Oтчёт по лабораторной работе №4

Архитектура вычеслительных систем

Кроитору Екатерина

Содержание

# 1 Цель работы

Освоение процедуры компиляции и сборки программ, написанных на ассем- блере NASM.

# 2 Задание

1. В соответствующем каталоге сделайте отчёт по лабораторной работе No4 в формате Markdown. В качестве отчёта необходимо предоставить отчёты в 3 форматах: pdf, docx и md.
2. Загрузите файлы на github.

# 3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. 1 приведено краткое описание стандартных каталогов Unix.

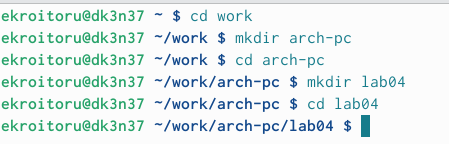
Таблица 1: Описание некоторых каталогов файловой системы GNU Linux

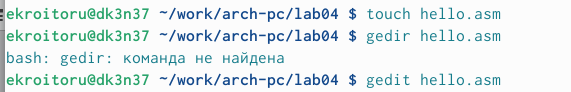
| Имя каталога | Описание каталога |
| --- | --- |
| / | Корневая директория, содержащая всю файловую |
| /bin | Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям |
| /etc | Общесистемные конфигурационные файлы и файлы конфигурации установленных программ |
| /home | Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя |
| /media | Точки монтирования для сменных носителей |
| /root | Домашняя директория пользователя root |
| /tmp | Временные файлы |
| /usr | Вторичная иерархия для данных пользователя |

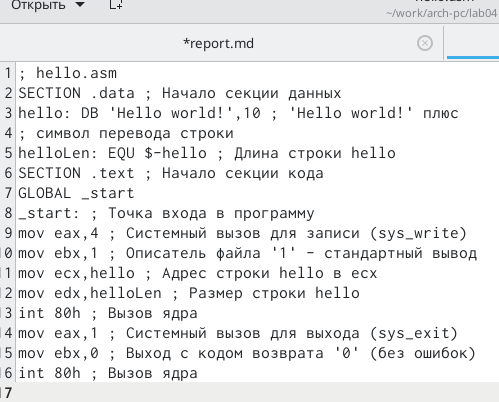
Более подробно об Unix см. в [1–6].

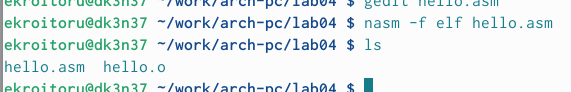
# 4 Выполнение лабораторной работы

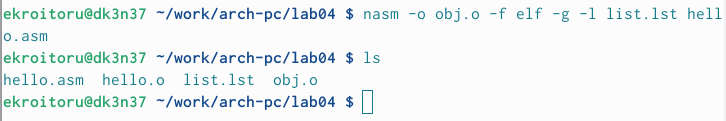
1. Создаём каталог для работы с программами на языке ассемблера NASM (рис. **¿fig:001?**)

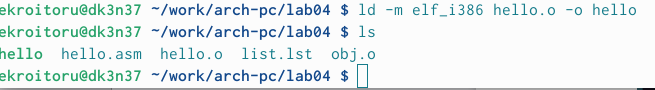
 2) Создаём текстовый файл с именем hello.asm и открываем этот файл с помо- щью любого текстового редактора gedit: (рис. **¿fig:002?**)

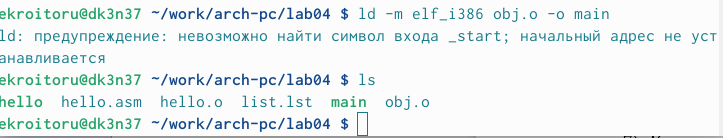
 3) водим в него следующий текст: (рис. **¿fig:003?**)

 4) NASM превращает текст программы в объектный код. Например, для компи- ляции приведённого выше текста программы «Hello World» необходимо написать следующее (рис. **¿fig:004?**) Т. к. текст программы набран без ошибок, транслятор преобразует текст про- граммы из файла hello.asm в объектный код, который записан в файл hello.o.

 5) С помощью команды ls проверим, что объектный файл был создан. У нас есть два файла hello.asm и hello.o. Следующая команда скомпилирует исходный файл hello.asm в obj.o , при этом формат выходного файла будет elf, и в него будут включены символы для отладки (опция -g), кроме того, создается файл листинга list.lst .Выполним следующую команду: (рис. **¿fig:005?**)

 6) Чтобы получить исполняемую программу, объектный файл необходимо пе- редать на обработку компоновщику, а потом с командой ls проверим содержи- мое: (рис. **¿fig:006?**)

 7) Ключ -o с последующим значением задаст в данном случае имя создавае- мого исполняемого файла. Выполним следующую команду Чтобы получить исполняемую программу, объектный файл необходимо пере- дать на обработку компоновщику, а потом с командой ls проверим содержимое: (рис. **¿fig:007?**)

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

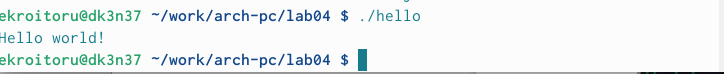
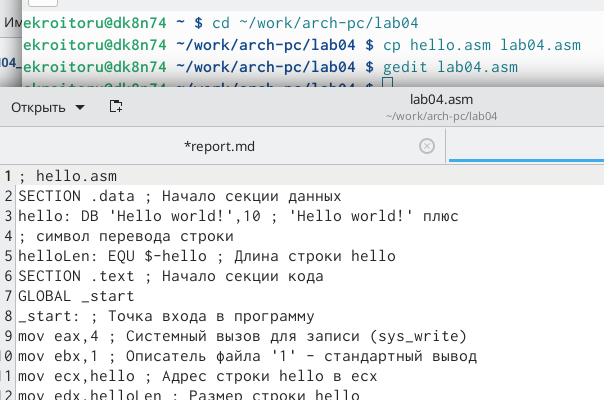


Рис. 1: файл

# 5 Выполнение самостоятельной работы

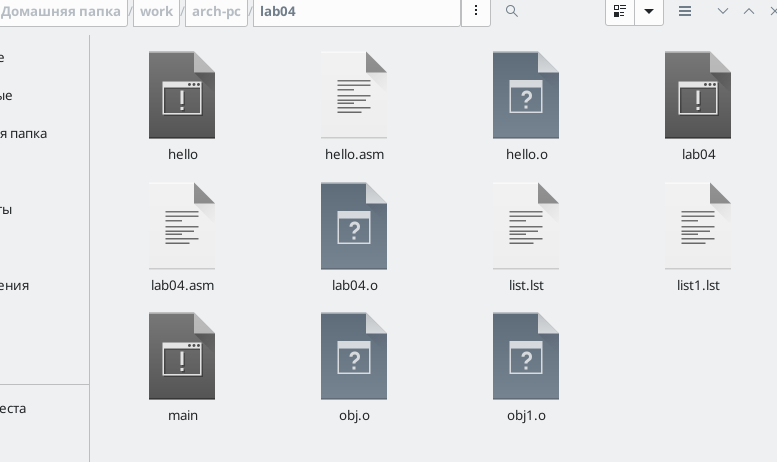
1. В каталоге ~/work/arch-pc/lab04 с помощью команды cp создали копию файла hello.asm с именем lab04.asm. (рис. **¿fig:009?**)

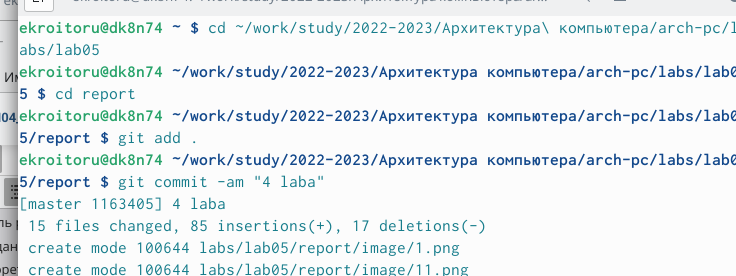
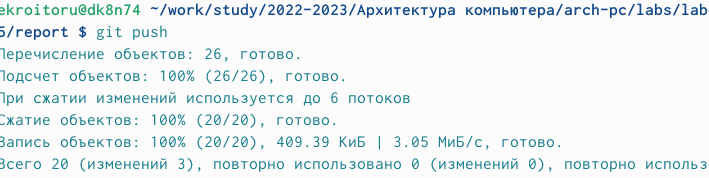
 2) С помощью текстового редактора вносим изменения в текст программы в файле lab04.asm так, чтобы вместо Hello world! на экран выводилась строка с фамилией и именем. Для этого вместо “Hello world” пишем своё имя. (рис. **¿fig:010?**)

самостоятельная работа.png 3) Проводим схожие действия с лабораторной работой, но изменяем название файлов. (рис. **¿fig:011?**)

 4) Оттранслируем полученный текст программы lab04.asm в объектный файл и запустим, получим вывод фамилии и имени. (рис. **¿fig:012?**)

самостоятельная работа.png 5) Переносим файлы в основную папку lab04: (рис. **¿fig:013?**)

 6) Загружаем файлы на GitHub при помощи команд. (рис. **¿fig:014?**) (рис. **¿fig:015?**)

  # Выводы

Я освоил процедуру компиляции и сборки программ, написанных на ассембле- ре NASM.

# Список литературы

1. GNU Bash Manual [Электронный ресурс]. Free Software Foundation, 2016. URL: https://www.gnu.org/software/bash/manual/.
2. Newham C. Learning the bash Shell: Unix Shell Programming. O’Reilly Media,
3. 354 с.
4. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.
5. Robbins A. Bash Pocket Reference. O’Reilly Media, 2016. 156 с.
6. Таненбаум Э. Архитектура компьютера. 6-е изд. СПб.: Питер, 2013. 874 с.
7. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.}

1. GNU Bash Manual [Электронный ресурс]. Free Software Foundation, 2016. URL: <https://www.gnu.org/software/bash/manual/>.

2. Newham C. [Learning the bash Shell: Unix Shell Programming](http://www.amazon.com/Learning-bash-Shell-Programming-Nutshell/dp/0596009658). O’Reilly Media, 2005. 354 с.

3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.

4. Robbins A. [Bash Pocket Reference](https://www.ncbi.nlm.nih.gov/pubmed/25246403). O’Reilly Media, 2016. 156 с.

5. Таненбаум Э. Архитектура компьютера. 6-е изд. СПб.: Питер, 2013. 874 с.

6. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.