

Отчет лабораторной работы №4

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

Баранов Никита Дмитриевич

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
3.1	Hello world	7
3.2	Транслятор NASM	8
3.3	Расширенный синтаксис командной строки NASM	8
3.4	Компоновщик LD	9
3.5	Запуск исполняемого файла	9
3.6	Задание для самостоятельной работы	9
4	Выводы	12

Список иллюстраций

3.1	Создаем каталоги и текстовый файл с помощью команды <code>mkdir</code> , <code>touch</code> , открываем файл с помощью команды <code>gedit</code>	7
3.2	Вносим текст в файл	8
3.3	Используем команду <code>nasm</code> и проверяем создался ли файл	8
3.4	Преобразуем файл <code>hello.asm</code> в <code>obj.o</code> и проверяем с помощью команды <code>ls</code>	8
3.5	Используем команду <code>ls</code> и <code>ld</code>	9
3.6	Используем команду <code>ls</code> и <code>ld</code>	9
3.7	Используем команду <code>./hello</code> и видим результат выполнения программы	9
3.8	Копируем файл, изменяя его имя	9
3.9	Открываем файл с помощью команды <code>gedit</code>	10
3.10	Редактируем файл, чтобы компилировались мои ФИО	10
3.11	Выполняем действия, которые проделывали ранее	10
3.12	Копируем файлы	11
3.13	Загружаем файлы на гитхаб	11

Список таблиц

1 Цель работы

Освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM.

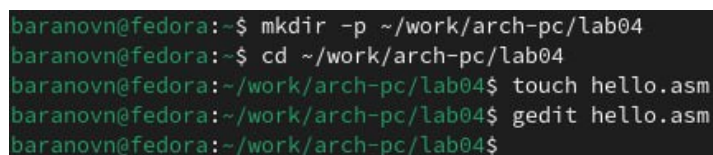
2 Задание

Написать две программы на языке ассемблера NASM - Hello world и lab4

3 Выполнение лабораторной работы

3.1 Hello world

Создаем каталог для работы с программами на языке ассемблера NASM и создаем текстовый файл, открываем этот файл с помощью текстового редактора(рис. fig. 3.1).



```
baranovn@fedora:~$ mkdir -p ~/work/arch-pc/lab04
baranovn@fedora:~$ cd ~/work/arch-pc/lab04
baranovn@fedora:~/work/arch-pc/lab04$ touch hello.asm
baranovn@fedora:~/work/arch-pc/lab04$ gedit hello.asm
baranovn@fedora:~/work/arch-pc/lab04$
```

Рис. 3.1: Создаем каталоги и текстовый файл с помощью команды mkdir, touch, открываем файл с помощью команды gedit

Редактируем файл(рис. fig. 3.2).

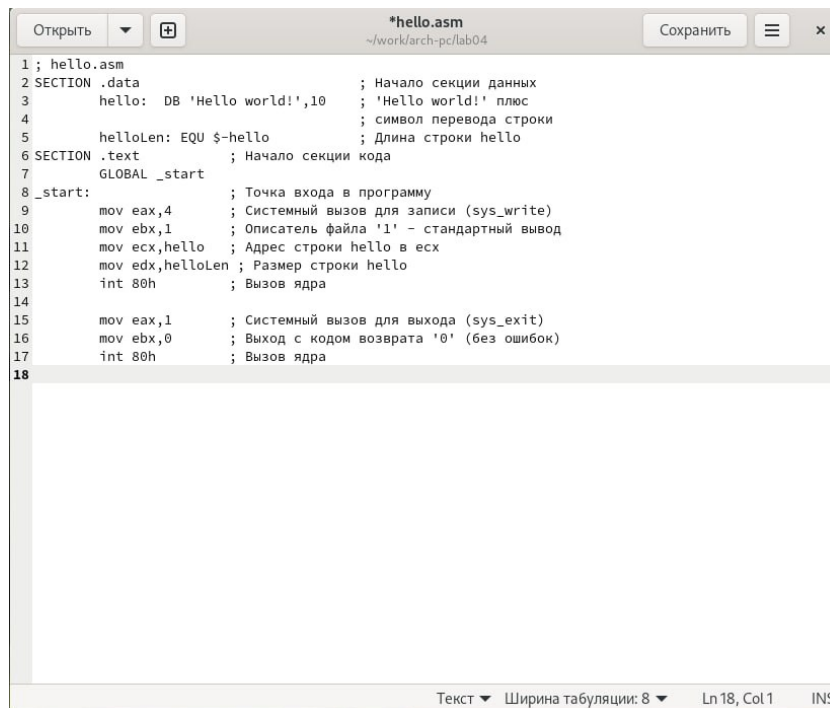


Рис. 3.2: Вносим текст в файл

3.2 Транслятор NASM

Преобразуем текст программы в объектный код(рис. fig. 3.3)

```

baranov@fedora:~/work/arch-pc/lab04$ nasm -f elf hello.asm
baranov@fedora:~/work/arch-pc/lab04$ ls
hello.asm  hello.o
baranov@fedora:~/work/arch-pc/lab04$

```

Рис. 3.3: Используем команду nasm и проверяем создался ли файл

3.3 Расширенный синтаксис командной строки NASM

Компилируем исходный файл(рис. fig. 3.4)

```

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

```

Рис. 3.4: Преобразуем файл hello.asm в obj.o и проверяем с помощью команды ls

3.4 Компоновщик LD

Передаем объектный файл на обработку компоновщику и проверяем что файл был создан(рис. fig. 3.5)

```
baranov@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello
baranov@fedora:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst obj.o
```

Рис. 3.5: Используем команду ls и ld

Передаем объектный файл на обработку компоновщику и проверяем что файл был создан(рис. fig. 3.6)

```
baranov@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main
baranov@fedora:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst main obj.o
```

Рис. 3.6: Используем команду ls и ld

3.5 Запуск исполняемого файла

Запускаем ранее созданный файл(рис. fig. 3.7)

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

Рис. 3.7: Используем команду ./hello и видим результат выполнения программы

3.6 Задание для самостоятельной работы

В каталоге ~/work/arch-pc/lab04 с помощью команды cp создайте копию файла hello.asm с именем lab4.asm(рис. fig. 3.8)

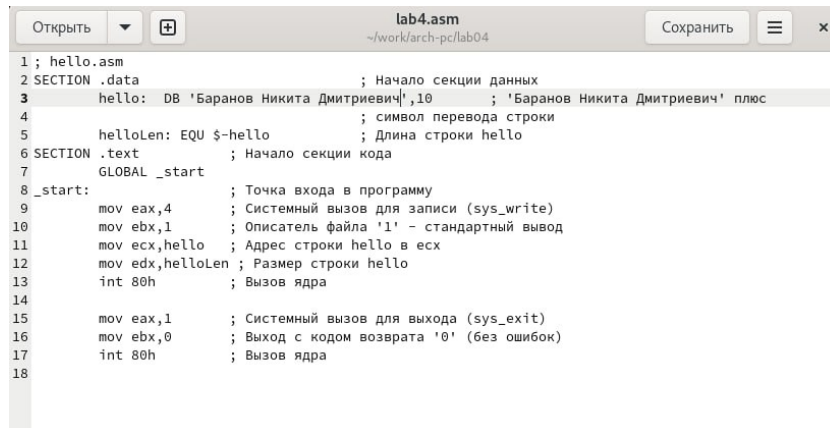
```
baranov@fedora:~/work/arch-pc/lab04$ cp hello.asm lab4.asm
baranov@fedora:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o lab4.asm list.lst main obj.o
```

Рис. 3.8: Копируем файл, изменяя его имя

С помощью любого текстового редактора внесите изменения в текст программы в файле lab4.asm так, чтобы вместо Hello world! на экран выводилась строка с вашими фамилией и именем(рис. fig. 3.9)(рис. fig. 3.10)

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

Рис. 3.9: Открываем файл с помощью команды gedit



```
1; hello.asm
2SECTION .data ; Начало секции данных
3    hello: DB 'Баранов Никита Дмитриевич',10 ; 'Баранов Никита Дмитриевич' плюс
4           ; символ перевода строки
5    helloLen: EQU $-hello ; Длина строки hello
6SECTION .text ; Начало секции кода
7    GLOBAL _start
8_start:
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
15   mov eax,1 ; Системный вызов для выхода (sys_exit)
16   mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
17   int 80h ; Вызов ядра
18
```

Рис. 3.10: Редактируем файл, чтобы компилировались мои ФИО

Оттранслируйте полученный текст программы lab4.asm в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл(рис. fig. 3.11)

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

Рис. 3.11: Выполняем действия, которые проделывали ранее

Скопируйте файлы hello.asm и lab4.asm в Ваш локальный репозиторий в каталог ~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/. Загрузите файлы на Github.(рис. fig. 3.12)(рис. fig. 3.13)

```

baranovnfedora:~/work/arch-pc/lab04$ cp hello.asm ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab04/
baranovnfedora:~/work/arch-pc/lab04$ cp lab4.asm ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab04/
baranovnfedora:~/work/arch-pc/lab04$ cd ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab04/
baranovnfedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello.asm lab4.asm presentation report

```

Рис. 3.12: Копируем файлы

```

baranovnfedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ git add .
baranovnfedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ git commit -am 'feat(main): add files lab-4'
[master 72a128a] feat(main): add files lab-4
2 files changed, 36 insertions(+)
create mode 100644 labs/lab04/hello.asm
create mode 100644 labs/lab04/lab4.asm
baranovnfedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ git push
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 1.01 КиБ | 1.01 МБ/с, готово.
Total 6 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To github.com:BaranovN/study_2024-2025_arhpc.git
8bd6c78..72a128a master -> master
baranovnfedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$

```

Рис. 3.13: Загружаем файлы на гитхаб

4 Выводы

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