

# **Создание и процесс обработки на языке ассемблера NASM**

**Лабораторная работа №4**

Приходько Иван Иванович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
<b>3</b>	<b>Выполнение задания для самостоятельной работы</b>	<b>9</b>
<b>4</b>	<b>Выводы</b>	<b>11</b>

# Список иллюстраций

2.1	Переход в папку lab4 . . . . .	6
2.2	Создание hello.asm . . . . .	6
2.3	Открытие hello.asm . . . . .	6
2.4	Редактирование hello.asm . . . . .	7
2.5	Компиляция файла с помощью NASM . . . . .	7
2.6	Проверка созданный файлов . . . . .	7
2.7	Использование полной команды NASM . . . . .	7
2.8	Сборка исполняемого файла . . . . .	7
2.9	Сборка в main . . . . .	8
2.10	Запуск hello . . . . .	8
3.1	Копирование hello.asm . . . . .	9
3.2	Открытие файла . . . . .	9
3.3	Редактирование файла . . . . .	9
3.4	Работа с lab4.asm . . . . .	10
3.5	Отправка файлов на Github . . . . .	10

## **Список таблиц**

# 1 Цель работы

Научится писать базовую программу на языке ассемблера NASM, компилировать их в объектные файлы и собирать из них готовые к запуску программы.

## 2 Выполнение лабораторной работы

Для начала перейдем в рабочий каталог (рис. 2.1).

```
ivanprihodko@fedora:~$ cd ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab04/  
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 2.1: Переход в папку lab4

Созаддим файл hello с расширением asm, чтобы писать в нем код (рис. 2.2).

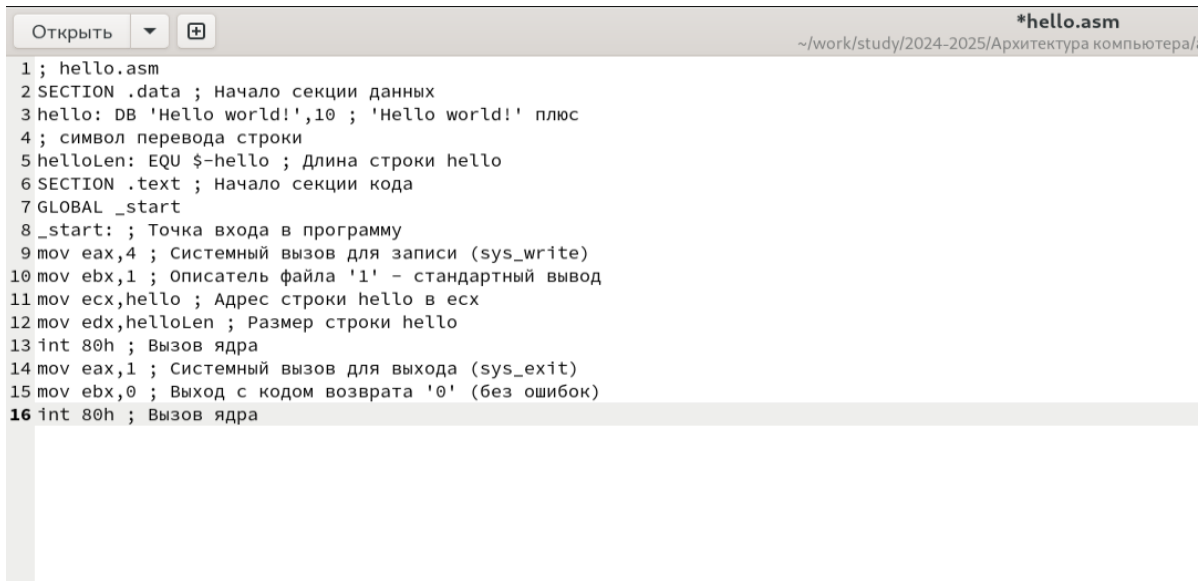
```
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ touch hello.asm  
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 2.2: Создание hello.asm

Теперь откроем данный файл и вставим в него данный код (рис. 2.3 и 2.4).

```
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ gedit hello.asm
```

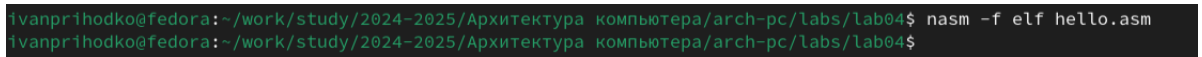
Рис. 2.3: Открытие hello.asm



```
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. 2.4: Редактирование hello.asm

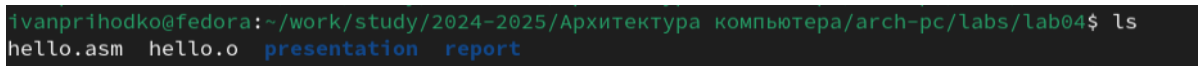
Теперь превратим наш файл в объектный (рис. 2.5).



```
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ nasm -f elf hello.asm
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 2.5: Компиляция файла с помощью NASM

Проверим (рис. 2.6).



```
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello.asm hello.o presentation report
```

Рис. 2.6: Проверка созданный файлов

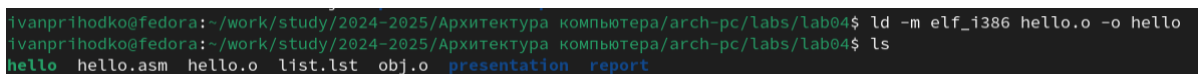
Используем полный вариант команды NASM (рис. 2.7).



```
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.asm
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello.asm hello.o list.lst obj.o presentation report
```

Рис. 2.7: Использование полной команды NASM

Используем компоновщик ld для создания исполняемого файла (рис. 2.8).



```
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ld -m elf_i386 hello.o -o hello
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello hello.asm hello.o list.lst obj.o presentation report
```

Рис. 2.8: Сборка исполняемого файла

Теперь соберем файл obj.o в main (рис. 2.9).

```
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ld -m elf_i386 obj.o -o main
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello hello.asm hello.o list.lst main obj.o presentation report
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 2.9: Сборка в main

Запустим файл (рис. 2.10).

```
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ./hello
Hello world!
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 2.10: Запуск hello



### 3 Выполнение задания для самостоятельной работы

Скопируем файл hello.asm в рабочий каталог (рис. 3.1).

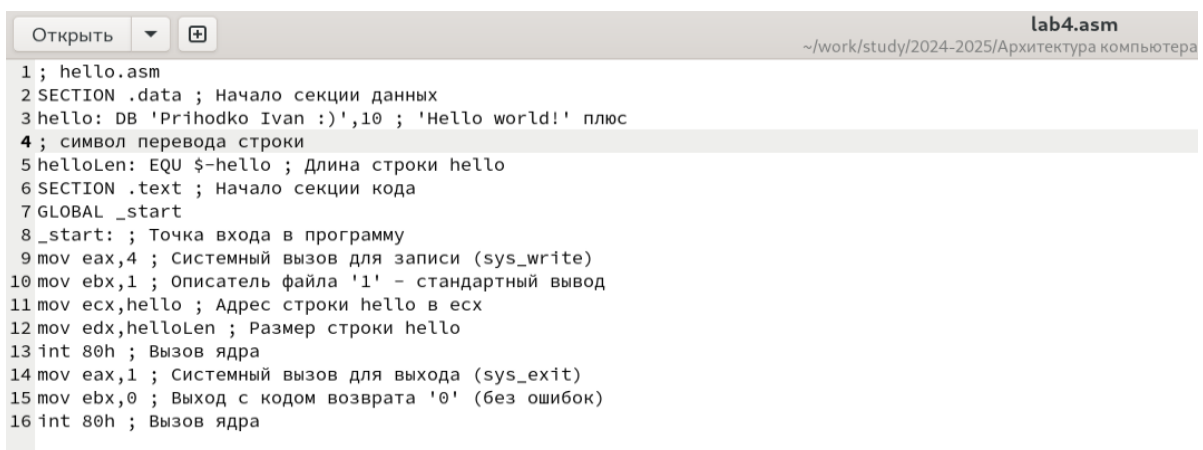
```
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ cp hello.asm lab4.asm
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 3.1: Копирование hello.asm

Внесем изменения в файл и напишем вместо Hello world! свое имя и фамилию (рис. 3.2 и 3.3).

```
ivanprihodko@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ gedit lab4.asm
```

Рис. 3.2: Открытие файла



```
lab4.asm
~/work/study/2024-2025/Архитектура компьютера

1 ; hello.asm
2 SECTION .data ; Начало секции данных
3 hello: DB 'Prihodko Ivan :)',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 есх
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. 3.3: Редактирование файла

Повторим действия также как и с hello.asm. Скомпилируем файл, соберем его в объектный и запустим (рис. 3.4).

```
ivanprihodko@fedora: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ nasm -f elf lab4.asm
ivanprihodko@fedora: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ld -m elf_i386 lab4.o -o lab4
ivanprihodko@fedora: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ./lab4
Prihodko Ivan :)
```

Рис. 3.4: Работа с lab4.asm

Не забудем отправить все на Github (рис. 3.5).

```
ivanprihodko@fedora: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ git add .
ivanprihodko@fedora: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ git commit -am 'feat(main): added files hello.asm and lab4.asm'
[master da9bcfe] feat(main): added files hello.asm and lab4.asm
9 files changed, 49 insertions(+)
create mode 100755 labs/lab04/hello
create mode 100644 labs/lab04/hello.asm
create mode 100644 labs/lab04/hello.o
create mode 100755 labs/lab04/lab4
create mode 100644 labs/lab04/lab4.asm
create mode 100644 labs/lab04/lab4.o
create mode 100644 labs/lab04/list.lst
create mode 100755 labs/lab04/main
create mode 100644 labs/lab04/obj.o
ivanprihodko@fedora: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ git push
Перечисление объектов: 16, готово.
Подсчет объектов: 100% (16/16), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (13/13), готово.
Запись объектов: 100% (13/13), 3.33 КИБ | 3.33 МИБ/с, готово.
Total 13 (delta 7), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (7/7), completed with 2 local objects.
To github.com:SunHermit67/study_2024-2025_arch-pc.git
8bd5114..da9bcfe master -> master
ivanprihodko@fedora: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 3.5: Отправка файлов на Github

## 4 Выводы

В результате проделанной работы появилось понимание того, как нужно создавать исполняемый файл на языке ассемблера.