

Отчет по лабораторной работе №5

Архитектура вычислительных систем

Белослюдов Иван Евгеньевич

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Выполнение самостоятельной работы	12
4	Выводы	15

Список иллюстраций

2.1	Переходим в каталог	5
2.2	Проверка	7
2.3	Проверка	8
2.4	Загрузка	9
2.5	Исправляем текст	10
2.6	Исправляем текст	11
3.1	Создаем	12
3.2	Проверка	13
3.3	Проверка	13
3.4	Проверка	13
3.5	Проверка	14

1 Цель работы

Приобретение практических навыков работы в Midnight Commander. Освоение инструкций языка ассемблера `mov` и `int`. # Задание

1. Создайте копию файла `lab6-1.asm`. Внесите изменения в программу (без использования внешнего файла `in_out.asm`), так чтобы она работала по следующему алгоритму: • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введенную строку на экран.
2. Получите исполняемый файл и проверьте его работу. На приглашение ввести строку введите свою фамилию.
3. Создайте копию файла `lab6-2.asm`. Исправьте текст программы с использование подпрограмм из внешнего файла `in_out.asm`, так чтобы она работала по следующему алгоритму: • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введенную строку на экран.

2 Выполнение лабораторной работы

1)Откроем Midnight Commander. Пользуясь клавишами перейдем в каталог ~/work/arch- pc. С помощью клавиши F7 создаем папку lab05 и перейдем в созданный каталог.

Левая панель	Файл	Команда	Настройки	Прав.
< ...dy/2022-2023/Архитектура компьютера/arch-pc -.[^]>				
.и	Имя	Размер	Время правки	
/..		-ВВЕРХ-	окт	7 18:00
/.git		2048	ноя	23 11:25
/config		2048	окт	7 18:01
/lab05		2048	дек	7 12:16
/labs		2048	окт	7 18:02
/template		2048	окт	7 18:01
.gitattributes		1765	окт	7 18:01
.gitignore		4637	окт	7 18:01
.gitmodules		278	окт	7 18:01
CHANGELOG.md		2126	окт	7 18:01
COURSE		8	окт	7 18:02
LICENSE		18657	окт	7 18:01
Makefile		815	окт	7 18:01
README.en.md		152	окт	7 18:01
README.git-flow.md		5653	окт	7 18:01
README.md		4477	окт	7 18:01
prepare		0	окт	7 18:02

Рис. 2.1: Переходим в каталог

2)Пользуясь строкой ввода и командой touch создаем файл lab5.asm


```

lab5.asm          [-M--] 20 L:[ 1+35 36/ 36] *(2434/2434b) <EOF>
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
;----- Объявление переменных -----
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
;----- Текст программы -----
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
;----- Системный вызов 'write' -----
; После вызова инструкции 'int 80h' на экран будет
; выведено сообщение из переменной 'msg' длиной 'msgLen'
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра
;----- системный вызов 'read' -----
; После вызова инструкции 'int 80h' программа будет ожидать ввода
; строки, которая будет записана в переменную 'buf1' размером 80
байт
mov eax, 3 ; Системный вызов для чтения (sys_read)
mov ebx, 0 ; Дескриптор файла 0 - стандартный ввод
mov ecx, buf1 ; Адрес буфера под вводимую строку
mov edx, 80 ; Длина вводимой строки
int 80h ; Вызов ядра
;----- Системный вызов 'exit' -----
; После вызова инструкции 'int 80h' программа завершит работу
mov eax,1 ; Системный вызов для выхода (sys_exit)
mov ebx,0 ; Выход с кодом возврата 0 (без ошибок)
int 80h ; Вызов ядра

```

Рис. 2.2: Проверка

- 4) Оттранслируем текст программы /lab5.asm в объективный файл. Выполним компоновку объектного файла и запустим получившийся исполняемый файл.

Левая панель	Файл	Команда	Настройки	Правая панель
<-	...	2-2023/Архитектура компьютера/arch-pc/lab05	-.[^]>	<- ~
.и	Имя	Размер	Время правки	'и
/..		-ВВЕРХ-	дек 7 12:16	/..
/image		2048	дек 7 12:30	/.texlive2022
*lab5		8744	дек 7 12:36	/.ssh
lab5.asm		2432	дек 7 12:34	/.pki
lab5.o		752	дек 7 12:37	/.mozilla
				/.local
				/.kde4
				/.gnupg
				/.config
				/.cache
				/Шаблоны
				/Рабочий стол
				/Общедоступные
				/Музыка
				/Изображения
				/Загрузки
				/Документы
				/Видео
				/work
				~public_html
				/public
				/21.09
				.xsession-erro
				.xsession-erro
				.profile
				.gtkrc-2.0
				.gitconfig
				.dmrc
				.bashrc
				.bash_profile
				.bash_history
				.Xauthority
				.ICEauthority
				Лабораторная%2

Рис. 2.3: Проверка

- Загрузим файл in_out.asm с тусиса. В одной из панелей mc откроем каталог с файлом lab5-1.asm. Скопируем файл in_out.asm в каталог с файлом lab5-1.asm. С помощью клавиши f5 создаем копию файла lab5-1.asm с именем lab5-2.asm

Левая панель				Правая панель			
Файл				Настройки			
Команда				[.*]>			
Имя	Размер	Время	Правки	Имя	Размер	Время	Правки
..	-BBERX-	дек 7 12:16		..	-BBERX-	ноя 15 12:23	
/image	2048	дек 7 12:30		/.cache	2048	окт 12 11:05	
lab5.o	2048	дек 7 12:34		/.config	4096	дек 9 15:14	
lab5-2.asm	1222	дек 7 13:08		/.gnupg	2048	дек 9 15:04	
lab5.asm	2432	дек 7 13:34		/.kde4	2048	окт 28 15:18	
lab5.o	752	дек 7 12:37		/.local	2048	сен 14 10:32	
				/.mozilla	2048	сен 14 10:46	
				/.pki	2048	сен 14 10:47	
				/.ssh	2048	окт 7 18:01	
				/.texlive2022	2048	окт 26 13:59	
				/21.09	2048	окт 5 11:50	
				/public	2048	сен 2 21:16	
				~public_html	18	сен 2 21:16	
				/work	2048	окт 7 17:58	
				/Видео	2048	сен 14 10:32	
				/Документы	2048	сен 14 10:32	
				/Загрузки	2048	дек 7 12:46	
				/Изображения	2048	сен 28 11:46	
				/Музыка	2048	сен 14 10:32	
				/Общедоступные	2048	сен 14 10:32	
				/Рабочий стол	2048	окт 28 15:18	
				/Шаблоны	2048	сен 14 10:32	
				.ICEauthority	0	сен 28 11:26	
				.Xauthority	1428	дек 9 15:04	
				.bash_history	8501	дек 7 13:30	
				.bash_profile	245	сен 3 2012	
				.bashrc	124	апр 23 2012	
				.dmrc	32	окт 28 18:33	
				.gitconfig	146	окт 7 17:56	
				.gtkrc-2.0	264	окт 28 15:27	
				.profile	536	сен 12 2016	
				.xsession-errors	0	дек 9 15:04	
				.xsession-errors.old	0	дек 8 18:51	
				lab5.asm	2432	дек 7 13:34	
				Лабораторная%20раб-та%20№1.doc_0.odt	1042813	окт 5 13:24	

Рис. 2.4: Загрузка

- 6) Исправим текст программы в файле lab6-2.asm с использованием под-программ из внешнего файла in_out.asm.

```

/afs/.dk.sci.pfu.edu.ru/home/i/e/iebeloslyudov/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05/lab5-2.asm
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
#include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в 'EAX'
call sprintf ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в 'ECX'
mov edx, 80 ; запись длины вводимого сообщения в 'EDX'
call read ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения

```

Рис. 2.5: Исправляем текст

- 7) В lab5-2.asm заменил подпрограмму sprintf на printf, создал файл и выявил разницу - она заключается в выводе текста.

```

/afs/.dk.sci.pfu.edu.ru/home/i/e/iebeloslyudov/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05/lab5-2.asm
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
%include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в 'EAX'
call sprint ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в 'EAX'
mov edx, 80 ; запись длины вводимого сообщения в 'EBX'
call sread ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения

```

Рис. 2.6: Исправляем текст

3 Выполнение самостоятельной работы

- 1) Создаем копию файла lab5.asm. Внесем изменения в программу. Получил исполняемый файл и проверил его на работу. На приглашение ввести строку ввел свою фамилию.

Левая панель				Правая панель			
Файл				Файл			
Команда				Команда			
Настройки				Настройки			
...				...			
Имя	Размер	Время	Правки	Имя	Размер	Время	Правки
..	-ВВЕРХ-	дек	7 12:16	..	-ВВЕРХ-	дек	7 12:16
/image	2048	дек	7 12:30	/image	2048	дек	7 12:30
in_out.asm	3942	дек	7 12:45	in_out.asm	3942	дек	7 12:45
lab5	8744	дек	7 12:42	lab5	8744	дек	7 12:42
lab5-2.asm	1222	дек	8 10:42	lab5-2.asm	1222	дек	8 10:42
lab5.asm	2432	дек	7 12:45	lab5.asm	2432	дек	7 12:45
lab5.o	752	дек	7 12:37	lab5.o	752	дек	7 12:37
lab5copy.asm	1222	дек	8 10:42	lab5copy.asm	1222	дек	8 10:42

Рис. 3.1: Создаем

```
iebeloslyudov@dk5n51: ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ nasm -f elf lab5copy.asm
iebeloslyudov@dk5n51: ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ld -m elf_i386 -o lab5copy lab5copy.o
iebeloslyudov@dk5n51: ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ldd -m elf_i386 -o lab5copy lab5copy.o
bash: ldd: команда не найдена
iebeloslyudov@dk5n51: ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ld -m elf_i386 -o lab5copy lab5copy.o
iebeloslyudov@dk5n51: ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ./lab5copy
Введите строку: Белослудов
iebeloslyudov@dk5n51: ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $
```

Рис. 3.2: Проверка

- 2) Создаем копию файла lab5-2.asm Исправил текст программы с использованием подпрограмм из внешнего файла in_out.asm

lab05: mc									
Левая панель		Файл		Команда		Настройки		Правая панель	
...2-2023/Архитектура компьютера/arch-pc/lab05									
..	Имя	Размер	Время	правки	Имя	Размер	Время	правки	Имя
../		-BBERX-	дек	7 12:16	../	-BBERX-	дек	7 12:16	../
/image		2048	дек	7 12:30	/image	2048	дек	7 12:30	/image
in_out.asm		2048	дек	7 12:16	in_out.asm	2048	дек	7 12:16	in_out.asm
lab05		8740	дек	7 12:16	lab05	8740	дек	7 12:16	lab05
lab5-2.asm		1228	дек	9 16:42	lab5-2.asm	1228	дек	9 16:42	lab5-2.asm
lab5-2copy.asm		1224	дек	9 17:12	lab5-2copy.asm	1224	дек	9 17:12	lab5-2copy.asm
lab5.asm		2132	дек	7 12:37	lab5.asm	2132	дек	7 12:37	lab5.asm
lab5.o		752	дек	7 12:37	lab5.o	752	дек	7 12:37	lab5.o
lab5copy		9080	дек	9 17:00	lab5copy	9080	дек	9 17:00	lab5copy
lab5copy.asm		1228	дек	9 17:00	lab5copy.asm	1228	дек	9 17:00	lab5copy.asm
lab5copy.o		1312	дек	9 17:04	lab5copy.o	1312	дек	9 17:04	lab5copy.o

Рис. 3.3: Проверка

- 3) Создаем исполняемый файл и проверил работу.

```
lab05: mc
/afs/.dk.sci.pfu.edu.ru/home/i/e/iebeloslyudov/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05/lab5-2copy
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
#include 'in_out.asm'; подключение внешнего файла
SECTION .data; Секция инициализированных данных
msg: DB 'Введите строку: ',0h; сообщение
SECTION .bss; Секция не инициализированных данных
buf1: RESB 80; Буфер размером 80 байт
SECTION .text; Код программы
GLOBAL _start; Начало программы
_start:; Точка входа в программу
mov eax, msg; запись адреса выводимого сообщения в 'EAX'
call sprintf; вызов подпрограммы печати сообщения
mov ecx, buf1; запись адреса переменной в 'EAX'
mov edx, 80; запись длины вводимого сообщения в 'EDX'
call sread; вызов подпрограммы ввода сообщения
call quit; вызов подпрограммы завершения
```

Рис. 3.4: Проверка

```
iebeloslyudov@dk5n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ nasm -f elf lab5-2copy.asm
iebeloslyudov@dk5n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ld -m elf_i386 -o lab5-2copy lab5-2copy.o
iebeloslyudov@dk5n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ./lab5-2copy
Введите строку:
Белослюдов
iebeloslyudov@dk5n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $
```

Рис. 3.5: Проверка

4 Выводы

В ходе выполнения лабораторной работы №5 я приобрел практические навыки работы в Midnight Commander. И освоил инструкции языка ассемблера `mov` и `int`.