

Лабораторная работа №5

Архитектура компьютеров

Кирилюк Светлана Алексеевна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	11

Список иллюстраций

2.1	Создание каталога	6
2.2	Переход в каталог	6
2.3	Создание текстового файла	6
2.4	Открытие файла	6
2.5	Ввод текста	7
2.6	Компиляция текста	7
2.7	Компиляция файла	7
2.8	Отправка на обработку	7
2.9	Ввод команды	7
2.10	Запуск файла	7
2.11	Создание копии файла	8
2.12	Переименование файла	8
2.13	Открытие файла	8
2.14	Ввод текста	8
2.15	Компиляция текста	8
2.16	Компиляция файла	9
2.17	Отправка на обработку	9
2.18	Ввод команды	9
2.19	Запуск файла	9
2.20	Копируем lab5.asm	9
2.21	Копируем hello.asm	9
2.22	Загрузка на Github	10

Список таблиц

1 Цель работы

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

2 Выполнение лабораторной работы

Сначала я создала каталог для работы с программами на языке ассемблера NASM (рис. 2.1), перешла в него (рис. 2.2).

```
sakirilyuk@dk8n68 ~ $ mkdir -p ~/work/arch-pc/lab05  
sakirilyuk@dk8n68 ~ $
```

Рис. 2.1: Создание каталога

```
sakirilyuk@dk8n68 ~ $ cd ~/work/arch-pc/lab05  
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $
```

Рис. 2.2: Переход в каталог

Создала текстовый файл (рис. 2.3) и открыла его с помощью текстового редактора (рис. 2.4) и ввела в него текст (рис. 2.5).

```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ touch hello.asm  
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $
```

Рис. 2.3: Создание текстового файла

```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ gedit hello.asm  
]
```

Рис. 2.4: Открытие файла

```

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

```

Рис. 2.5: Ввод текста

Затем я скомпилировала написанный текст с помощью команды (рис. 2.6), скомпилировала файл (рис. 2.7), отправила файл на обработку (рис. 2.8) и выполнила следующую команду (рис. 2.9).

```

sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ nasm -f elf hello.asm
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ ls
hello.asm  hello.o

```

Рис. 2.6: Компиляция текста

```

sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ nasm -o obj.o -f elf -g -l list.lst hel
lo.asm
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ ls
hello.asm  hello.o  list.lst  obj.o

```

Рис. 2.7: Компиляция файла

```

sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ ld -m elf_i386 hello.o -o hello
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ ls
hello  hello.asm  hello.o  list.lst  obj.o

```

Рис. 2.8: Отправка на обработку

```

sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ ld -m elf_i386 obj.o -o main
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $

```

Рис. 2.9: Ввод команды

И, наконец, я запустила файл с помощью команды (рис. 2.10).

```

sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ ./hello
Hello world!

```

Рис. 2.10: Запуск файла

#Выполнение задания для самостоятельной работы

В каталоге ~/work/arch-pc/lab05 с помощью команды `cp` я создала копию файла `hello.asm` (рис. 2.11) и переименовала его (рис. 2.12).

```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ cp hello.asm ~/work/arch-pc/lab05
```

Рис. 2.11: Создание копии файла

```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ mv hello.asm lab5.asm
```

Рис. 2.12: Переименование файла

Затем с помощью текстового редактора внесла изменения в текст программы в файле `lab5.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с моими фамилией и именем (рис. 2.13), (рис. 2.14), (рис. 2.15), (рис. 2.16), (рис. 2.17), (рис. 2.18), (рис. 2.19).

```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ gedit lab5.asm
```

Рис. 2.13: Открытие файла

```
1 SECTION .data
2 hello: DB 'Кирилюк Светлана',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
```

Рис. 2.14: Ввод текста

```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ nasm -f elf lab5.asm
```

Рис. 2.15: Компиляция текста


```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ nasm -o obj.o -f elf -g -l list.lst lab5.asm
```

Рис. 2.16: Компиляция файла

```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ ld -m elf_i386 lab5.o -o lab5
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $
```

Рис. 2.17: Отправка на обработку

```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ ld -m elf_i386 obj.o -o main
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $
```

Рис. 2.18: Ввод команды

```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ ./lab5
Кирилюк Светлана
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $
```

Рис. 2.19: Запуск файла

Скопировала файлы hello.asm и lab5.asm в свой локальный репозиторий в каталог ~/work/study/2022-2023/“Архитектура компьютера”/arch-pc/labs/lab05/ (рис. 2.20), (рис. 2.21).

```
sakirilyuk@dk8n68 ~ $ cp ~/work/arch-pc/lab05/lab5.asm ~/work/study/2022-2023/“Архитектура компьютера”/arch-pc/labs/lab05/
```

Рис. 2.20: Копируем lab5.asm

```
sakirilyuk@dk8n68 ~/work/arch-pc/lab05 $ cp ~/work/arch-pc/lab05/hello.asm ~/work/study/2022-2023/“Архитектура компьютера”/arch-pc/labs/lab05/
```

Рис. 2.21: Копируем hello.asm

Загрузила всё на Github (рис. 2.22).

```

sakirilyuk@dk8n68 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ gitt add /
bash: gitt: команда не найдена
sakirilyuk@dk8n68 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git add .
sakirilyuk@dk8n68 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git commit -am 'feat(main): add files lab-5'
[master c2efc89] feat(main): add files lab-5
2 files changed, 30 insertions(+)
create mode 100644 labs/lab05/hello.asm
create mode 100644 labs/lab05/lab5.asm
sakirilyuk@dk8n68 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git push
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 1.05 КиБ | 1.05 МиБ/с, готово.
Всего 6 (изменений 2), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:sakirilyuk/study_2022-2023_arh-pc.git
f9fb3a6..c2efc89 master -> master

```

Рис. 2.22: Загрузка на Github

3 Выводы

В ходе лабораторной работы я освоила процедуры компиляции и сборки программ, написанных на ассемблере NASM.