Лабораторная работа №6. Арифметические операции в NASM

**Танатти Елена Геннадьевна**

Содержание

# 1 Цель работы

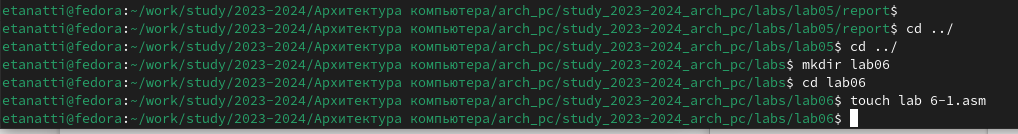
Освоение арифметических инструкций языка ассемблера NASM

# 2 Задание

1. Создайте каталог для программам лабораторной работы № 6, перейдите в него и создайте файл lab6-1.asm: mkdir cd ~/work/arch-pc/lab06 ~/work/arch-pc/lab06 touch lab6-1.asm
2. Рассмотрите примеры программ вывода символьных и численных значений. Программы будут выводить значения записанные в регистр eax. Введите в файл lab6-1.asm текст программы из листинга 6.1. В данной программе в ре гистр eax записывается символ 6 (mov eax,‘6’), в регистр ebx символ 4 (mov ebx,‘4’). Далее к значению в регистре eax прибавляем значение регистра ebx (add eax,ebx, ре- зультат сложения запишется в регистр eax). Далее выводим результат. Так как для работы функции sprintLF в регистр eax должен быть записан адрес, необходимо использовать до- полнительную переменную. Для этого запишем значение регистра eax в переменную buf1 (mov [buf1],eax), а затем запишем адрес переменной buf1 в регистр eax (mov eax,buf1) и вызовем функцию sprintLF.
3. Создайте исполняемый файл и запустите его. nasm -f elf lab6-1.asm ld -m elf\_i386 -o lab6-1 lab6-1.o ./lab6-1

# 3 Выполнение лабораторной работы

1. Создали каталог для программам лабораторной работы № 6, перейдите в него и создайте файл lab6-1.asm:

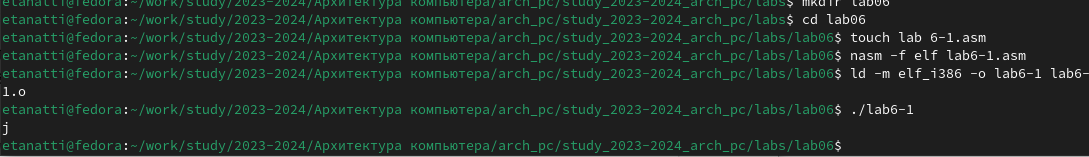


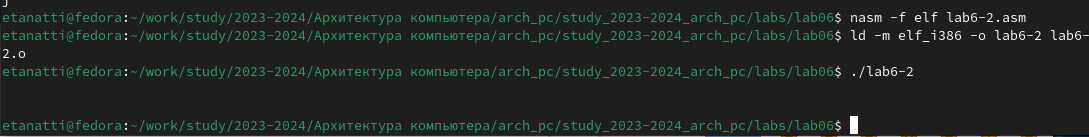
Создали каталог и добавили файл

1. Введите в файл lab6-1.asm текст программы из листинга



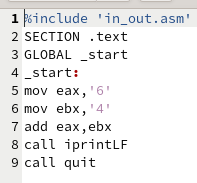
Текст программы

1. Создаем исполняемый файл и запускаем его, ожидаем в результате увидеть j 
2. Изменим текст программы и вместо символов, запишем в регистры числа. Исправляем текст программы, создаем исполняемый файд и запускаем его



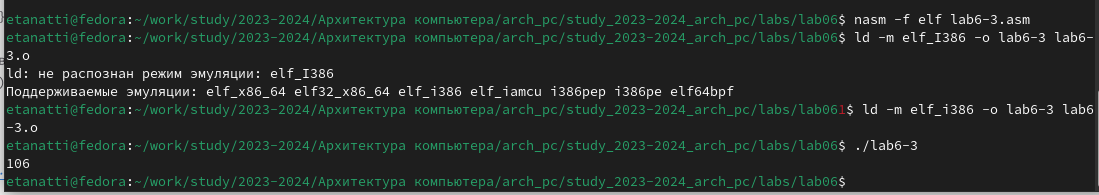
Исправили текст программы, записали в 6-2

1. Создадим новый файл - lab6-3.asm, используя in\_out.asm:



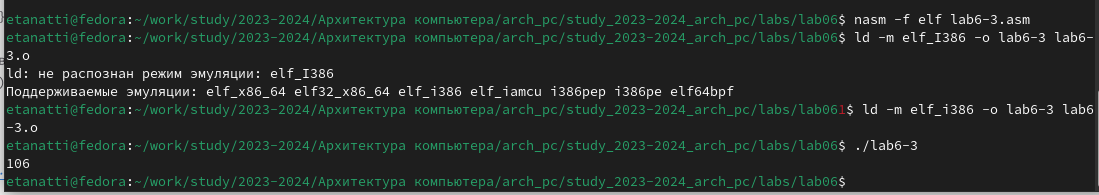
Новый asm файл с in\_out.asm

1. Создаем исполняемый файл и запускаем его

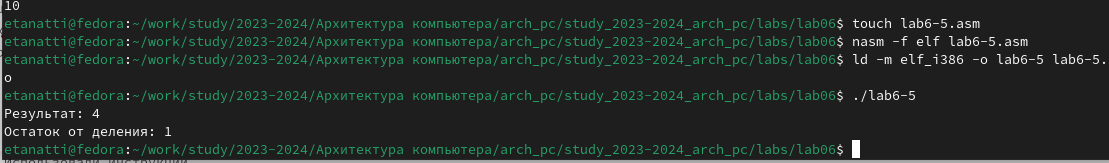


Получаем результат нового файла

1. По аналогии меняем значения в файле, запишем его как 6-4 и запустим:



Запустили полученный файл

1. В качестве примера выполнения арифметических операций в NASM приведем про- грамму вычисления арифметического выражения f(x) = (5x2 + 3)/3 Создадим новый файл lab6-5.asm, запишем в него код программы: По аналогии создаем исполняемый файл и запускаем 

# 4 Выводы

Кажется, нельзя сказать, что освоили - немного коснулись арифметических инструкций языка ассемблера NASM

# 5 Список литературы

1. 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)