

Шаблон отчёта по лабораторной работе 4

язык ассемблера NASM

Ниязов С.М.

Содержание

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

1. Цель работы
2. Задание
3. Теоретическое введение
4. Выполнение лабораторной работы
5. Вывод

Цель работы

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

Задание

1. Программа Hello world!
 - 1.1 создать каталог для работы с программами на языке ассемблера NASM
 - 1.2 перейти в созданный каталог
 - 1.3 создать текстовый файл с именем hello.asm
 - 1.4 открыть этот файл
 - 1.5 ввести в него указанный текст
2. Транслятор Nasm
 - 2.1 выполнить компиляцию в объектный код
3. Расширенный синтаксис
 - 3.1 выполнить компиляцию исходного файла

4. Компоновщик LD
 - 4.1 передать объектный файл на обработку компоновщику
5. Запустить исполняемый файл
6. Задания для самостоятельной работы
 - 6.1 создать копию файла hello.asm с именем lab4.asm
 - 6.2 изменить скопированный файл, чтобы выводилась строка с именем и фамилией
 - 6.3 оттранслировать полученный текст программы lab4.asm в объектный файл
 - 6.4 скопировать файлы hello.asm и lab4.asm в локальный репозиторий

Теоретическое введение

В процессе создания ассемблерной программы можно выделить четыре шага:

- Набор текста программы в текстовом редакторе и сохранение её в отдельном файле. Каждый файл имеет свой тип (или расширение), который определяет назначение файла. Файлы с исходным текстом программ на языке ассемблера имеют тип `asm`.
- Трансляция — преобразование с помощью транслятора, например `nasm`, текста программы в машинный код, называемый объектным. На данном этапе также может быть получен листинг программы, содержащий кроме текста программы различную дополнительную информацию, созданную транслятором. Тип объектного файла — `o`, файла листинга — `lst`.
- Компоновка или линковка — этап обработки объектного кода компоновщиком (`ld`), который принимает на вход объектные файлы и собирает по ним исполняемый файл. Исполняемый файл обычно не имеет расширения. Кроме того, можно получить файл карты загрузки программы в ОЗУ, имеющий расширение `map`.
- Запуск программы. Конечной целью является работоспособный исполняемый файл.

Ошибки на предыдущих этапах могут привести к некорректной работе программы, поэтому может присутствовать этап отладки программы при помощи специальной программы — отладчика. При нахождении ошибки необходимо провести коррекцию программы, начиная с первого шага.

Выполнение лабораторной работы

1. Программа Hello world!(см рис 1)
 - 1.1 создать каталог для работы с программами на языке ассемблера NASM(см рис 1)
 - 1.2 перейти в созданный каталог(см рис 1)
 - 1.3 создать текстовый файл с именем hello.asm(см рис 1)
 - 1.4 открыть этот файл(см рис 1)

1.5 ввести в него указанный текст(см рис 1)

2. Транслятор Nasm(см рис 1)

2.1 выполнить компиляцию в объектный код(см рис 1)

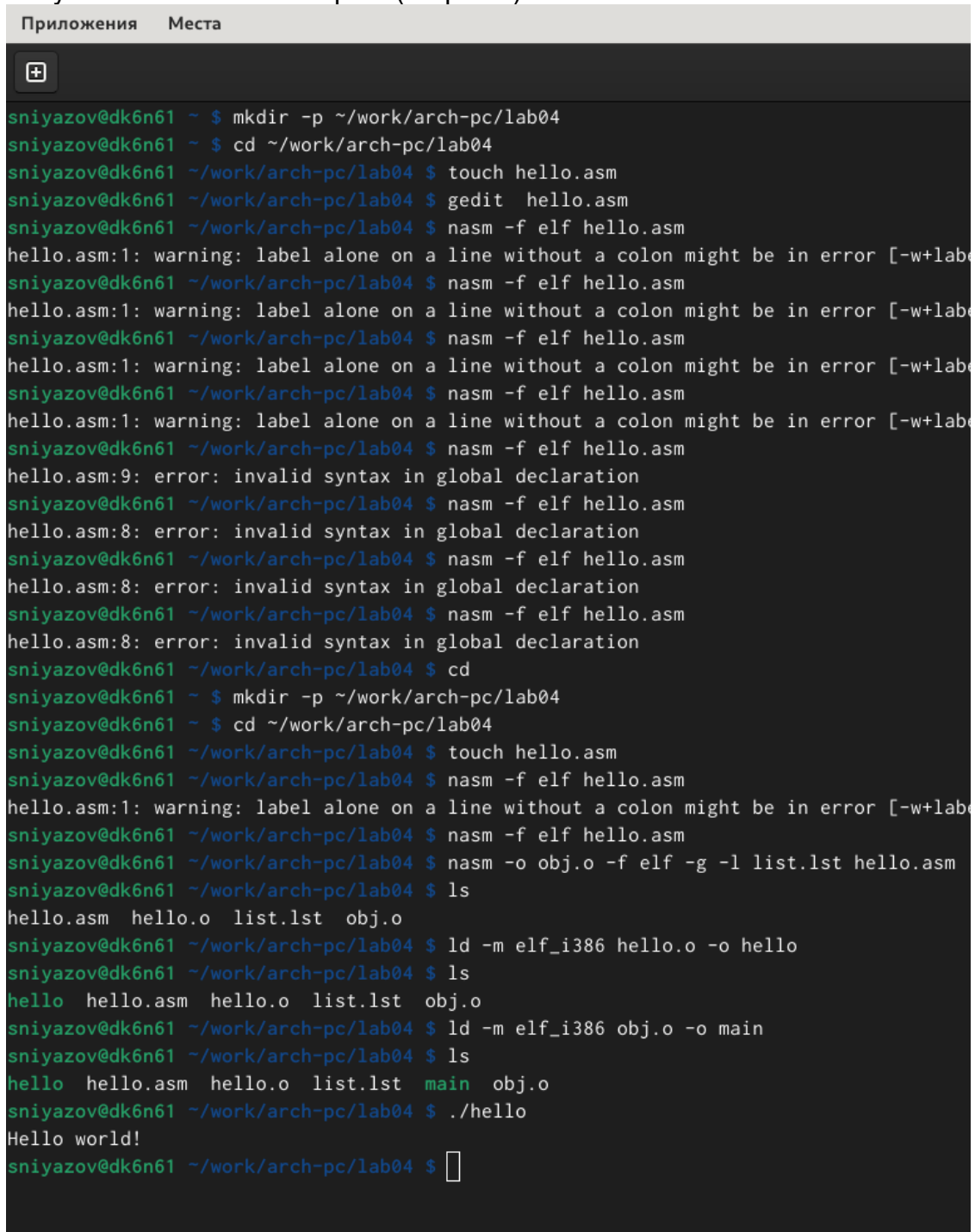
3. Расширенный синтаксис(см рис 1)

3.1 выполнить компиляцию исходного файла(см рис 1)

4. Компоновщик LD(см рис 1)

4.1 передать объектный файл на обработку компоновщику(см рис 1)

5. Запустить исполняемый файл(см рис 1)



```
Приложения Места
[+]
sniyazov@dk6n61 ~ $ mkdir -p ~/work/arch-pc/lab04
sniyazov@dk6n61 ~ $ cd ~/work/arch-pc/lab04
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ touch hello.asm
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ gedit hello.asm
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
hello.asm:1: warning: label alone on a line without a colon might be in error [-w+lab
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
hello.asm:1: warning: label alone on a line without a colon might be in error [-w+lab
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
hello.asm:1: warning: label alone on a line without a colon might be in error [-w+lab
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
hello.asm:1: warning: label alone on a line without a colon might be in error [-w+lab
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
hello.asm:9: error: invalid syntax in global declaration
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
hello.asm:8: error: invalid syntax in global declaration
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
hello.asm:8: error: invalid syntax in global declaration
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
hello.asm:8: error: invalid syntax in global declaration
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ cd
sniyazov@dk6n61 ~ $ mkdir -p ~/work/arch-pc/lab04
sniyazov@dk6n61 ~ $ cd ~/work/arch-pc/lab04
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ touch hello.asm
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
hello.asm:1: warning: label alone on a line without a colon might be in error [-w+lab
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst hello.asm
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ ls
hello.asm hello.o list.lst obj.o
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ ld -m elf_i386 hello.o -o hello
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ ls
hello hello.asm hello.o list.lst obj.o
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj.o -o main
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ ls
hello hello.asm hello.o list.lst main obj.o
sniyazov@dk6n61 ~/work/arch-pc/lab04 $ ./hello
Hello world!
sniyazov@dk6n61 ~/work/arch-pc/lab04 $
```

Выполнение лабораторной работы (рис 1)

6. Задания для самостоятельной работы(см рис 2)

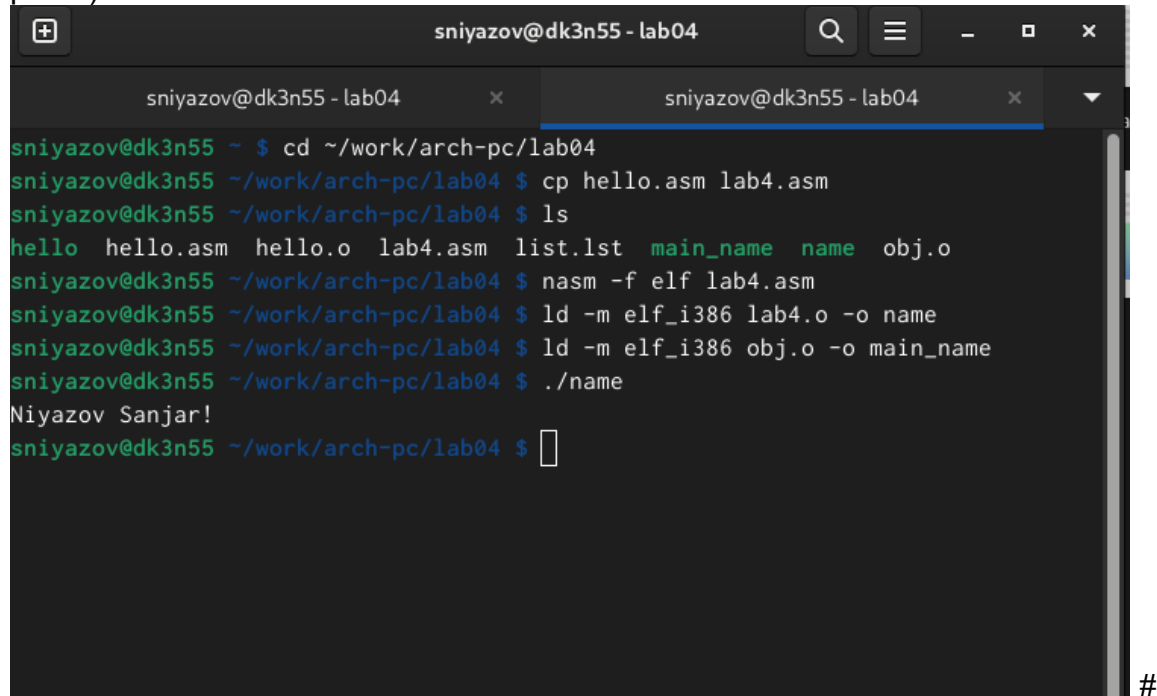
6.1 создать копию файла hello.asm с именем lab4.asm(см рис 2)

6.2 изменить скопированный файл, чтобы выводилась строка с именем и фамилией(см рис 2)

6.3 оттранслировать полученный текст программы lab4.asm в объектный

файл(см рис 2)

6.4 скопировать файлы hello.asm и lab4.asm в локальный репозиторий(см рис 2)

A terminal window titled 'sniyazov@dk3n55 - lab04' with two tabs. The active tab shows the following commands and output:

```
sniyazov@dk3n55 ~ $ cd ~/work/arch-pc/lab04
sniyazov@dk3n55 ~/work/arch-pc/lab04 $ cp hello.asm lab4.asm
sniyazov@dk3n55 ~/work/arch-pc/lab04 $ ls
hello  hello.asm  hello.o  lab4.asm  list.lst  main_name  name  obj.o
sniyazov@dk3n55 ~/work/arch-pc/lab04 $ nasm -f elf lab4.asm
sniyazov@dk3n55 ~/work/arch-pc/lab04 $ ld -m elf_i386 lab4.o -o name
sniyazov@dk3n55 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj.o -o main_name
sniyazov@dk3n55 ~/work/arch-pc/lab04 $ ./name
Niyazov Sanjar!
sniyazov@dk3n55 ~/work/arch-pc/lab04 $
```

Выводы

Я ознакомился с созданием и процессом обработки программ на языке ассемблера NASM

Список литературы