

Отчёт по лабораторной работе №4

Архитектура вычислительных систем

Савурская Полина Александровна

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
3.1	Самостоятельная работа.	8
4	Выводы	11

Список иллюстраций

3.1	Команда 'mkdir'	6
3.2	Команды 'touch и gedit'	6
3.3	Команда 'nasm'	7
3.4	Команды 'nasm -o и ls'	7
3.5	Команды 'ld и ls'	7
3.6	Команда './hello'	8
3.7	Команда 'cp'	8
3.8	Команда 'gedit'	8
3.9	Команда 'компоновка и запуск'	9
3.10	Команда 'копирование элементов'	9
3.11	Команда 'загрузка файлов на Github'	10

1 Цель работы

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

2 Задание

1. В каталоге `~/work/arch-рс/lab05` с помощью команды `ср` создайте копию файла `hello.asm` с именем `lab5.asm`
2. С помощью любого текстового редактора внесите изменения в текст программы в файле `lab5.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с вашими фамилией и именем.
3. Оттранслируйте полученный текст программы `lab5.asm` в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл.
4. Скопируйте файлы `hello.asm` и `lab5.asm` в Ваш локальный репозиторий в каталог `~/work/study/2022-2023/“Архитектура компьютера”/arch-рс/labs/lab05/`. Загрузите файлы на Github.

3 Выполнение лабораторной работы

Создадим каталог для работы с программами на языке ассемблера NASM с помощью команды `mkdir ~/work/arch-pc/lab04` (рис. 3.1)

```
asavurskaya@dk8n74 ~$ mkdir -p ~/work/arch-pc/lab04
asavurskaya@dk8n74 ~$ cd ~/work/arch-pc/lab04
```

Рис. 3.1: Команда 'mkdir'

В созданном каталоге создаем текстовый файл с именем `hello.asm` с помощью команды `touch hello.asm`. Откроем этот файл с помощью текстового редактора, например, `gedit` (команда `gedit hello.asm`). После введем текст, данный в шаблоне выполнения Л/Р. (рис. 3.2)

```
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ touch hello.asm
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ gedit hello.asm
```

^Z

Рис. 3.2: Команды 'touch и gedit'

NASM превращает текст программы в объектный код. Для компиляции приведённого выше текста программы «Hello World» необходимо написать команду `nasm -f elf hello.asm` (рис. 3.3)

```
[1]+ Остановлен gedit hello.asm
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ls
hello.asm hello.o
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -
```

Рис. 3.3: Команда 'nasm'

Выполним следующую команду: `nasm -o obj.o -f elf -g -l list.lst hello.asm`. Данная команда скомпилирует исходный файл `hello.asm` в `obj.o`. С помощью команды `ls` проверим, что файлы были созданы. (рис. 3.4)

```
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst hello.asm
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ls
hello.asm hello.o list.lst obj.o
```

Рис. 3.4: Команды 'nasm -o и ls'

Объектный файл необходимо передать на обработку компоновщику командой `ld -m elf_i386 hello.o -o hello`. С помощью команды `ls` проверим, что исполняемый файл `hello` был создан. Выполним следующую команду: `ld -m elf_i386 obj.o -o main` и проверим ее выполнение с помощью команды `ls` (рис. 3.5)

```
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ld -m elf_i386 hello.o -o hello
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ls
hello hello.asm hello.o list.lst obj.o
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj.o -o main
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ls
hello hello.asm hello.o list.lst main obj.o
```

Рис. 3.5: Команды 'ld и ls'

Запустим на выполнение созданный исполняемый файл, находящийся в текущем каталоге набрав в командной строке: `./hello` (рис. 3.6)

```

pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ./hello
Hello, world!
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $

```

Рис. 3.6: Команда './hello'

3.1 Самостоятельная работа.

В каталоге ~/work/arch-pc/lab04 с помощью команды `cp` создайте копию файла `hello.asm` с именем `lab4.asm` (рис. 3.7)

```

-----
hello.asm  hello.o
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ cp hello.asm lab4.asm
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ls
hello  hello.asm  hello.o  lab4.asm  list.lst  main  obj.o
-----

```

Рис. 3.7: Команда 'cp'

С помощью любого текстового редактора (я использую `gedit`) внесем изменения в текст программы в файле `lab4.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с вашими фамилией и именем. (рис. 3.8)

```

pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ gedit lab4.asm
Failed to register: GDBus.Error:org.freedesktop.DBus.Error.No
[1]+  Убито                  gedit hello.asm
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ gedit lab4.asm

```

Рис. 3.8: Команда 'gedit'

Оттранслируем полученный текст программы `lab4.asm` в объектный файл. Выполним компоновку объектного файла и запустим получившийся исполняемый файл с помощью команд, которые мы использовали ранее. (рис. 3.9)


```

pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst lab4.asm
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ld -m elf_i386 lab4.o -o lab4
ld: невозможно найти lab4.o: Нет такого файла или каталога
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ls
hello  hello.asm  hello.o  lab4.asm  list.lst  main  obj.o
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ls
hello  hello.asm  hello.o  lab4.asm  list.lst  main  obj.o
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ nasm -f elf l
lab4.asm  list.lst
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ nasm -f elf lab4.asm
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ls
hello  hello.asm  hello.o  lab4.asm  lab4.o  list.lst  main  obj.o
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ld -m elf_i386 lab4.o -o lab4
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ls
hello  hello.asm  hello.o  lab4  lab4.asm  lab4.o  list.lst  main  obj.o
pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ ./lab4
Savurskaya Polina

```

Рис. 3.9: Команда ‘компоновка и запуск’

Скопируем файлы hello.asm и lab4.asm в локальный репозиторий в каталог ~/work/study/2022-2023/“Архитектура компьютера”/arch-pc/labs/lab04/. (рис. 3.10)

```

pasavurskaya@dk8n74 ~/work/arch-pc/lab04 $ cd ~/work/arch-pc/
pasavurskaya@dk8n74 ~/work/arch-pc $ pwd
/afs/.dk.sci.pfu.edu.ru/home/p/a/pasavurskaya/work/arch-pc
pasavurskaya@dk8n74 ~/work/arch-pc $ ls
lab04  lab05
pasavurskaya@dk8n74 ~/work/arch-pc $ ls ~
c++      hello.o  mkdir   public_html  work    Документы  Изображения  Общедоступн
hello.asm lab05    public  temp         Видео   Загрузки   Музыка        'Рабочий сто
pasavurskaya@dk8n74 ~/work/arch-pc $ mc

```

Рис. 3.10: Команда ‘копирование элементов’

Загрузим файлы на Github. (рис. 3.11)

```
pasavurskaya@dk8n74 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git add .
pasavurskaya@dk8n74 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git commit -am 'lab04'
[master 19f7a74] lab04
 9 files changed, 49 insertions(+)
 create mode 100755 labs/lab04/hello
 create mode 100644 labs/lab04/hello.asm
 create mode 100644 labs/lab04/hello.o
 create mode 100755 labs/lab04/lab4
 create mode 100644 labs/lab04/lab4.asm
 create mode 100644 labs/lab04/lab4.o
 create mode 100644 labs/lab04/list.lst
 create mode 100755 labs/lab04/main
 create mode 100644 labs/lab04/obj.o
pasavurskaya@dk8n74 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git push
Перечисление объектов: 16, готово.
```

Рис. 3.11: Команда ‘загрузка файлов на Github’

4 Выводы

Я освоила процедуры компиляции и сборки программ, написанных на ассемблере NASM.