Отчет по лабораторной работе №4

дисциплина: Архитектура компьютера

Михайлова Регина Алексеевна

Содержание

# 1 Цель работы

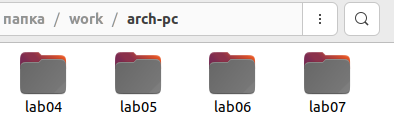
Освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM.

# 2 Выполнение лабораторной работы

1. Рассмотрим пример простой программы на языке ассемблера NASM. Традиционно первая программа выводит приветственное сообщение Hello world! на экран. Создадим каталог для работы с программами на языке ассемблера NASM (рис. ??, ??): mkdir -p ~/work/arch-pc/lab04

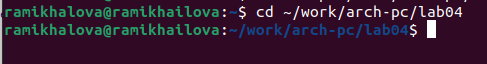
Создание каталога

Создание каталога



Созданный каталог

Перейдем в созданный каталог (рис. ??): cd ~/work/arch-pc/lab04

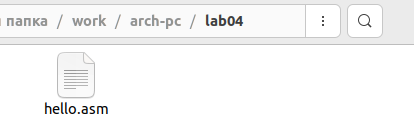


Переход в созданный каталог

Создадим текстовый файл с именем hello.asm (рис. ??, ??): touch hello.asm

Создание файла hello.asm

Создание файла hello.asm



Созданный файл hello.asm

Откроем этот файл с помощью любого текстового редактора, например, gedit (рис. ??): gedit hello.asm

Открытие файла hello.asm

Открытие файла hello.asm

и введем в него следующий текст:

; hello.asm  
SECTION .data ; Начало секции данных  
hello: DB 'Hello world!',10 ; 'Hello world!' плюс  
; символ перевода строки  
helloLen: EQU $-hello ; Длина строки hello  
SECTION .text ; Начало секции кода  
GLOBAL \_start  
\_start: ; Точка входа в программу  
mov eax,4 ; Системный вызов для записи (sys\_write)  
mov ebx,1 ; Описатель файла '1' - стандартный вывод  
mov ecx,hello ; Адрес строки hello в ecx  
mov edx,helloLen ; Размер строки hello  
int 80h ; Вызов ядра  
mov eax,1 ; Системный вызов для выхода (sys\_exit)  
mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)  
int 80h ; Вызов ядра

1. NASM превращает текст программы в объектный код. Например, для компиляции приве- дённого выше текста программы «Hello World» (рис. ??) необходимо написать: nasm -f elf hello.asm

Компиляция программы «Hello World»

Компиляция программы «Hello World»

Если текст программы набран без ошибок, то транслятор преобразует текст программы из файла hello.asm в объектный код, который запишется в файл hello.o. Таким образом, имена всех файлов получаются из имени входного файла и расширения по умолчанию. При наличии ошибок объектный файл не создаётся, а после запуска транслятора появятся сообщения об ошибках или предупреждения. С помощью команды ls проверим, что объектный файл был создан (рис. ??).

hello.o успешно создан

hello.o успешно создан

1. Полный вариант командной строки nasm выглядит следующим образом: nasm [-@ косвенный\_файл\_настроек] [-o объектный\_файл] [-f формат\_объектного\_файла] [-l листинг] [параметры…] [–] исходный\_файл

Выполним следующую команду (рис. ??): nasm -o obj.o -f elf -g -l list.lst hello.asm

Компиляция hello.asm в obj.o

Компиляция hello.asm в obj.o

Данная команда скомпилирует исходный файл hello.asm в obj.o (опция -o позволяет задать имя объектного файла, в данном случае obj.o), при этом формат выходного файла будет elf, и в него будут включены символы для отладки (опция -g), кроме того, будет создан файл листинга list.lst (опция -l). С помощью команды ls проверим, что файлы были созданы (рис. ??)

obj.o и list.lst успешно созданы

obj.o и list.lst успешно созданы

1. Чтобы получить исполняемую программу, объектный файл необходимо передать на обработку компоновщику (рис. ??): ld -m elf\_i386 hello.o -o hello С помощью команды ls проверим, что исполняемый файл hello был создан (рис. ??).

Передача объектного файла компоновщику

Передача объектного файла компоновщику

Выполним следующую команду (рис. ??): ld -m elf\_i386 obj.o -o main

Исполняемый файл будет иметь имя main (рис. ??)

Создание файла main

Создание файла main

1. Запустить на выполнение созданный исполняемый файл (рис. ??), находящийся в текущем каталоге, можно, набрав в командной строке: ./hello

Исполняемый файл успешно выполнен

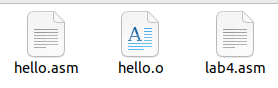
Исполняемый файл успешно выполнен

# 3 Выполнение заданий для самостоятельной работы

1. В каталоге ~/work/arch-pc/lab04 с помощью команды cp создадим копию файла hello.asm с именем lab4.asm (рис. ??), после чего копия файла будет успешно создана (рис. ??)

Копирование файла

Копирование файла



Копия файла создана

1. С помощью текстового редактора внесем изменения в текст программы в файле lab4.asm так, чтобы вместо Hello world! на экран выводилась строка с фамилией и именем:

SECTION .data   
name: DB 'Mikhaylova Regina',10  
nameLen: EQU $-name  
  
SECTION .text   
GLOBAL \_start  
  
\_start:   
mov eax,4   
mov ebx,1   
mov ecx,name   
mov edx,nameLen   
int 80h   
mov eax,1   
mov ebx,0   
int 80h

1. Оттранслируем полученный текст программы lab4.asm в объектный файл (рис. ??). Выполним компоновку объектного файла (рис. ??) и запустим получившийся исполняемый файл (рис. ??).

Преобразование в объектный файл

Преобразование в объектный файл

Компоновка объектного файла

Компоновка объектного файла

Запуск исполняемого файла

Запуск исполняемого файла

# 4 Вывод

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

# 5 Список литературы

1. GDB: The GNU Project Debugger. — URL: https://www.gnu.org/software/gdb/.
2. GNU Bash Manual. — 2016. — URL: https://www.gnu.org/software/bash/manual/.
3. Midnight Commander Development Center. — 2021. — URL: https://midnight-commander. org/.
4. NASM Assembly Language Tutorials. — 2021. — URL: https://asmtutor.com/.
5. Newham C. Learning the bash Shell: Unix Shell Programming. — O’Reilly Media, 2005. — 354 с. — (In a Nutshell). — ISBN 0596009658. — URL: http://www.amazon.com/Learning- bash-Shell-Programming-Nutshell/dp/0596009658.
6. Robbins A. Bash Pocket Reference. — O’Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
7. The NASM documentation. — 2021. — URL: https://www.nasm.us/docs.php.
8. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
9. Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
10. Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс,
11. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
12. Расширенный ассемблер: NASM. — 2021. — URL: https://www.opennet.ru/docs/RUS/nasm/.
13. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ- Петербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
14. Столяров А. Программирование на языке ассемблера NASM для ОС Unix. — 2-е изд. — М. : МАКС Пресс, 2011. — URL: http://www.stolyarov.info/books/asm\_unix.
15. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
16. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер, 2015. — 1120 с. — (Классика Computer Science).