

Шаблон отчёта по лабораторной работе

10

Талебу Тенке Франк Устон НКАбд-05-23

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы :	6
2.1	Выводы по результатам выполнения заданий :	9
3	Задание для самостоятельной работы :	10
3.1	Выводы по результатам выполнения заданий :	12
4	Выводы	13
	Список литературы	14

Список иллюстраций

2.1	Ресунок	6
2.2	Ресунок	7
2.3	Ресунок	8
2.4	Ресунок	8
2.5	Ресунок	9
3.1	Ресунок	12

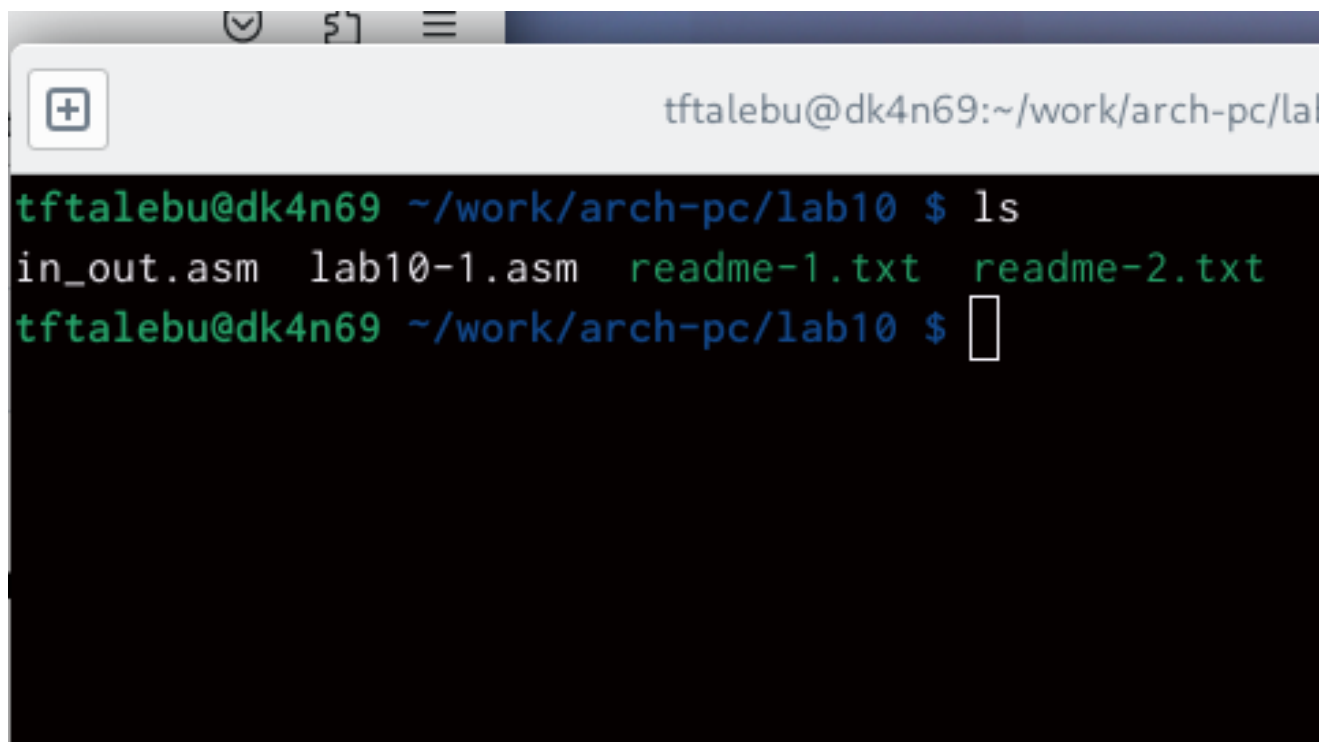
Список таблиц

1 Цель работы

- В этой лабораторной работе мы научимся писать программы с использованием подпрограмм и познакомимся со способами отладки с использованием GDB и его основными функциями

2 Выполнение лабораторной работы :

- Здесь мы начали с создания каталога для программы лабораторной работы No10, а затем переместились в десятый каталог лаборатории “~/work/arch-pc/lab10”, после чего мы создали файл “lab10-1.asm” и “readme-1.txt” и “readme-2.txt”.(рис. [2.1])



The image shows a terminal window with a dark background. The title bar at the top indicates the user is 'tftalebu' on host 'dk4n69' in the directory '~/work/arch-pc/lab10'. The terminal shows the command 'ls' being executed, resulting in the output: 'in_out.asm lab10-1.asm readme-1.txt readme-2.txt'. The prompt is ready for the next command.

```
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ls
in_out.asm lab10-1.asm readme-1.txt readme-2.txt
tftalebu@dk4n69 ~/work/arch-pc/lab10 $
```

Рис. 2.1: Рисунок

- Затем мы заполнили код нашей программы в файле lab9-1.asm.(рис. [2.2])

Открыть ▾ + lab10-1.asm ~/\work/arch-pc/lab10 Сохранить ≡ ▾

```
1 ;-----
2 ; Запись в файл строки введенной на запрос
3 ;-----
4 %include 'in_out.asm'
5 SECTION .data
6 filename db 'readme-1.txt', 0h ; Имя файла
7 msg db 'Введите строку для записи в файл: ', 0h ; Сообщение
8 SECTION .bss
9 contents resb 255 ; переменная для вводимой строки
10 SECTION .text
11 global _start
12 _start:
13 ; --- Печать сообщения `msg`
14 mov eax,msg
15 call sprint
16 ; ---- Запись введенной с клавиатуры строки в `contents`
17 mov ecx, contents
18 mov edx, 255
19 call sread
20 ; --- Открытие существующего файла (`sys_open`)
21 mov ecx, 2 ; открываем для записи (2)
22 mov ebx, filename
23 mov eax, 5
24 int 80h
25 ; --- Запись дескриптора файла в `esi`
26 mov esi, eax
27 ; --- Расчет длины введенной строки
28 mov eax, contents ; в `eax` запишется количество
29 call slen ; введенных байтов
30 ; --- Записываем в файл `contents` (`sys_write`)
31 mov edx, eax
32 mov ecx, contents
33 mov ebx, esi
34 mov eax, 4
35 int 80h
36 ; --- Закрываем файл (`sys_close`)
37 mov ebx, esi
38 mov eax, 6
39 int 80h
40 call quit
```

Рис. 2.2: Ресунок

- После этого мы скомпилировали файл, создали исполняемый файл и проверили его работу.(рис. [2.3])

```
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ cat readme-1.txt
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ nasm -f elf -g -l lab10-1.lst lab10-1.asm
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ld -m elf_i386 -o lab10-1 lab10-1.o
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ./lab10-1
Введите строку для записи в файл: Hello world!
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ cat readme-1.txt
Hello world!
tftalebu@dk4n69 ~/work/arch-pc/lab10 $
```

Рис. 2.3: Ресунок

- Используя команду chmod, мы изменили права доступа к исполняемому файлу lab10-1, запретив его выполнение.(рис. [2.4])

```
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ chmod 000 lab10-1
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ls
in_out.asm  lab10-1.asm  lab10-1.o    readme-2.txt
lab10-1     lab10-1.lst  readme-1.txt
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ls -l
итого 34
-rw-r--r-- 1 tftalebu studsci 3942 ноя  6 12:40 in_out.asm
----- 1 tftalebu studsci 9764 дек 11 12:36 lab10-1
-rw-r--r-- 1 tftalebu studsci 1142 дек 11 12:36 lab10-1.asm
-rw-r--r-- 1 tftalebu studsci 13448 дек 11 12:36 lab10-1.lst
-rw-r--r-- 1 tftalebu studsci 2544 дек 11 12:36 lab10-1.o
-rw-r--r-- 1 tftalebu studsci  13 дек 11 12:36 readme-1.txt
-rw-r--r-- 1 tftalebu studsci   0 дек 11 12:21 readme-2.txt
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ./lab11-1
bash: ./lab11-1: Нет такого файла или каталога
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ./lab10-1
bash: ./lab10-1: Отказано в доступе
tftalebu@dk4n69 ~/work/arch-pc/lab10 $
```

Рис. 2.4: Ресунок

- После этого мы попытались запустить исполняемый файл, но без каких-либо результатов, и это потому, что у нас нет доступа для чтения этого файла.
- На этом шаге с помощью команды `chmod` мы изменили права доступа к файлу `lab 10-1.asm` с исходным кодом программы, добавив права на выполнение. После этого мы попытались выполнить файл.
- Файл не был выполнен, потому что это файл с форматом `.asm`, который содержит код, но не скомпилированный.
- На этом шаге мы предоставили доступ к файлу `readme-1.txt` в соответствии с имеющимся у нас вариантом (рис. [2.5])

```
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ chmod 125 readme-1.txt
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ls -l
итого 34
-rw-r--r-- 1 tftalebu studsci 3942 ноя 6 12:40 in_out.asm
----- 1 tftalebu studsci 9764 дек 11 12:52 lab10-1
-rw-r--r-- 1 tftalebu studsci 1289 дек 11 12:52 lab10-1.asm
-rw-r--r-- 1 tftalebu studsci 13715 дек 11 12:52 lab10-1.lst
-rw-r--r-- 1 tftalebu studsci 2544 дек 11 12:52 lab10-1.o
---x-w-r-x 1 tftalebu studsci 13 дек 11 12:53 readme-1.txt
-rw-r--r-- 1 tftalebu studsci 0 дек 11 12:21 readme-2.txt
tftalebu@dk4n69 ~/work/arch-pc/lab10 $
```

Рис. 2.5: Ресунок

2.1 Выводы по результатам выполнения заданий :

- В этой части работы мы узнали, как работать с отладчиком GDB, и получили более близкое представление о том, как работают подпрограммы.

3 Задание для самостоятельной работы :

- Здесь мы написали программу, которая работает по следующему алгоритму (рис. [??])

Открыть test.asm Сохранить

~/work/arch-pc/lab10

```
1 %include 'in_out.asm'
2
3 section .data
4     nameRequest: db "Как вас зовут? - ", 0
5     filename: db "testname.txt", 0
6     iam: db "Меня зовут "
7     iamLength: equ $-iam
8
9 section .bss
10    name: resb 255
11
12 section .text
13     global _start
14
15 _start:
16     mov eax, nameRequest
17     call sprint
18
19     mov ecx, name
20     mov edx, 255
21     call sread
22
23     mov ecx, 0777o
24     mov ebx, filename
25     mov eax, 8
26     int 80h
27
28     call _openfile
29
30     mov edx, iamLength
31     mov ecx, iam
32     mov ebx, eax
33     mov eax, 4
34     int 80h
35
36     call _closefile
37
38     call _openfile
39
40     mov edx, 2
41     mov ecx, 0
42     mov ebx, eax
```

- Вывод приглашения “Как Вас зовут?” - ввести с клавиатуры свои фамилию и имя - создать файл с именем name.txt - записать в файл сообщение “Меня зовут”
- дописать в файл строку введенную с клавиатуры - закрыть файл
- Наконец, мы создали исполняемый файл и проверяем его работу. Проверьте наличие файла и его содержимого с помощью команд ls и cat.(рис. [3.1])

```
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ touch test.asm
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ touch testname.txt
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ nasm -f elf -g -l test.lst test.asm
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ld -m elf_i386 -o test test.o
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ./test
Как вас зовут? - Taleubou Tenkeu Frank
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ ls
in_out.asm  lab10-1.asm  lab10-1.o    readme-2.txt  test.asm  testname.txt
lab10-1     lab10-1.lst  readme-1.txt  test          test.lst  test.o
tftalebu@dk4n69 ~/work/arch-pc/lab10 $ cat testname.txt
Меня зовут Taleubou Tenkeu Frank
tftalebu@dk4n69 ~/work/arch-pc/lab10 $
```

Рис. 3.1: Ресунок

3.1 Выводы по результатам выполнения заданий :

- В этой части мы использовали полученные навыки для создания программ, работающих с файлами.

4 Выводы

- В этой лабораторной работе мы узнаем, как писать программы, которые работают с файлами (создание, запись, модификация и удаление) в Nasm

Список литературы