

# Отчёт по лабораторной работе №9

## Дисциплина: Архитектура компьютера

Барсегян Вардан Левонович НПИбд-01-22

### Содержание

Цель работы.....	2
Выполнение лабораторной работы.....	3
Реализация циклов в NASM.....	3
Обработка аргументов командной строки.....	5
Задание для самостоятельной работы.....	10
Выводы.....	13

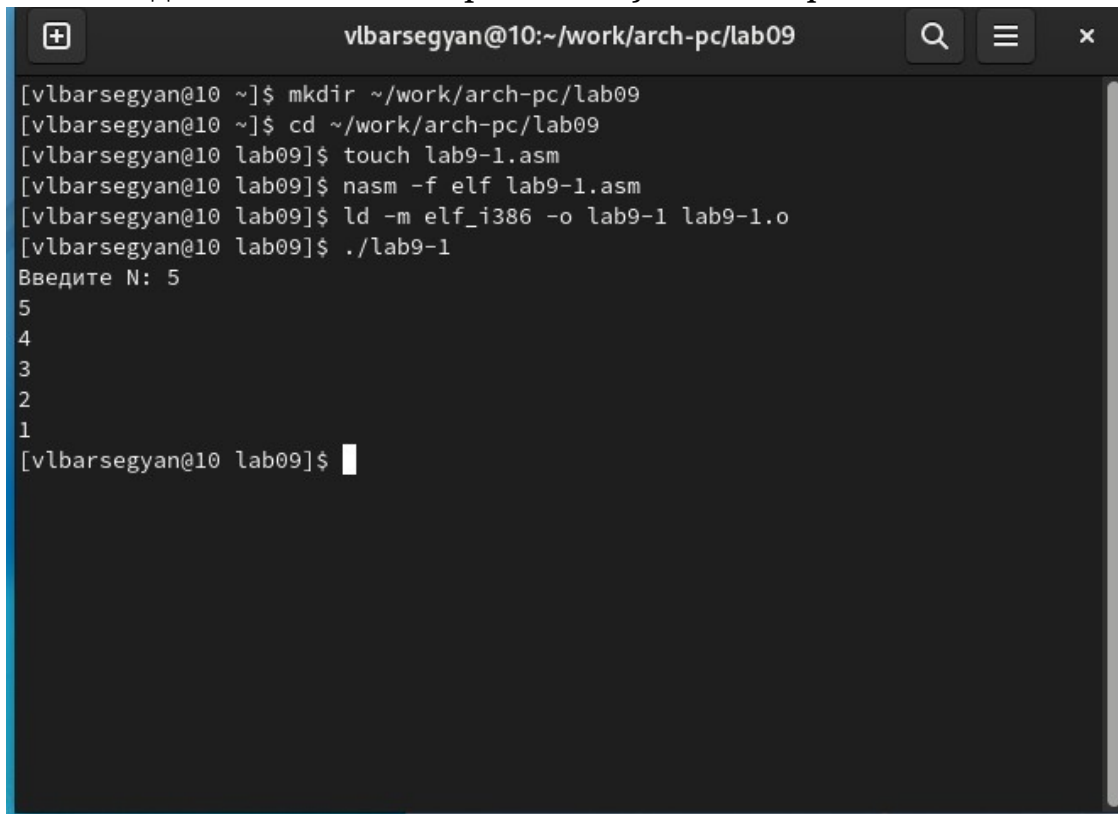
## Цель работы

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

## Выполнение лабораторной работы

### Реализация циклов в NASM

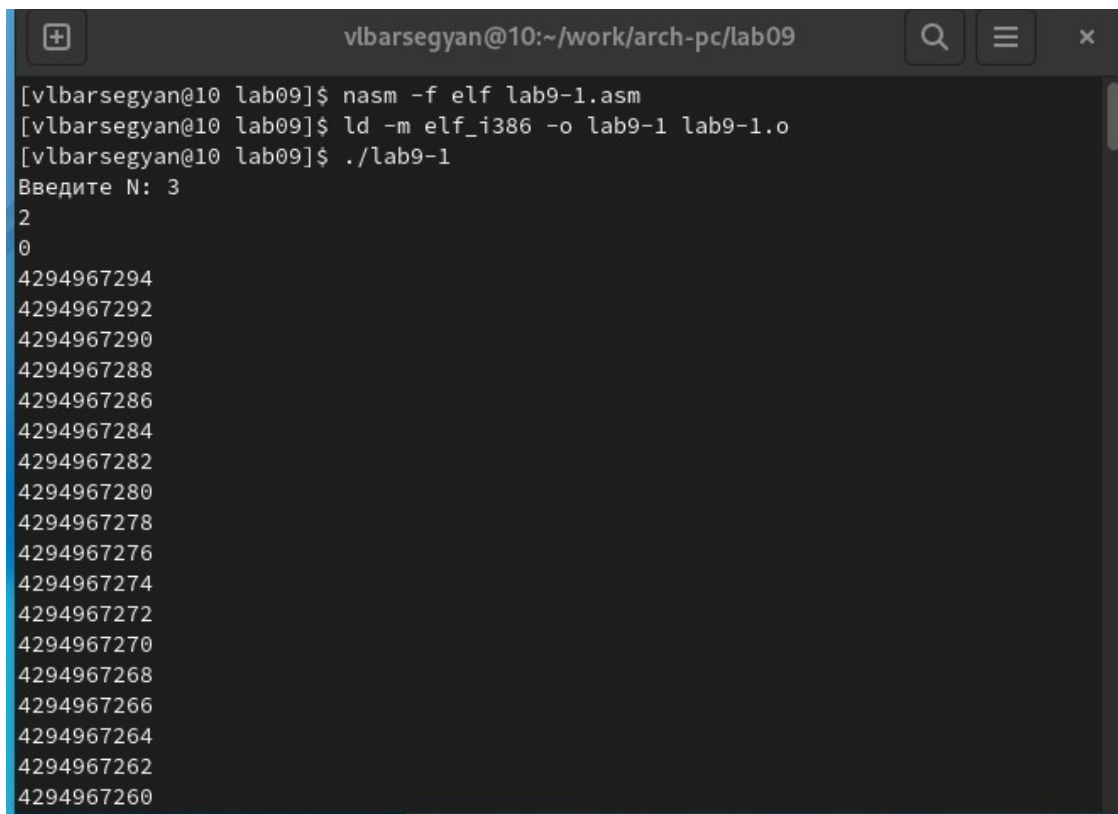
1. Создаю каталог для программ лабораторной работы № 9, перехожу в него и создаю файл *lab9-1.asm*. В этот файл вставляю текст программы, создаю исполняемый файл и запускаю его (рис. 1)



```
vlbarsegyan@10:~/work/arch-pc/lab09
[vlbarsegyan@10 ~]$ mkdir ~/work/arch-pc/lab09
[vlbarsegyan@10 ~]$ cd ~/work/arch-pc/lab09
[vlbarsegyan@10 lab09]$ touch lab9-1.asm
[vlbarsegyan@10 lab09]$ nasm -f elf lab9-1.asm
[vlbarsegyan@10 lab09]$ ld -m elf_i386 -o lab9-1 lab9-1.o
[vlbarsegyan@10 lab09]$ ./lab9-1
Введите N: 5
5
4
3
2
1
[vlbarsegyan@10 lab09]$
```

Рис. 1: Создание каталога, файла, запуск исполняемого файла

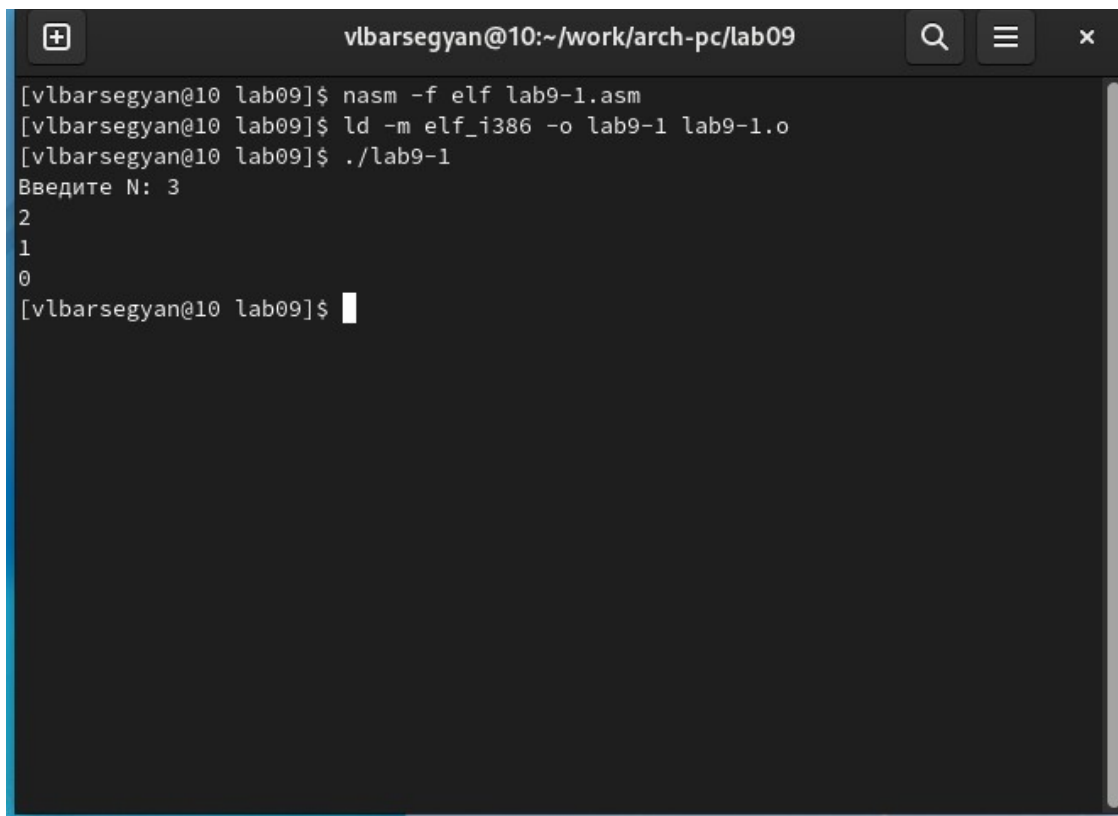
2. Изменяю текст программы и запускаю исполняемый файл (рис. 2). Теперь программа работает некорректно, и число проходов цикла не соответствует значению *N*, введенному с клавиатуры



```
vlbarsegyan@10:~/work/arch-pc/lab09
[vlbarsegyan@10 lab09]$ nasm -f elf lab9-1.asm
[vlbarsegyan@10 lab09]$ ld -m elf_i386 -o lab9-1 lab9-1.o
[vlbarsegyan@10 lab09]$ ./lab9-1
Введите N: 3
2
0
4294967294
4294967292
4294967290
4294967288
4294967286
4294967284
4294967282
4294967280
4294967278
4294967276
4294967274
4294967272
4294967270
4294967268
4294967266
4294967264
4294967262
4294967260
```

Рис. 2: Изменение файла и запуск исполняемого файла

3. Изменяю текст программы и запускаю исполняемый файл (рис. 3). Теперь число проходов цикла соответствует значению  $N$ , введенному с клавиатуры

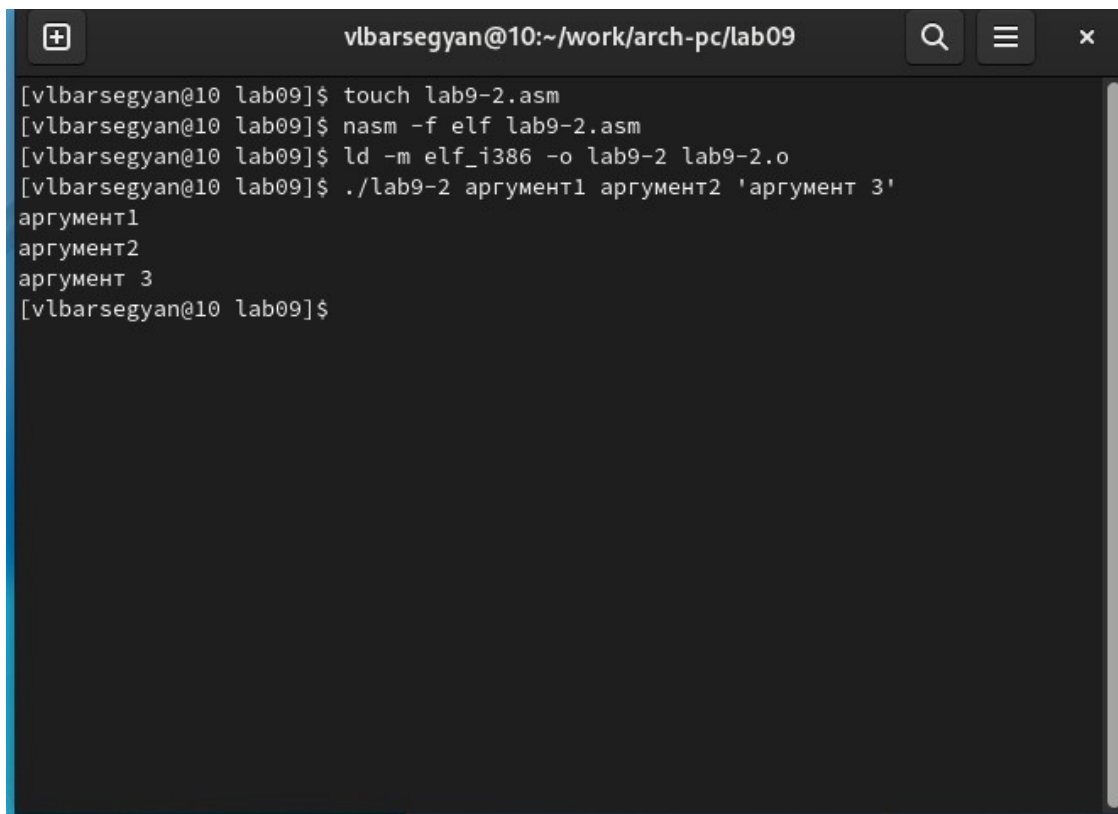


```
vlbarsegyan@10:~/work/arch-pc/lab09
[vlbarsegyan@10 lab09]$ nasm -f elf lab9-1.asm
[vlbarsegyan@10 lab09]$ ld -m elf_i386 -o lab9-1 lab9-1.o
[vlbarsegyan@10 lab09]$ ./lab9-1
Введите N: 3
2
1
0
[vlbarsegyan@10 lab09]$
```

Рис. 3: Изменение файла и запуск исполняемого файла

#### Обработка аргументов командной строки

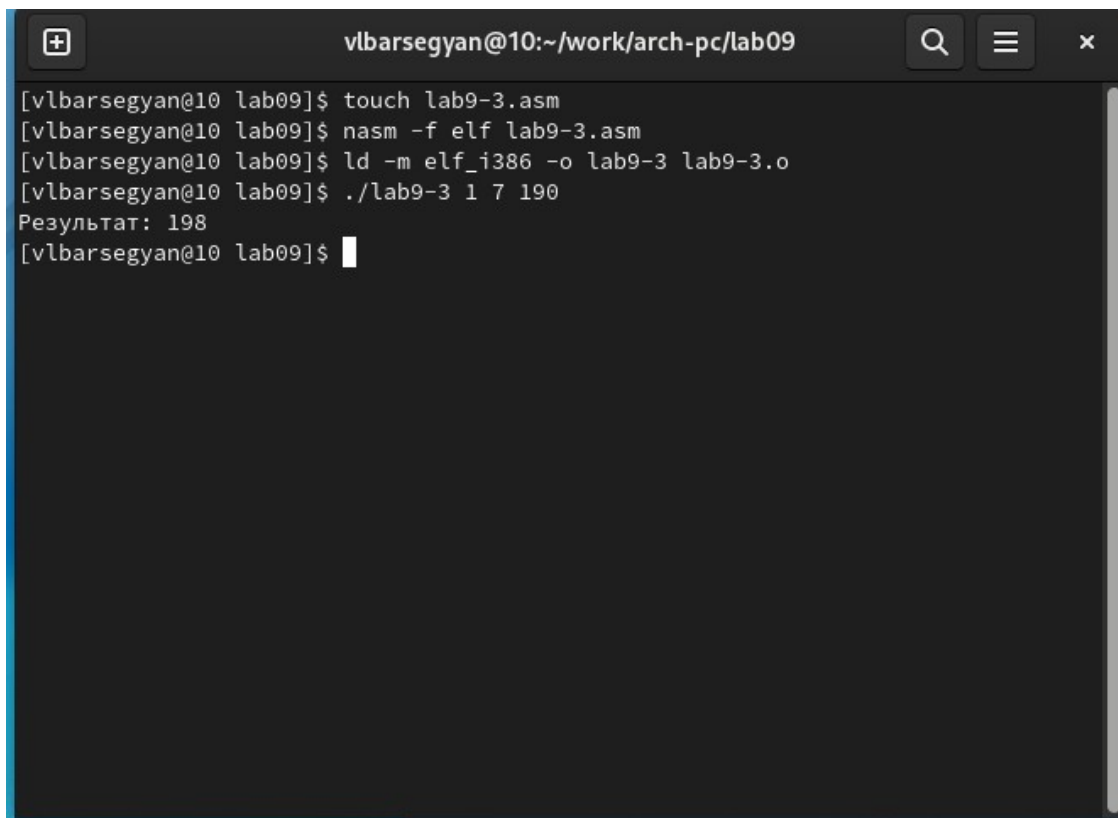
4. Создаю файл lab9-2.asm в каталоге `~/work/arch-pc/lab09` и ввожу в него текст программы из листинга 9.2. Запускаю исполняемый файл с указанием аргументов (рис. 4). Программа обработала поочередно все аргументы

A terminal window with a dark background and light text. The window title bar shows the user 'vlbarsegyan@10' and the directory '~/work/arch-pc/lab09'. The terminal contains the following commands and output:

```
[vlbarsegyan@10 lab09]$ touch lab9-2.asm
[vlbarsegyan@10 lab09]$ nasm -f elf lab9-2.asm
[vlbarsegyan@10 lab09]$ ld -m elf_i386 -o lab9-2 lab9-2.o
[vlbarsegyan@10 lab09]$ ./lab9-2 аргумент1 аргумент2 'аргумент 3'
аргумент1
аргумент2
аргумент 3
[vlbarsegyan@10 lab09]$
```

Рис. 4: Создание нового файла, запуск исполняемого файла

5. Создаю файл lab9-3.asm в каталоге `~/work/arch-pc/lab09` и ввожу в него текст программы из листинга 9.3. Запускаю исполняемый файл с указанием аргументов (рис. 5). Программа вывела сумму всех введенных аргументов

A terminal window with a dark background and light text. The title bar at the top shows the user 'vlbarsegyan@10' and the directory '~/work/arch-pc/lab09'. There are icons for search, menu, and close on the right. The terminal content shows a series of commands: 'touch lab9-3.asm', 'nasm -f elf lab9-3.asm', 'ld -m elf\_i386 -o lab9-3 lab9-3.o', and './lab9-3 1 7 190'. The output of the last command is 'Результат: 198'. The prompt is '[vlbarsegyan@10 lab09]\$' followed by a cursor.

```
[vlbarsegyan@10 lab09]$ touch lab9-3.asm
[vlbarsegyan@10 lab09]$ nasm -f elf lab9-3.asm
[vlbarsegyan@10 lab09]$ ld -m elf_i386 -o lab9-3 lab9-3.o
[vlbarsegyan@10 lab09]$ ./lab9-3 1 7 190
Результат: 198
[vlbarsegyan@10 lab09]$
```

*Рис. 5: Создание нового файла, запуск исполняемого файла*

6. Изменяю текст программы из листинга 9.3 для вычисления произведения аргументов командной строки (рис. 6). Запускаю исполняемый файл и проверяю его работу (рис. 7)

```
%include 'in_out.asm'
```

```
SECTION .data
```

```
msg db "Результат: ",0
```

```
SECTION .text
```

```
global _start
```

```
_start:
```

```
pop ecx ; Извлекаем из стека в `ecx` количество  
; аргументов (первое значение в стеке)
```

```
pop edx ; Извлекаем из стека в `edx` имя программы  
; (второе значение в стеке)
```

```
sub ecx,1 ; Уменьшаем `ecx` на 1 (количество  
; аргументов без названия программы)
```

```
mov esi, 1 ; Используем `esi` для хранения произведения
```

```
next:
```

```
cmp ecx,0h ; проверяем, есть ли еще аргументы  
jz _end ; если аргументов нет выходим из цикла  
; (переход на метку `_end`)
```

```
pop eax ; иначе извлекаем следующий аргумент из стека  
call atoi ; преобразуем символ в число
```

```
mul esi; умножаем к промежуточному произведению  
mov esi, eax;
```

```
loop next ; переход к обработке следующего аргумента
```

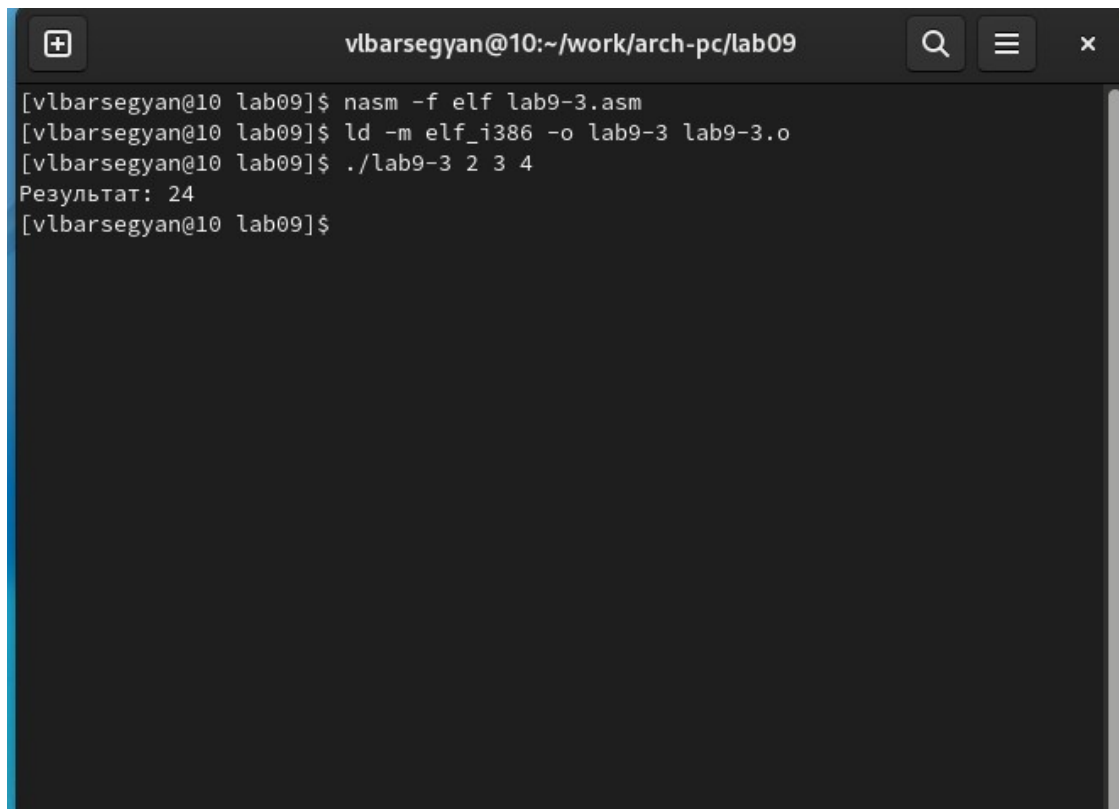
```
_end:
```

```
mov eax, msg ; вывод сообщения "Результат: "  
call sprint
```

```
mov eax, esi ; записываем сумму в регистр `eax`  
call iprintLF ; печать результата  
call quit ; завершение программы
```



*Рис. 6: Текст новой программы*

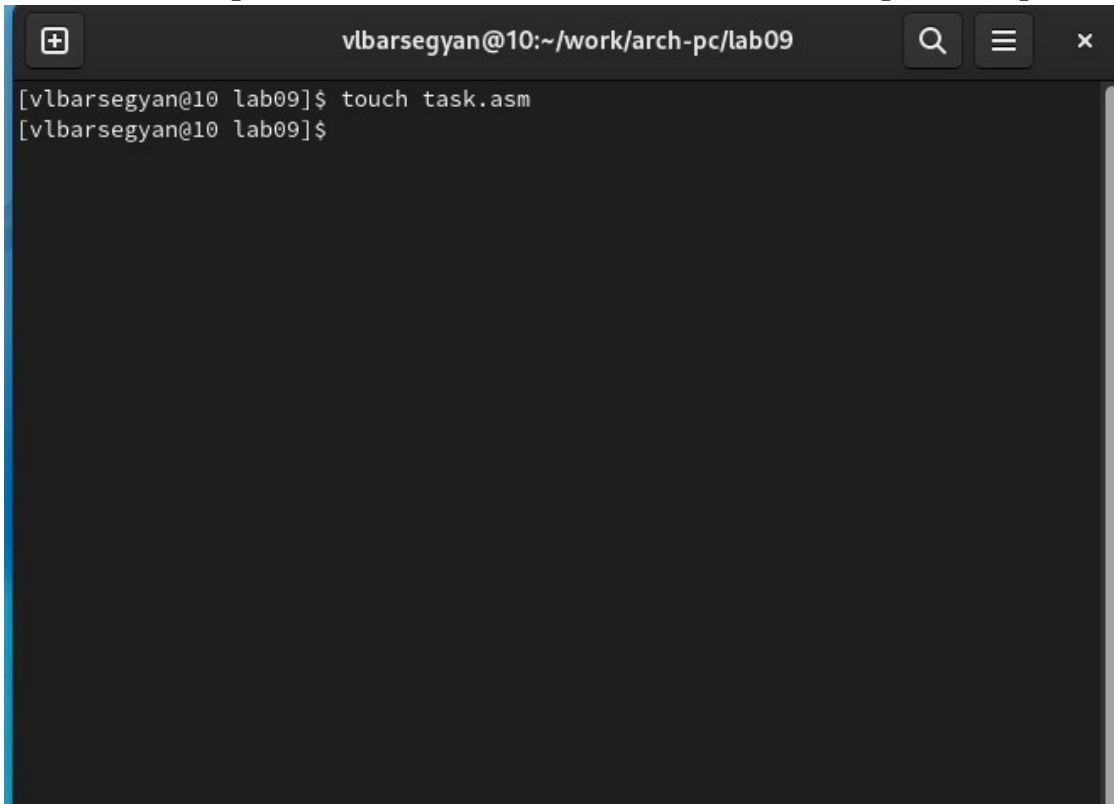


```
vlbarsegyan@10:~/work/arch-pc/lab09
[vlbarsegyan@10 lab09]$ nasm -f elf lab9-3.asm
[vlbarsegyan@10 lab09]$ ld -m elf_i386 -o lab9-3 lab9-3.o
[vlbarsegyan@10 lab09]$ ./lab9-3 2 3 4
Результат: 24
[vlbarsegyan@10 lab09]$
```

*Рис. 7: Запуск исполняемого файла*

## Задание для самостоятельной работы

1. Создаю файл для выполнения самостоятельной работы (рис. 8)

A terminal window with a dark background. The title bar shows the user 'vlbarsegyan@10' and the directory '~/work/arch-pc/lab09'. There are icons for search, menu, and close on the right. The terminal shows two lines of text: '[vlbarsegyan@10 lab09]\$ touch task.asm' followed by a new line '[vlbarsegyan@10 lab09]\$'.

```
vlbarsegyan@10:~/work/arch-pc/lab09
[vlbarsegyan@10 lab09]$ touch task.asm
[vlbarsegyan@10 lab09]$
```

Рис. 8: Создания файла для С/Р

2. Пишу программу для выполнения варианта 6 (рис. 9)

```

%include 'in_out.asm'

SECTION .data
func db "Функция:  $f(x) = 4x - 3$ ", 0
msg db "Результат: ", 0

SECTION .text
global _start

_start:
mov eax, func
call sprintLF
pop ecx
pop edx
sub ecx, 1
mov esi, 4 ; Используем `esi` как множитель для x
mov edi, 0 ; в edi храним общую сумму

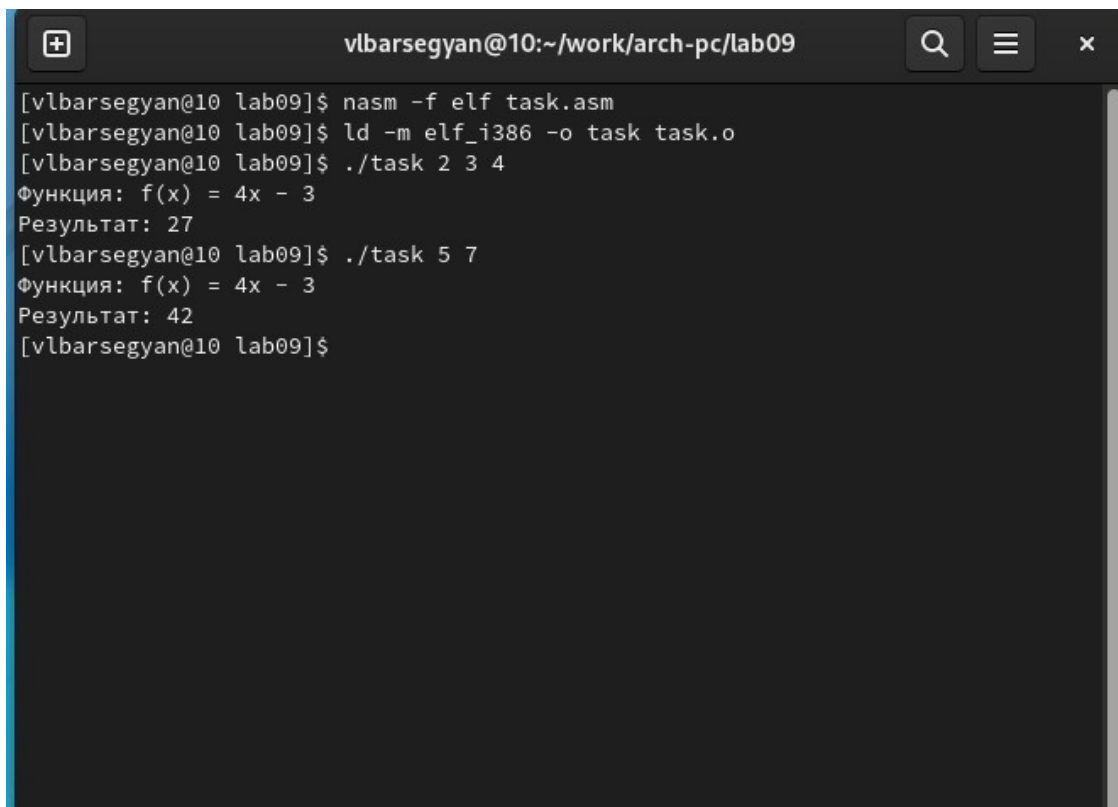
next:
cmp ecx, 0h
jz _end
pop eax
call atoi
mul esi
sub eax, 3
add edi, eax
loop next

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

```

Рис. 9: Текст программы для C/P

3. Создаю исполняемый файл и проверяю корректность работы программы (рис. 10)



A terminal window with a dark background and light gray text. The window title bar shows the user 'vlbarsegyan@10' and the directory '~/work/arch-pc/lab09'. The terminal contains the following commands and output:

```
[vlbarsegyan@10 lab09]$ nasm -f elf task.asm
[vlbarsegyan@10 lab09]$ ld -m elf_i386 -o task task.o
[vlbarsegyan@10 lab09]$ ./task 2 3 4
Функция:  $f(x) = 4x - 3$ 
Результат: 27
[vlbarsegyan@10 lab09]$ ./task 5 7
Функция:  $f(x) = 4x - 3$ 
Результат: 42
[vlbarsegyan@10 lab09]$
```

*Рис. 10: Запуск исполняемого файла и его проверка*

## Выводы

Я приобрел навыки написания программ с использованием циклов и обработкой аргументов командной строки. Написал программу для вычисления суммы значений от заданной функции, в которой аргументы вводятся с командной строки