

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

дисциплина: *Архитектура компьютера*

Студент:

Группа:

МОСКВА

2022 г.

Цел работу:

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

Ход работы:

1. В каталоге ~/work/arch-pc/lab05 с помощью команды `cp` создайте копию файла `hello.asm` с именем `lab5.asm`
2. С помощью любого текстового редактора внесите изменения в текст программы в файле `lab5.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с вашими фамилией и именем.
3. Оттранслируйте полученный текст программы `lab5.asm` в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл.
4. Скопируйте файлы `hello.asm` и `lab5.asm` в Ваш локальный репозиторий в каталог ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab05/. Загрузите файлы на Github.

```
[root@localhost-live report]# cd ~/work/study/2022-2023/ComputerArchitecture/arch-pc/labs/lab05
[root@localhost-live lab05]# touch hello.asm
[root@localhost-live lab05]# mcedit hello.asm

[root@localhost-live lab05]# nasm -f elf hello.asm
[root@localhost-live lab05]# ls
hello.asm  hello.o  presentation  report
[root@localhost-live lab05]# ld -m elf_i386 hello.o hello
ld: cannot find hello: No such file or directory
[root@localhost-live lab05]# nasm -o obj.o -f elf -g -l list.lst hello.asm
[root@localhost-live lab05]# ls
hello.asm  hello.o  list.lst  obj.o  presentation  report
[root@localhost-live lab05]# ld -m elf_i386 hello.o hello
ld: cannot find hello: No such file or directory
[root@localhost-live lab05]# ld -m elf_i386 hello.o -o hello
[root@localhost-live lab05]# ls
hello  hello.asm  hello.o  list.lst  obj.o  presentation  report
[root@localhost-live lab05]# ld -m elf_i386 obj.o -o main
[root@localhost-live lab05]# ls
hello  hello.asm  hello.o  list.lst  main  obj.o  presentation  report
[root@localhost-live lab05]# ./hello
Hello world!
[root@localhost-live lab05]#
```

```
hello.asm  [-----]  7 L: [ 1+14 15/ 15] *(188 / 188b) <EOF>
SECTION .data
hello: DB 'Hello world!', 10
helloLen: EQU $-hello
SECTION .text
GLOBAL _start
_start:
mov eax,4
mov ebx,1
mov ecx,hello
mov edx,helloLen
int 80h

mov eax,1
mov ebx,0
int 80h
```

Контрольные вопросы:

1. Какие основные отличия ассемблерных программ от программ на языках высокого уровня?

Язык высокого уровня практически не привязан ни к конкретному процессору, ни к операционной системе (если не используются специфические директивы). Язык ассемблера – это низкоуровневый язык программирования, на котором вы пишете свои программы. Для каждого процессора существует свой язык ассемблера.

2. В чём состоит отличие инструкции от директивы на языке ассемблера?

Инструкция непосредственно преобразуется в то, что может выполнить процессор. Директива – это то, что ассемблер может интерпретировать и что-то говорит о том, как должны быть собраны инструкции. Директивы – это никогда не то, что увидит процессор.

3. Перечислите основные правила оформления программ на языке ассемблера.

Правила оформления ассемблерных программ. При наборе программ на языке ассемблера придерживайтесь следующих правил: * директивы набирайте большими буквами, инструкции – малыми; * пишите текст широко –

не скупердяйничайте; * не выходите за край экрана, т.е. не делайте текст шире 80 знаков - его не удобно будет редактировать и печатать; * для отступов пользуйтесь табуляцией (клавиша TAB); * блоки комментариев задавайте с одинаковым отступом.

4. Каковы этапы получения исполняемого файла?

Создание исполняемого файла издавна производилось в три этапа: (1) обработка исходного кода препроцессором, (2) компиляция в объектный код и (3) компоновка объектных модулей, включая модули из объектных библиотек, в исполняемый файл. Это классическая схема для компилируемых языков.

5. Каково назначение этапа трансляции?

Трансляция исходного текста программы состоит в преобразовании предложений исходного языка в коды машинных команд и выполняется с помощью транслятора с языка ассемблера (т. е. с помощью программы ассемблера).

7. Какие файлы могут создаваться при трансляции программы, какие из них создаются по умолчанию?

Если трансляция прошла успешно, на экран выводится сообщение «Конец сеанса» и создаются файлы `p.obj`, `p.exe` и `p`.

8. Каковы форматы файлов для `nasm` и `ld`?

Плоский бинарный формат.