# Отчёта по лабораторной работе №6

**Арифметические операции в NASM.**

Гурбанов Сарча

# Содержание

1. [Цель работы](#_bookmark0) 4
2. [Задание](#_bookmark1) 5
3. [Выполнение лабораторной работы](#_bookmark2) 6
   1. [Cимвольные и численные данные в NASM](#_bookmark2) . . . . . . . . . . . . . 6
   2. [Выполнение арифметических операций в NASM](#_bookmark7) 11
   3. [Ответы на вопросы по программе](#_bookmark10) 14
   4. [Задание для самостоятельной работы](#_bookmark11) 15
4. [Выводы](#_bookmark14) 18

**Список иллюстраций**

1. [Создаем каталог с помощью команды mkdir и файл с помощью](#_bookmark2) [команды touch](#_bookmark2) . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
2. [Заполняем файл](#_bookmark3) . . . . . . . . . . . . . . . . . . . . . . . . . . . 7
3. [Запускаем файл и смотрим на его работу](#_bookmark3) . . . . . . . . . . . . . . 7
4. [Изменяем файл](#_bookmark4) . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
5. [Запускаем файл и смотрим на его работу](#_bookmark4) . . . . . . . . . . . . . . 8
6. [Создаем файл](#_bookmark4) . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
7. [Заполняем файл](#_bookmark5) . . . . . . . . . . . . . . . . . . . . . . . . . . . 9
8. [Смотрим на работу программы](#_bookmark5) . . . . . . . . . . . . . . . . . . . 9
9. [Изменяем файл](#_bookmark6) 10
10. [Смотрим на работу программы](#_bookmark6) 10
11. [Изменяем файл](#_bookmark7) 11
12. [Смотрим на работу программы](#_bookmark7) 11
13. [Создаем файл](#_bookmark7) 11
14. [Заполняем файл](#_bookmark8) 12
15. [Смотрим на результат работы программы](#_bookmark8) 12
16. [Редактируем файл](#_bookmark9) . . . . . . . . . . . . . . . . . . . . . . . . . . 13
17. [Смотрим на результат работы программы](#_bookmark9) 13
18. [Создаем файл](#_bookmark9) 13
19. [Заполняем файл](#_bookmark10) 14
20. [Проверяемс результат работы программы](#_bookmark10) 14
21. [Создаем файл](#_bookmark11) 15
22. [Заполняем файл](#_bookmark12) 16
23. [Проверяем работу программы](#_bookmark12) 16
24. [Проверяем работу программы](#_bookmark13) 17

# Цель работы

Освоить арифметических инструкций языка ассемблера NASM и написать программы для вычисления арифметических выражений с неизвестной.

# Задание

Написать программы для решения выражений.

# Выполнение лабораторной работы

## Cимвольные и численные данные в NASM

Создаем каталог для программ ЛБ6, и в нем создаем файл (рис. [3.1).](#_bookmark2)

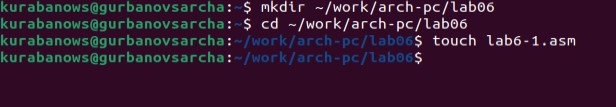


Рис. 3.1: Создаем каталог с помощью команды mkdir и файл с помощью команды touch

Открываем файл в Midnight Commander и заполняем его в соответствии с листингом 6.1 (рис. [3.2).](#_bookmark3)

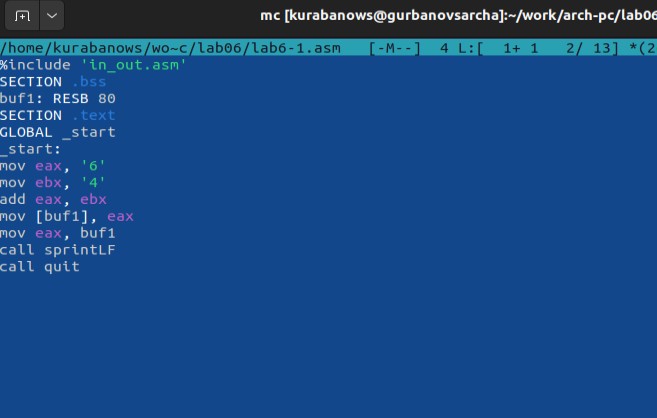


Рис. 3.2: Заполняем файл

Создаем исполняемый файл и запускаем его (рис. [3.3).](#_bookmark3)

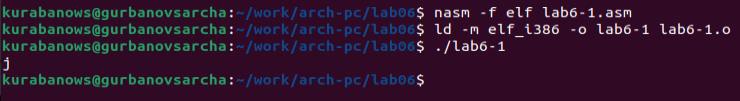


Рис. 3.3: Запускаем файл и смотрим на его работу

Снова открываем файл для редактирования и убиравем кавычки с числовых

значений (рис. [3.4).](#_bookmark4)

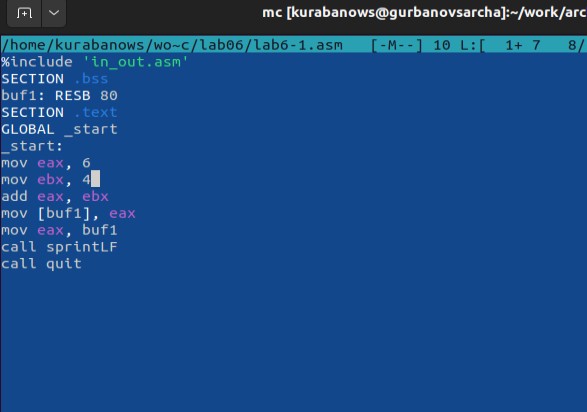


Рис. 3.4: Изменяем файл

Создаем исполняемый файл и запускаем его (рис. [3.5).](#_bookmark4)

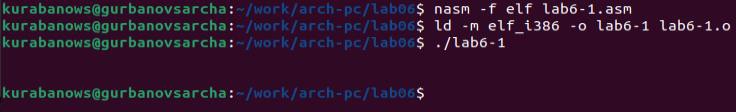


Рис. 3.5: Запускаем файл и смотрим на его работу

Создаем новый файл в каталоге (рис. [3.6).](#_bookmark4)



Рис. 3.6: Создаем файл

Заполняем файл в соответствии с листингом 6.2 (рис. [3.7).](#_bookmark5)

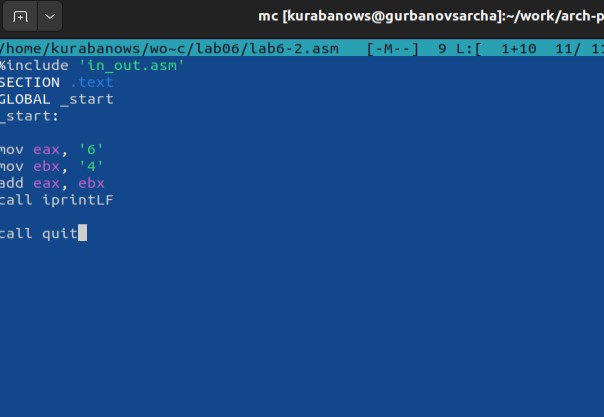


Рис. 3.7: Заполняем файл

Создаем исполняемый файл и запускаем его (рис. [3.8).](#_bookmark5)

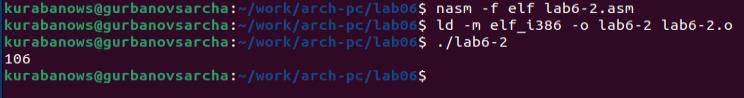


Рис. 3.8: Смотрим на работу программы

Снова открываем файл для редактирования и убиравем кавычки с числовых

значений (рис. [3.9).](#_bookmark6)

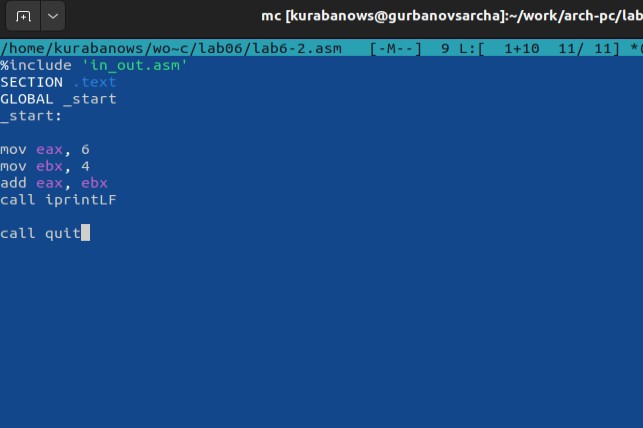


Рис. 3.9: Изменяем файл

Создаем исполняемый файл и запускаем его (рис. [3.10).](#_bookmark6)

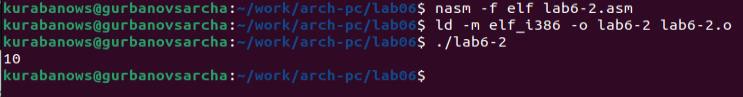


Рис. 3.10: Смотрим на работу программы

Снова открываем файл для редактирования и меняем iprintLF на iprint (рис.

[3.11).](#_bookmark7)

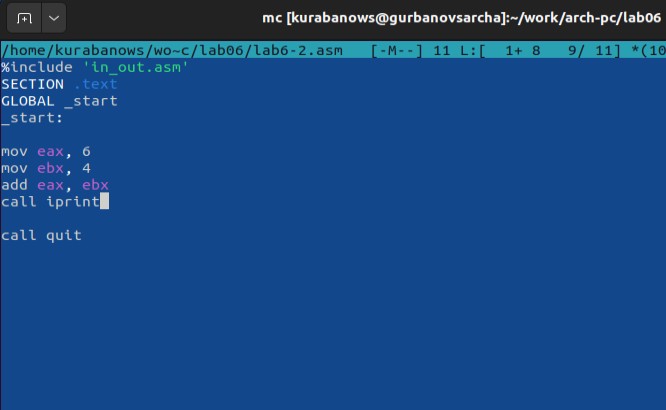


Рис. 3.11: Изменяем файл

Создаем исполняемый файл и запускаем его (рис. [3.12).](#_bookmark7)

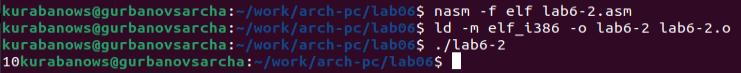


Рис. 3.12: Смотрим на работу программы

Вывод функций iprintLF и iprint отличаются только тем, что LF переносит на новую строку.

## Выполнение арифметических операций в NASM

Создаем новый файл в каталоге (рис. [3.13).](#_bookmark7)



Рис. 3.13: Создаем файл

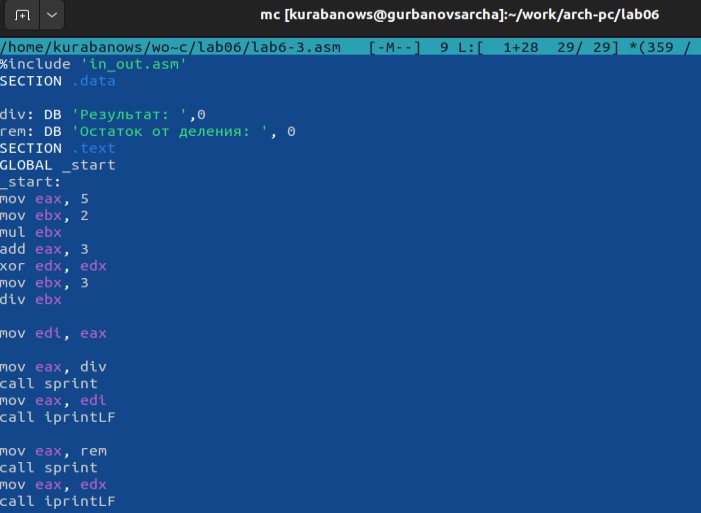
Открываем файл и редактируем в соответствии с листингом 6.3 (рис. [3.14).](#_bookmark8)

Рис. 3.14: Заполняем файл

Создаем исполняемый файл и запускаем его (рис. [3.15).](#_bookmark8)

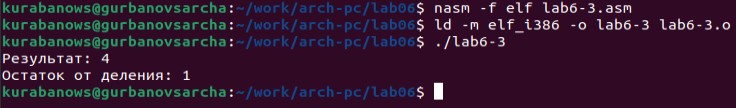


Рис. 3.15: Смотрим на результат работы программы

Открываем файл и редактируем его для вычисления выражения f(�) = (4 �6 + 2)/5 (рис. [3.16).](#_bookmark9)

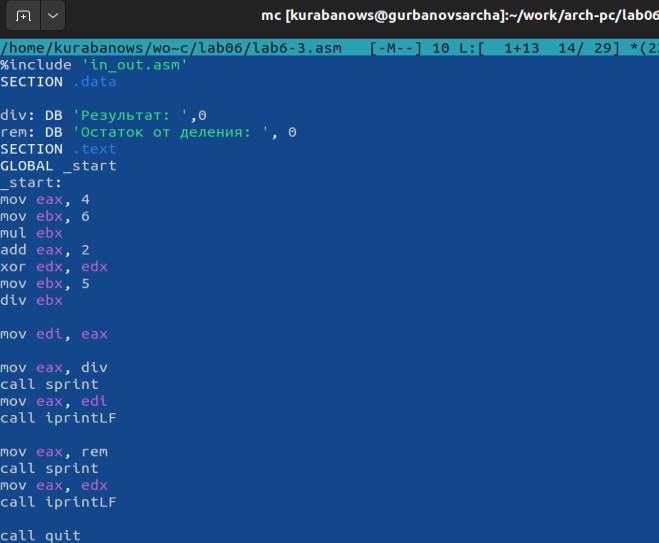


Рис. 3.16: Редактируем файл

Компилируем файл и запускаем программу (рис. [3.17).](#_bookmark9)

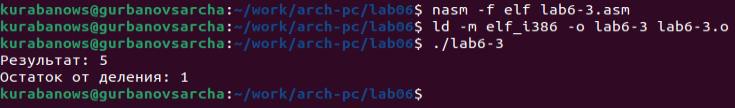


Рис. 3.17: Смотрим на результат работы программы

Создаем новый файл в каталоге (рис. [3.18).](#_bookmark9)



Рис. 3.18: Создаем файл

Открываем файл и редактируем в соответствии с листингом 6.4 (рис. [3.19).](#_bookmark10)

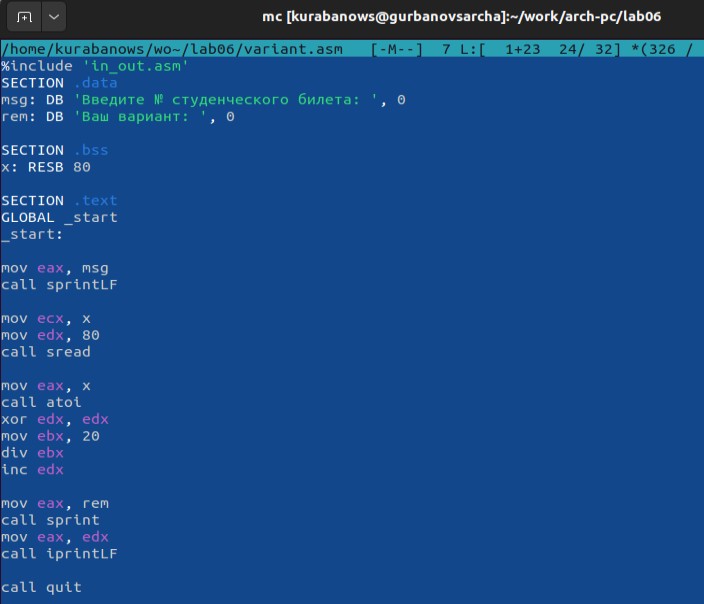


Рис. 3.19: Заполняем файл

Компилируем файл и запускаем его (рис. [3.20).](#_bookmark10)

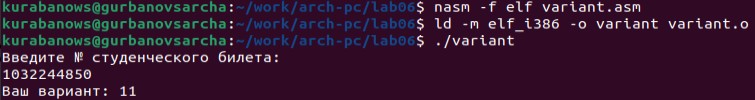


Рис. 3.20: Проверяемс результат работы программы

## Ответы на вопросы по программе

* + 1. Строка “mov eax,rem” и строка “call sprint” отвечают за вывод на экран сообщения ‘Ваш вариант:’.
    2. Эти инструкции используются для чтения строки с вводом данных от поль- зователя. Начальный адрес строки сохраняется в регистре ecx, а количество символов в строке (максимальное количество символов, которое может

быть считано) сохраняется в регистре edx. Затем вызывается процедура sread, которая выполняет чтение строки.

* + 1. Инструкция “call atoi” используется для преобразования строки в целое число. Она принимает адрес строки в регистре eax и возвращает полученное число в регистре eax.
    2. Строка “xor edx,edx” обнуляет регистр edx перед выполнением деления. Строка “mov ebx,20” загружает значение 20 в регистр ebx. Строка “div ebx” выполняет деление регистра eax на значение регистра ebx с сохранением частного в регистре eax и остатка в регистре edx.
    3. Остаток от деления записывается в регистр edx.
    4. Инструкция “inc edx” используется для увеличения значения в регистре edx на 1. В данном случае, она увеличивает остаток от деления на 1.
    5. Строка “mov eax,edx” передает значение остатка от деления в регистр eax. Строка “call iprintLF” вызывает процедуру iprintLF для вывода значения на экран вместе с переводом строки.

## Задание для самостоятельной работы

Создаем новый файл в каталоге (рис. [3.21).](#_bookmark11)



Рис. 3.21: Создаем файл

Открываем его и заполняем, чтобы решалось выражение f(x)=�^3 �1/3 + 21(рис.

[3.22).](#_bookmark12)

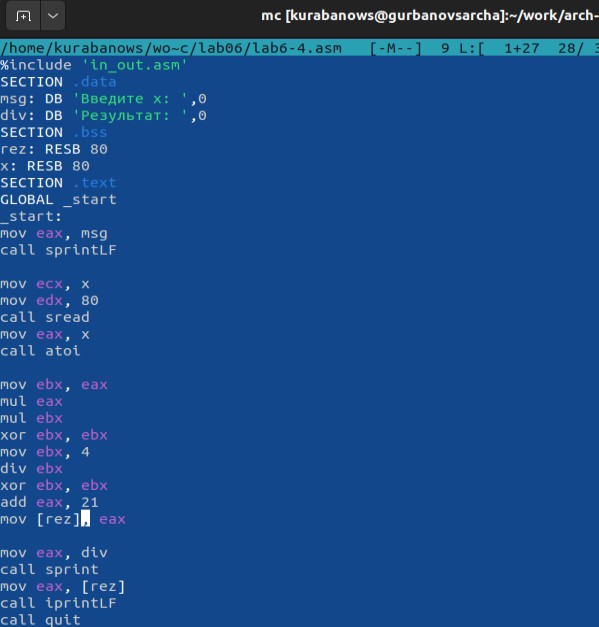


Рис. 3.22: Заполняем файл

Компилируем программу и проверяем для x=1 (рис. [3.23).](#_bookmark12)

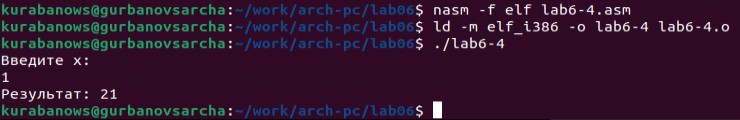


Рис. 3.23: Проверяем работу программы

Компилируем программу и проверяем для x=3 (рис. [3.24).](#_bookmark13)

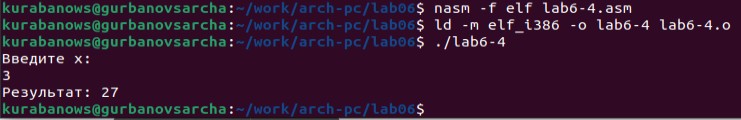


Рис. 3.24: Проверяем работу программы

# Выводы

Мы приобрели навыки создания исполнительных файлов для решения выра- жений и освоили арифметические инструкции в NASM.