Лабораторная работа №5

Создание и процесс обработки программ на языке ассемблера NASM

Рассолова Маргарита Сергеевна

Содержание

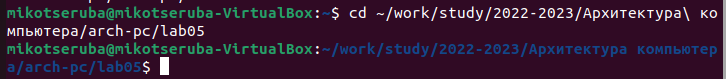
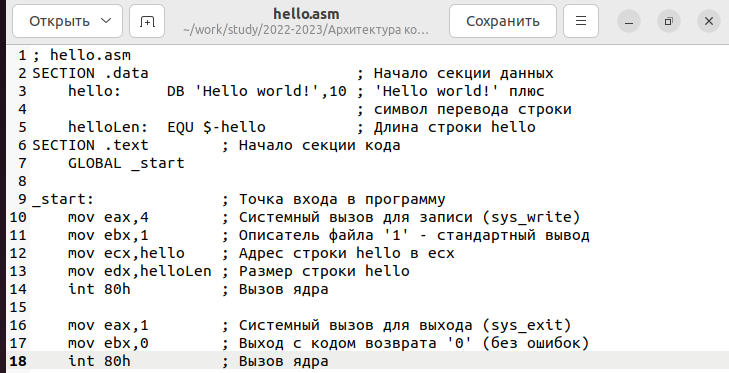
# 1 Цель работы

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

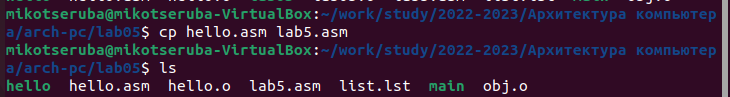
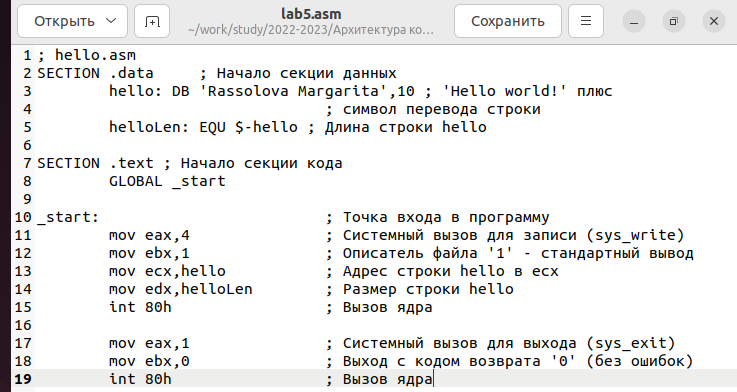
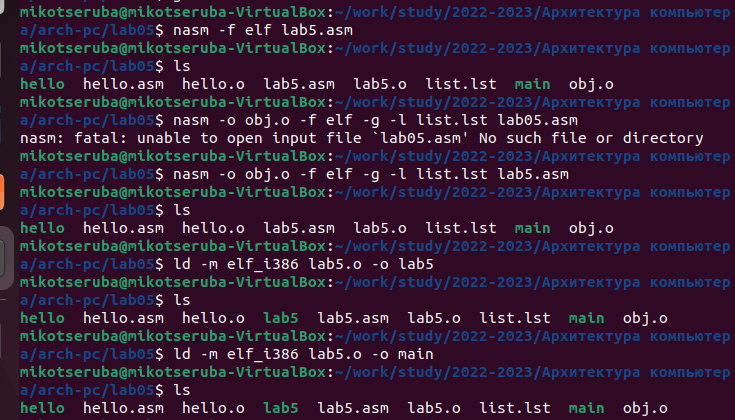
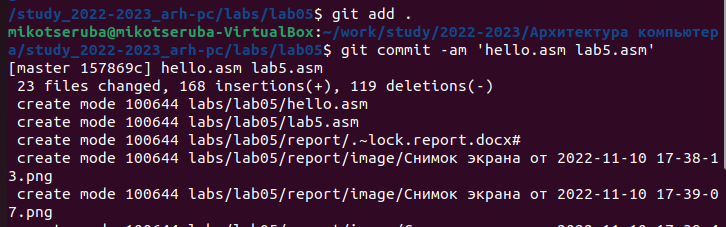
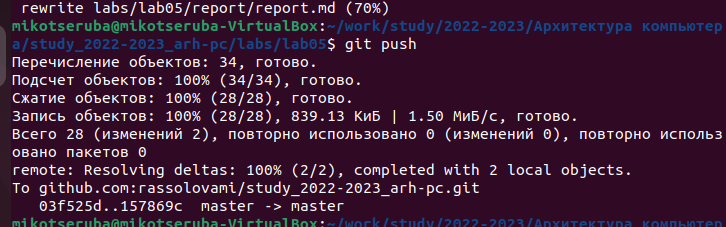
# 2 Теоретическое введение

NASM (Netwide Assembler) — свободный (LGPL и лицензия BSD) ассемблер для архитектуры Intel x86. Используется для написания 16-, 32- и 64-разрядных программ. NASM может работать на платформах, отличных от x86, таких как SPARC и PowerPC, однако код он генерирует только для x86 и x86-64. NASM успешно конкурирует со стандартным в Linux- и многих других UNIX-системах ассемблером gas. Считается, что качество документации у NASM выше, чем у gas. Кроме того, ассемблер gas по умолчанию использует AT&T-синтаксис, ориентированный на процессоры не от Intel, в то время как NASM использует вариант традиционного для x86-ассемблеров Intel-синтаксиса; Intel-синтаксис используется всеми ассемблерами для DOS/Windows, например, MASM, TASM, fasm.

# 3 Выполнение лабораторной работы

1. Создала каталог для работы с программами на языке ассемблера NASM с помощью команды mkdir. Перешла в него с помощью команды cd. (рис. **¿fig:001?**, рис. **¿fig:002?**) Создание каталога 
2. Создала текстовый файл с именем hello.asm с помощью команды touch. (рис. **¿fig:003?**) Создание файла
3. Открыла файл с помощью текстового редактора gedit. Ввела туда текст из файла к лабораторной работе. (рис. **¿fig:004?**, рис. **¿fig:005?**) Открытие файла 
4. Провела компиляцию текста программы с помощью команды nasm -f elf hello.asm. Проверила с помощью команды ls. (рис. **¿fig:006?**, рис. **¿fig:007?**) Компиляция текста Проверка 1
5. Скомпилировала файл с помощью команды nasm -o obj.o -elf -g -l list.lst hello.asm. Проверила с помощью команды ls. (рис. **¿fig:008?**, рис. **¿fig:009?**) Компиляция файла Проверка 2
6. Передала файл на обработку компоновщику с помощью команды ld -m elf\_i386 hello.o -o hello. (рис. **¿fig:010?**) Передача файла на обработку
7. Также выполнила команду ld -m elf\_i386 obj.o -o main. (рис. **¿fig:011?**) Заданное имя исполняемого файла
8. Запустила на выполнение созданный исполняемый файл с помощью команды ./hello. (рис. **¿fig:012?**) Запуск файла

# 4 Самостоятельная работа

1. В каталоге ~/work/study/arch-pc/lab05 с помощью команды cp создала копию файла hello.asm с именем lab5.asm. Проверила с помощью команды ls. (рис. **¿fig:013?**) 
2. С помощью текстового редактора gedit открыла файл, изменила “Hello world!” на свое имя и фамилию. (рис. **¿fig:014?**, рис. **¿fig:015?**) Запуск файла с помощью текстового редактора 
3. Оттранслировала полученный текст программы, выполнила компоновку объектного файла с помощью команд, описанных в 4-7 пунктах выполнения лабораторной работы. (рис. **¿fig:016?**) 
4. Запустила получившийся файл с помощью команды ./lab5. (рис. **¿fig:017?**) Запуск измененного файла
5. Отправила файлы на github. (рис. **¿fig:018?**, рис. **¿fig:019?**)  

# 5 Выводы

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