

**Лабораторная работа №6.
Арифметические операции в NASM**

Танатти Елена Геннадьевна

Содержание

1	Цель работы	3
2	Задание	4
3	Выполнение лабораторной работы	5
4	Выводы	9
5	Список литературы	10

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 еах. Введите в файл lab6-1.asm текст программы из листинга 6.1. В данной программе в регистр еах записывается символ 6 (`mov еах,'6'`), в регистр ебх символ 4 (`mov ебх,'4'`). Далее к значению в регистре еах прибавляем значение регистра ебх (`add еах,ебх`, результат сложения запишется в регистр еах). Далее выводим результат. Так как для работы функции `sprintf` в регистр еах должен быть записан адрес, необходимо использовать дополнительную переменную. Для этого запишем значение регистра еах в переменную `buf1` (`mov [buf1],еах`), а затем запишем адрес переменной `buf1` в регистр еах (`mov еах,buf1`) и вызовем функцию `sprintf`.
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 программ лабораторной работы № 6, перейдите в него и создайте файл lab6-1.asm:

```
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab05/report$  
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab05/report$ cd ../  
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab05$ cd ../  
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs$ mkdir lab06  
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs$ cd lab06  
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ touch lab 6-1.asm  
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$
```

Рис. 3.1: Создали каталог и добавили файл

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

```

%include 'in_out.asm'
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax,'6'
mov ebx,'4'
add eax,ebx
mov [buf1],eax
mov eax,buf1
call sprintf
call quit

```

Рис. 3.2: Текст программы

3. Создаем исполняемый файл и запускаем его, ожидаем в результате увидеть j

```

etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs$ mkdir lab06
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs$ cd lab06
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ touch lab-1.asm
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ nasm -f elf lab6-1.asm
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ld -m elf_i386 -o lab6-1 lab6-1.o
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ./lab6-1
j
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$

```

4. Изменим текст программы и вместо символов, запишем в регистры числа.
Исправляем текст программы, создаем исполняемый файл и запускаем его

```

etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ nasm -f elf lab6-2.asm
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ld -m elf_i386 -o lab6-2 lab6-2.o
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ./lab6-2

```

Рис. 3.3: Исправили текст программы, записали в 6-2

5. Создадим новый файл - lab6-3.asm, используя in_out.asm:

```

1 |%include 'in_out.asm'
2 SECTION .text
3 GLOBAL _start
4 _start:
5 mov eax, '6'
6 mov ebx, '4'
7 add eax, ebx
8 call iprintLF
9 call quit

```

Рис. 3.4: Новый asm файл с in_out.asm

6. Создаем исполняемый файл и запускаем его

```

etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ nasm -f elf lab6-3.asm
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
ld: не распознан режим эмуляции: elf_i386
Поддерживаемые эмуляции: elf_x86_64 elf32_x86_64 elf_i386 elf_iamcu i386pep i386pe elf64bpf
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ./lab6-3
106
etanatti@fedora:~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$

```

Рис. 3.5: Получаем результат нового файла

7. По аналогии меняем значения в файле, запишем его как lab6-4 и запустим:

```
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ nasm -f elf lab6-3.asm
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
ld: не распознан режим эмуляции: elf_i386
Поддерживаемые эмуляции: elf_x86_64 elf32_x86_64 elf_i386 elf_iamcu i386pep i386pe elf64bpf
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ld -m elf_i386 -o lab6-3 lab6-3.o
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ./lab6-3
106
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$
```

Рис. 3.6: Запустили полученный файл

8. В качестве примера выполнения арифметических операций в NASM приведем программу 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

```
10
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ touch lab6-5.asm
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ nasm -f elf lab6-5.asm
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ld -m elf_i386 -o lab6-5 lab6-5.o
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$ ./lab6-5
Результат: 4
Остаток от деления: 1
etanatti@fedora: ~/work/study/2023-2024/Архитектура компьютера/arch_pc/study_2023-2024_arch_pc/labs/lab06$
```


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.
12. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
13. Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.
14. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ- Петербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
15. 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.
16. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science)
17. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер, 2015 — 1120 с. — (Классика Computer Science)