

Отчёт по лабораторной работе 4

Дисциплина: Архитектура компьютера

Вячеслав Кочконян

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Программа Hello world!	6
2.2	Транслятор NASM	7
2.3	Компоновщик LD	8
2.4	Выполнение заданий для самостоятельной работы.	9
3	Выводы	11

Список иллюстраций

2.1	Создание каталога и файла	6
2.2	Программа hello.asm	7
2.3	Трансляция hello.asm	7
2.4	Трансляция hello.asm с дополнительными опциями	8
2.5	Линковка программы	8
2.6	Линковка программы	8
2.7	Запуск программ	9
2.8	Код программы в файле lab4.asm	9
2.9	Запуск программы lab4.asm	10

Список таблиц

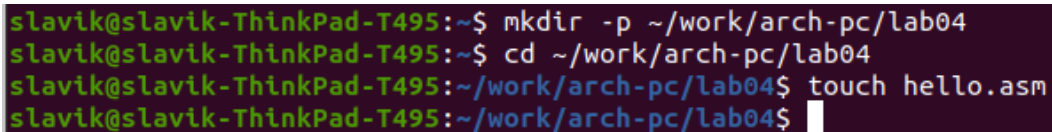
1 Цель работы

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

2 Выполнение лабораторной работы

2.1 Программа Hello world!

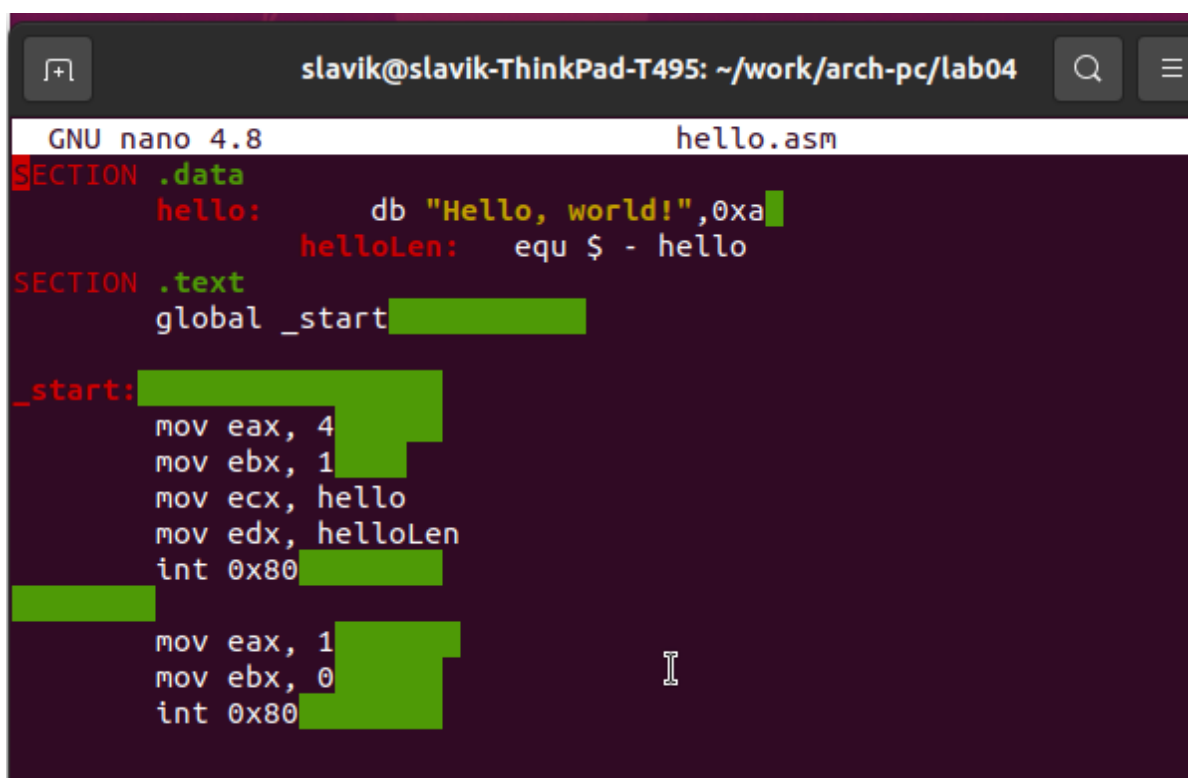
Создаю каталог lab04 командой mkdir, перехожу в него с помощью команды cd, создаю файл hello.asm. (рис. 2.1)



```
slavik@slavik-ThinkPad-T495:~$ mkdir -p ~/work/arch-pc/lab04
slavik@slavik-ThinkPad-T495:~$ cd ~/work/arch-pc/lab04
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ touch hello.asm
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
```

Рис. 2.1: Создание каталога и файла

Открыл файл и написала код программы по заданию.(рис. 2.2)



```
slavik@slavik-ThinkPad-T495: ~/work/arch-pc/lab04
GNU nano 4.8 hello.asm
SECTION .data
hello:      db "Hello, world!",0xa
helloLen:   equ $ - hello

SECTION .text
global _start

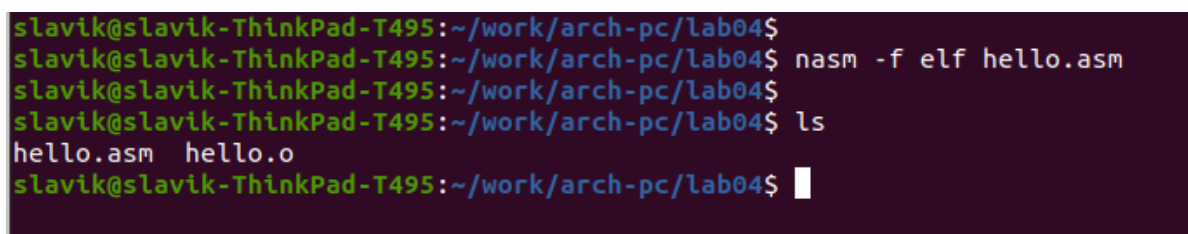
_start:
    mov eax, 4
    mov ebx, 1
    mov ecx, hello
    mov edx, helloLen
    int 0x80

    mov eax, 1
    mov ebx, 0
    int 0x80
```

Рис. 2.2: Программа hello.asm

2.2 Транслятор NASM

Транслирую файл командой `nasm`. Получился объектный файл `hello.o` (рис. 2.3)



```
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ nasm -f elf hello.asm
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ls
hello.asm hello.o
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
```

Рис. 2.3: Трансляция hello.asm

Транслирую файл командой `nasm` с дополнительными опциями. (рис. 2.4)
Получился файл листинга `list.lst`, объектный файл `obj.o`, в программу добавилась отладочная информация.

```

slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ nasm -f elf hello.asm
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ls
hello.asm hello.o
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.asm
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ls
hello.asm hello.o list.lst obj.o
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$

```

Рис. 2.4: Трансляция hello.asm с дополнительными опциями

2.3 Компоновщик LD

Выполняю линковку командой ld и получил исполняемый файл. (рис. 2.5)

```

slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst obj.o
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$

```

Рис. 2.5: Линковка программы

Еще раз выполняю линковку для объектного файла obj.o и получаю исполняемый файл main.(рис. 2.6)

```

slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst obj.o
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst main obj.o
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$

```

Рис. 2.6: Линковка программы

Запускаю исполняемые файлы.(рис. 2.7)

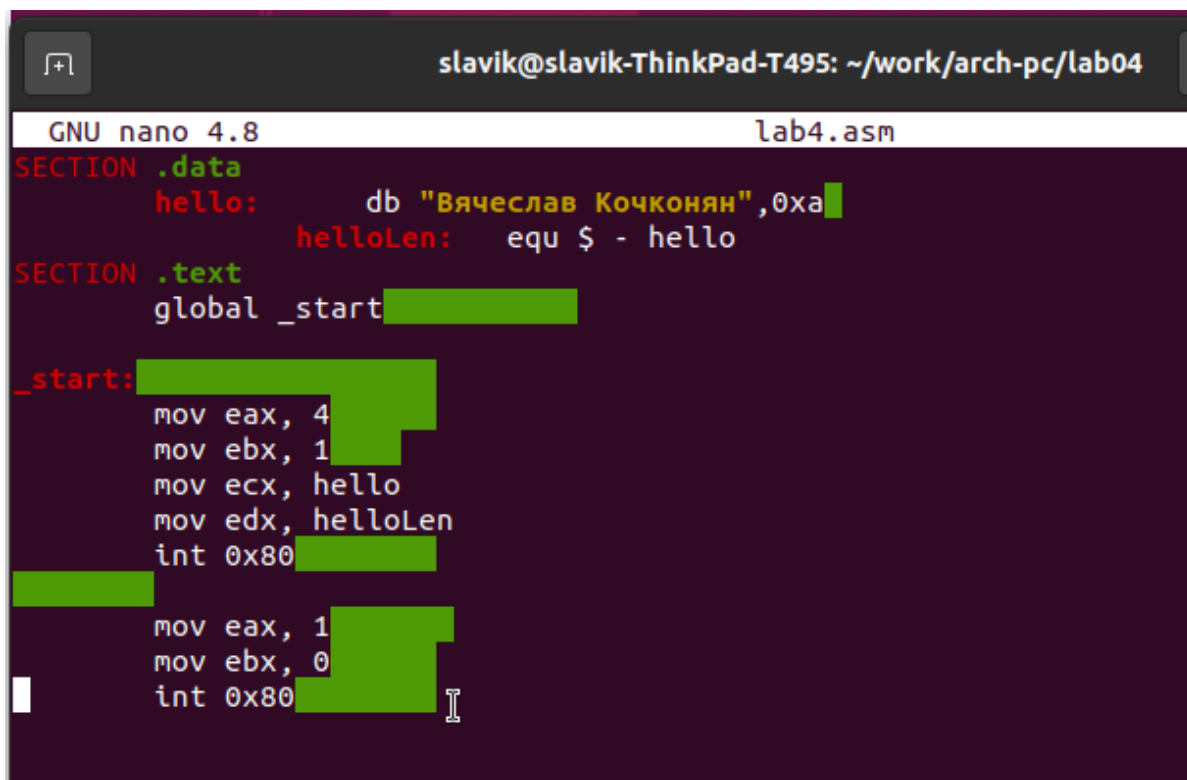

```
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$  
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ./main  
Hello, world!  
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ./hello  
Hello, world!  
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
```

Рис. 2.7: Запуск программ

2.4 Выполнение заданий для самостоятельной работы.

Копирую программу в новый файл.

Изменяю сообщение Hello world на свое имя (рис. 2.8) и запускаю новую программу. (рис. 2.9)



```
slavik@slavik-ThinkPad-T495: ~/work/arch-pc/lab04  
GNU nano 4.8 lab4.asm  
SECTION .data  
    hello:      db "Вячеслав Кочконян",0xa  
    helloLen:   equ $ - hello  
SECTION .text  
    global _start  
_start:  
    mov eax, 4  
    mov ebx, 1  
    mov ecx, hello  
    mov edx, helloLen  
    int 0x80  
  
    mov eax, 1  
    mov ebx, 0  
    int 0x80
```

Рис. 2.8: Код программы в файле lab4.asm

```
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$  
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ nasm -f elf lab4.asm  
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o lab4  
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$ ./lab4  
Вячеслав Кочкоян  
slavik@slavik-ThinkPad-T495:~/work/arch-pc/lab04$
```

Рис. 2.9: Запуск программы lab4.asm

3 Выводы

При выполнении данной лабораторной работы я освоил процесс компиляции и сборки программ, написанных на ассемблере `nasm`.