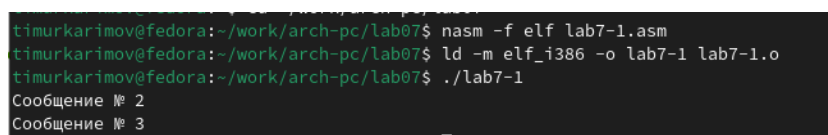
□2. Ввод текста программы и запуск

- В файл lab7-1.asm введен текст программы(рис. 4.2).
- Программа была успешно запущена, и результат соответствовал ожиданиям(рис. 4.3).



```
GNU nano 7.2 /home/timurkarimov/work/arch-pc/lab07/lab7-1.asm
%include 'in_out.asm' ; подключение внешнего файла
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 ; Вывод на экран строки
call sprintf ; 'Сообщение № 1'
_label2:
mov eax, msg2 ; Вывод на экран строки
call sprintf ; 'Сообщение № 2'
_label3:
mov eax, msg3 ; Вывод на экран строки
call sprintf ; 'Сообщение № 3'
_end:
call quit ; вызов подпрограммы завершения
[ Прочитано 20 строк ]
^G Справка ^O Записать ^W Поиск ^K Вырезать ^T Выполнить ^C Позиция
^X Выход ^R ЧитФайл ^\ Замена ^U Вставить ^J Выровнять ^/_ К строке
```

Рис. 4.2: Текст в файле lab7-1.asm



```
timurkarimov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
timurkarimov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
timurkarimov@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 2
Сообщение № 3
```

Рис. 4.3: Запуск программы lab7-1

3. Изменение текста программы

- Изменен текст программы для получения нужного результата(рис. 4.4).
- Создан исполняемый файл и проведена проверка работы программы(рис. 4.5).

```

%include 'in_out.asm' ; подключение внешнего файла
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 ; Вывод на экран строки
call sprintf ; 'Сообщение № 1'
jmp _end
_label2:
mov eax, msg2 ; Вывод на экран строки
call sprintf ; 'Сообщение № 2'
jmp _label1
_label3:
mov eax, msg3 ; Вывод на экран строки
call sprintf ; 'Сообщение № 3'

```

Почитано 22 строки

Рис. 4.4: Изменение текста программы

```

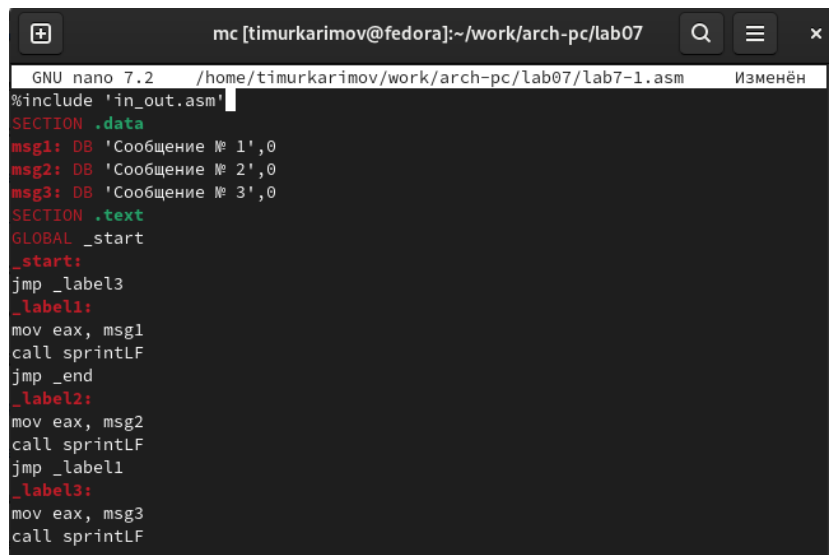
timurkarimov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
timurkarimov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
timurkarimov@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 2
Сообщение № 1
timurkarimov@fedora:~/work/arch-pc/lab07$

```

Рис. 4.5: Проверка работы программы

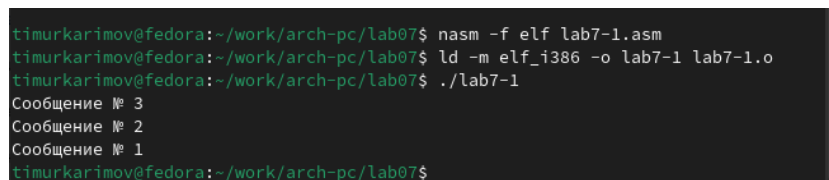
□4. Последующие изменения

- Текст программы изменен для последовательного вывода сообщений: сначала 3, затем 2, затем 1.(рис. 4.6).
- Программа была запущена и проверена на корректность работы.(рис. 4.7).



```
mc [timurkarimov@fedora]:~/work/arch-pc/lab07
GNU nano 7.2 /home/timurkarimov/work/arch-pc/lab07/lab7-1.asm
%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
```

Рис. 4.6: Изменение текста



```
timurkarimov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-1.asm
timurkarimov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-1 lab7-1.o
timurkarimov@fedora:~/work/arch-pc/lab07$ ./lab7-1
Сообщение № 3
Сообщение № 2
Сообщение № 1
timurkarimov@fedora:~/work/arch-pc/lab07$
```

Рис. 4.7: Запуск программы

5. Создание нового файла

- Создан файл lab8-2.asm и написан текст программы для сравнения чисел(рис. 4.8).

```

GNU nano 7.2 /home/timurkarimov/work/arch-pc/lab07/lab7-2.asm Изменён
#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:
; ----- Вывод сообщения 'Введите B: '
mov eax,msg1
call sprint
; ----- Ввод 'B'
mov ecx,B
mov edx,10
call sread
; ----- Преобразование 'B' из символа в число
Имя файла для записи: /home/timurkarimov/work/arch-pc/lab07/lab7-2.asm
^G Справка      M-D Формат DOS  M-A Доп. в начало  M-B Резерв. копия
^C Отмена       M-M Формат Mac  M-P Доп. в конец  ^T Обзор

```

Рис. 4.8: Текст программы для сравнения чисел

□6. Проверка работы новой программы

- Введены два различных числа для тестирования работы программы(рис. 4.9).

```

timurkarimov@fedora:~/work/arch-pc/lab07$ touch lab7-2.asm
timurkarimov@fedora:~/work/arch-pc/lab07$ mc

timurkarimov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-2.asm
timurkarimov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-2 lab7-2.o
timurkarimov@fedora:~/work/arch-pc/lab07$ ./lab7-2
Введите B: 47
Наибольшее число: 50
timurkarimov@fedora:~/work/arch-pc/lab07$ ./lab7-2
Введите B: 67
Наибольшее число: 67

```

Рис. 4.9: Программа для сравнения чисел

□7. Анализ файла листинга

- Создан файл листинга lab7-2.lst и открыт для анализа(рис. 4.10).

```

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

timurkarimov@fedora:~/work/arch-pc/lab07$

```

Рис. 4.10: Файл листинга lab7-2.lst

□7.1. Объяснение строк кода из листинга.

- Строка 21(рис. 4.11):.
- Адрес: 00000101

- Машинный код: B8 [0A000000]
- Описание: `mov eax,B` — значение переменной B загружается в регистр `eax`.

```
21 00000101 B8[0A000000]          mov eax,B
33 00000105 E801555555          call 55555555
```

Рис. 4.11: Объяснения первой строки

- Строка 35(рис. 4.12):.
- Адрес: 00000134
- Машинный код: E863FFFFFF
- Описание: `call atoi` — символ из предыдущей строки преобразуется в число.

```
34 00000130 B8[00000000]          mov eax,max
35 00000135 E862FFFFFF          call atoi
36 00000139 43[00000000]          mov ebx,3
```

Рис. 4.12: Объяснения второй строки

- Строка 47(рис. 4.13):.
- Адрес: 00000162
- Машинный код: A1 [00000000]
- Описание: `mov eax,[max]` — значение переменной `max` загружается в регистр `eax`.

```
46 0000015E E801555555          call atoi
47 00000163 A1[00000000]          mov eax,[max]
48 00000167 43[00000000]          mov ebx,3
```

Рис. 4.13: Объяснения третьей строки

8. Ошибка при компиляции

- Удалена переменная `max` из строки `mov eax,max`.
- При попытке создать исполняемый файл возникла ошибка из-за отсутствия второго операнда(рис. 4.14).

```
[1]+ Остановлен /usr/bin/mc -P "$MC_PWD_FILE" "$@"
timurkarimov@fedora:~/work/arch-pc/lab07$ nasm -f elf -l lab7-2.lst lab7-2.asm
lab7-2.asm:34: error: invalid combination of opcode and operands
timurkarimov@fedora:~/work/arch-pc/lab07$
```

Рис. 4.14: Создание файла без одного операнда

9. Анализ ошибки

- Файл листинга указал на место ошибки и объяснил ее причины(рис. 4.15).

```
33                                     check_b:
34                                     mov eax,
34                                     ***** error: invalid combination of opcode and
35 00000130 E867FFFFFF call atoi ; Вызов подпрограммы перевода
36 00000135 A3[00000000] mov [max],eax ; запись преобразованного
```

Рис. 4.15: Файл листинга без одного операнда

#Самостоятельная работа.

- Написана программа для нахождения меньшего из трех чисел с вводом чисел с клавиатуры (вариант 18, числа: 30, 73, 83)(рис. 4.16). Программа успешно вывела меньшее из этих чисел(рис. 4.17).

```
GNU nano 7.2 /home/timurkarimov/work/arch-pc/lab07/lab7-3.asm
#include 'in_out.asm'

section .data
msga db 'Введите A: ',0h
msgb db 'Введите B: ',0h
msgc db 'Введите C: ',0h
msgl db "Наименьшее число: ",0h
section .bss
min resb 10
A resb 10 ; резервируем место не
B resb 10 ; только под B,
C resb 10 ; но и под A и C
section .text
global _start
_start:
; ----- Ввод A, B, C
mov eax,msga
call sprint
mov ecx,A
[ Прочитано 86 строк ]
^G Справка ^O Записать ^W Поиск ^K Вырезать ^T Выполнить ^C Позиция
^X Выход ^R ЧитФайл ^\ Замена ^U Вставить ^J Выровнять ^/_ К строке
```

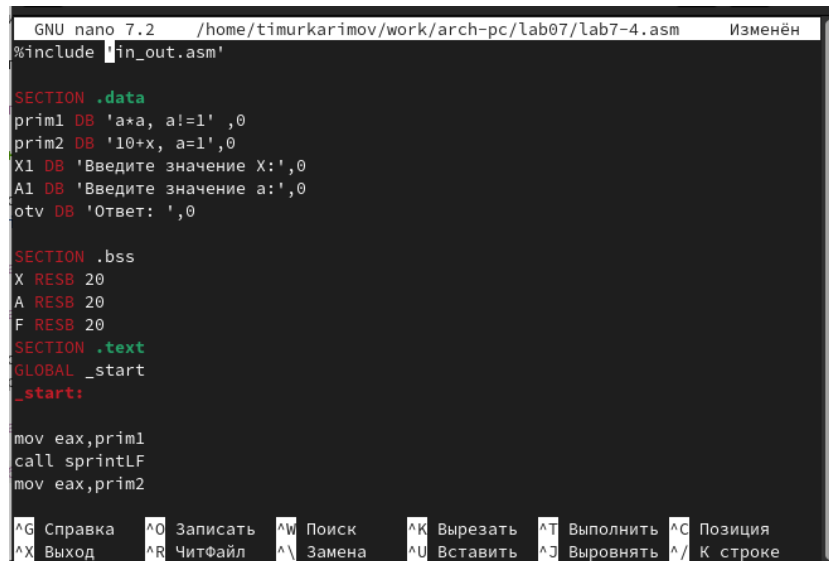
Рис. 4.16: Текст программы для нахождения меньшего числа

```
timurkarimov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-3.asm
timurkarimov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-3 lab7-3.o
timurkarimov@fedora:~/work/arch-pc/lab07$ ./lab7-3
Введите A: 30
Введите B: 83
Введите C: 73
Наименьшее число: 30
timurkarimov@fedora:~/work/arch-pc/lab07$
```

Рис. 4.17: Результат работы программы

- Написана программа для вычисления выражения при введенных значениях

X и A с выводом формулы в начале работы программы (вариант 18)(рис. 4.18)(рис. 4.19).



```
GNU nano 7.2 /home/timurkarimov/work/arch-pc/lab07/lab7-4.asm  Изменён
#include "in_out.asm"

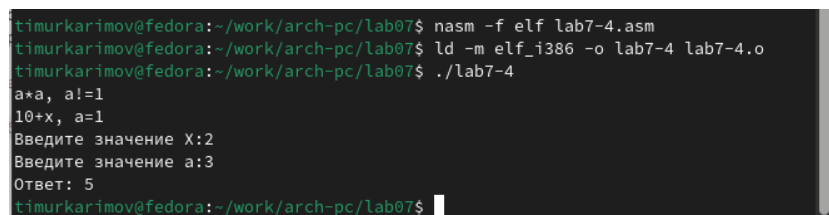
SECTION .data
prim1 DB 'a*a, a!=1',0
prim2 DB '10+x, a=1',0
X1 DB 'Введите значение X:',0
A1 DB 'Введите значение a:',0
otv DB 'Ответ: ',0

SECTION .bss
X RESB 20
A RESB 20
F RESB 20
SECTION .text
GLOBAL _start
_start:

mov eax,prim1
call sprintLF
mov eax,prim2

^G Справка  ^O Записать  ^W Поиск    ^K Вырезать  ^T Выполнить ^C Позиция
^X Выход    ^R ЧитФайл  ^\ Замена   ^U Вставить  ^J Выводить  ^/_ К строке
```

Рис. 4.18: Текст программы для вычисления выражения



```
timurkarimov@fedora:~/work/arch-pc/lab07$ nasm -f elf lab7-4.asm
timurkarimov@fedora:~/work/arch-pc/lab07$ ld -m elf_i386 -o lab7-4 lab7-4.o
timurkarimov@fedora:~/work/arch-pc/lab07$ ./lab7-4
a*a, a!=1
10+x, a=1
Введите значение X:2
Введите значение a:3
Ответ: 5
timurkarimov@fedora:~/work/arch-pc/lab07$
```

Рис. 4.19: Проверка работы программы

5 Выводы

В ходе выполнения работы были изучены команды условного и безусловного перехода, а также приобретены навыки написания программ с использованием этих команд. Работа с ассемблером позволила глубже понять основы низкоуровневого программирования и взаимодействия с аппаратным обеспечением компьютера.

Список литературы

1. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.
2. Robbins A. Bash Pocket Reference. O'Reilly Media, 2016. 156 с.
3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.
4. Newham C. Learning the bash Shell: Unix Shell Programming. O'Reilly Media, 2005. 354 с.