

Шаблон отчёта по лабораторной работе №6

Арифметические операции в NASM.

Саидова Маржина Авдулвохидовна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	18
	Список литературы	19

Список иллюстраций

4.1	каталог лаб6	8
4.2	текст программы из листинга 6.1	9
4.3	запуск исполняемый файл	9
4.4	исправленный текст (листинг 6.1)	10
4.5	запуск файла	10
4.6	запуск lab6-2	11
4.7	текст из листинга 6.2	11
4.8	исправленный текст	12
4.9	запуск файла	12
4.10	запуск файла с заменой	13
4.11	запуск файла lab6-3.asm	13
4.12	изменения текста	14
4.13	проверка работы файла	14
4.14	листинг 6.4	15
4.15	запуск файла	15
4.16	проверка работы файла	16
4.17	текст программы	17

Список таблиц

3.1	Описание некоторых каталогов файловой системы GNU Linux . . .	7
-----	---	---

1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

2 Задание

1. Создать каталог для программ лабораторной работы № 6.
2. Создать файл lab6-1.asm и ввести в файл lab6-1.asm текст программы из листинга 6.1. Создать исполняемый файл и запустить его.
3. Исправить текста программы (Листинг 6.1), создать исполняемый файл и запустить его.
4. Создать файл lab6-2.asm и ввести в него текст программы из листинга 6.2. Исправить текст программы (Листинг 6.2). Создать исполняемый файл и запустите его.
5. Заменить функцию `iprintLF` на `iprint`. Создать исполняемый файл и запустить его.
6. Создать файл lab6-3.asm. Ввести текст программы из листинга 6.3.
7. Изменить текст программы для вычисления выражения $\pi(\pi) = (4 \cdot 6 + 2)/5$. Создать исполняемый файл и проверить его работу.
8. Создать файл variant.asm в каталоге. Ввести в файл variant.asm текст программы из 6.4 листинга. Создать исполняемый файл и запустить его.
9. Ответить на вопросы.
10. Самостоятельная работа

3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. 3.1 приведено краткое описание стандартных каталогов Unix.

Таблица 3.1: Описание некоторых каталогов файловой системы GNU Linux

Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую систему
/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 про Unix см. в [1–4].

4 Выполнение лабораторной работы

1. Создать каталог для программ лабораторной работы № 6.

```
masaidova@dk8n60 ~ $ mkdir ~/work/arch-pc/lab06
masaidova@dk8n60 ~ $ cd ~/work/arch-pc/lab06
masaidova@dk8n60 ~/work/arch-pc/lab06 $ touch lab6-1.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ nasm -f elf lab6-1.asm
```

Рис. 4.1: каталог лаб6

2. Создать файл lab6-1.asm и ввести в файл lab6-1.asm текст программы из листинга 6.1. Создать исполняемый файл и запустить его.


```
1 %include 'in_out.asm'
2 SECTION .bss
3 buf1: RESB 80
4 SECTION .text
5 GLOBAL _start
6 _start:
7 mov eax, '6'
8 mov ebx, '4'
9 add eax, ebx
10 mov [buf1], eax
11 mov eax, buf1
12 call sprintLF
13 call quit
```

Рис. 4.2: текст программы из листинга 6.1

```
masaidova@dk8n60 ~/work/arch-pc/lab06 $ nasm -f elf lab6-1.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-1 lab6-1.o
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ./lab6-1

masaidova@dk8n60 ~/work/arch-pc/lab06 $
```

Рис. 4.3: запуск исполняемый файл

3. Исправить текста программы (Листинг 6.1), создать исполняемый файл и запустить его.

```

1 %include 'in_out.asm'
2 SECTION .bss
3 buf1: RESB 80
4 SECTION .text
5 GLOBAL _start
6 _start:
7 mov eax,6
8 mov ebx,4
9 add eax,ebx
0 mov [buf1],eax
1 mov eax,buf1
2 call sprintLF
3 call quit

```

Рис. 4.4: исправленный текст (листинг 6.1)

```

masaidova@dk8n60 ~/work/arch-pc/lab06 $ nasm -f elf lab6-1.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-1 lab6-1.o
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ./lab6-1

masaidova@dk8n60 ~/work/arch-pc/lab06 $ █

```

Рис. 4.5: запуск файла

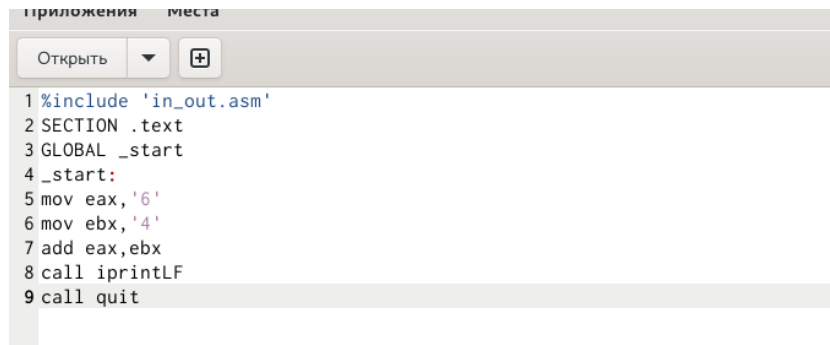
4. Создать файл lab6-2.asm и ввести в него текст программы из листинга 6.2.

```

masaidova@dk8n60 ~/work/arch-pc/lab06 $ touch ~/work/arch-pc/lab06/lab6-2.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ nasm -f elf lab6-2.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-2 lab6-2.o
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ./lab6-2
106
masaidova@dk8n60 ~/work/arch-pc/lab06 $

```

Рис. 4.6: запуск lab6-2



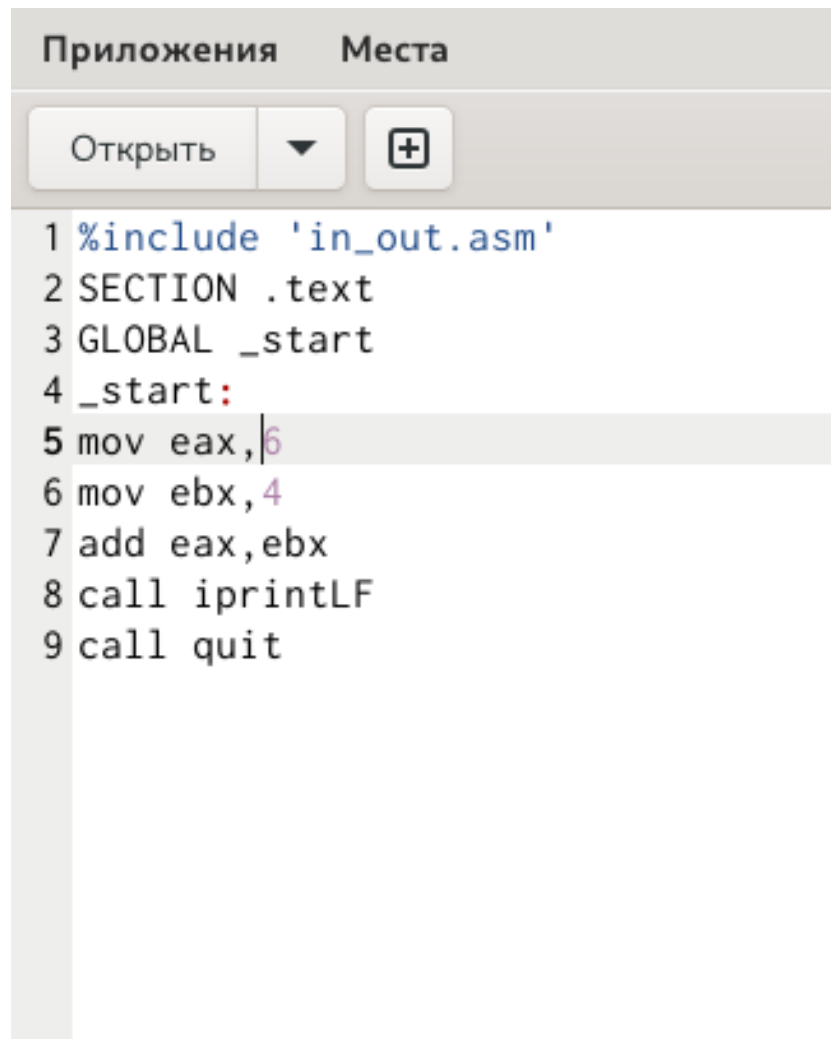
```

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

```

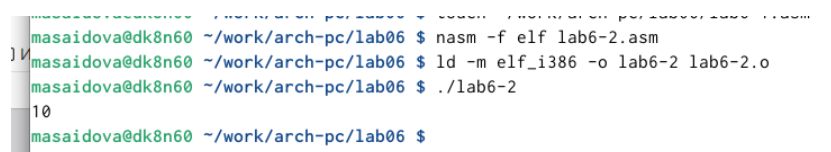
Рис. 4.7: текст из листинга 6.2

5. Исправить текст программы (Листинг 6.2). Создать исполняемый файл и запустите его.



```
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. 4.8: исправленный текст



```
masaidova@dk8n60 ~/work/arch-pc/lab06 $ nasm -f elf lab6-2.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-2 lab6-2.o
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ./lab6-2
10
masaidova@dk8n60 ~/work/arch-pc/lab06 $
```

Рис. 4.9: запуск файла

6. Заменить функцию `iprintLF` на `iprint`. Создать исполняемый файл и запустить его.

```

masaidova@dk8n60 ~/work/arch-pc/lab06 $ nasm -f elf lab6-2.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-2 lab6-2.o
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ./lab6-2
10masaidova@dk8n60 ~/work/arch-pc/lab06 $

```

Рис. 4.10: запуск файла с заменой

7. Создать файл lab6-3.asm. Ввести текст программы из листинга 6.3. Запустить файл.

```

masaidova@dk8n60 ~/work/arch-pc/lab06 $ nasm -t elf lab6-3.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-3 lab6-3.o
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ./lab6-3
Результат: 4
Остаток от деления: 1
masaidova@dk8n60 ~/work/arch-pc/lab06 $

```

Рис. 4.11: запуск файла lab6-3.asm

8. Изменить текст программы для вычисления выражения $\pi(\pi) = (4 \cdot \pi + 2)/5$. Создать исполняемый файл и проверить его работу.

```

1 ;-----
2 ; Программа вычисления выражения
3 ;-----
4 %include 'in_out.asm' ; подключение внешнего файла
5 SECTION .data
6 div: DB 'Результат: ',0
7 rem: DB 'Остаток от деления: ',0
8 SECTION .text
9 GLOBAL _start
10 _start:
11 ; ---- Вычисление выражения
12 mov eax,4 ; EAX=4
13 mov ebx,6 ; EBX=6
14 mul ebx ; EAX=EAX*EBX
15 add eax,2 ; EAX=EAX+2
16 xor edx,edx ; обнуляем EDX для корректной работы div
17 mov ebx,5 ; EBX=5
18 div ebx ; EAX=EAX/5, EDX=остаток от деления
19 mov edi,eax ; запись результата вычисления в 'edi'
20 ; ---- Вывод результата на экран
21 mov eax,div ; вызов подпрограммы печати
22 call sprint ; сообщения 'Результат: '
23 mov eax,edi ; вызов подпрограммы печати значения
24 call iprintLF ; из 'edi' в виде символов
25 mov eax,rem ; вызов подпрограммы печати
26 call sprint ; сообщения 'Остаток от деления: '
27 mov eax,edx ; вызов подпрограммы печати значения
28 call iprintLF ; из 'edx' (остаток) в виде символов
29 call quit ; вызов подпрограммы завершения

```

Рис. 4.12: изменения текста

```

masaidova@dk8n60 ~/work/arch-pc/lab06 $ nasm -f elf lab6-3.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o lab6-3 lab6-3.o
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ./lab6-3
Результат: 5
Остаток от деления: 1
masaidova@dk8n60 ~/work/arch-pc/lab06 $

```

Рис. 4.13: проверка работы файла

9. Создать файл `variant.asm` в каталоге. Ввести в файл `variant.asm` текст программы из 6.4 листинга. Создать исполняемый файл и запустить его.

```

1 ;-----
2 ; Программа вычисления варианта
3 ;-----
4 %include 'in_out.asm'
5 SECTION .data
6 msg: DB 'Введите № студенческого билета: ',0
7 rem: DB 'Ваш вариант: ',0
8 SECTION .bss
9 x: RESB 80
10 SECTION .text
11 GLOBAL _start
12 _start:
13 mov eax, msg
14 call sprintLF
15 mov ecx, x
16 mov edx, 80
17 call sread
18 mov eax, x ; вызов подпрограммы преобразования
19 call atoi ; ASCII кода в число, 'eax=x'
20 xor edx, edx
21 mov ebx, 20
22 div ebx
23 inc edx
24 mov eax, rem
25 call sprint
26 mov eax, edx
27 call iprintLF
28 call quit

```

Рис. 4.14: листинг 6.4

```

masaidova@dk8n60 ~/work/arch-pc/lab06 $ touch ~/work/arch-pc/lab06/variant.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ nasm -f elf variant.asm
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ld -m elf_i386 -o variant variant.o
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ./variant
Введите № студенческого билета:
1132240683
Ваш вариант: 4
masaidova@dk8n60 ~/work/arch-pc/lab06 $

```

Рис. 4.15: запуск файла

10. Ответить на вопросы:

Какие строки листинга 6.4 отвечают за вывод на экран сообщения ‘Ваш вариант:’? `mov eax, rem` `call sprint` 2 Для чего используются следующие инструкции? `mov ecx, x` `mov edx, 80` `call sread` Для получения данных с клавиатуры. Для чего используется инструкция “`call atoi`”? Для преобразования ASCII кода в число

Какие строки листинга 6.4 отвечают за вычисления варианта? `xor edx, edx` `mov`

ebx, 20 div ebx inc edx В какой регистр записывается остаток от деления при выполнении инструкции “div ebx”? Для чего используется инструкция “inc edx”? Увеличение edx на 1 Какие строки листинга 6.4 отвечают за вывод на экран результата вычислений? mov eax, edx call iprintlF

11. Самостоятельная работа

```
masaidova@dk8n60 ~/work/arch-pc/lab06 $ ./sr
Формула 4/3*(x - 1) + 5
Введите x:
4
Результат: 9
Остаток от деления: 0

masaidova@dk8n60 ~/work/arch-pc/lab06 $ ./sr
Формула 4/3*(x - 1) + 5
Введите x:
10
Результат: 17
Остаток от деления: 0

masaidova@dk8n60 ~/work/arch-pc/lab06 $
```

Рис. 4.16: проверка работы файла


```

#include 'in_out.asm'

SECTION .data
    func: DB "Формула  $\frac{4}{3} \cdot (x - 1) + 5$ ", 0
    msg: DB "Введите x: ", 0
    res: DB "Результат: ", 0
    rem_msg: DB "Остаток от деления: ", 0
    newline: db 0xA

SECTION .bss
    x: RESB 80
    result: resd 1
    remainder: resd 1

SECTION .text
GLOBAL _start
_start:

    mov eax, func
    call sprintf

    mov eax, msg
    call sprintf

    mov ecx, x
    mov edx, 80
    call sread

    mov eax, x
    call atoi

    ; Вычисление  $\frac{4}{3} \cdot (x - 1) + 5$  (с остатком)
    sub eax, 1      ;  $x - 1$ 
    mov ebx, eax     ; Сохраняем  $(x - 1)$ 
    mov eax, 4
    mul ebx          ;  $4 \cdot (x - 1)$ 
    mov ebx, 3       ; Делим на 3
    cdq              ; Расширяем eax до edx:eax (для idiv)
    idiv ebx         ; eax = результат деления, edx = остаток
    add eax, 5       ;  $+ 5$ 

```

Рис. 4.17: текст программы

5 Выводы

Освоила арифметические инструкции языка ассемблера NASM.

Список литературы

1. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.
2. Robbins A. Bash Pocket Reference. O'Reilly Media, 2016. 156 с.
3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.
4. Newham C. Learning the bash Shell: Unix Shell Programming. O'Reilly Media, 2005. 354 с.