

Лабораторная работа №9

Программирование цикла. Обработка аргументов командной строки.

Воробчук Лилия Андреевна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Самостоятельная работа	19
4	Выводы	22

Список иллюстраций

2.1	создание файла	6
2.2	ввод листинга	7
2.3	запуск	8
2.4	измененная программа	9
2.5	запуск	10
2.6	внесение изменений	11
2.7	исполнение файла	12
2.8	создание файла	13
2.9	запуск	14
2.10	ввод листинга	15
2.11	исполнение	16
2.12	измененный текст программы	17
2.13	запуск	18
3.1	программа для функции	20
3.2	запуск	21

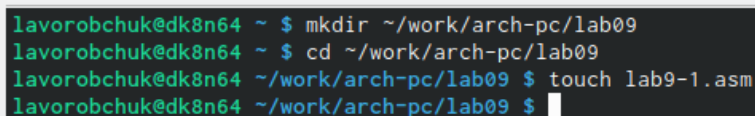
Список таблиц

1 Цель работы

Приобретение навыков написания программ с использованием циклов и обработкой аргументов командной строки.

2 Выполнение лабораторной работы

1. Создаю каталог для программ лабораторной работы №9, перехожу в него и создаю файл lab9-1.asm (рис. 2.1)



```
lavorobchuk@dk8n64 ~ $ mkdir ~/work/arch-pc/lab09
lavorobchuk@dk8n64 ~ $ cd ~/work/arch-pc/lab09
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ touch lab9-1.asm
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $
```

Рис. 2.1: создание файла

2. Ввожу в файл lab9-1.asm текст программы из листинга 9.1. (рис. 2.2)

```

1 %include 'in_out.asm'
2 SECTION .data
3 msg1 db 'Введите N: '
4 SECTION .bss
5 N: resb 10
6 SECTION .text
7 global _start
8 _start:
9
10 mov eax,msg1
11 call sprint
12
13 mov ecx, N
14 mov edx, 10
15 call sread
16
17 mov eax,N
18 call atoi
19 mov [N],eax
20
21 mov ecx,[N]
22 label:
23
24 mov [N],ecx
25 mov eax,[N]
26 call iprintLF
27 loop label
28
29 call quit

```

Рис. 2.2: ввод листинга

3. Создаю исполняемый файл и проверяю его работу. (рис. 2.3)

```
lavorobchuk@k8n64 ~/work/arch-pc/lab09 $ nasm -f elf lab9-1.asm
lavorobchuk@k8n64 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab9-1 lab9-1.o
lavorobchuk@k8n64 ~/work/arch-pc/lab09 $ ./lab9-1
Введите N: 12
12
11
10
9
8
7
6
5
4
3
2
1
lavorobchuk@k8n64 ~/work/arch-pc/lab09 $
```

Рис. 2.3: запуск

4. Меняю текст программы lab9-1, добавив изменение значение регистра есх в цикле. В данном случае число проходов цикла соответствует значению N введенному с клавиатуры (рис. 2.4)


```

1 %include 'in_out.asm'
2 SECTION .data
3 msg1 db 'Введите N: '
4 SECTION .bss
5 N: resb 10
6 SECTION .text
7 global _start
8 _start:
9
10 mov eax,msg1
11 call sprint
12
13 mov ecx, N
14 mov edx, 10
15 call sread
16
17 mov eax,N
18 call atoi
19 mov [N],eax
20
21 mov ecx,[N]
22 label:
23
24 sub ecx,1
25 mov [N],ecx
26 mov eax,[N]
27 call iprintLF
28 loop label
29
30 call quit

```

Рис. 2.4: измененная программа

5. Создаю исполняемый файл и проверяю его работу. (рис. 2.5)

```
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ nasm -f elf lab9-1.asm
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab9-1 lab9-1.o
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ./lab9-1
Введите N: 12
11
9
7
5
3
1
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $
```

Рис. 2.5: запуск

6. Вношу изменения в текст программы, добавив команды `push` и `pop` для сохранения значения счетчика цикла `loop`. В данном случае число проходов цикла не соответствует значению `N` введенному с клавиатуры (рис. 2.6)

```

1 %include 'in_out.asm'
2 SECTION .data
3 msg1 db 'Введите N: '
4 SECTION .bss
5 N: resb 10
6 SECTION .text
7 global _start
8 _start:
9
10 mov eax,msg1
11 call sprint
12
13 mov ecx, N
14 mov edx, 10
15 call sread
16
17 mov eax,N
18 call atoi
19 mov [N],eax
20
21 mov ecx,[N]
22 label:
23
24 push ecx
25 sub ecx,1
26 mov [N],ecx
27 mov eax,[N]
28 call iprintLF
29 pop ecx
30 loop label
31
32 call quit

```

Рис. 2.6: внесение изменений

7. Создаю исполняемый файл и запускаю его (рис. 2.7)

```
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ nasm -f elf lab9-1.asm
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab9-1 lab9-1.o
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ./lab9-1
Введите N: 12
11
10
9
8
7
6
5
4
3
2
1
0
```

Рис. 2.7: исполнение файла

8. Создаю файл lab9-2.asm в каталоге и ввожу в него текст программы из листинга 9.2.(рис. 2.8)

```
1 %include 'in_out.asm'
2 SECTION .text
3 global _start
4 _start:
5 pop ecx
6
7 pop edx
8
9 sub ecx, 1
10 next:
11 cmp ecx, 0
12 jz _end
13
14 pop eax
15 call sprintf
16 loop next
17
18 _end:
19 call quit
```

Рис. 2.8: создание файла

9. Создаю исполняемый файл и запускаю его, указав аргументы (рис. 2.9)

```
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ nasm -f elf lab9-2.asm
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab9-2 lab9-2.o
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ./lab9-2
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ 4 8 '5'
bash: 4: команда не найдена
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ./lab9-2 4 8 '7'
4
8
7
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $
```

Рис. 2.9: запуск

10. Создаю файл lab9-3.asm и ввожу в него текст программы из листинга 9.3 (рис. 2.10)

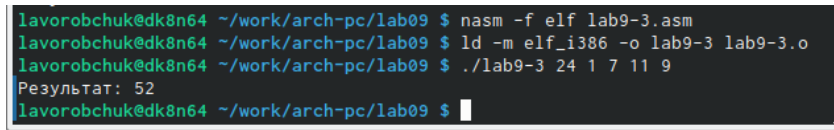
```

1 %include 'in_out.asm'
2 SECTION .data
3 msg db "Результат: ",0
4 SECTION .text
5 global _start
6 _start:
7 pop ecx
8
9 pop edx
10
11 sub ecx,1
12
13 mov esi, 0
14
15 next:
16 cmp ecx,0h
17 jz _end
18
19 pop eax
20 call atoi
21 add esi,eax
22
23 loop next
24
25 _end:
26 mov eax, msg
27 call sprint
28 mov eax, esi
29 call iprintLF
30 call quit

```

Рис. 2.10: ввод листинга

11. Создаю исполняемый файл и запускаю его, указав аргументы (рис. 2.11)



```
lavorobchuk@k8n64 ~/work/arch-pc/lab09 $ nasm -f elf lab9-3.asm
lavorobchuk@k8n64 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab9-3 lab9-3.o
lavorobchuk@k8n64 ~/work/arch-pc/lab09 $ ./lab9-3 24 1 7 11 9
Результат: 52
lavorobchuk@k8n64 ~/work/arch-pc/lab09 $
```

Рис. 2.11: исполнение

12. Изменяю текст программы из листинга 9.3 для вычисления произведения аргументов командной строки (рис. 2.12)


```

1 %include 'in_out.asm'
2
3 SECTION .data
4 msg DB "Результат: ",0
5
6 SECTION .text
7 GLOBAL _start
8
9 _start:
10
11 pop ecx
12
13 pop edx
14
15 sub ecx,1
16
17 mov esi,1
18 mov eax,1
19
20 next:
21 cmp ecx,0
22 jz _end
23
24 pop eax
25 call atoi
26 mov ebx,eax
27 mov eax,esi
28 mul ebx
29 mov esi,eax
30 loop next
31
32
33
34 _end:
35 mov eax,msg
36 call sprint
37 mov eax,esi
38 call iprintLF
39
40 call quit

```

Рис. 2.12: измененный текст программы

13. Создаю исполняемый файл и запускаю его, указав аргументы (рис. 2.13)

```
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ nasm -f elf lab9-3.asm
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o lab9-3 lab9-3.o
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ./lab9-3 31 7 11 9 3
Результат: 64449
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $
```

Рис. 2.13: запуск

3 Самостоятельная работа

1. Пишу программу, которая находит сумму значений функции $f(x) = 4x + 3$ (рис. 3.1)

```

1 %include 'in_out.asm'
2 SECTION .data
3 primer: DB 'Функция: f(x)= 4x + 3',0
4 result: DB 'Результат: ', 0
5
6 SECTION .text
7 GLOBAL _start
8
9 _start:
10 mov eax, primer
11 call sprintf
12
13 pop ecx
14 pop edx
15 sub ecx, 1
16 mov esi, 0
17
18 next:
19 cmp ecx, 0h
20 jz _end
21 pop eax
22 call atoi
23 mov ebx, 4
24 mul ebx
25 add eax, 3
26 add esi, eax
27 loop next
28
29 _end:
30
31 mov eax, result
32 call sprintf
33 mov eax, esi
34 call iprintLF
35 call quit
36

```

Рис. 3.1: программа для функции

2. Создаю исполняемый файл и запускаю его, указав аргументы (рис. 3.2)

```
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ nasm -f elf tawt.asm
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ld -m elf_i386 -o tawt tawt.o
lavorobchuk@dk8n64 ~/work/arch-pc/lab09 $ ./tawt 5 3 8 19
Функция:  $f(x) = 4x + 3$ 
Результат: 152
```

Рис. 3.2: запуск

4 Выводы

В ходе выполнения данной лабораторной работы я приобрела навыки написания программ с использованием циклов и обработкой аргументов командной строки.