Лабораторная работа №6

Арифметические операции в NASM.

Жукова Арина Александровна

Содержание

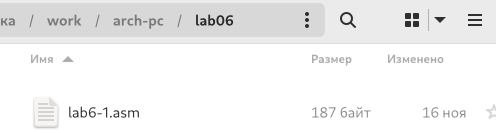
# 1 Цель работы

Целью данной лабораторной работы является освоение арифметических инструкций языка ассемблера NASM.

# 2 Выполнение лабораторной работы

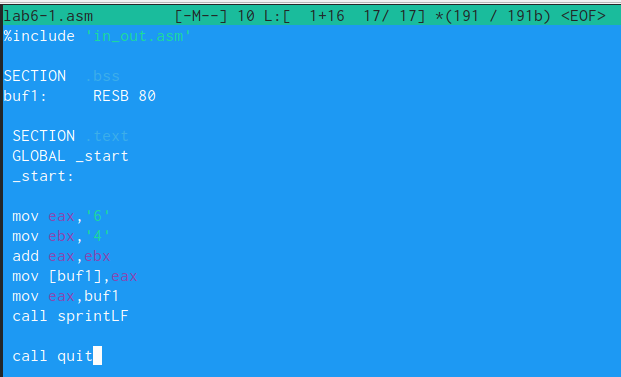
## 2.1 Символьные и численные данные в NASM

1. Создаём файл lab6-1.asm в новом каталоге для программ лабораторной работы №6 (рис. ??).



Создание файла

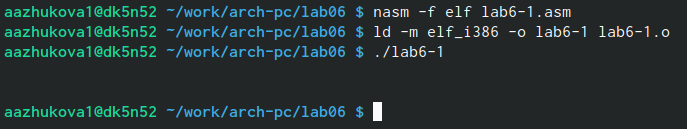
1. Вводим в файл lab6-1.asm текст программы из данного листинга 6.1 (рис. ??).



Ввод текста программы

В программе в регистр eax записывается символ 6 (‘6’), в ebx - 4 (‘4’). Далее значения складываютcя, результат сложения записывается в регистр eax. Для вывода результата при помощи команды sprintLF необходимо, чтобы в регистре eax был записан адрес, для этого используем дополнительную переменную buf1.

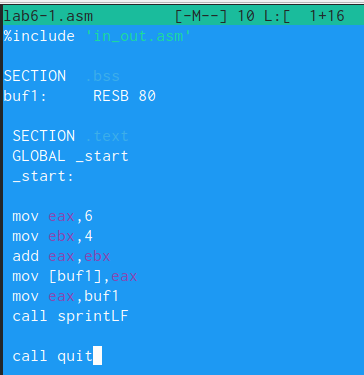
Создаём исполняем файл и проверяем его работу (рис. ??).



Создание исполняемого файла и проверка его работы

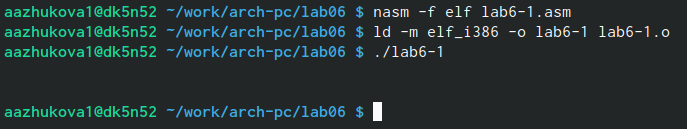
При выводе программы отображается символ j, так как программа выводит символ, соотвествующий сумме двоичных кодов символа 4 и 6 по системе ASCII.

1. Изменяем текст программы и вместо символов (‘4’ и ‘6’), запишем в регистры числа (4 и 6) (рис. ??)



Изменение текста программы

Создаём исполняемый файл и запускаем его (рис. ??).



Создание и запуск исполняемого файла

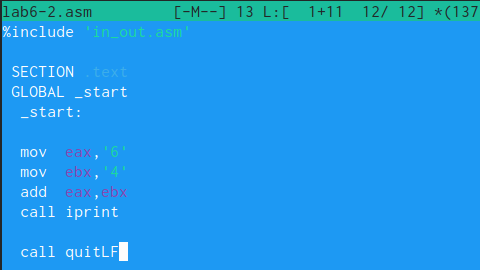
Теперь выводится символ с кодом 10 - перевод строки. Этот символ выводится на экран пустой строкой.

1. Создаём новый файл lab6-2.asm при помощи команды touch (рис. ??).

Создание файла lab6-2.asm

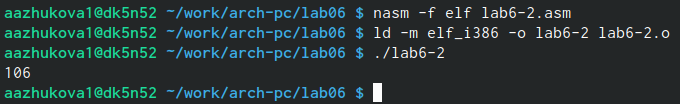
Создание файла lab6-2.asm

Вводим в него текст программы для вывода значения регистра eax (рис. ??).



Редактирование файла

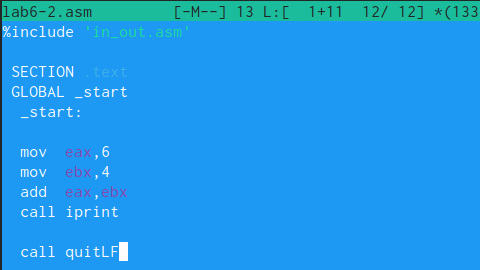
Создаём и запускаем исполняемый файл (рис. ??).



Создание и проверка работы файла

Программа выводит число 106, так как программа выводит сумму кодов символов ‘4’ и ‘6’. Однако функция iprintLF позволяет вывести на экран число, а не символ, кодом которого является это число.

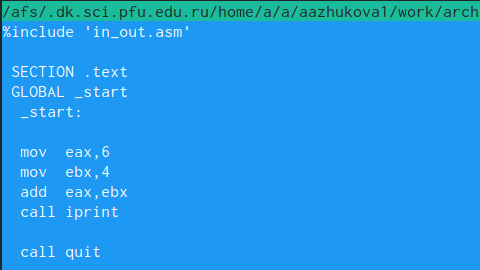
1. Заменяем символы на числа в тексте программы файла lab6-2.asm (рис. ??).



Редактирование файла

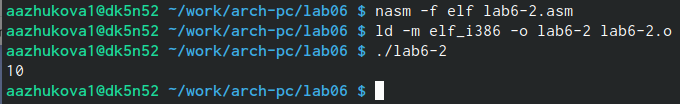
Теперь программа складывает не коды, соответсвующие символам, а сами числа, поэтому мы получаем вывод 10.

Заменяем в тексте программы функцию iprintLF на iprint (рис. ??).



Редактирование файла

Далее создаём и запускаем исполняемый файл (рис. ??).



Создание и проверка работы файла

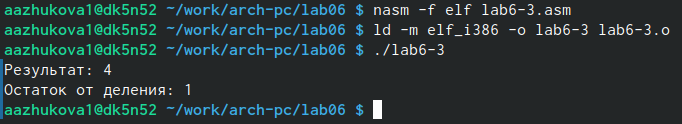
## 2.2 Выполнение арифметических операций в NASM

1. Создадим файл lab6-3.asm в каталоге ~/work/arch-pc/lab06 и введем в него текст листинга 6.3 (рис. ??).



Ввод текста программы

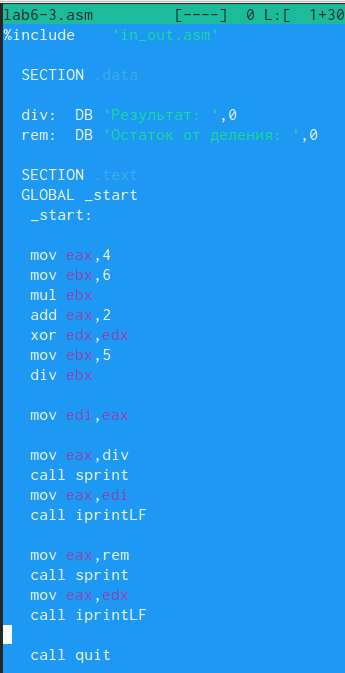
Создаём исполняемый файл и запускаем его (рис. ??).



Создание и проверка работы файла

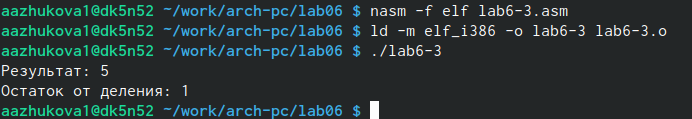
Изменим текст программы для вычисления выражения

(рис. ??).



Редактирование файла

Создаём исполняемый файл и проверяем его работу (рис. ??).



Создание и проверка работы файла

1. Рассмотрим программу вычисления варианта задания по номеру студенческого билета, работающую по следующему алгоритму:

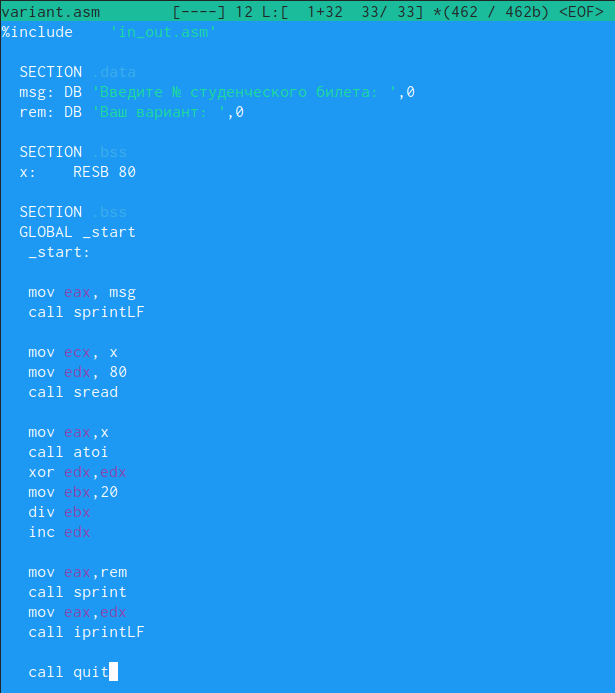
• вывести запрос на введение № студенческого билета

• вычислить номер варианта по формуле:

, где Sn – номер студенческого билета (В данном случае a mod b – это остаток от деления a на b).

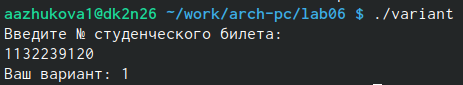
• вывести на экран номер варианта.

Создаём файл variant.asm в каталоге ~/work/arch-pc/lab06, вводим текст листинга 6.4 в файл variant.asm (рис. ??).



Ввод текста программы

Создаём исполняемый файл и запустим его (рис. ??).



Создание и проверка работы файла

### 2.2.1 Ответы на вопросы

1. За вывод на экран сообщения ‘Ваш вариант:’ отвечают строки:

mov eax,rem  
call sprint

1. Данные инструкции используются для: mov ecx, x - адрес вводимой строки x вкладывается в регистр ecx mov edx, 80 - запись в регистр call sread вызов подпрограммы из внешнего файла, обеспечивающий ввод данных с клавиатуры.
2. call atoi используется для вызова подпрограммы из внешнего файла, который преобразует ascii-код символа в целое числои записывает результат в регистр.
3. За вычисление варианта отвечают строки:

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

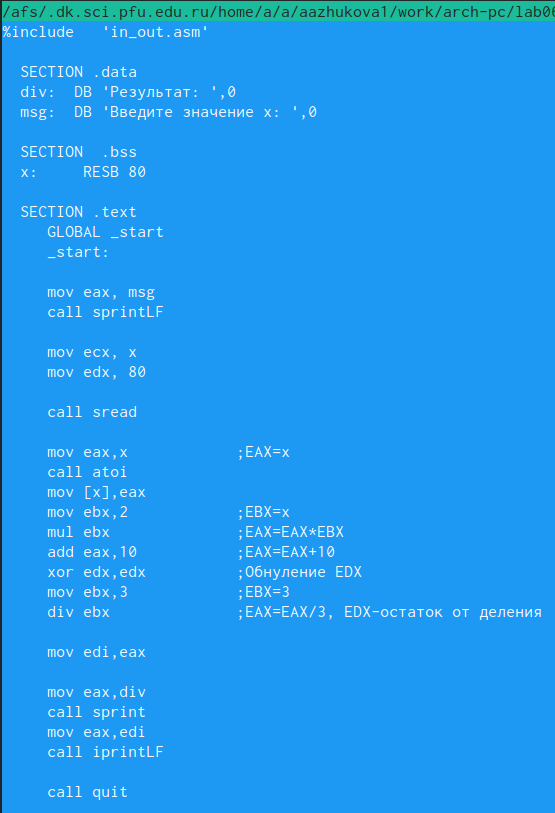
1. При выполнении инструкции div ebx остаток от деления записывается в регистр edx.
2. Инструкция inc edx увеличивает значение регистра edx на 1.
3. За вывод на экран результата вычислений отвечает строки:

mov eax,edx  
call iprintLF

## 2.3 Выполнение заданий для самостоятельной работы

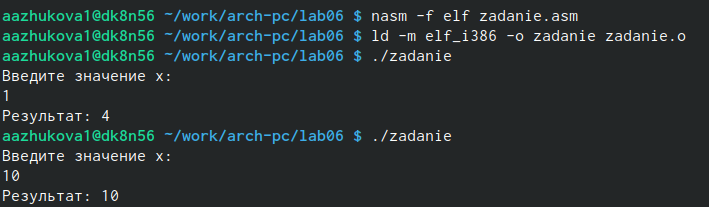
Создаём файл zadanie.asm и вводим в него текст программы для вычисления выражения

. Программа должна выводить выражение для вычисления, выводить запрос на ввод значения x, вычислять заданное выражение в зависимости от введенного x, выводить результат вычислений (рис. ??).



Ввод текста программы

Создаём исполняемый файл и запустим его (рис. ??).



Создание и проверка работы файла

# 3 Выводы

При выполнении данной лабораторной работы нами были освоены арифметические инструкции языка ассемблера NASM.

# Список литературы