

Отчет по лабораторной работе №4

Архитектура вычислительных систем

Диана Олеговна Шаяхметова

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Самостоятельная работа	11
6	Выводы	12
	Список литературы	13

Список иллюстраций

4.1	переход в каталог	8
4.2	открытие файла в gedit	8
4.3	ввод текста	9
4.4	компиляция текста	9
4.5	компиляция текста	9
4.6	передача файла компоновщику	9
4.7	ввод команды main	10
4.8	запуск файла	10
5.1	создание копии файла	11

Список таблиц

1 Цель работы

Целью работы является 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 Теоретическое введение

4 Выполнение лабораторной работы

1. Переходим в каталог lab04 и создаем текстовый файл с именем hello.asm.

```
doshayakhmetova@dk6n51 ~ $ cd ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab04  
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ touch hello.asm
```

Рис. 4.1: переход в каталог

2. Открываем этот файл с помощью gedit.

```
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ gedit hello.asm
```

Рис. 4.2: открытие файла в gedit

3. Вводим данный текст.


```

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. 4.3: ввод текста

4. Компилируем данный текст и проверяем, что файлы созданы с помощью команды `ls`.

```

doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ nasm -f elf hello.asm
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ls
hello.asm  hello.o  presentation  report

```

Рис. 4.4: компиляция текста

5. Компилируем файл `hello.asm` в `obj.o`.

```

doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ nasm -o obj.o -f elf -g -l list.lst hello.asm

```

Рис. 4.5: компиляция текста

6. Чтобы получить исполняемую программу, передаем объектный файл на обработку компоновщику.

```

doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ld -m elf_i386 hello.o -o hello

```

Рис. 4.6: передача файла компоновщику

7. Вводим команду `main` и получаем переименованный файл.

```
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ld -m elf_i386 obj.o -o main
```

Рис. 4.7: ввод команды `main`

8. Запускаем на выполнение созданный исполняемый файл.

```
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ./hello  
Hello world!
```

Рис. 4.8: запуск файла

5 Самостоятельная работа

1. С помощью команды `cp` создаем копию файла `hello.asm` и переименовываем его в `lab04.asm`.

```
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ cp hello.asm lab04.asm
```

Рис. 5.1: создание копии файла

2. В `gedit` вносим изменения в текст программы файла, заменяя `Hello World` на свои фамилию и имя, транслируем полученный текст в объектный файл, выполняем его компоновку и запускаем получившийся файл.

```
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ gedit lab04.asm
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ nasm -f elf lab04.asm
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ nasm -o obj.o -f elf -g -l list.lst lab04.asm
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ls
hello  hello.asm  hello.o  lab04.asm  lab04.o  list.lst  main  obj.o  presentation  report
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ld -m elf_i386 lab04.o -o lab04
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ld -m elf_i386 obj.o -o lab04
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ./lab04
Shayakhmetova Diana
doshayakhmetova@dk6n51 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $
```

6 Выводы

Я освоила процедуру компиляции и сьюрки программ, написанных в ассемблере NASM.

Список литературы