

Отчет по лабораторной работе № 6

Архитектура вычислительных систем

Кулагина Анна Сергеевна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	12
	Список литературы	13

Список иллюстраций

Список таблиц

1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

2 Задание

1. Написать программу вычисления выражения $y = f(x)$. Программа должна выводить выражение для вычисления, выводить запрос на ввод значения x , вычислять заданное выражение в зависимости от введенного x , выводить результат вычислений. Вид функции $f(x)$ выбрать из таблицы 6.3 вариантов заданий в соответствии с номером полученным при выполнении лабораторной работы. Создайте исполняемый файл и проверьте его работу для значений x_1 и x_2 из 6.3.

3 Выполнение лабораторной работы

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. ??) 1. Создадим каталог для программ лабораторной работы No 6, перейдем в него и создадим файл lab6-1.asm

```
askulagina@dk2n22 ~ $ mkdir ~/work/arch-pc/lab06
askulagina@dk2n22 ~ $ cd ~/work/arch-pc/lab06
askulagina@dk2n22 ~/work/arch-pc/lab06 $ touch lab6-1.asm
askulagina@dk2n22 ~/work/arch-pc/lab06 $
```

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. ??) 2. Введем в файл lab6-1.asm текст программы из листинга 7.1. В данной программе в регистр eax записывается символ 6, в регистр ebx символ 4. К значению в регистре eax прибавляем значение регистра ebx. Далее выводим результат. Так как для работы функции `sprintf` в регистр eax должен быть записан адрес, необходимо использовать дополнительную переменную. Для этого запишем значение регистра eax в переменную `buf1`, а затем запишем адрес переменной `buf1` в регистр eax и вызовем функцию `sprintf`.

```

...dk.sci.pfu.edu.ru/home/a/s/askulagina/work/arch-pc/lab06/lab6-1.asm
%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 sprintLF
call quit

```

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на

```

askulagina@dk2n22 ~/work/arch-pc/lab06 $ nasm -f elf lab6-1.asm
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-1 lab6-1.o
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ./lab6-1
bash: ./lab6-1: Нет такого файла или каталога
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ./lab7-1
bash: ./lab7-1: Нет такого файла или каталога
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ./lab6-1
j

```

иллюстрацию (рис. ??)

3.Далее изменим текст программы и вместо символов, запишем в регистры числа. Исправим текст программы (Листинг 1) заменив строки на `mov eax,6` `mov ebx,4`.

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на

```

%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 sprintLF
call quit

```

иллюстрацию (рис. ??)

4.Преобразуем текст программы из Листинга 7.1 с использованием этих функций.Создадим файл `lab6-2.asm` в каталоге `~/work/arch-pc/lab06` и введем в него текст программы из листинга 7.2.Создадим исполняемый файл и запустим его.

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на

```
%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
```

иллюстрацию (рис. ??)

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на

```
askulagina@dk2n22 ~/work/arch-pc/lab06 $ mc
askulagina@dk2n22 ~/work/arch-pc/lab06 $ nasm -f elf lab6-2.asm
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-2 lab6-2.o
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ./lab6-2
10askulagina@dk2n22 ~/work/arch-pc/lab06 $
```

иллюстрацию (рис. ??)

5. Аналогично предыдущему примеру изменим символы на числа. Заменяя строки на `mov eax,6` `mov ebx,4` Создадим исполняемый файл и запустим его. Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстра-

```
askulagina@dk2n22 ~/work/arch-pc/lab06 $ nasm -f elf lab6-2.asm
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-2 lab6-2.o
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ./lab6-2
106askulagina@dk2n22 ~/work/arch-pc/lab06 $
```

цию (рис. ??)

6. В качестве примера выполнения арифметических операций в NASM приведем программу вычисления арифметического выражения $\frac{(5 \times 2 + 3)}{3}$. Создадим файл `lab6-3.asm` в каталоге `~/work/arch-pc/lab06`. Введем текст программы из листинга 7.3 и введем его `lab6-3.asm`. Создадим исполняемый файл и запустим его. Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. ??)

```
askulagina@dk2n22 ~/work/arch-pc/lab06 $ nasm -f elf lab6-3.asm
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-3 lab6-3.o
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ./lab6-3
Результат: 4
Остаток от деления: 1
askulagina@dk2n22 ~/work/arch-pc/lab06 $
```

Изменим текст программы для вычисления выражения $\text{X}(\text{X}) = (4 \times 6 + 2)/5$. Создадим исполняемый файл и проверим его работу. Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. ??)

```
askulagina@dk2n22 ~/work/arch-pc/lab06 $ nasm -f elf lab6-3.asm
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-3 lab6-3.o
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ./lab6-3
Результат: 5
Остаток от деления: 1
askulagina@dk2n22 ~/work/arch-pc/lab06 $
```

7. В качестве другого примера рассмотрим программу вычисления варианта задания по номеру студенческого билета, работающую по алгоритму. Создадим файл `variant.asm` в каталоге `~/work/arch-pc/lab06`. Введем текст программы из листинга 7.4 и введем в файл `variant.asm`. Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис.

```
askulagina@dk2n22 ~/work/arch-pc/lab06 $ nasm -f elf variant.asm
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o variant variant.o
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ./variant
Введите № студенческого билета:
1132221543
Ваш вариант: 4
askulagina@dk2n22 ~/work/arch-pc/lab06 $
```

Ответы на вопросы: 1. За вывод на экран сообщения 'Ваш вариант:' отвечают строки 2. `nasm` используется для `mov` используется

Ход выполнения самостоятельной работы: Напишем программу вычисления выражения $\text{X} = 4/3 (\text{X} - 1) + 5$ Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. ??)

```
askulagina@dk2n22 ~/work/arch-pc/lab06 $ nasm -f elf variant4.asm
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o variant4 variant4.o
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ./variant4
Введите x
4
Результат 5
askulagina@dk2n22 ~/work/arch-pc/lab06 $ ./variant4
Введите x
10
Результат 7
askulagina@dk2n22 ~/work/arch-pc/lab06 $
```

4 Выводы

Я приобрела практические навыки и умения по разработке командных файлов и освоила арифметические инструкции языка ассемблера NASM.

Список литературы