Арифметические операции в NASM

Лабораторная работа №6

Владимир Романович Козомазов

Содержание

Список иллюстраций

Список таблиц

# 1 Цель работы

* Изучить работу с численными и символьными данными в языке ассемблера NASM.
* Изучить арифметические операции в языке ассемблера NASM и поработать с ними на практике.

# 2 Выполнение лабораторной работы

## 2.1 Описание выполняемого задания

* Написать программы для работы с численными и символьными данными и выводом их на экран.
* Написать программы для работы с арифметическими операциями и вычисления значения арифметического выражения.

## 2.2 Выполнение заданий лабораторной работы

### 2.2.1 Символьные и численые данные в NASM

Создал каталог для программ лабораторной работы №6, перешёл в него и создал файл lab6-1.asm (рис. 1)

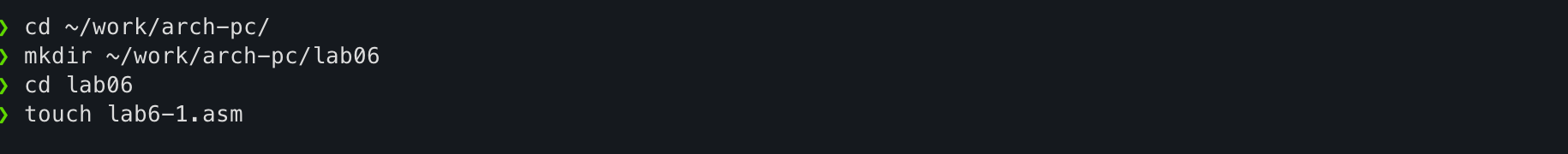


Рис. 1: Создание директории для программ лабораторной работы

Ввёл в файл lab6-1.asm текст программыы из листинга 6.1 (рис. 2)



Рис. 2: Ввод текста из листинга 6.1 в файл lab6-1.asm

Создал исполняемы файл и запустил его (рис. 3)

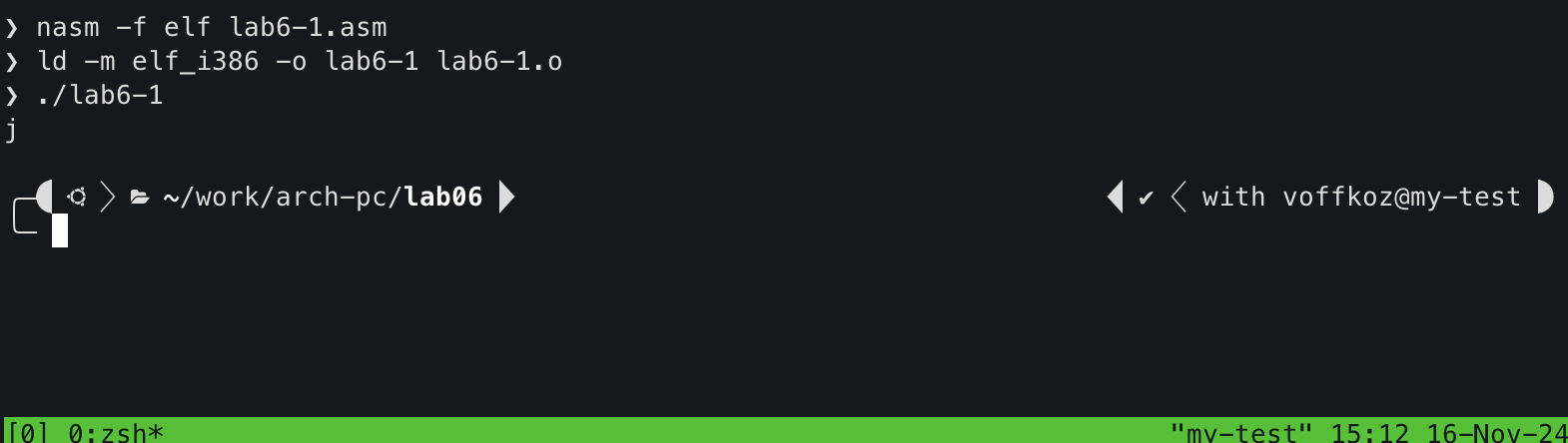


Рис. 3: Трансялция и запуск программы lab6-1

Изменил текст программы, вместо символов подставил числа (рис. 4)

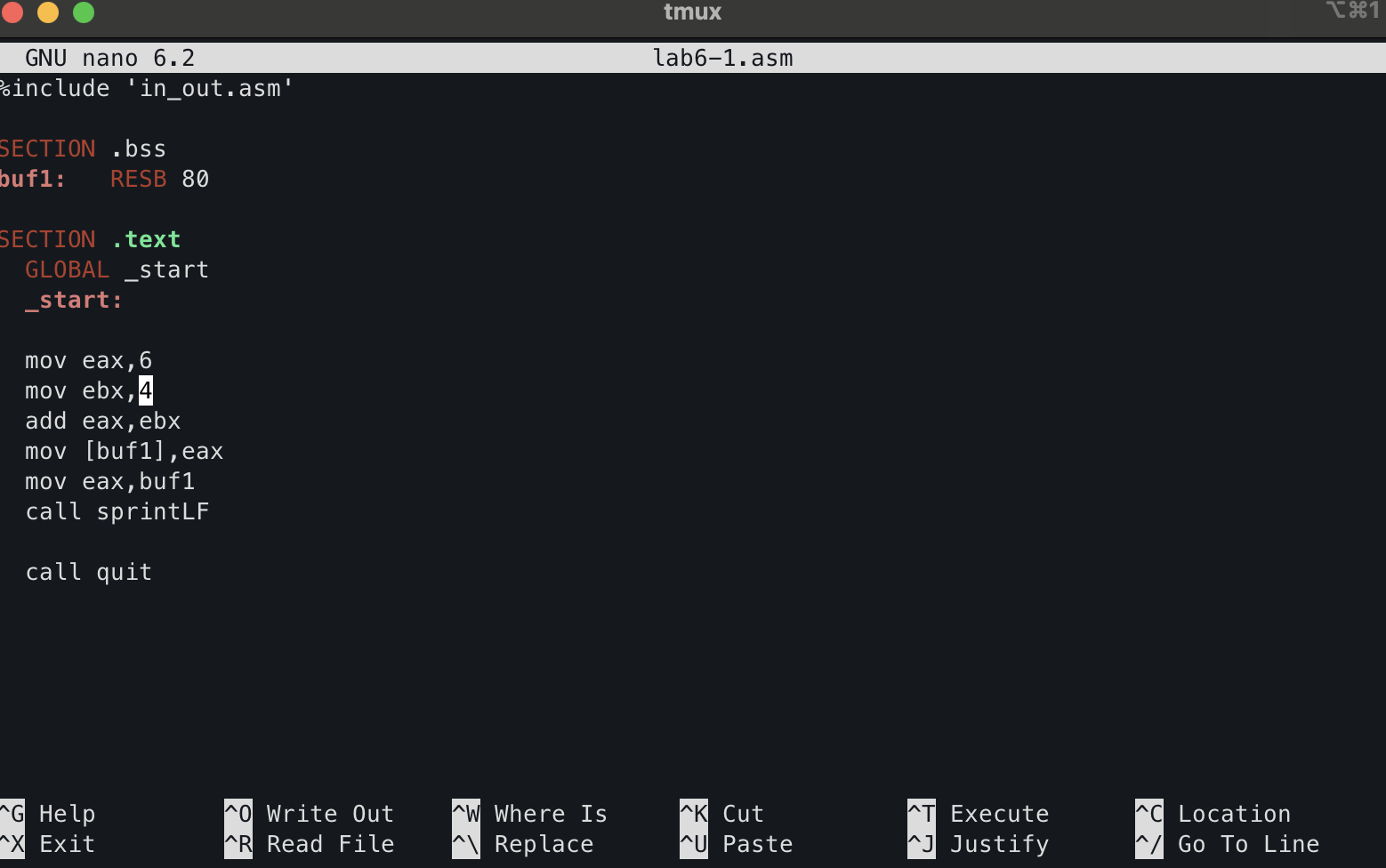


Рис. 4: Изменения кода в файле lab6-1.asm

Повторно создал исполняемый файл и запустил его (рис. 5). В результате выполнения програмы на экран вывелась пустая строка, что говорит о том, что на экран вывелся символ с кодом 10 - LF.

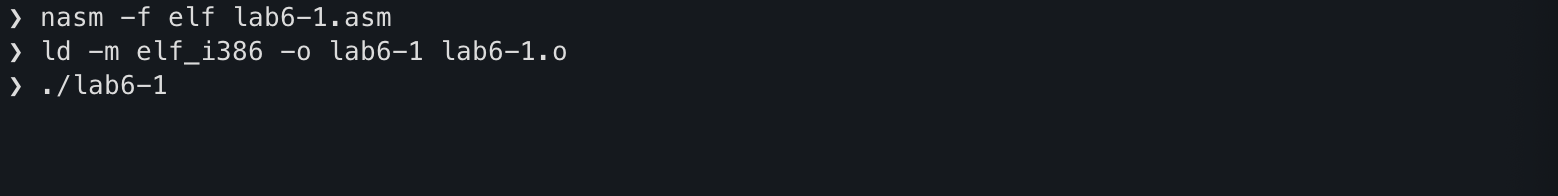


Рис. 5: Повторный запуск програмы lab6-1

Создал файл lab6-2.asm и ввёл в него код из листинга 6.2 (рис. 6)

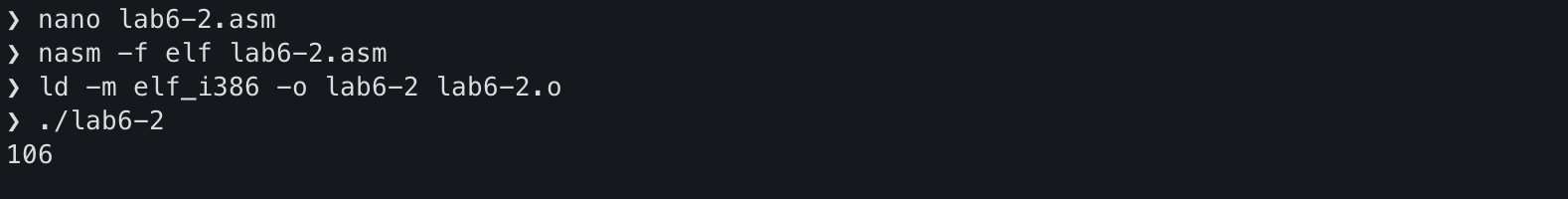


Рис. 6: Ввод кода програмы из листинга 6.2 в файл lab6-2

Аналогично предыдущему примеру изменил текст программы lab6-2.asm, заменив символы на числа (рис. 7)



Рис. 7: Изменения кода в файле lab6-2

Повторно скомпилировал и запустил програму lab6-2 (рис. 8). В результате получили, как и ожидалось, число 10

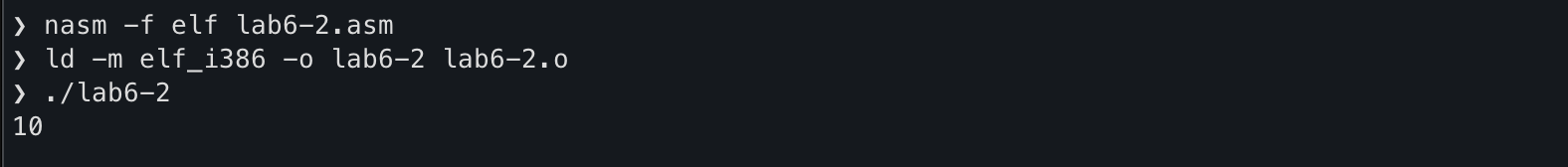


Рис. 8: Повторный запуск програмы lab6-2

Заменил функцию iprintLF на iprint (рис. 9)

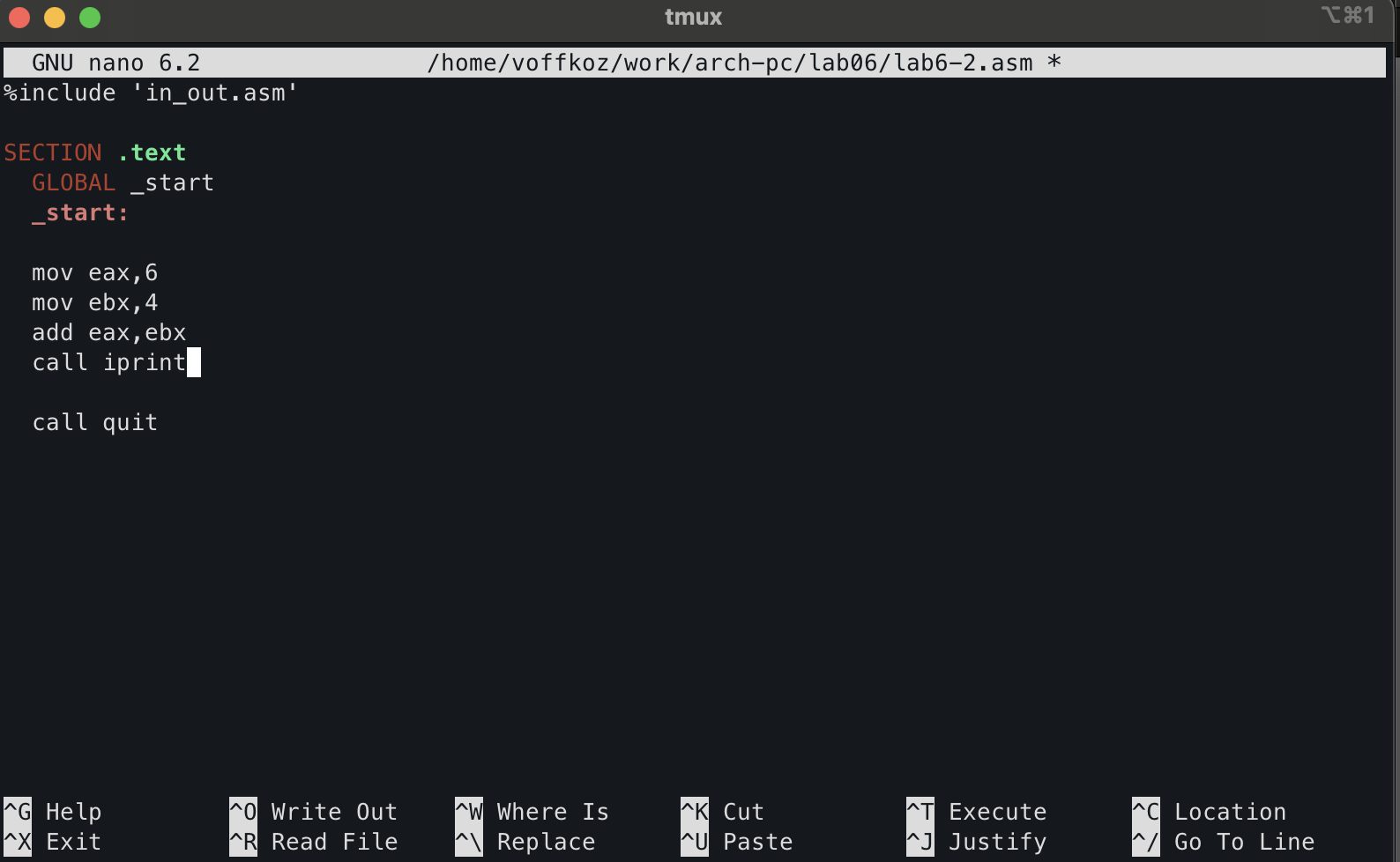


Рис. 9: Замена функции iprintLF на iprint в файле lab6-2

Повторно скомпилировал и запустил програму lab6-2 (рис. 10). Вывод программы отличается тем, что после её выполнения не происходит переход на новую строку.

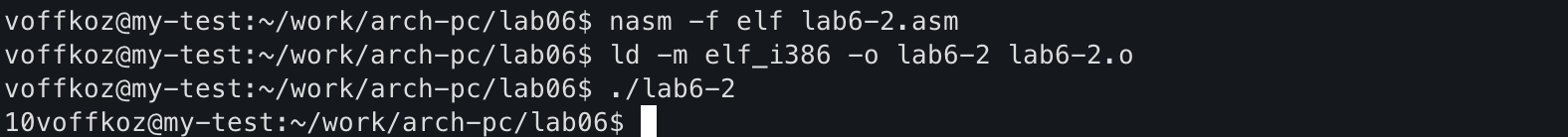


Рис. 10: Запуск программы lab6-2 с функцией iprint вместо iprintLF

### 2.2.2 Выполнение арифметических операций в NASM

Создал файл lab6-3.asm и ввёл в него код из листинга 6.3 (рис. 11)

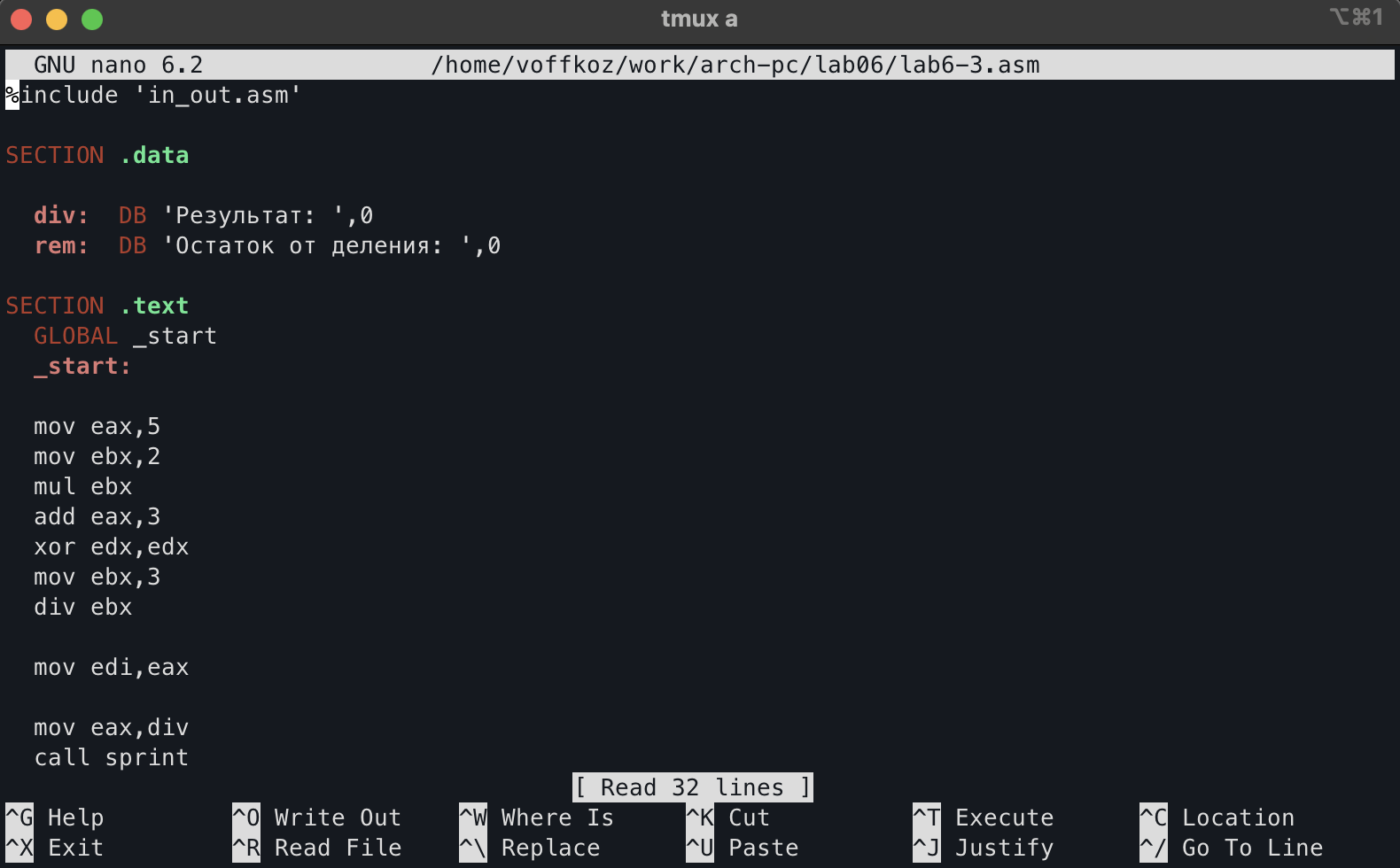


Рис. 11: Ввод кода из листинга 6.3 в файл lab6-3.asm

Скомпилировал и запустил программу lab6-3 (рис. 12)

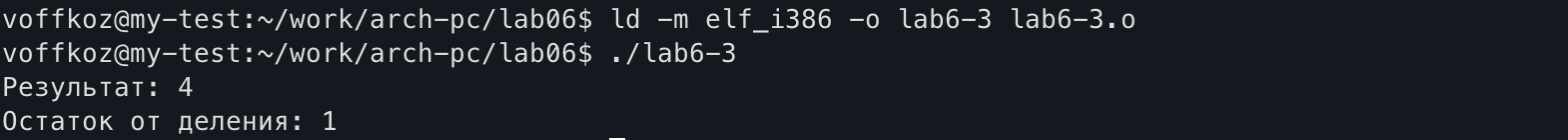


Рис. 12: Запуск программы lab6-3

Изменил текст файла lab6-3.asm так, чтобы он вычислял значение выражения (4 \* 6 + 2)/5 (рис. 13)

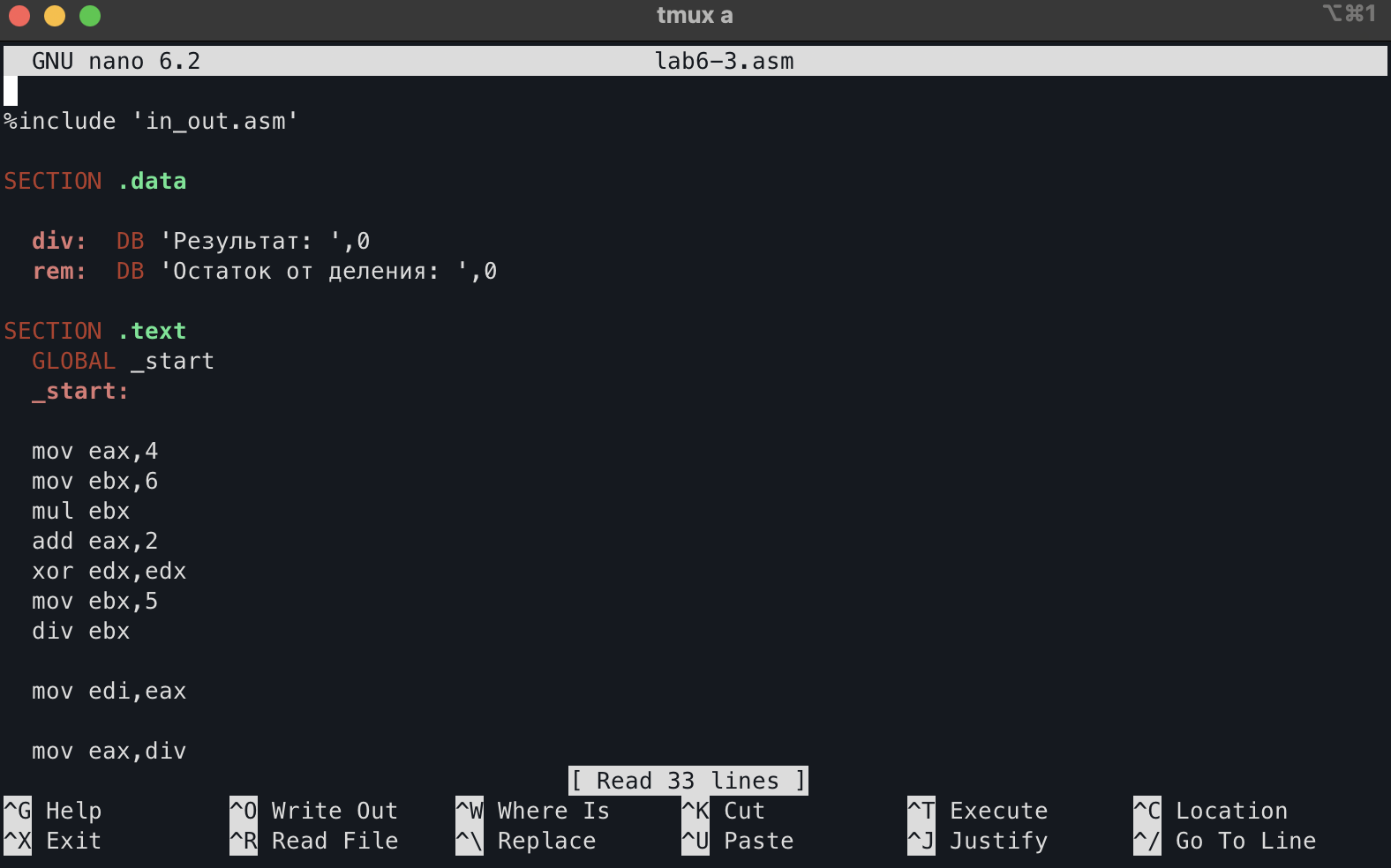


Рис. 13: Изменение текста программы в файле `lab6-3.asm’

Повторно скомпилировал и запустил программу lab6-3 (рис. 14)

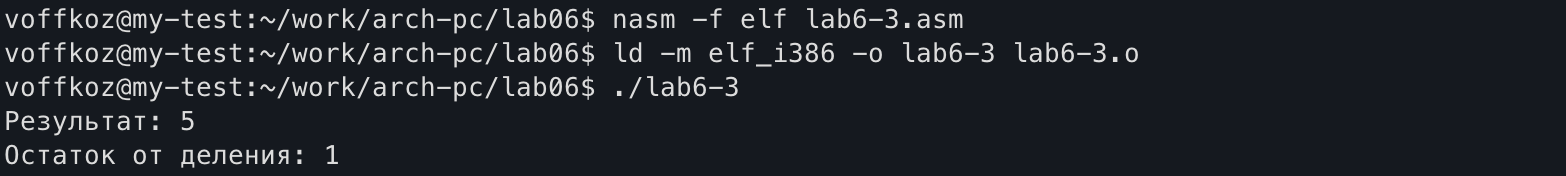


Рис. 14: Запуск программы `lab6-3’

Для написания программы вычисления варианта задания по номеру студенческого билета создал файл variant.asm и ввёл в него код из листинга 6.4 (рис. 15)

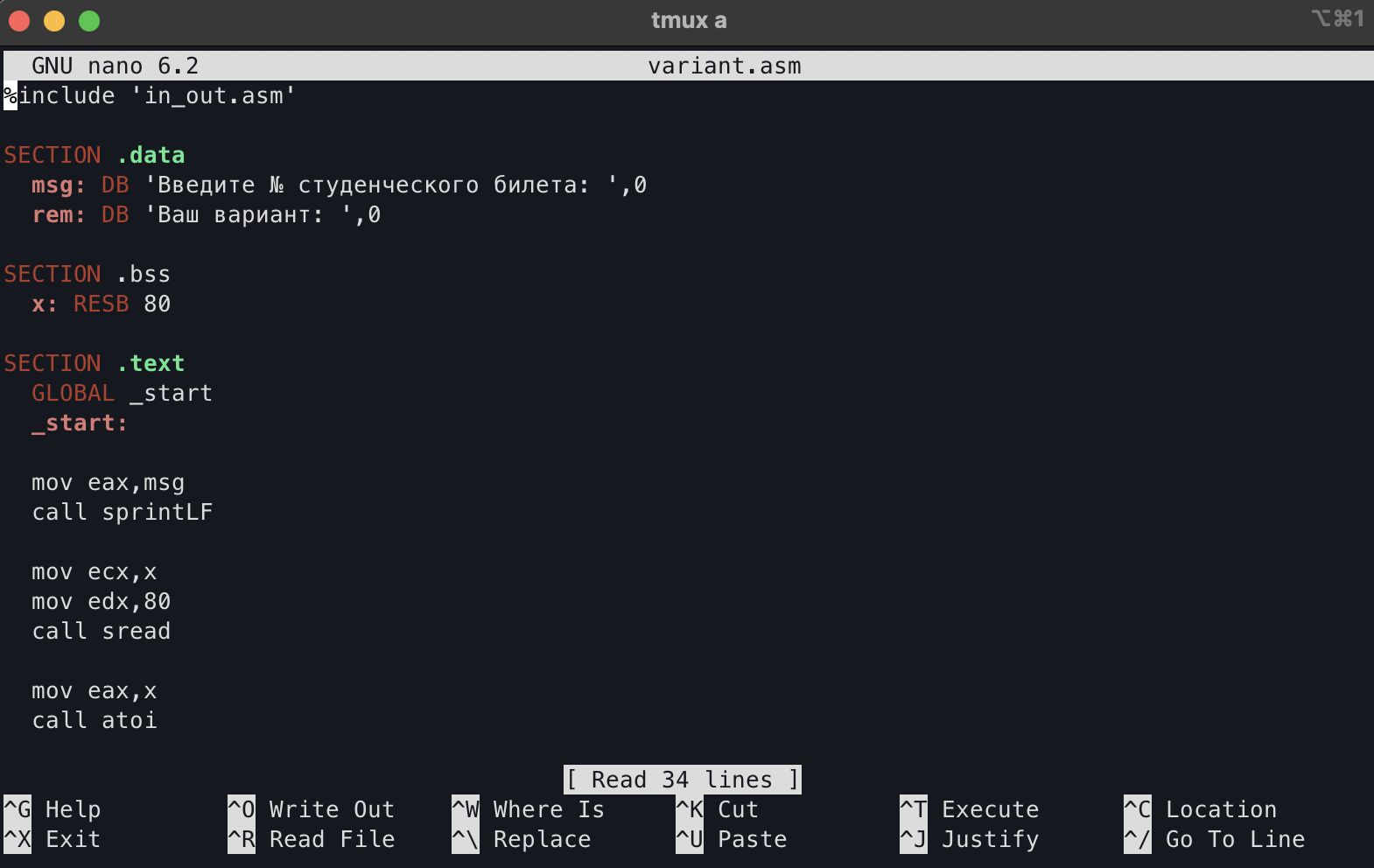


Рис. 15: Создание нового файла variant.asm и ввод кода из листинга 6.4

Скомпилировал и запустил программу variant (рис. 16)

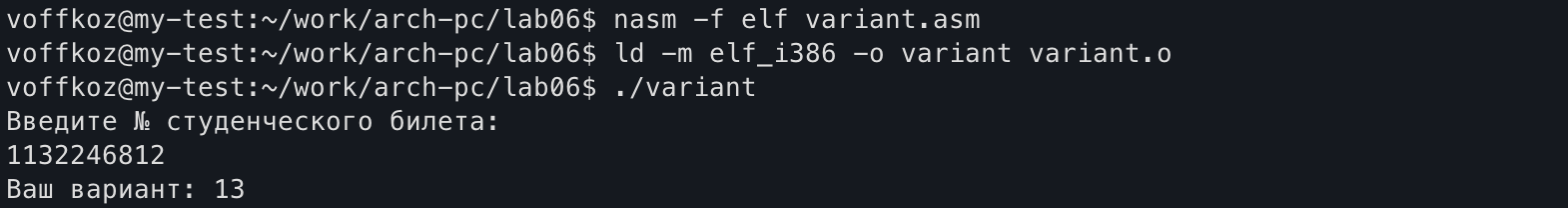


Рис. 16: Запуск программы variant и вывод результата работы на экран

### 2.2.3 Ответы на вопросы по заданию:

1. За вывод на экран сообщения “Ваш вариант:” отвечают следующие строки:

* mov eax,rem  
  call sprint

1. Инструкции

* mov ecx,x  
  mov edx,80  
  call sread
* используются для запроса у пользователя данных, которые он вводит с клавиатуры и сохранения их в переменную x.

1. Инструкция call atoi используется для преобразования значения, которое ввёл пользователь с клавиатуры из строки в число, с которыми можно производить арифметические операции.
2. За вычисление варианта отвечают следующие строки:

* xor edx,edx  
  mov ebx,20  
  div ebx  
  inc edx

1. Остаток от деления при выполнении инструкции div ebx записывается в регистр edx.
2. Инструкция inc edx используется для того, чтобы увеличить остаток от деления на единицу.
3. За вывод на экран результата вычислений отвечают следующие строки:

* mov eax,rem  
  call sprint  
  mov eax,edx  
  call iprintLF

## 2.3 Выводы по результатам выполнения заданий

В ходе выполнения заданий лабораторной работы были созданы программы для сложения символов и чисел и вывода их на экран. Была продемонстрирована разница между работой с символами и числами, а также были исользованы различные функции для вывода их на экран, в частности такие функции, как iprint, iprintLF и atoi. Была продемонстрирована разница между выводом функций iprintLF и iprint: вывод функция iprint отличается от функции iprintLF тем, что после её выполнения не происходит переход на новую строку.

Также в ходе выполнения лабораторной работы был преобретён навык вычисления арифметических выражений, а также освоен метод деления с остатком, который очень часто используется в программировании.

# 3 Выполнение самостоятельного задания

## 3.1 Описание выполняемого самостоятельного задания

Написать программу для вычисления выражения . Программа должна выводить выражения для вычисления, выводить запрос на ввод значения , вычислять заданное выражение в зависимости от введённого , выводить результат вычислений.

## 3.2 Выполнение заданий для самостоятельной работы

Создал файл calc\_function.asm и написал код для выполнения задания (рис. 17)

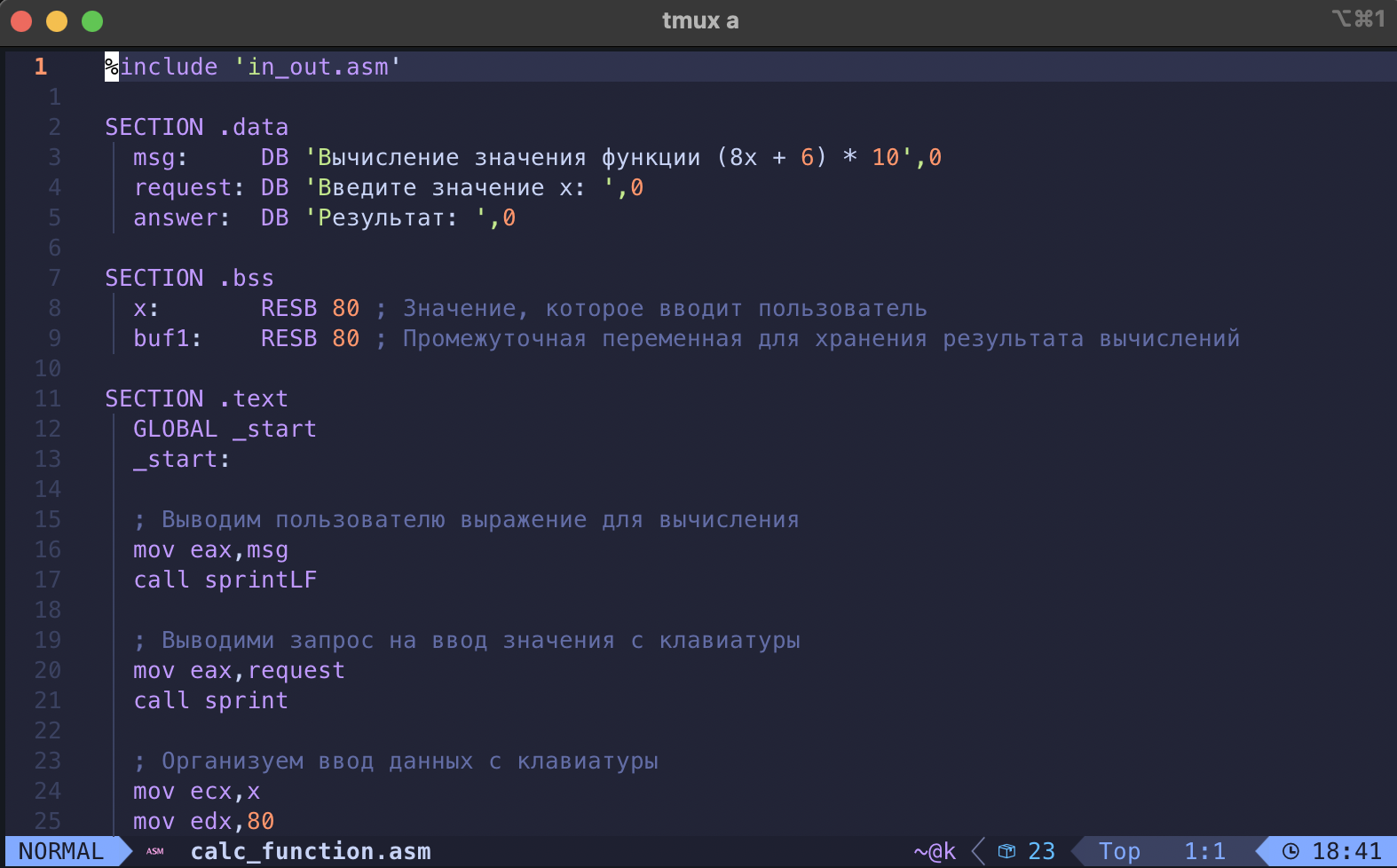


Рис. 17: Создание файла calc\_function.asm и написание кода для выполнения задания

Проверка программы calc\_function на правильность вычислений (рис. 18)

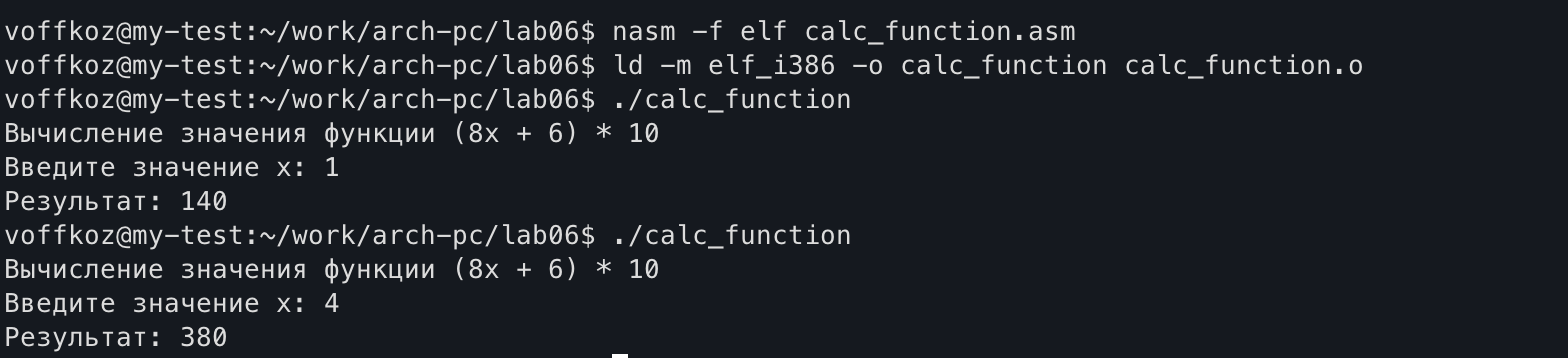


Рис. 18: Результаты запуска программы calc\_function и вывод результата работы на экран

## 3.3 Выводы по результатам выполнения самостоятельного задания

В ходе выполнения задания для самостоятельной работы была разработана программа для вычисления значения заданной функции при различный значения . При написании программы были применены методы и техники, освоенные в ходе выполнения лабораторной работы, в частности метод обработки ввода пользователя, выводы сообщений на экран и вычисление математических выражений.

## 3.4 Листинги написанных программ

Листинг программы lab6-1.asm

%include 'in\_out.asm'  
  
SECTION .bss  
buf1: RESB 80  
  
SECTION .text  
 GLOBAL \_start  
 \_start:  
  
 mov eax,6  
 mov ebx,4  
 add eax,ebx  
 mov [buf1],eax  
 mov eax,buf1  
 call sprintLF  
  
 call quit

Листинг программы lab6-2.asm

%include 'in\_out.asm'  
  
SECTION .text  
 GLOBAL \_start  
 \_start:  
  
 mov eax,6  
 mov ebx,4  
 add eax,ebx  
 call iprint  
  
 call quit

Листинг программы lab6-3.asm

%include 'in\_out.asm'  
  
SECTION .data  
  
 div: DB 'Результат: ',0  
 rem: DB 'Остаток от деления: ',0  
  
SECTION .text  
 GLOBAL \_start  
 \_start:  
  
 mov eax,4  
 mov ebx,6  
 mul ebx  
 add eax,2  
 xor edx,edx  
 mov ebx,5  
 div ebx  
  
 mov edi,eax  
  
 mov eax,div  
 call sprint  
 mov eax,edi  
 call iprintLF  
  
 mov eax,rem  
 call sprint  
 mov eax,edx  
 call iprintLF  
  
 call quit

Листинг программы variant.asm

%include 'in\_out.asm'  
  
SECTION .data  
 msg: DB 'Введите № студенческого билета: ',0  
 rem: DB 'Ваш вариант: ',0  
  
SECTION .bss  
 x: RESB 80  
  
SECTION .text  
 GLOBAL \_start  
 \_start:  
  
 mov eax,msg  
 call sprintLF  
  
 mov ecx,x  
 mov edx,80  
 call sread  
  
 mov eax,x  
 call atoi  
  
 xor edx,edx  
 mov ebx,20  
 div ebx  
 inc edx  
  
 mov eax,rem  
 call sprint  
 mov eax,edx  
 call iprintLF  
  
 call quit

Листинг программы calc\_function.asm

%include 'in\_out.asm'  
  
SECTION .data  
 msg: DB 'Вычисление значения функции (8х + 6) \* 10',0  
 request: DB 'Введите значение х: ',0  
 answer: DB 'Результат: ',0  
  
SECTION .bss  
 x: RESB 80 ; Значение, которое вводит пользователь  
 buf1: RESB 80 ; Промежуточная переменная для хранения результата вычислений  
  
SECTION .text  
 GLOBAL \_start  
 \_start:  
  
 ; Выводим пользователю выражение для вычисления  
 mov eax,msg  
 call sprintLF  
  
 ; Выводими запрос на ввод значения с клавиатуры  
 mov eax,request  
 call sprint  
  
 ; Организуем ввод данных с клавиатуры  
 mov ecx,x  
 mov edx,80  
 call sread  
  
 ; Преобразуем значение, введённое пользователем, в число  
 mov eax,x  
 call atoi  
  
 ; Вычисляем выражение  
 mov ebx,8  
 mul ebx  
 mov ebx,6  
 add eax,ebx  
 mov ebx,10  
 mul ebx  
  
 ; Сохраняем результат вычислений в переменную buf1  
 mov [buf1],eax  
  
 ; Выводим результат вычисления пользователю  
 mov eax,answer  
 call sprint  
 mov eax,[buf1]  
 call iprintLF  
  
 call quit

# 4 Выводы

В ходе выполнения лабораторной работы №6 удалось достичь всех поставленных целей, а именно:

* Была изучена работа с численными и символьными данными в языке ассемблера NASM, а также написаны программы для закрепления изученного материала, в частности было продемонстрированно различие в работе с символами и числами, а также использованы разные функции для вывода символов и чисел на экран.
* Была изучена работа с арифметическими операциями в зыке ассемблера NASM, были написаны программы для вычисления заданных арифметических выражений, были изучены такие команды как mul, div, inc, dec, neg, add, sub.