Отчёт по лабораторной работе №6

Архитектура компьютера НММбд-03-24

Туева Анастасия Юрьевна

Содержание

# 1 Цель работы

Цель данной лабораторной работы - освоить арифметческие инструкции языка ассемблера NASM.

# 2 Задание

1. Символьные и численные данные в NASM
2. Выполнение арифметических операций в NASM
3. Выполнение заданий для самостоятельной работы

# 3 Выполнение лабораторной работы

1. Символьные и численные данные в NASM

С помощью команды “touch” создаю файл “lab6-1.asm” (рис. 1).



Рис. 1: Создание файла

Открываем созданный файл lab6-1.asm, вставляем в него программу вывода значения регистра “eax” (рис. 2).

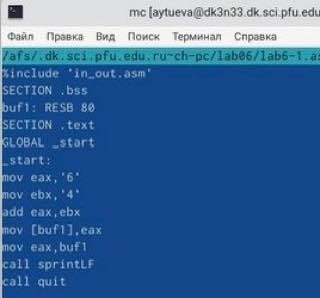


Рис. 2: Редактирование файла

Создаем исполняемый файл программы и запускаем его (рис. 3). Программа выводит символ j, потому что программа вывела символ, соответствующий по системе ASCII сумме двоичных кодов символов 4 и 6.

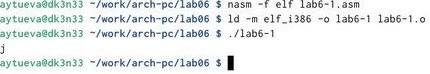


Рис. 3: Запуск исполняемого файла

Изменяем в тексте программы символы “6” и “4” на цифры 6 и 4 (рис. 4).

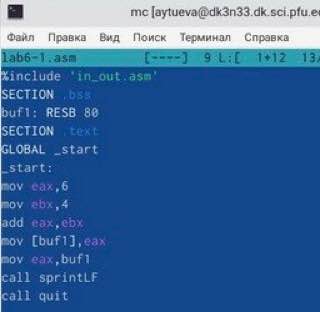


Рис. 4: Редактирование файла

Создаем новый исполняемый файл программы и запускаем его (рис. 5). Теперь вывелся символ с кодом 10, это символ перевода строки. Этот символ не отображается при выводе на экран.

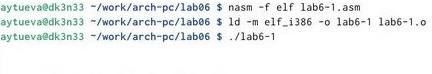


Рис. 5: Запуск исполняемого файла

Создаем новый файл lab6-2.asm с помощью команды “touch”.

Вводим в файл текст другой программы для вывода значения регистра “eax” (рис. 6).

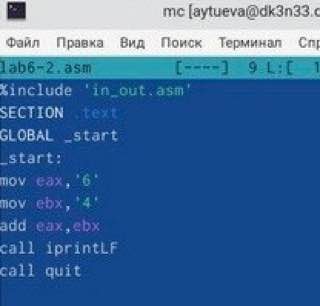


Рис. 6: Редактирование файла

Создаем и запускаем исполняемый файл “lab6-2” (рис. 7). Теперь выводится число 106.

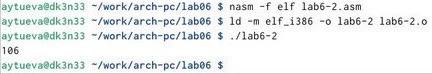


Рис. 7: Запуск исполняемого файла

Заменяем в тексте программы в файле “lab6-2.asm” символы “6” и “4” на числа 6 и 4 (рис. 8).

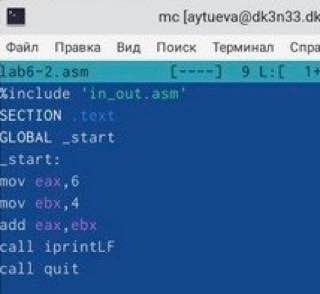


Рис. 8: Редактирование файла

Создаем и запускаем новый исполняемый файл (рис. 9).. Теперь программа складывает не соответствующие символам коды в системе ASCII, а сами числа, поэтому выводится 10.

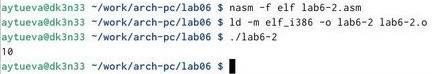


Рис. 9: Запуск исполняемого файла

Заменяем в тексте программы функцию “iprintLF” на “iprint” (рис. 10).

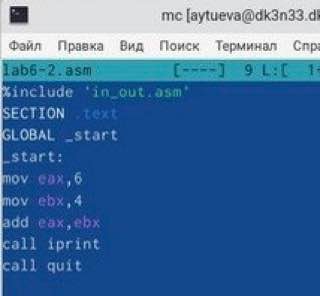


Рис. 10: Редактирование файла

Создаем и запускам новый исполняемый файл (рис. 11). Вывод не изменился.

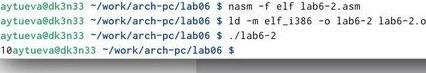


Рис. 11: Запуск исполняемого файла

1. Выполнение арифметических операций в NASM

Создаем файл “lab6-3.asm” с помощью команды “touch” (рис. 12).



Рис. 12: Создание файла

Вводим в созданный файл текст программы для вычисления значения выражения “f(x) = (5 \* 2 + 3)/3” (рис. 13).

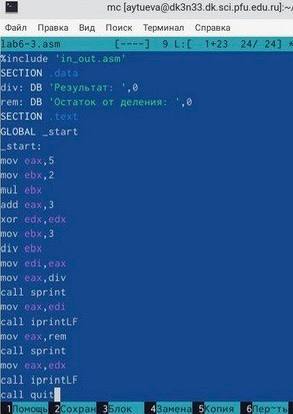


Рис. 13: Редактирование файла

Создаем исполняемый файл и запускаю его (рис. 14).

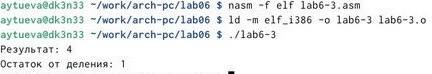


Рис. 14: Запуск исполняемого файла

Изменяем программу так, чтобы она вычисляла значение выражения “f(x) = (4 \* 6 + 2)/5” (рис. 15).

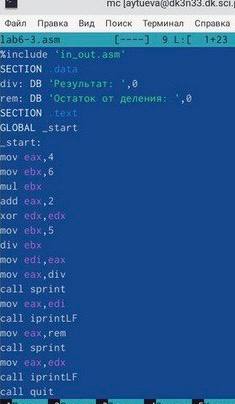


Рис. 15: Изменение программы

Создаем и запускаю новый исполняемый файл (рис. 16).

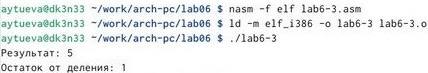


Рис. 16: Запуск исполняемого файла

Создаем файл “variant.asm” с помощью команды “touch” (рис. 17).



Рис. 17: Создание файла

Вводим в файл текст программы для вычисления варианта задания по номеру студенческого билета (рис. 18).



Рис. 18: Редактирование файла

Создаем и запускаем исполняемый файл (рис. 19). Вводим номер своего студенческого билета с клавиатуры, программа выводит, что мой вариант - 11.

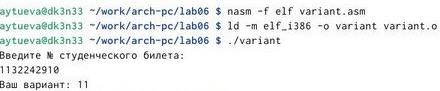


Рис. 19: Запуск исполняемого файла

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

mov eax,rem call sprint

1. Инструкция “mov ecx, x” используется, чтобы положить адрес вводимой строки x в регистр “ecx” “mov edx, 80” - запись в регистр “edx” длины вводимой строки “call sread” - вызов подпрограммы из внешнего файла, обеспечивающей ввод сообщения с клавиатуры.
2. “call atoi” используется для вызова подпрограммы из внешнего файла, которая преобразует ascii-код символа в целое число и записывает результат в регистр “eax”
3. За вычисления варианта отвечают строки:

xor edx,edx ; обнуление edx для корректной работы div mov ebx,20 ; ebx = 20 div ebx ; eax = eax/20, edx - остаток от деления inc edx ; edx = edx + 1

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 результатов вычислений отвечают строки:

mov eax,edx call iprintLF

# 4 Выполнение самостоятельной работы

Создаем файл “lab6-4.asm” с помощью команды “touch” (рис. 20).



Рис. 20: Создание файла

Открываем созданный файл для редактирования, вводим в него текст программы для вычисления значения выражения “10 \* (x + 1) - 10” (рис. 21). Это выражение было под вариантом 11.

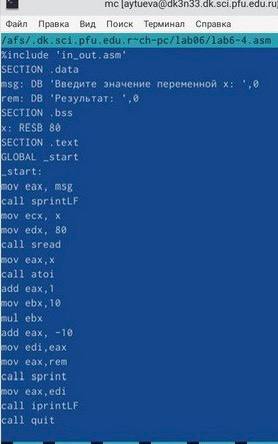


Рис. 21: Написание программы

Создаем и запускаем исполняемый файл (рис. 22). При вводе значения 1, вывод - 10, а при вводе значения 7, вывод - 70.



Рис. 22: Запуск исполняемого файла

# 5 Выводы

При выполнении данной лабораторной работы я освоила арифметические инструкции языка ассемблера NASM.