

Лабораторная работа №8.

Программирование цикла. Обработка аргументов командной строки.

Митрофанов Тимур Александрович

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Задание для самостоятельной работы	17
4	Выводы	21

Список иллюстраций

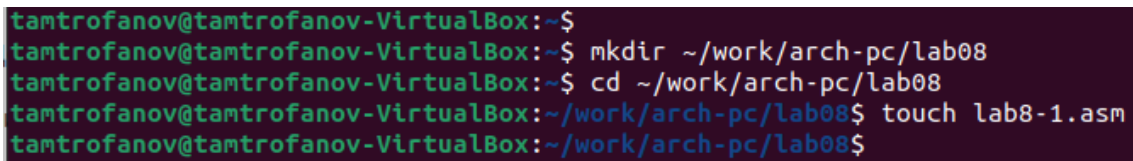
2.1	Создание подкаталога <i>lab08</i> и файла <i>lab8-1.asm</i>	5
2.2	Внесённый код в файл <i>lab8-1.asm</i>	6
2.3	Воспроизведение файла <i>lab8-1.asm</i>	7
2.4	Изменённый файл <i>lab8-1.asm</i>	8
2.5	Воспроизведение файла <i>lab8-1.asm</i>	9
2.6	Изменённый файл <i>lab8-1.asm</i>	10
2.7	Воспроизведение файла <i>lab8-1.asm</i>	11
2.8	Создание файла <i>lab8-2.asm</i>	11
2.9	Внесённый код в файл <i>lab8-2.asm</i>	12
2.10	Воспроизведение файла <i>lab8-2.asm</i>	12
2.11	Создание файла <i>lab8-3.asm</i>	13
2.12	Внесённый код в файл <i>lab8-3.asm</i>	14
2.13	Воспроизведение файла <i>lab8-3.asm</i>	14
2.14	Изменённый файл <i>lab8-3.asm</i>	15
2.15	Воспроизведение файла <i>lab8-3.asm</i>	16
3.1	Создание файла <i>lab8-4.asm</i>	17
3.2	Внесённый код в файл <i>lab8-4.asm</i>	18
3.3	Воспроизведение файла <i>lab8-4.asm</i>	20

1 Цель работы

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

2 Выполнение лабораторной работы

Создал подкаталог *lab08* и в нём файла *lab8-1.asm* (рис. 2.1).

A screenshot of a terminal window with a dark background and light-colored text. The text shows a series of commands being entered at a prompt. The prompt is 'tamtrofanov@tamtrofanov-VirtualBox:~\$'. The commands are: 'mkdir ~/work/arch-pc/lab08', 'cd ~/work/arch-pc/lab08', and 'touch lab8-1.asm'. The prompt changes to '~/work/arch-pc/lab08\$' after the 'cd' command.

```
tamtrofanov@tamtrofanov-VirtualBox:~$  
tamtrofanov@tamtrofanov-VirtualBox:~$ mkdir ~/work/arch-pc/lab08  
tamtrofanov@tamtrofanov-VirtualBox:~$ cd ~/work/arch-pc/lab08  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ touch lab8-1.asm  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$
```

Рис. 2.1: Создание подкаталога *lab08* и файла *lab8-1.asm*

В файл *lab8-1.asm* внёс код программы из листинга 8.1(рис. 2.2). Скопмпелировал и проверил работу программы из файла *lab8-1.asm*(рис. 2.3).

Открыть ▾

+

*lab8-1.asm
~/work/arch-pc/lab08

Сохран

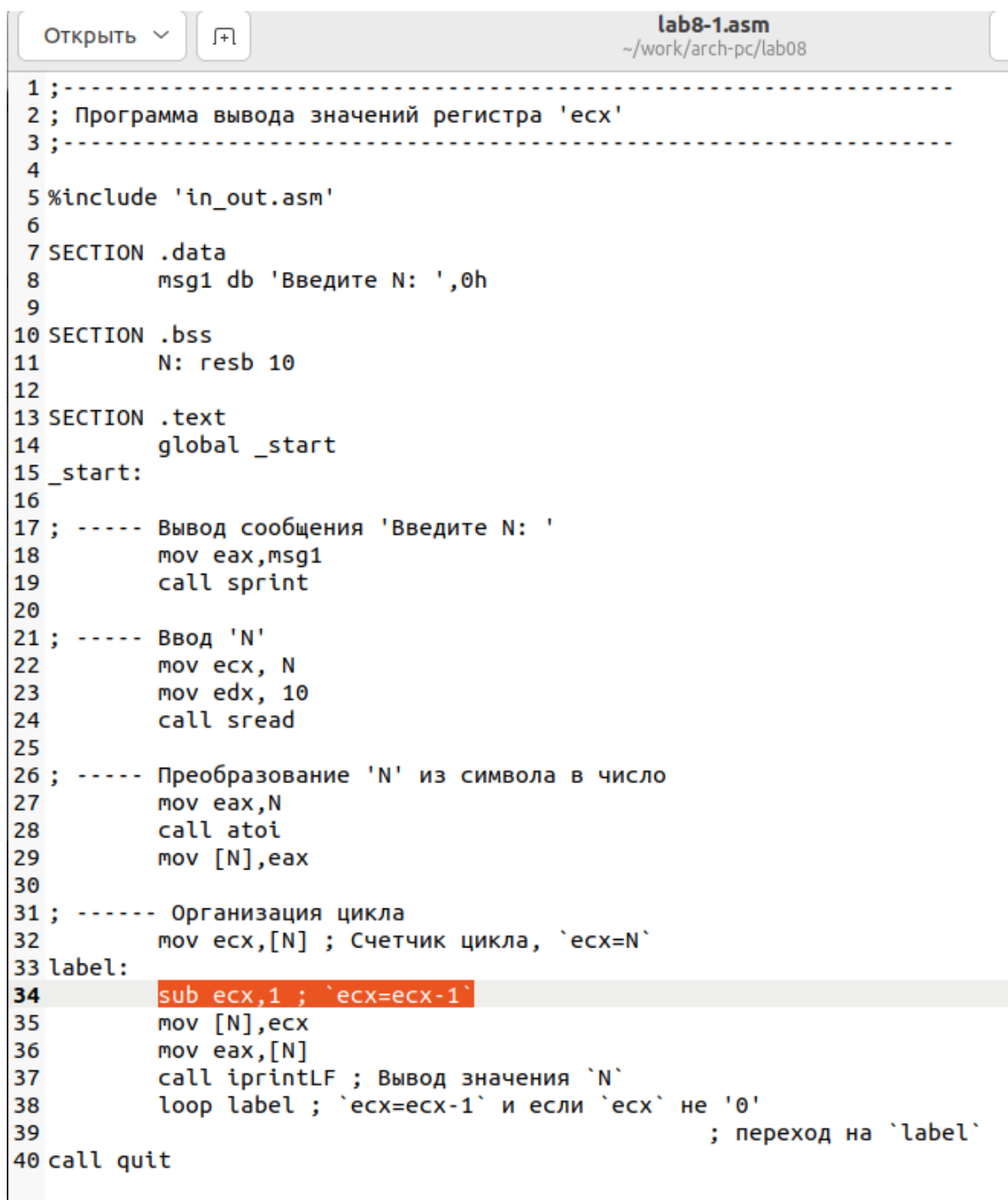
```
1 ;-----  
2 ; Программа вывода значений регистра 'ecx'  
3 ;-----  
4  
5 %include 'in_out.asm'  
6  
7 SECTION .data  
8     msg1 db 'Введите N: ',0h  
9  
10 SECTION .bss  
11     N: resb 10  
12  
13 SECTION .text  
14     global _start  
15 _start:  
16  
17 ; ----- Вывод сообщения 'Введите N: '  
18     mov eax,msg1  
19     call sprint  
20  
21 ; ----- Ввод 'N'  
22     mov ecx, N  
23     mov edx, 10  
24     call sread  
25  
26 ; ----- Преобразование 'N' из символа в число  
27     mov eax,N  
28     call atoi  
29     mov [N],eax  
30  
31 ; ----- Организация цикла  
32     mov ecx,[N] ; Счетчик цикла, `ecx=N`  
33 label:  
34     mov [N],ecx  
35     mov eax,[N]  
36     call iprintLF ; Вывод значения `N`  
37     loop label ; `ecx=ecx-1` и если `ecx` не `0`  
38                                     ; переход на `label`  
39 call quit
```

Рис. 2.2: Внесённый код в файл *lab8-1.asm*

```
tamtrofanov@tamtrofanov-VirtualBox: ~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ nasm -f elf lab8-1.asm  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ld -m elf_i386 -o lab8-1 lab8-1.o  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ./lab8-1  
Введите N: 5  
5  
4  
3  
2  
1
```

Рис. 2.3: Воспроизведение файла *lab8-1.asm*

Внёс необходимые изменения в файл *lab8-1.asm*(рис. 2.4). Скомпелировал и проверил работу изменённой программы из файла *lab8-1.asm*(рис. 2.5). В данном случае переход счётчика в цикле происходит через один, а соответственно и циклов в 2 раза меньше.



```
1 ;-----
2 ; Программа вывода значений регистра 'ecx'
3 ;-----
4
5 %include 'in_out.asm'
6
7 SECTION .data
8     msg1 db 'Введите N: ',0h
9
10 SECTION .bss
11     N: resb 10
12
13 SECTION .text
14     global _start
15 _start:
16
17 ; ----- Вывод сообщения 'Введите N: '
18     mov eax,msg1
19     call sprint
20
21 ; ----- Ввод 'N'
22     mov ecx, N
23     mov edx, 10
24     call sread
25
26 ; ----- Преобразование 'N' из символа в число
27     mov eax,N
28     call atoi
29     mov [N],eax
30
31 ; ----- Организация цикла
32     mov ecx,[N] ; Счетчик цикла, `ecx=N`
33 label:
34     sub ecx,1 ; `ecx=ecx-1`
35     mov [N],ecx
36     mov eax,[N]
37     call iprintLF ; Вывод значения `N`
38     loop label ; `ecx=ecx-1` и если `ecx` не '0'
39                                     ; переход на `label`
40 call quit
```

Рис. 2.4: Изменённый файл *lab8-1.asm*


```
tamtrofanov@tamtrofanov-VirtualBox: ~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ nasm -f elf lab8-1.asm  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ld -m elf_i386 -o lab8-1 lab8-1.o  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ./lab8-1  
Введите N: 6  
5  
3  
1  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$
```

Рис. 2.5: Воспроизведение файла *lab8-1.asm*

Внёс необходимые изменения в файл *lab8-1.asm*(рис. 2.6). Скомпелировал и проверил работу изменённой программы из файла *lab8-1.asm*(рис. 2.7). В данном случае число переходов соответствует числу введённому с клавиатуры.

Открыть ▾

lab8-1.asm
~/work/arch-pc/lab08

```
1 ;-----  
2 ; Программа вывода значений регистра 'ecx'  
3 ;-----  
4  
5 %include 'in_out.asm'  
6  
7 SECTION .data  
8     msg1 db 'Введите N: ',0h  
9  
10 SECTION .bss  
11     N: resb 10  
12  
13 SECTION .text  
14     global _start  
15 _start:  
16  
17 ; ----- Вывод сообщения 'Введите N: '  
18     mov eax,msg1  
19     call sprint  
20  
21 ; ----- Ввод 'N'  
22     mov ecx, N  
23     mov edx, 10  
24     call sread  
25  
26 ; ----- Преобразование 'N' из символа в число  
27     mov eax,N  
28     call atoi  
29     mov [N],eax  
30  
31 ; ----- Организация цикла  
32     mov ecx,[N] ; Счетчик цикла, `ecx=N`  
33 label:  
34     push ecx ; добавление значения ecx в стек  
35     sub ecx,1  
36     mov [N],ecx  
37     mov eax,[N]  
38     call iprintLF  
39     pop ecx ; извлечение значения ecx из стека  
40  
41     loop label  
42  
43     call quit
```

Рис. 2.6: Изменённый файл *lab8-1.asm*

```

tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ nasm -f elf lab8-1.asm
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ld -m elf_i386 -o lab8-1 lab8-1.o
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ./lab8-1
Введите N: 6
5
4
3
2
1
0
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$

```

Рис. 2.7: Воспроизведение файла *lab8-1.asm*

Создал файл *lab8-2.asm*(рис. 2.8).

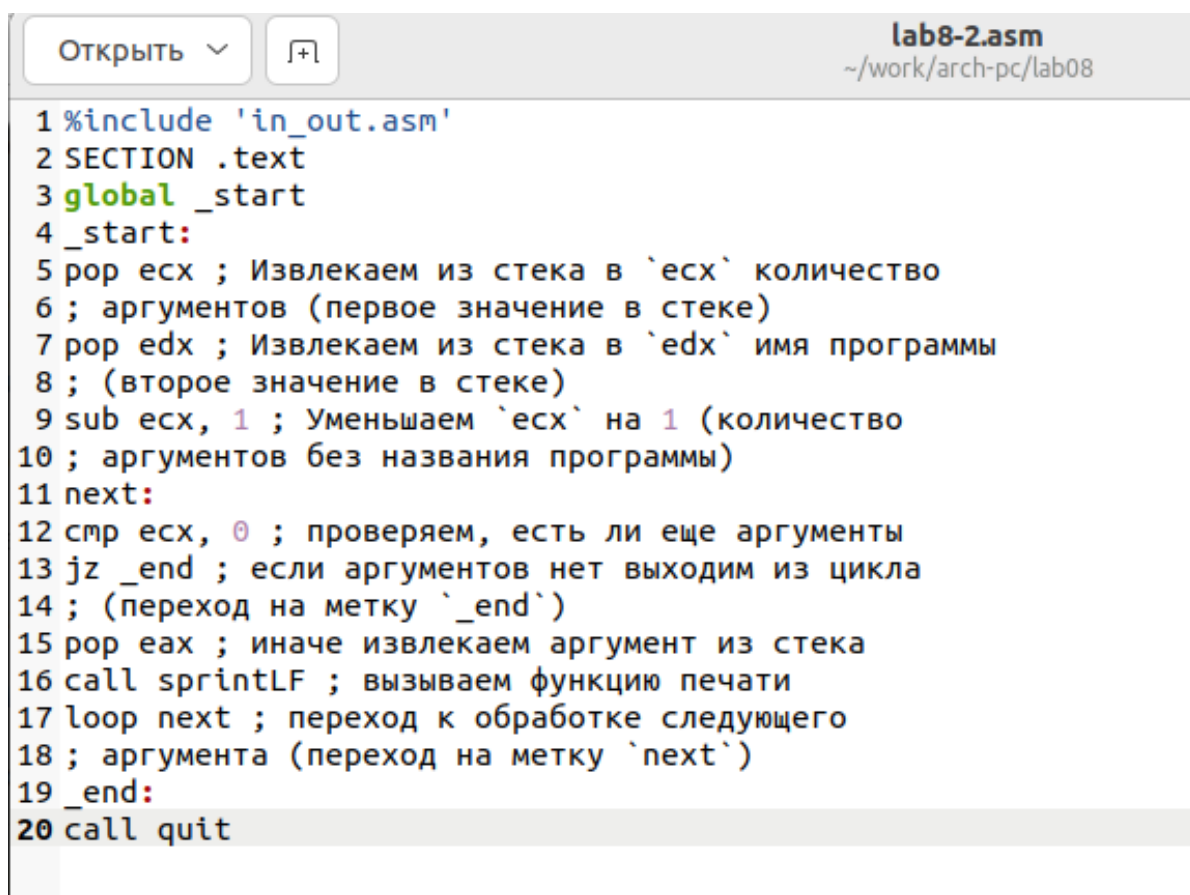
```

tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ touch lab8-2.asm
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$

```

Рис. 2.8: Создание файла *lab8-2.asm*

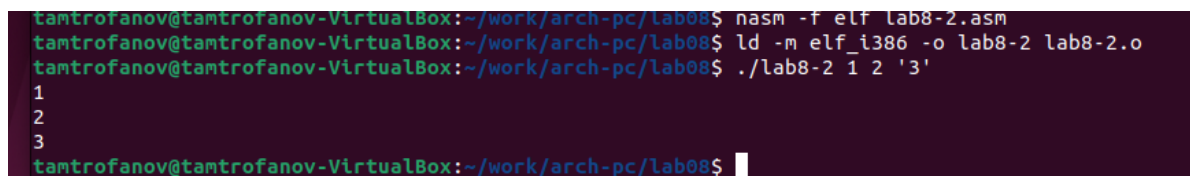
В файл *lab8-2.asm* внёс код программы из листинга 8.2(рис. 2.9). Скопмпелировал и проверил работу программы из файла *lab8-2.asm*(рис. 2.10). Программой были обработаны все 3 аргумента.



```
lab8-2.asm
~/work/arch-pc/lab08

1 %include 'in_out.asm'
2 SECTION .text
3 global _start
4 _start:
5 pop ecx ; Извлекаем из стека в `ecx` количество
6 ; аргументов (первое значение в стеке)
7 pop edx ; Извлекаем из стека в `edx` имя программы
8 ; (второе значение в стеке)
9 sub ecx, 1 ; Уменьшаем `ecx` на 1 (количество
10 ; аргументов без названия программы)
11 next:
12 cmp ecx, 0 ; проверяем, есть ли еще аргументы
13 jz _end ; если аргументов нет выходим из цикла
14 ; (переход на метку `_end`)
15 pop eax ; иначе извлекаем аргумент из стека
16 call sprintLF ; вызываем функцию печати
17 loop next ; переход к обработке следующего
18 ; аргумента (переход на метку `next`)
19 _end:
20 call quit
```

Рис. 2.9: Внесённый код в файл *lab8-2.asm*



```
tantrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ nasm -f elf lab8-2.asm
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ld -m elf_i386 -o lab8-2 lab8-2.o
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ./lab8-2 1 2 '3'
1
2
3
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$
```

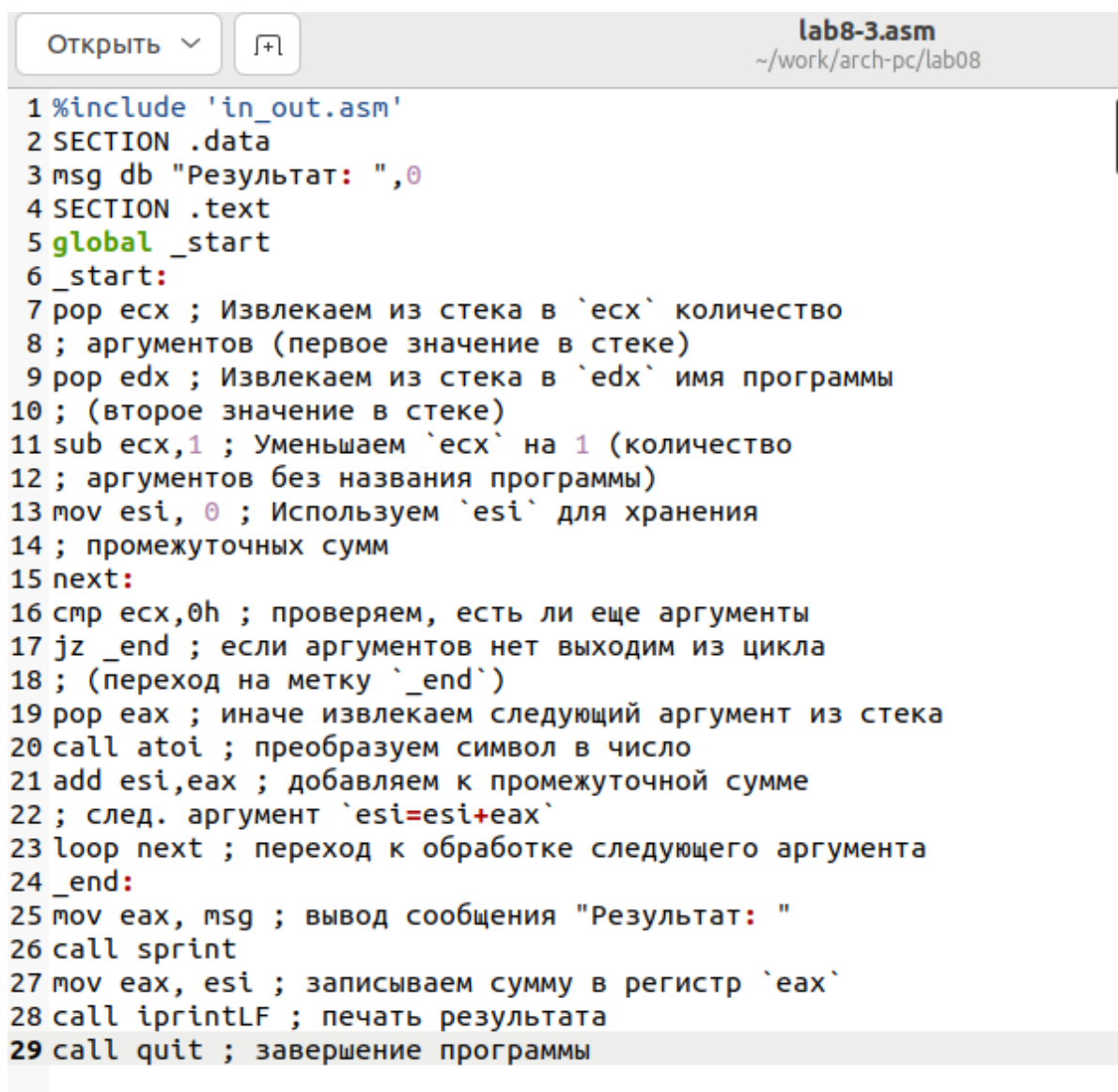
Рис. 2.10: Воспроизведение файла *lab8-2.asm*

Создал файл *lab8-3.asm*(рис. 2.11).

```
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ touch lab8-3.asm  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$
```

Рис. 2.11: Создание файла *lab8-3.asm*

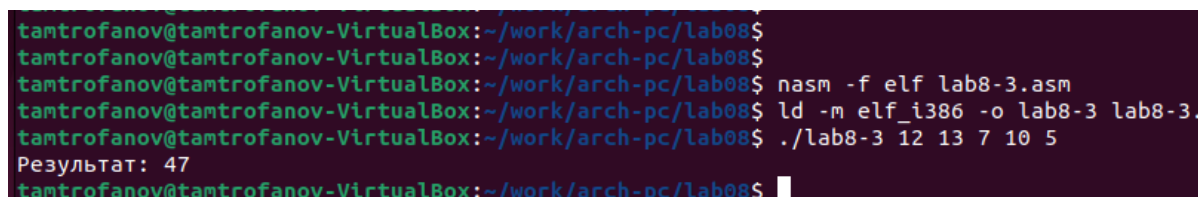
В файл *lab8-3.asm* внёс код программы из листинга 8.3(рис. 2.12). Скопмпелировал и проверил работу изменённый программы из файла *lab8-3.asm*(рис. 2.13).



```
lab8-3.asm
~/work/arch-pc/lab08

1 %include 'in_out.asm'
2 SECTION .data
3 msg db "Результат: ",0
4 SECTION .text
5 global _start
6 _start:
7 pop ecx ; Извлекаем из стека в `ecx` количество
8 ; аргументов (первое значение в стеке)
9 pop edx ; Извлекаем из стека в `edx` имя программы
10 ; (второе значение в стеке)
11 sub ecx,1 ; Уменьшаем `ecx` на 1 (количество
12 ; аргументов без названия программы)
13 mov esi, 0 ; Используем `esi` для хранения
14 ; промежуточных сумм
15 next:
16 cmp ecx,0h ; проверяем, есть ли еще аргументы
17 jz _end ; если аргументов нет выходим из цикла
18 ; (переход на метку `_end`)
19 pop eax ; иначе извлекаем следующий аргумент из стека
20 call atoi ; преобразуем символ в число
21 add esi,eax ; добавляем к промежуточной сумме
22 ; след. аргумент `esi=esi+eax`
23 loop next ; переход к обработке следующего аргумента
24 _end:
25 mov eax, msg ; вывод сообщения "Результат: "
26 call sprint
27 mov eax, esi ; записываем сумму в регистр `eax`
28 call iprintLF ; печать результата
29 call quit ; завершение программы
```

Рис. 2.12: Внесённый код в файл *lab8-3.asm*

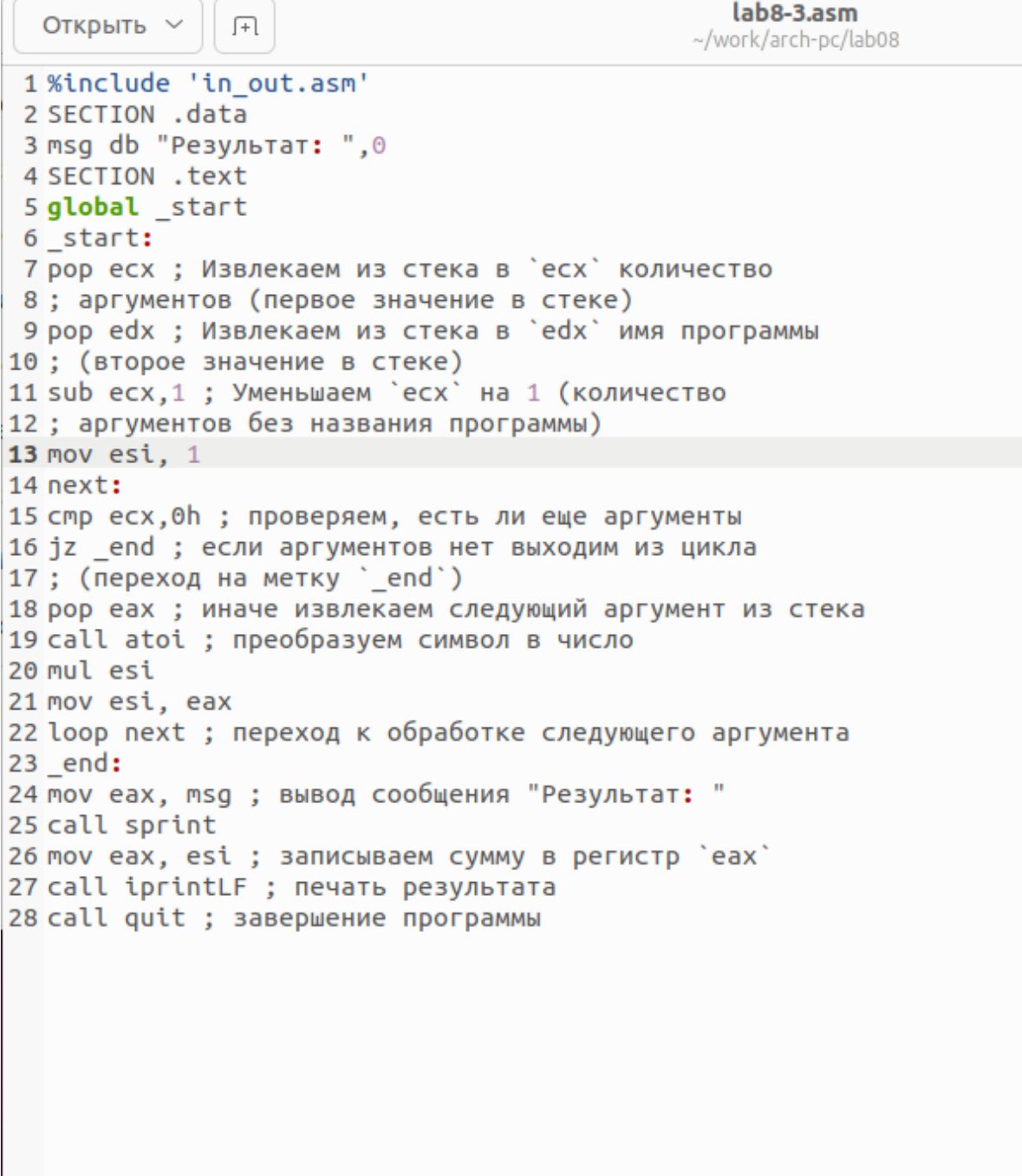


```
tantrofanov@tantrofanov-VirtualBox:~/work/arch-pc/lab08$
tantrofanov@tantrofanov-VirtualBox:~/work/arch-pc/lab08$
tantrofanov@tantrofanov-VirtualBox:~/work/arch-pc/lab08$ nasm -f elf lab8-3.asm
tantrofanov@tantrofanov-VirtualBox:~/work/arch-pc/lab08$ ld -m elf_i386 -o lab8-3 lab8-3.o
tantrofanov@tantrofanov-VirtualBox:~/work/arch-pc/lab08$ ./lab8-3 12 13 7 10 5
Результат: 47
tantrofanov@tantrofanov-VirtualBox:~/work/arch-pc/lab08$
```

Рис. 2.13: Воспроизведение файла *lab8-3.asm*

Внес необходимые изменения в файл *lab8-3.asm* соответствии с заданием(рис.

2.14). Скопмпелировал и проверил работу изменённой программы из файла *lab8-3.asm* (рис. 2.15).



```
1 %include 'in_out.asm'
2 SECTION .data
3 msg db "Результат: ",0
4 SECTION .text
5 global _start
6 _start:
7 pop ecx ; Извлекаем из стека в `ecx` количество
8 ; аргументов (первое значение в стеке)
9 pop edx ; Извлекаем из стека в `edx` имя программы
10 ; (второе значение в стеке)
11 sub ecx,1 ; Уменьшаем `ecx` на 1 (количество
12 ; аргументов без названия программы)
13 mov esi, 1
14 next:
15 cmp ecx,0h ; проверяем, есть ли еще аргументы
16 jz _end ; если аргументов нет выходим из цикла
17 ; (переход на метку `_end`)
18 pop eax ; иначе извлекаем следующий аргумент из стека
19 call atoi ; преобразуем символ в число
20 mul esi
21 mov esi, eax
22 loop next ; переход к обработке следующего аргумента
23 _end:
24 mov eax, msg ; вывод сообщения "Результат: "
25 call sprint
26 mov eax, esi ; записываем сумму в регистр `eax`
27 call iprintLF ; печать результата
28 call quit ; завершение программы
```

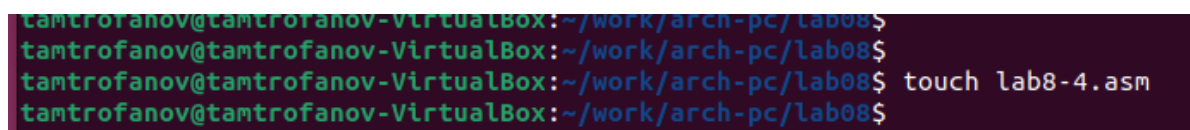
Рис. 2.14: Изменённый файл *lab8-3.asm*

```
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ nasm -f elf lab8-3.asm  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ld -m elf_i386 -o lab8-3 lab8-3.o  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ./lab8-3 2 5 10  
Результат: 100  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$
```

Рис. 2.15: Воспроизведение файла *lab8-3.asm*

3 Задание для самостоятельной работы

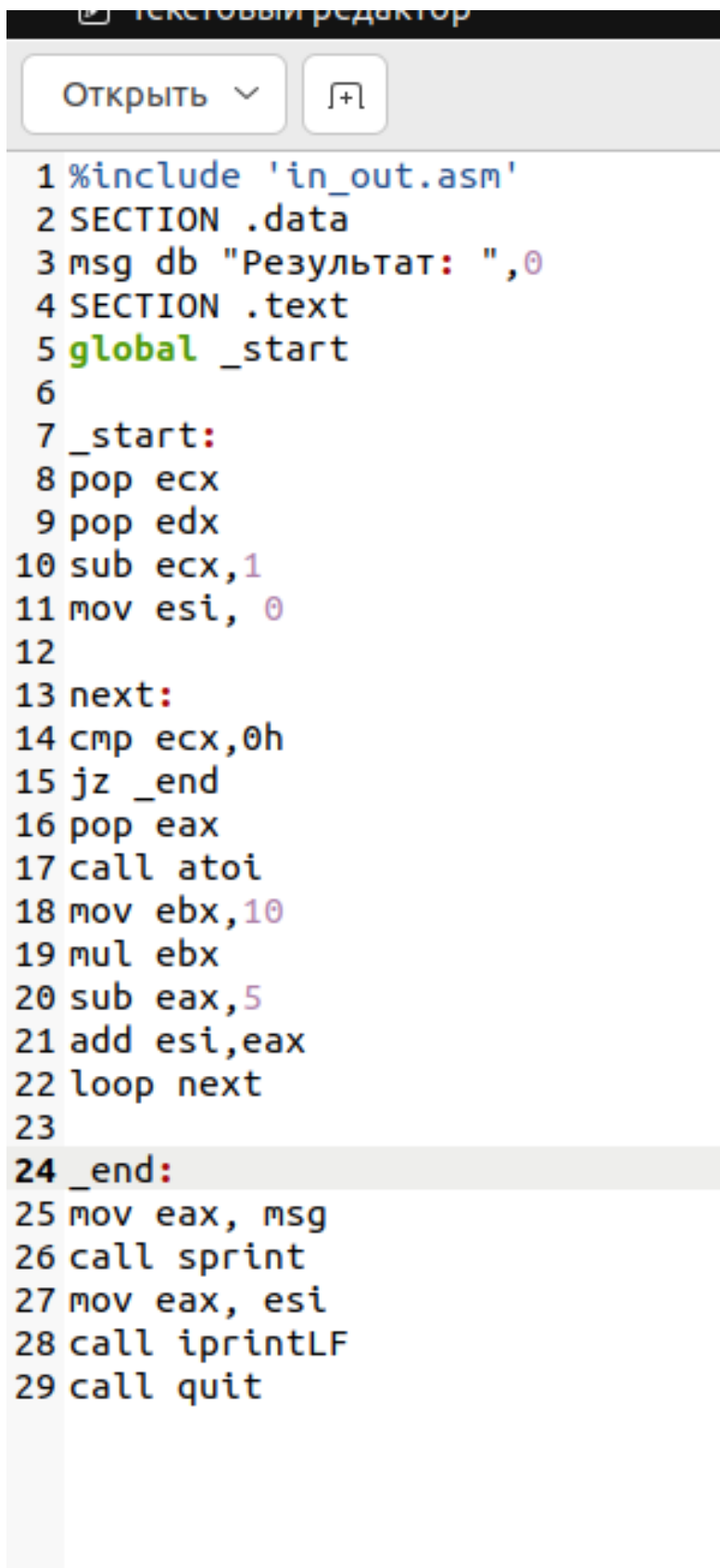
Создал файл *lab8-4.asm*(рис. 3.1).



```
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ touch lab8-4.asm  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$
```

Рис. 3.1: Создание файла *lab8-4.asm*

В файл *lab8-4.asm* внёс код соответствующий формуле моего варианта (3 вариант)(рис. 3.2). Скомпилировал и проверил работу программы из файла *lab8-4.asm*(рис. 3.3).



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

Рис. 3.2: Внесённый код в файл *lab8-4.asm*

```

#include 'in_out.asm'

SECTION .data
msg db "Результат: ",0

SECTION .text
global _start

_start:
pop ecx
pop edx
sub ecx,1
mov esi, 0

next:
cmp ecx,0h
jz _end
pop eax
call atoi
mov ebx,10
mul ebx
sub eax,5
add esi,eax
loop next

_end:
mov eax, msg
call sprint
mov eax, esi
call iprintLF
call quit

```

```
tamtrofanov@tamtrofanov-VirtualBox: ~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ nasm -f elf lab8-4.asm  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ld -m elf_i386 -o lab8-4 lab8-4.o  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ./lab8-4 1 2 3 4  
Результат: 80  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$ ./lab8-4 5 6 7 8  
Результат: 240  
tamtrofanov@tamtrofanov-VirtualBox:~/work/arch-pc/lab08$
```

Рис. 3.3: Воспроизведение файла *lab8-4.asm*

4 Выводы

Сегодня я приобрёл навыки написания программ с использованием циклов и обработкой аргументов командной строки.