

Отчет по лабораторной работе № 4

Дисциплина: Архитектура компьютера

Дроздова Дарья Игоревна

Содержание

1	Цель работы	3
2	Выполнение лабораторной работы	4
3	Выводы	8

1 Цель работы

Целью данной лабораторной работы является освоение процедуры компиляции и сборки программ, написанных на языке ассемблере NASM.

2 Выполнение лабораторной работы

1. Порядок выполнения лабораторной работы

1. Программа *Hello world!*

- Создаем каталог для работы с программами на языке ассемблера NASM и переходим в созданный каталог:

```
[didrozdova@fedora study_2022-2023_arh-pc]$ mkdir -p ~/work/arch-pc/lab05  
[didrozdova@fedora study_2022-2023_arh-pc]$ cd ~/work/arch-pc/lab05
```

- Создаем текстовый файл с именем `hello.asm` и открываем с помощью текстового редактора `gedit`:

```
[didrozdova@fedora lab05]$ touch hello.asm  
[didrozdova@fedora lab05]$ gedit hello.asm
```

- Вводим в файле следующий текст:



The screenshot shows a text editor window titled `*hello.asm` with the path `~/work/arch-pc/lab05`. The code inside is as follows:

```
1 SECTION .data  
2     hello: DB 'Hello world!',10  
3     helloLen: EQU $-hello  
4 SECTION .text  
5     GLOBAL _start  
6 _start:  
7     mov eax,4  
8     mov ebx,1  
9     mov ecx,hello  
10    mov edx,helloLen  
11    int 80h  
12    mov eax,1  
13    mov ebx,0  
14    int 80h
```

2. Транслятор *NASM*

- Для компиляции написанного в файле `hello.asm` текста программы «Hello World» необходимо написать следующий текст и, используя команду `ls`, проверить, что объектный файл создан:

```
[didrozdova@fedora lab05]$ nasm -f elf hello.asm
[didrozdova@fedora lab05]$ ls
hello.asm  hello.o
```

С помощью написанной выше команды транслятор преобразует текст программы из файла `hello.asm` в объектный код, который запишется в файл `hello.o`.

3. Расширенный синтаксис командной строки NASM

- Компилируем исходный файл `hello.asm` в `obj.o(-o)` и создаем файл листинга `list.lst(-l)` и с помощью команды `-ls` проверяем, что файлы были созданы:

```
[didrozdova@fedora lab05]$ nasm -o obj.o -f elf -g -l list.lst hello.asm
[didrozdova@fedora lab05]$ ls
hello.asm  hello.o  list.lst  obj.o
```

2. Компоновщик LD

- Чтобы получить исполняемую программу, объектный файл необходимо передать на обработку компоновщику и с помощью команды `-ls` проверить, что файл `hello` был создан:

```
[didrozdova@fedora lab05]$ ld -m elf_i386 hello.o -o hello
```

- Выполняем следующую команду:

```
[didrozdova@fedora lab05]$ ld -m elf_i386 obj.o -o main
```

Проверим результат выполненной команды:

```
hello  hello.asm  hello.o  list.lst  main  obj.o
```

Исполняемый файл по-прежнему имеет имя `hello`, объектный файл из которого собран этот исполняемый файл – `main`.

1. Запуск исполняемого файла

- Запускаем на выполнение созданный исполняемый файл, находящийся в текущем каталоге, набрав в командной строке:

```
[didrozdova@fedora lab05]$ ./hello
Hello world!
[didrozdova@fedora lab05]$ |
```

3. Задание для самостоятельной работы

- В каталоге ~/work/arch-pc/lab05 с помощью команды `cp` создаем копию файла `hello.asm` с именем `lab5.asm`:

```
[didrozdova@fedora lab05]$ cp hello.asm lab5.asm
[didrozdova@fedora lab05]$ ls
hello  hello.asm  hello.o  lab5.asm  list.lst  main  obj.o
```

- С помощью текстового редактора `gedit` меняем текст программы в файле `lab5.asm` так, чтобы вместо “Hello world!” на экран выводилась строка с собственным именем и фамилией:

```
[didrozdova@fedora lab05]$ gedit lab5.asm
```



```
1 SECTION .data
2     hello: DB 'Дроздова Дарья',10
3     helloLen: EQU $-hello
4 SECTION .text
5     GLOBAL _start
6 _start:
7     mov eax,4
8     mov ebx,1
9     mov ecx,hello
10    mov edx,helloLen
11    int 80h
12    mov eax,1
13    mov ebx,0
14    int 80h
```

- Оттранслируем полученный текст программы `lab5.asm` в объектный файл и запускаем получившийся исполняемый файл:

```
[didrozdova@fedora lab05]$ nasm -f elf lab5.asm
[didrozdova@fedora lab05]$ ls
hello  hello.asm  hello.o  lab5.o  lab5.asm  list.lst  main  obj.o

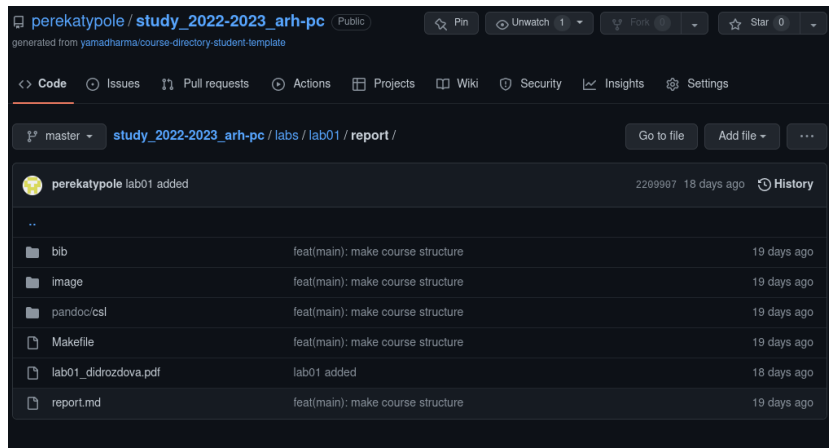
[didrozdova@fedora lab05]$ nasm -o obj_for_lab5.o -f elf -g -l list.lst lab5.asm
[didrozdova@fedora lab05]$ ls
hello  hello.o  lab5.o  main  obj.o
hello.asm  lab5.asm  list.lst  obj_for_lab5.o

[didrozdova@fedora lab05]$ ld -m elf_i386 lab5.o -o lab5
[didrozdova@fedora lab05]$ ls
hello  hello.o  lab5.asm  list.lst  obj_for_lab5.o
hello.asm  lab5  lab5.o  main  obj.o
```

```
[didrozdova@fedora lab05]$ ld -m elf_i386 obj_for_lab5.o -o main_lab5
[didrozdova@fedora lab05]$ ls
hello      hello.o  lab5.asm  list.lst  main_lab5  obj.o
hello.asm  lab5     lab5.o    main      obj_for_lab5.o
```

- Копируем файлы hello.asm и lab5.asm в собственный локальный репозиторий в каталог ~/work/study/2022-2023/“Архитектура компьютера”/study_2022-2023_arh-pc/labs/lab04/ и загружаем файлы на Github:

```
[didrozdova@fedora lab05]$ cp hello.asm lab5.asm ~/work/study/2022-2023/"Архитектура компьютера"/study_2022-2023_arh-pc/labs/lab04
```



3 Выводы

В ходе выполнения практических заданий “Лабораторной работы №4” я ознакомилась с синтаксисом ассемблера NASM, а также освоила процедуры компиляции и сборки программ.