Отчёт по лабораторной работе №6

Дисциплина: Архитектура компьютера

Бережной Иван Александрович

Содержание

# 1 Цель работы

Освоить арифметические инструкций языка ассемблера NASM.

# 2 Задание

1. Разбор символьных и численных данных в NASM
2. Выполнение арифметических операций в NASM
3. Ответы на вопросы
4. Задание для самостоятельной работы

# 3 Выполнение лабораторной работы

## 3.1 Разбор символьных и численных данных в NASM

Для начала создадим каталог lab06 для последующей работы. Для этогого воспользуемся командой mkdir. Перейдём в него и создадим файл lab6-1.asm (рис. [1](#fig:001)).

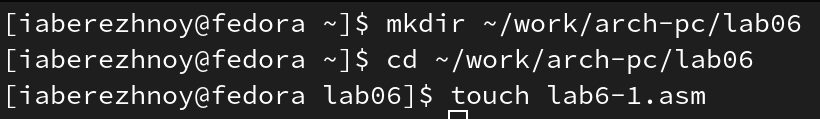


Figure 1: Подготовка к работе

Скопируем предложенный листинг в новый файл (рис. [2](#fig:002)). Создадим и запустим исполняемый файл (рис. [3](#fig:003)). В выводе видим букву “j”.

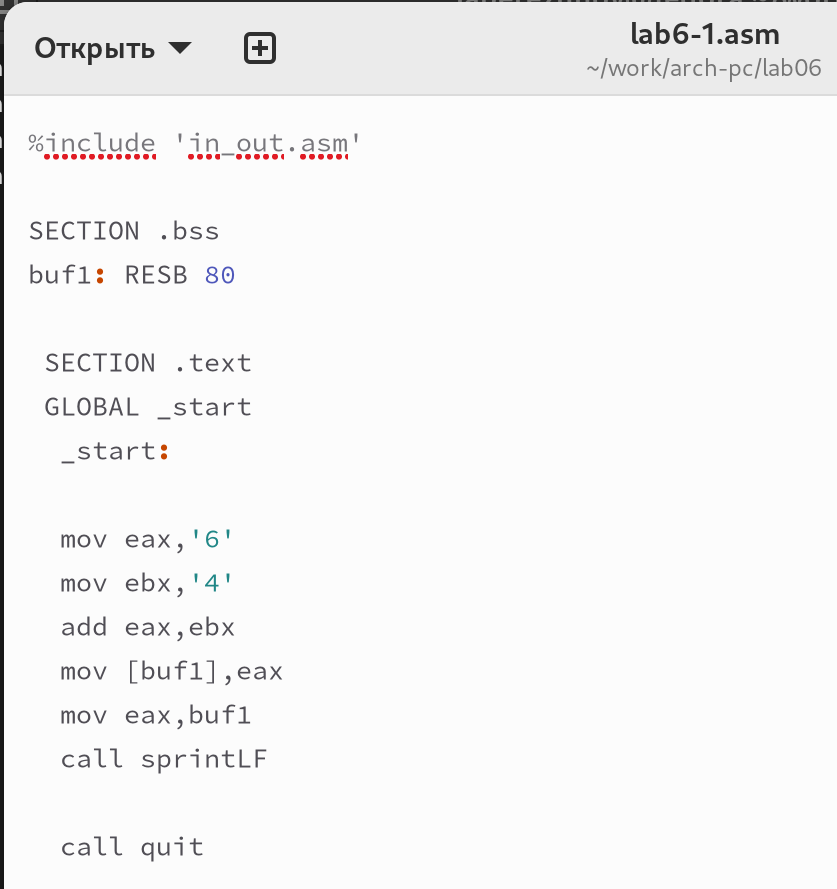


Figure 2: Копирование кода в lab6-1.asm

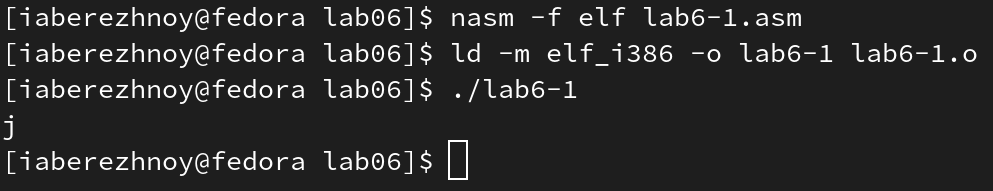


Figure 3: Запуск файла lab6-1.exe

Немного изменим текст программы, убрав одинарные кавычки в строках mov eax,'6' и mov ebx,'4' (рис. [4](#fig:004)). Также пересоздадим и запустим исполняемый файл - получили перевод строки (рис. [5](#fig:005)).

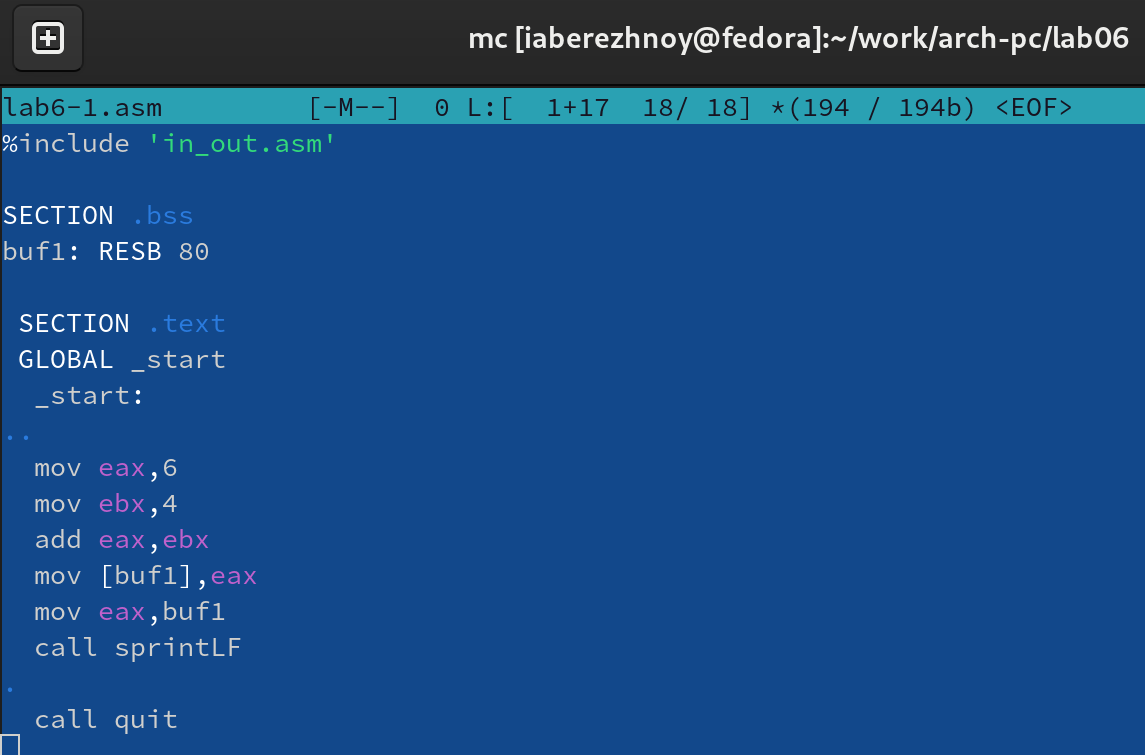


Figure 4: Редактирование lab6-1.exe

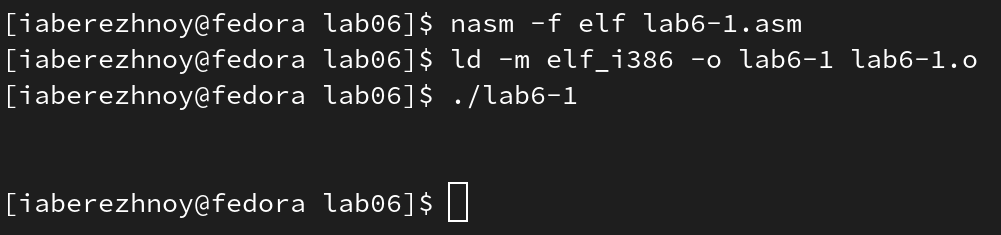


Figure 5: Запуск изменённого файла lab6-1.exe

Создадим файл lab6-2.asm и скопируем в него второй листинг (рис. [6](#fig:006)). Для него тоже создадим исполняемый файл и запустим его. В выводе получили 106 (рис. [7](#fig:007)).

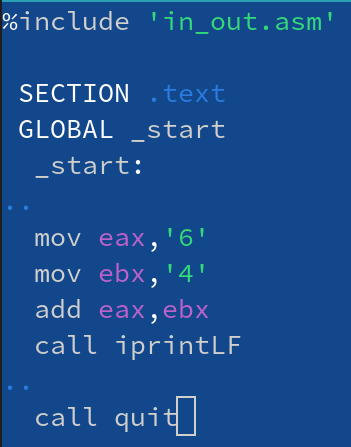


Figure 6: Копирование кода в lab6-2.asm

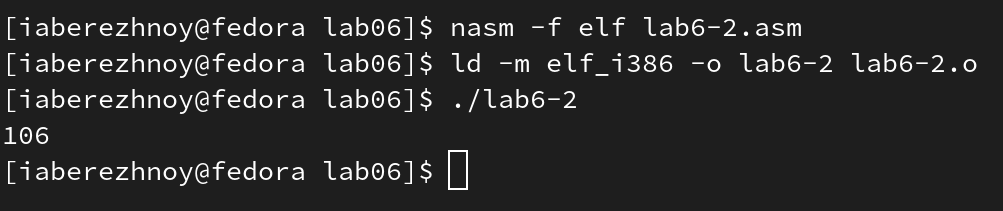


Figure 7: Запуск файла lab6-2.exe

Аналогично предыдущей программе изменим в этой пару строк (рис. [8](#fig:008)) и запустим уже изменённый файл. В результате получили 10 (рис. [9](#fig:009)).

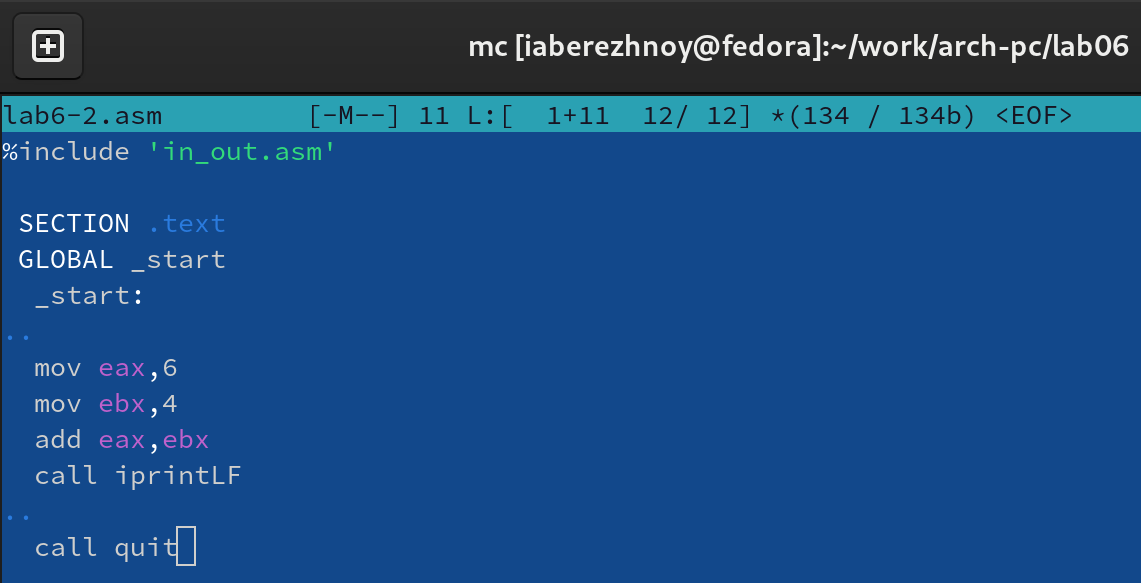


Figure 8: Редактирование lab6-2.asm

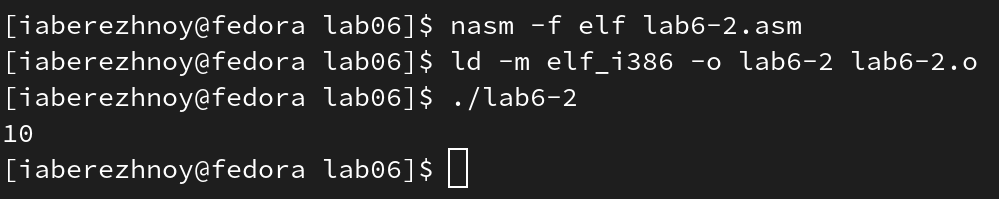


Figure 9: Запуск изменённого файла lab6-2.exe

Теперь заменим всё в этом же файле функцию iprintLF на iprint. Создадим исполняемый файл и запустим его. Получили всё тот же результат, но теперь отсутствует перевод строки (рис. [10](#fig:010)).

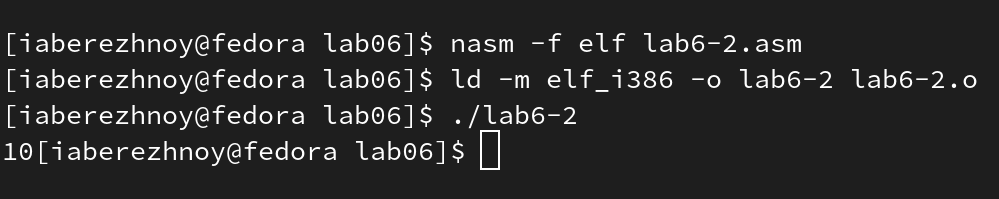


Figure 10: Запуск дважды изменённого файла lab6-2.exe

## 3.2 Выполнение арифметических операций в NASM

Создадим файл lab6-3.asm и скопируем в него третий код (рис. [11](#fig:011)). Создадим и запустим исполняемый файл - в результате получили 4, а остаток от деления 1 (рис. [12](#fig:012)).

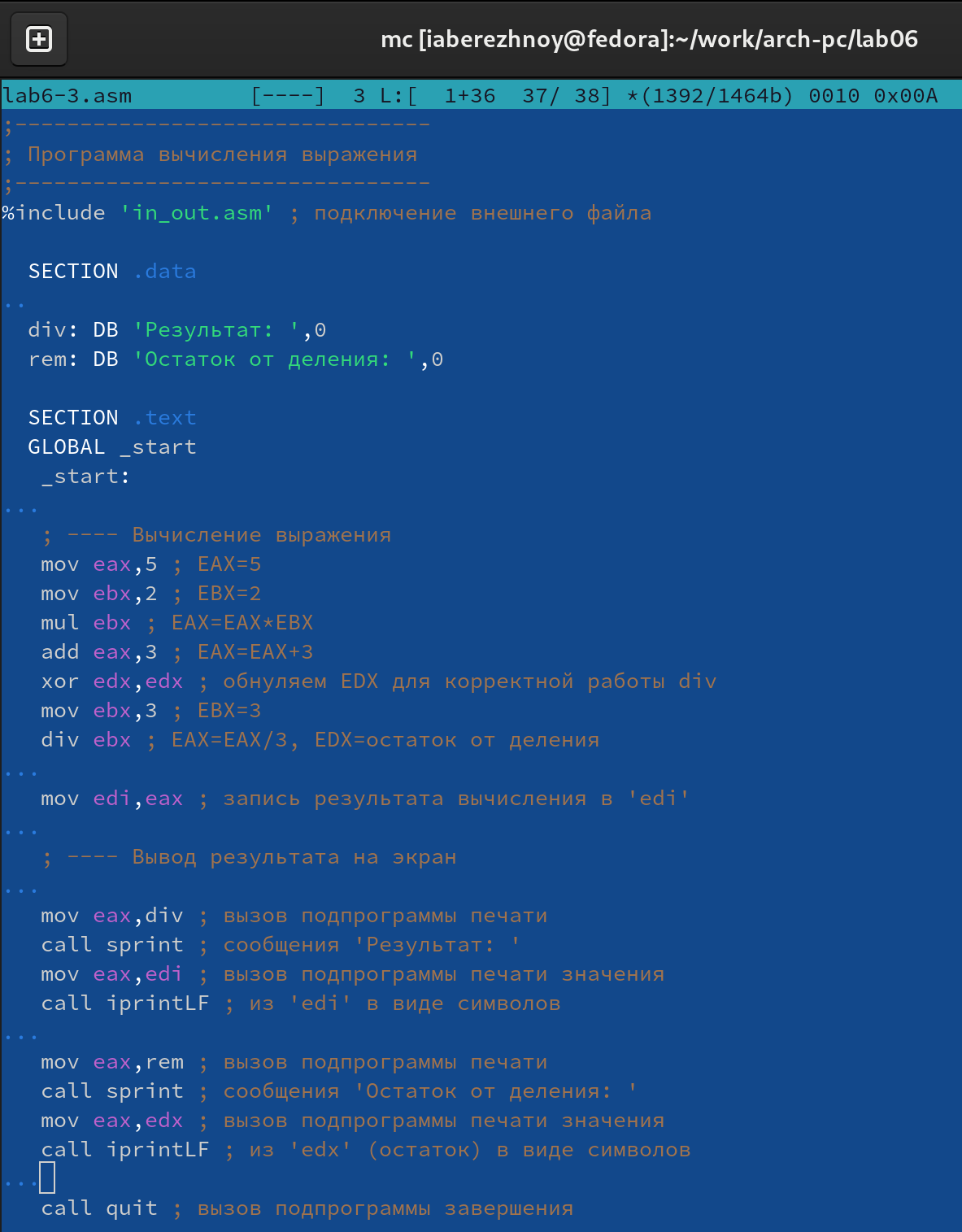


Figure 11: Копирование кода в lab6-3.asm

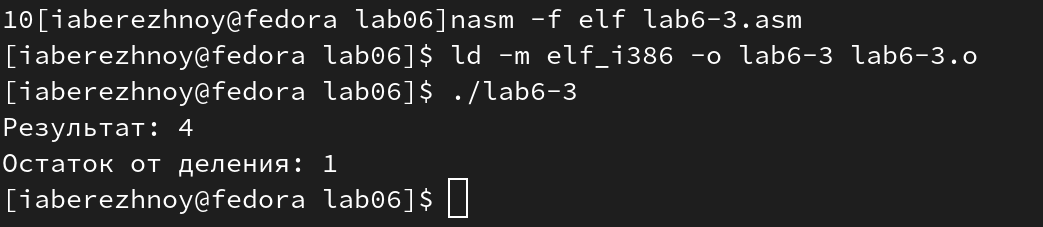


Figure 12: Запуск файла lab6-3.exe

Изменим текст программы так, чтобы вычислялось значение выражения (4\*6+2)/5. Проверим работу файла (рис. [13](#fig:013)).

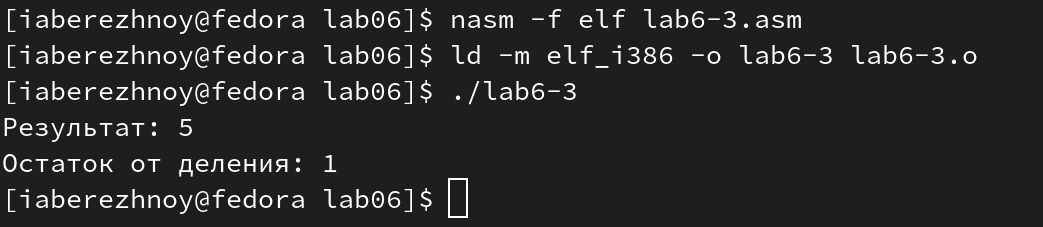


Figure 13: Запуск изменённого файла lab6-3.exe

Создадим файл variant.asm командой touch, скопируем в него листинг №4 (рис. [14](#fig:014)) и проверим работы исполняемого файла. Для этого введём 1132236041 - получили 2 (рис. [15](#fig:015)), а значит, в задании для самостоятельной работы мы будем выполнять вариант под номером 2.

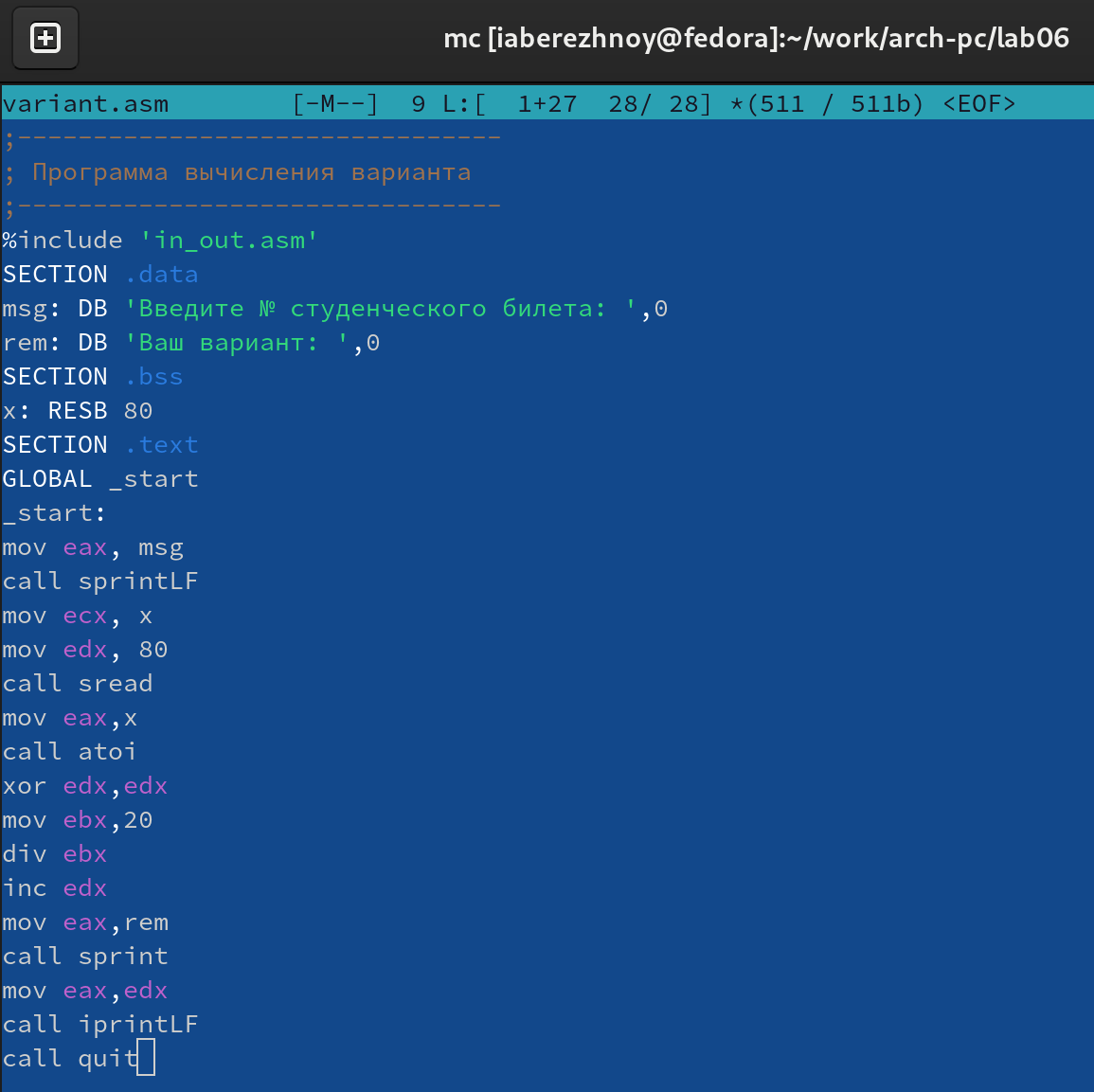


Figure 14: Копирование кода в variant.asm

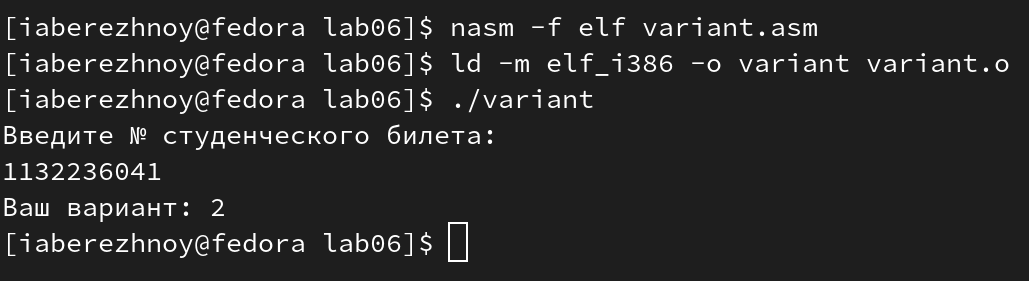


Figure 15: Запуск файла variant.exe

## 3.3 Ответы на вопросы

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

mov eax,rem  
call sprint

1. Первая инструкция используется, чтобы скопировать адрес вводимой строки в регистр ecx; Вторая инструкция используется, чтобы установить в регистр edx длинну вводимой строки; Третья инструкция используется, чтобы вызвать подпрограмму из внешнего файла, обеспечивающей ввод текста с клавиатуры.
2. Инструкция call atoi используется, чтобы преобразовать ascii-код символа в целое число.
3. За вычисление варианта в листинге 6.4 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 увеличения значения в регистре edx на 1.
3. За вывод результата вычислений на экран в листинге 6.4 отвечают следующие строки:

mov eax,edx  
call iprintLF

## 3.4 Задание для самостоятельной работы

Напишем программу для вычисления выражения y = (12x + 3)5 в файле lab6-4.asm (листинг 6.1). Проверим её работоспособность (рис. [16](#fig:016))

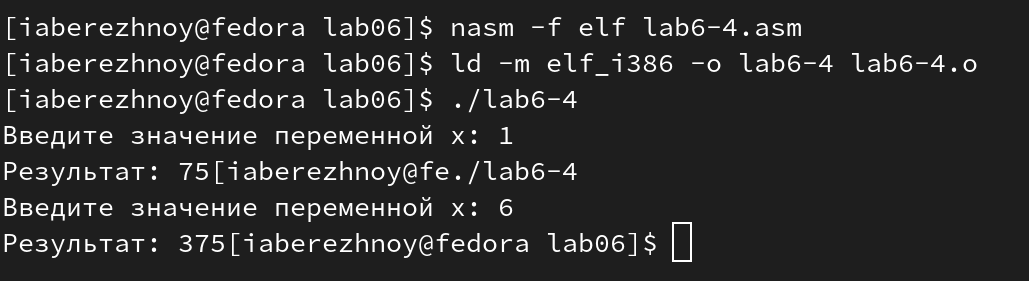


Figure 16: Проверка работоспособности программы

**Листинг 6.1. Программа вычисления выражения y = (12x + 3)5**

%include 'in\_out.asm'  
  
SECTION .data  
msg: DB 'Введите значение переменной x: ',0  
rem: DB 'Результат: ',0  
  
SECTION .bss  
x: RESB 80  
  
SECTION .text  
GLOBAL \_start  
  
\_start:  
 ; ---- Ввод значения переменной x  
 mov eax, msg  
 call sprint  
 mov ecx, x  
 mov edx, 80  
 call sread  
  
 ; ---- Вычисление выражения (12𝑥 + 3)5  
 mov eax, x  
 call atoi  
 mov ebx, 12  
 imul eax, ebx  
 add eax, 3  
 imul eax, 5  
 mov edi, eax  
  
 ; ---- Вывод результата на экран  
 mov eax, rem  
 call sprint  
 mov eax, edi  
 call iprint  
 call quit

# 4 Выводы

В ходе выполнения лабораторной работы мы разобрали символьные и численные данные в NASM, а также освоили арифметические операции языка ассемблера NASM.

# Список литературы

::: [Архитектура ЭВМ](https://esystem.rudn.ru/mod/resource/view.php?id=1030554)