

# ЛО4\_Логинова

Архитектура компьютеров

Логинова Дарья

## Содержание

Цель работы .....	1
Задание .....	1
Теоретическое введение .....	1
Выполнение лабораторной работы .....	2
Выводы.....	5
Список литературы.....	5

## Цель работы

Освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM.

## Задание

1. В каталоге `~/work/arch-pc/lab04` с помощью команды `cp` создайте копию файла `hello.asm` с именем `lab4.asm`
2. С помощью любого текстового редактора внесите изменения в текст программы в файле `lab4.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с вашими фамилией и именем.
3. Оттранслируйте полученный текст программы `lab4.asm` в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл.
4. Скопируйте файлы `hello.asm` и `lab4.asm` в Ваш локальный репозиторий в каталог `~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/`. Загрузите файлы на Github.

## Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. [-@tbl:std-dir] приведено краткое описание стандартных каталогов Unix.

## Описание некоторых каталогов файловой системы GNU Linux {#tbl:std-dir}

Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую
/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 [[@tanenbaum\\_book\\_modern-os\\_ru](#); [@robbins\\_book\\_bash\\_en](#); [@zarrelli\\_book\\_mastering-bash\\_en](#); [@newham\\_book\\_learning-bash\\_en](#)].

## Выполнение лабораторной работы

Создадим каталог для работы с программами на языке ассемблера NASM и перейдем в созданный каталог. [-@fig:001].

```
daloginova@dk8n54 ~ $ mkdir -p ~/work/arch-pc/lab04
daloginova@dk8n54 ~ $ cd ~/work/arch-pc/lab04
```

Рис.1

Создадим текстовый файл с именем hello.asm и откроем этот файл с помощью текстового редактора gedit.[-@fig:002]

```
daloginova@dk8n54 ~/work/arch-pc/lab04 $ touch hello.asm
daloginova@dk8n54 ~/work/arch-pc/lab04 $ gedit hello.asm
```

Рис.2

Введем в текстовый редактор следующий текст.[-@fig:003]

```

1 ; hello.asm
2 SECTION .data ; Начало секции данных
3 hello: DB 'Hello world!',10 ; 'Hello world!' плюс
4 ; символ перевода строки
5 helloLen: EQU $-hello ; Длина строки hello
6 SECTION .text ; Начало секции кода
7 GLOBAL _start
8 _start: ; Точка входа в программу
9 mov eax,4 ; Системный вызов для записи (sys_write)
10 mov ebx,1 ; Описатель файла '1' - стандартный вывод
11 mov ecx,hello ; Адрес строки hello в ecx
12 mov edx,helloLen ; Размер строки hello
13 int 80h ; Вызов ядра
14 mov eax,1 ; Системный вызов для выхода (sys_exit)
15 mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
16 int 80h ; Вызов ядра

```

Рис.3

Скомпилируем текст и с помощью команды ls проверим, что объектный файл был создан.[-@fig:004]

```

daloginova@dk8n54 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
daloginova@dk8n54 ~/work/arch-pc/lab04 $ ls
hello.asm hello.o

```

Рис.4

Скомпилируем исходный файл hello.asm в obj.o и с помощью команды ls проверим, что файлы были созданы. [-@fig:005]

```

daloginova@dk8n54 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst hello.asm
daloginova@dk8n54 ~/work/arch-pc/lab04 $ ls
hello.asm hello.o list.lst obj.o

```

Рис.5

Для получения исполняемой программы, передадим на обработку компоновщику объектный файл и с помощью команды ls проверим, что исполняемый файл hello был создан.[-@fig:006]

```

daloginova@dk8n54 ~/work/arch-pc/lab04 $ ld -m elf_i386 hello.o -o hello
daloginova@dk8n54 ~/work/arch-pc/lab04 $ ls
hello hello.asm hello.o list.lst obj.o

```

Рис.6

Выполним следующую команду и увидим формат командной строки LD набрав ld – help.[-@fig:007]

```

daloginova@dk8n54 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj.o -o main
6 daloginova@dk8n54 ~/work/arch-pc/lab04 $ ld --help
15 Использование ld [параметры] файл...
Параметры:
  -a КЛЮЧЕВОЕ СЛОВО                                Управление общей библиотекой для совместимости с HP/UX
  -A АРХИТЕКТУРА, --architecture АРХИТЕКТУРА      Задать архитектуру
  -b ЦЕЛЬ, --format ЦЕЛЬ

```

Рис.7

Запустим на выполнение созданный исполняемый файл, находящийся в текущем каталоге.[-@fig:008]

```

daloginova@dk8n54 ~/work/arch-pc/lab04 $ ./hello
Hello world!
daloginova@dk8n54 ~/work/arch-pc/lab04 $

```

Рис.8

В каталоге ~/work/arch-pc/lab04 с помощью команды cp создадим копию файла hello.asm с именем lab4.asm и с помощью текстового редактора внесем изменения в текст программы в файле lab4.asm так, чтобы на экран выводилась строка с фамилией и именем. [-@fig:009]

```

daloginova@dk8n54 ~/work/arch-pc/lab04 $ cp hello.asm lab4.asm
daloginova@dk8n54 ~/work/arch-pc/lab04 $ gedit lab4.asm
daloginova@dk8n54 ~/work/arch-pc/lab04 $ nasm -f elf lab4.asm
daloginova@dk8n54 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst lab4.asm
daloginova@dk8n54 ~/work/arch-pc/lab04 $ ld -m elf_i386 lab4.o -o lab4
daloginova@dk8n54 ~/work/arch-pc/lab04 $ ./lab4
Логина Дарья

```

Рис.9

Оттранспируем полученный текст программы lab4.asm в объектный файл. Выполним компоновку объектного файла и запустим получившийся исполняемый файл. Скопируем файлы hello.asm и lab4.asm в локальный репозиторий в каталог ~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/и Загрузим файлы на Github.[-@fig:010]

```

daloginova@dk8n54 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04 $ ls
hello.asm lab4.asm presentation report
daloginova@dk8n54 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04 $ git add .
daloginova@dk8n54 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04 $ git commit -am 'feat(main): added files lab4
'
[master fd32087] feat(main): added files lab4
2 files changed, 32 insertions(+)
create mode 100644 labs/lab04/hello.asm
create mode 100644 labs/lab04/lab4.asm
daloginova@dk8n54 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04 $ git push
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 966 байтов | 966.00 КиБ/с, готово.
Total 6 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To github.com:loginovva/study_2023-2024_arhpc-.git
4bdcc32..fd32087 master -> master
daloginova@dk8n54 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04 $

```

Рис.10

## Выводы

В ходе лабораторной работы я освоила процедуры компиляции и сборки программ, написанных на ассемблере NASM.

## Список литературы