

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ
НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 4

дисциплина: Архитектура компьютера

Студент: Виме Давид Тененте

Группа: НКАбд-01-24

Содержание

Список иллюстраций

3.1 Создаем каталоги с помощью команды mkdir.....	6
3.2 Переходим в каталог с помощью команды cd.....	6
3.3 Создаем текстовый файл hello.asm.....	6
3.4 Открываем файл и заполняем его по примеру.....	7
3.5 Используем команду nasm.....	7
3.6 Проверяем работу команды.....	7
3.7 Преобразуем файл hello.asm в obj.o.....	7
3.8 Проверяем создание файла командой ls.....	8
3.9 Используем команду ld.....	8
3.10 Используем команду ls.....	8
3.11 Используем команду ld, создавая файл main.....	8
3.12 Используем команду ls.....	8
3.13 Используем команду ./hello.....	9
3.14 Используем команду cp.....	9
3.15 Открываем файл в текстовом редакторе.....	9
3.16 Редактируем файл для своего имени и фамилии.....	9
3.17 Прописываем команды для работы файла и запускаем программу	10
3.18 Копируем файлы в каталог с ЛР4.....	10
3.19 Загружаем файлы.....	10

1 Цель работы

Освоить процедуры компиляции и сборки программ, познакомиться с языком ассемблера NASM.

2 Задание

Написать 2 программы(Hello world, lab4(Имя Фамилия))

3 Выполнение лабораторной работы

3.1 Программа Hello world!

Создаем каталог для работы с программами на языке ассемблера NASM (рис. 3.1).

```
davidwime@fedora:~$ mkdir -p ~/work/arch-pc/lab04
davidwime@fedora:~$
```

Рис. 3.1: Создаем каталоги с помощью команды mkdir

Переходим в созданный каталог (рис. 3.2).

```
davidwime@fedora:~$ cd ~/work/arch-pc/lab04
davidwime@fedora:~/work/arch-pc/lab04$
```

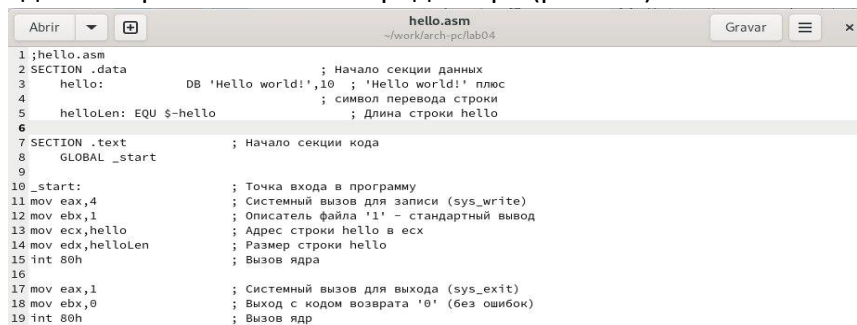
Рис. 3.2: Переходим в каталог с помощью команды cd

Создаем текстовый файл (рис. 3.3).

```
davidwime@fedora:~/work/arch-pc/lab04$ touch hello.asm
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.3: Создаем текстовый файл hello.asm

Открываем данный файл в текстовом редакторе (рис. 3.4).



The screenshot shows a text editor window titled 'hello.asm' with the following assembly code:

```
1 ;hello.asm
2 SECTION .data
3     hello:    DB 'Hello world!',10 ; 'Hello world!' плюс
4               ; символ перевода строки
5     helloLen: EQU $-hello         ; Длина строки hello
6
7 SECTION .text
8     GLOBAL _start
9
10 _start:
11     mov eax,4 ; Системный вызов для записи (sys_write)
12     mov ebx,1 ; Описатель файла '1' - стандартный вывод
13     mov ecx,hello ; Адрес строки hello в ecx
14     mov edx,helloLen ; Размер строки hello
15     int 80h ; Вызов ядра
16
17     mov eax,1 ; Системный вызов для выхода (sys_exit)
18     mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
19     int 80h ; Вызов ядра
```

Рис. 3.4: Открываем файл и заполняем его по примеру

3.2 Транслятор NASM

Преобразуем текст программы в объектный код (рис. 3.5).

```
davidwime@fedora:~/work/arch-pc/lab04$ nasm -f elf hello.asm
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.5: Используем команду nasm

Проверяем создался ли объектный файл с помощью команды ls (рис. 3.6).

```
davidwime@fedora:~/work/arch-pc/lab04$ ls
hello.asm hello.o
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.6: Проверяем работу команды

3.3 Расширенный синтаксис командной строки NASM

Компилируем исходный файл (рис. 3.7).

```
davidwime@fedora:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst hello
.asm
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.7: Преобразуем файл hello.asm в obj.o

Проверяем, как сработала команда (рис. 3.8).

```
davidwime@fedora:~/work/arch-pc/lab04$ ls
hello.asm hello.o list.lst obj.o
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.8: Проверяем создание файла командой ls

3.4 Компоновщик LD

Передаем объектный файл на обработку компоновщику (рис. 3.9).

```
davidwime@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.9: Используем команду ld

Проверяем создался ли исполняемый файл hello (рис. 3.10).

```
davidwime@fedora:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst obj.o
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.10: Используем команду ls

Передаем объектный файл на обработку компоновщику (рис. 3.11).

```
davidwime@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.11: Используем команду ld, создавая файл main

Проверяем создался ли исполняемый файл hello (рис. 3.12).

```
davidwime@fedora:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst main obj.o
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.12: Используем команду ls

3.5 Запуск исполняемого файла

Запускаем на выполнение созданный исполняемый файл (рис. 3.13).

```
davidwime@fedora:~/work/arch-pc/lab04$ ./hello
Hello world!
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.13: Используем команду ./hello

3.6 Задание для самостоятельной работы

Создаем копию файла hello.asm (рис. 3.14).

```
davidwime@fedora:~/work/arch-pc/lab04$ cp hello.asm lab4.asm
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.14: Используем команду cp

Открываем файл и редактируем его (рис. 3.15).

```
davidwime@fedora:~/work/arch-pc/lab04$ gedit lab4.asm
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.15: Открываем файл в текстовом редакторе

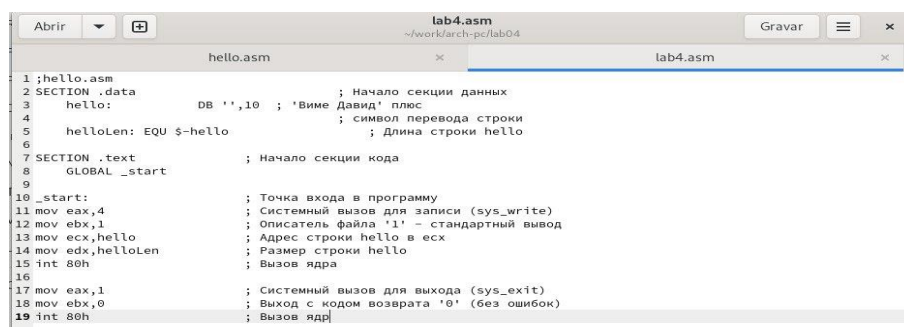


Рис. 3.16: Редактируем файл для своего имени и фамилии

Прописываем те же команды, что и с первой программой (рис. 3.17).

```
davidwime@fedora:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
davidwime@fedora:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst lab4.asm
davidwime@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o hello
davidwime@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o main
davidwime@fedora:~/work/arch-pc/lab04$ ./hello
Виме Давид
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.17: Прописываем команды для работы файла и запускаем программу

Копируем файлы в локальный репозиторий (рис. 3.18).


```
davidwime@fedora:~/work/arch-pc/lab04$ cp hello.asm ~/work/study/2024-2025/"Архитектура компьютера"/study_2024-2025_arh-pc-/labs/lab04
davidwime@fedora:~/work/arch-pc/lab04$ cp lab4.asm ~/work/study/2024-2025/"Архитектура компьютера"/study_2024-2025_arh-pc-/labs/lab04
davidwime@fedora:~/work/arch-pc/lab04$
```

Рис. 3.18: Копируем файлы в каталог с ЛР4

Переходим в каталог лабораторных работ и загружаем файлы на Github (рис. 3.19).

Рис. 3.19: Загружаем файлы

4 Выводы

Мы познакомились с языком ассемблера NASM и создали две работающих программы.