Отчёт по лабораторной работе №6

дисциплина: архитектура компьютера

Бражко Александра Александровна

Содержание

# 1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

# 2 Задание

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

# 3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. 1 приведено краткое описание стандартных каталогов Unix.

Таблица 1: Описание некоторых каталогов файловой системы GNU Linux

| Имя каталога | Описание каталога |
| --- | --- |
| / | Корневая директория, содержащая всю файловую |
| /bin | Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям |
| /etc | Общесистемные конфигурационные файлы и файлы конфигурации установленных программ |
| /home | Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя |
| /media | Точки монтирования для сменных носителей |
| /root | Домашняя директория пользователя root |
| /tmp | Временные файлы |
| /usr | Вторичная иерархия для данных пользователя |

Более подробно про Unix см. в [1–4].

# 4 Символьные и численные данные в NASM

Создаём каталог для программ лабораторной работы № 6, переходим в него и создаём файл-lab6.asm (рис. 1).

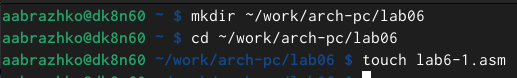


Рис. 1: Создание и переход в каталог, создание файла

Копируем в текущий каталог файл in\_out.asm (рис. 2).

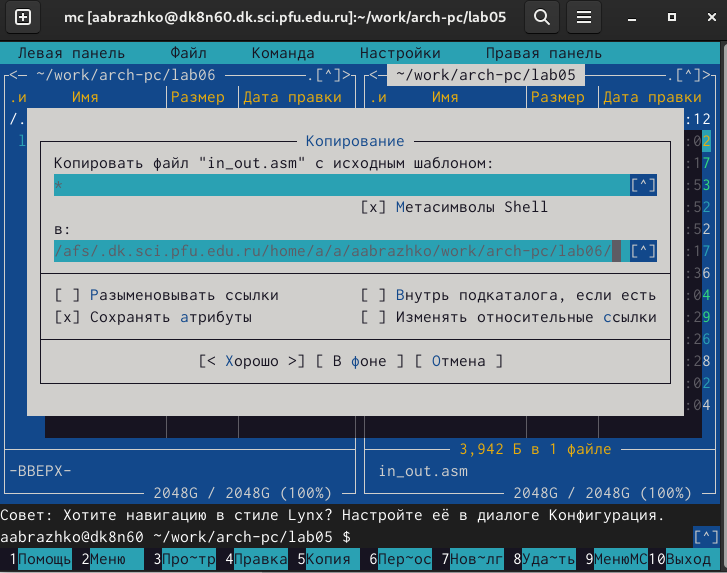


Рис. 2: Копирование

Открываем созданный файл lab6-1.asm, вставляем в него программу вывода значения регистра eax (рис. 3).

|  |
| --- |
| Рис. 3: Ввод программы |

Рис. 3: Ввод программы

Создаём и запускаем исполняемый файл. Программа в итоге должна вывести j (рис. 4).

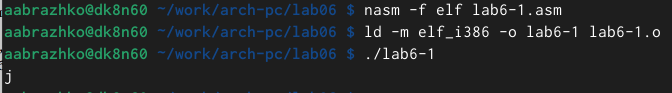


Рис. 4: Работа программы

В программе изменяем ‘4’ и ‘6’ на 4 и 6 (рис. 5).

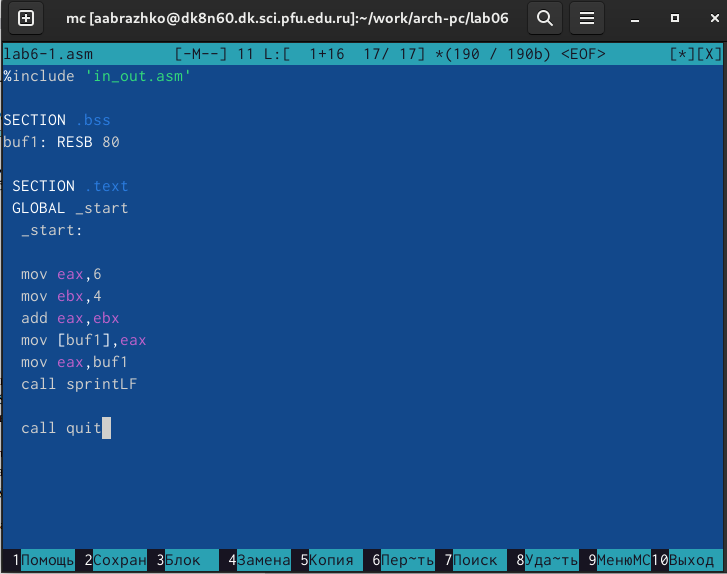


Рис. 5: Замена

Создаём и запускаем исполняемый файл. Программа в итоге должна вывести символ с кодом 10, но этот символ не отображается при выводе на экран (рис. 6).

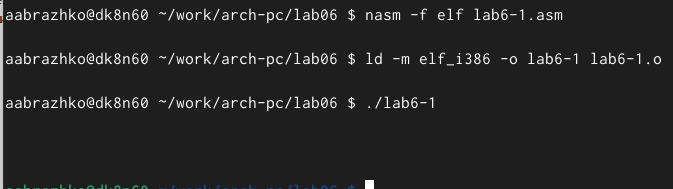


Рис. 6: Работа программы

Создаём файл lab6-2.asm в каталоге ~/work/arch-pc/lab06 (рис. 7).

Рис. 7: Создание файла

Рис. 7: Создание файла

Вводим в него текст программы из листинга 6.2 (рис. 8).

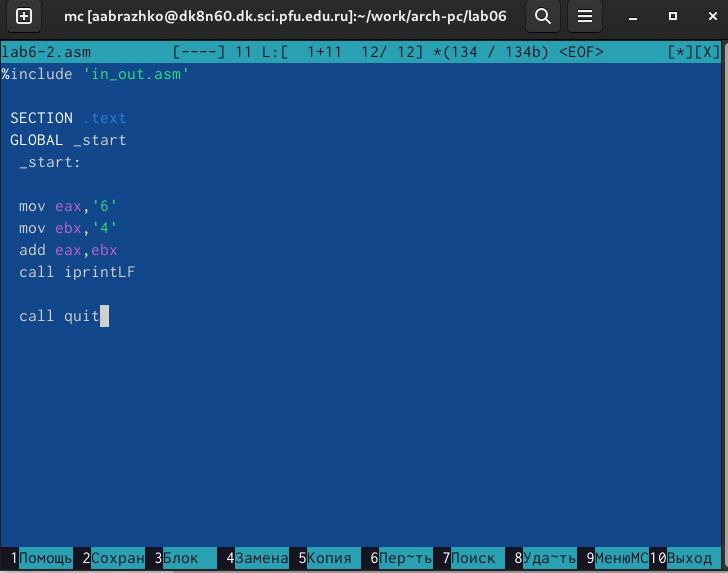


Рис. 8: Ввод программы

Создаём и запускаем исполняемый файл. Программа в итоге должна вывести 106 (рис. 9).

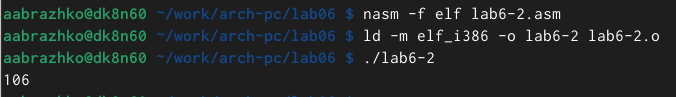


Рис. 9: Работа программы

В программе изменяем ‘4’ и ‘6’ на 4 и 6 (рис. 10).

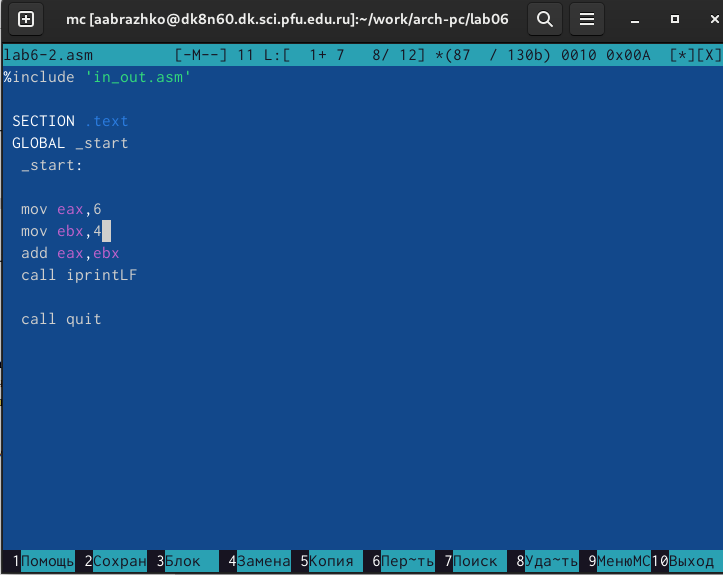


Рис. 10: Замена

Создаём и запускаем исполняемый файл. Программа в итоге должна вывести 10 (рис. 11).

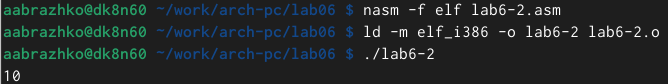


Рис. 11: Работа программы

Заменяем функцию iprintLF на iprint. Создаём исполняемый файл и запускаем его. Разницей между ipintLF и iprint является количество строк вывода (рис. 12).

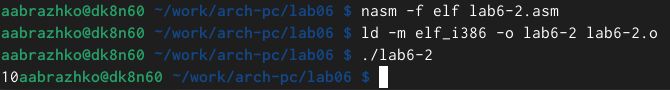


Рис. 12: Результат после корректировки

# 5 Выполнение арифметических операций в NASM

Создаём файл lab6-3.asm в каталоге ~/work/arch-pc/lab06 (рис. 13).

Рис. 13: Создание файла

Рис. 13: Создание файла

Открываем созданный файл lab6-3.asm, вставляем в него текст программы из листинга 6.3 (рис. 14).

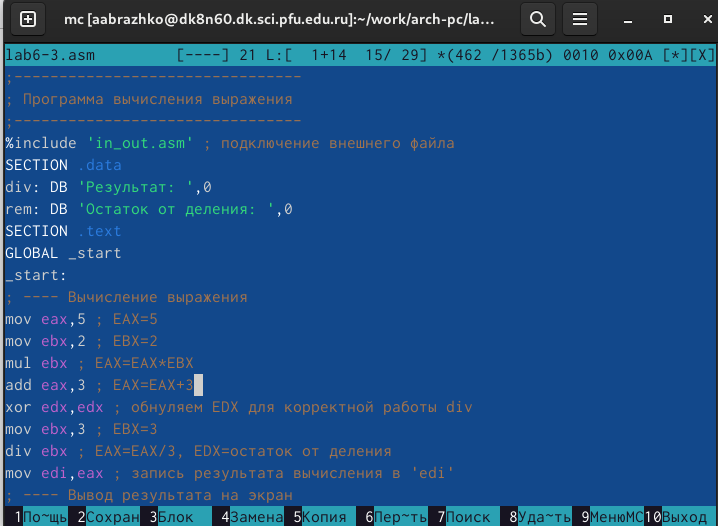


Рис. 14: Открытие файла и написание программы

Создаём и запускаем исполняемый файл (рис. 15).

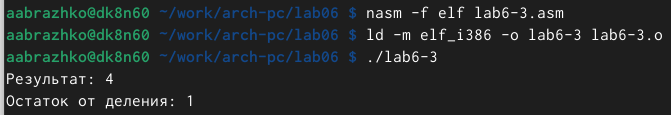


Рис. 15: Работа команд

Изменяяем текст программы для вычисления выражения 𝑓(𝑥) = (4 ∗ 6 + 2)/5 (рис. 16).

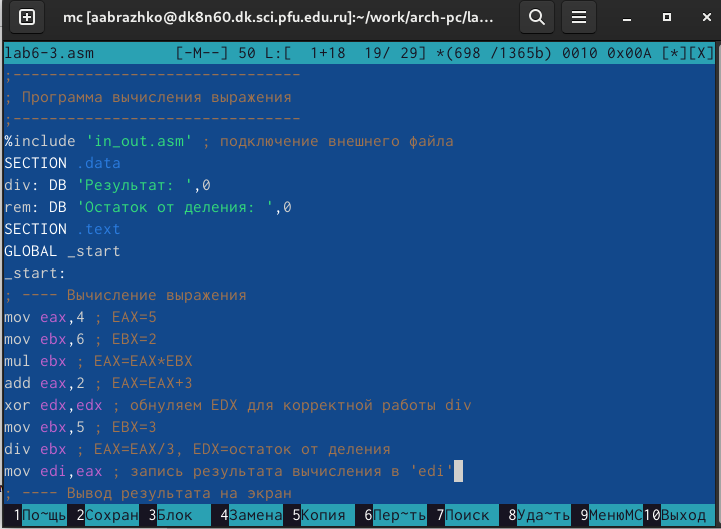


Рис. 16: Изменяем текст

Создаём и запускаем исполняемый файл (рис. 17).

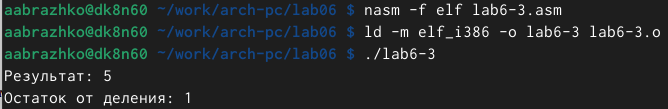


Рис. 17: Работа команд

Создаём файл variant.asm в каталоге ~/work/arch-pc/lab06 (рис. 18).

Рис. 18: Создание файла

Рис. 18: Создание файла

Открываем созданный файл variant.asm, вставляем в него текст программы из листинга 6.4 (рис. 19).

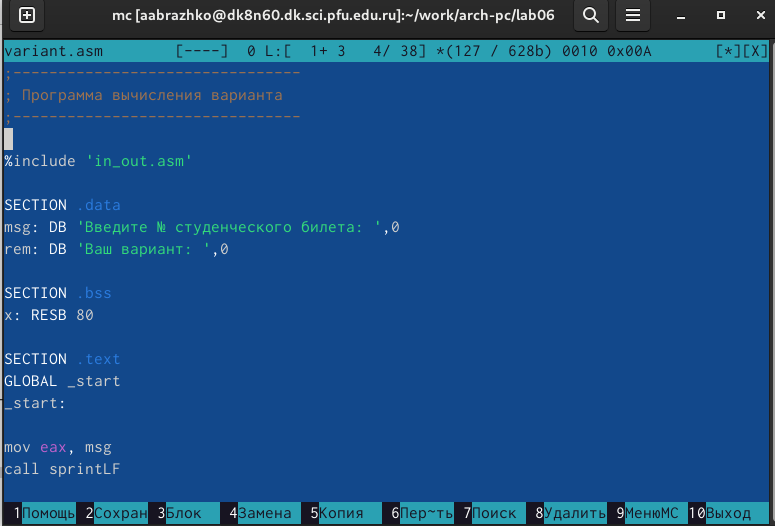


Рис. 19: Открытие файла и написание программы

Создаём и запускаем исполняемый файл (рис. 20).

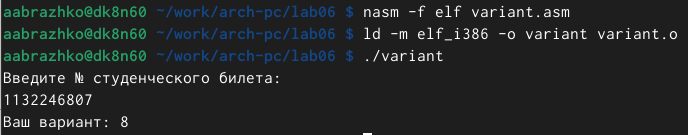


Рис. 20: Работа команд

# 6 Ответы на вопросы

1. Какие строки листинга 6.4 отвечают за вывод на экран сообщения ‘Ваш вариант:’?

mov eax,rem call sprint

1. Для чего используется следующие инструкции? mov ecx, x mov edx, 80 call sread

инструкция mov ecx, x используется для того, 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.

1. Для чего используется инструкция “call atoi”?

Call atoi используется для вызова подпрограммы из внешнего файла, которая преобразует ascii-код символа в целое число и записывает результат в регистр eax.

1. Какие строки листинга 6.4 отвечают за вычисления варианта?

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

1. В какой регистр записывается остаток от деления при выполнении инструкции “div ebx”?

При выполнении инструкции div ebx остаток от деления записывается в регистр edx.

1. Для чего используется инструкция “inc edx”?

Инструкция inc edx увеличивает значение регистра edx на 1.

1. Какие строки листинга 6.4 отвечают за вывод на экран результата вычислений?

mov eax,edx call iprintLF

# 7 Выполнение заданий для самостоятельной работы

Создаём файл lab6-4.asm в каталоге ~/work/arch-pc/lab06 (рис. 21).

Рис. 21: Создание файла

Рис. 21: Создание файла

Открываем созданный файл lab6-4.asm и вводим в него текст программы для вычисления значения выражения для 8 варианта: (11+x)\*2-6 (рис. 22).

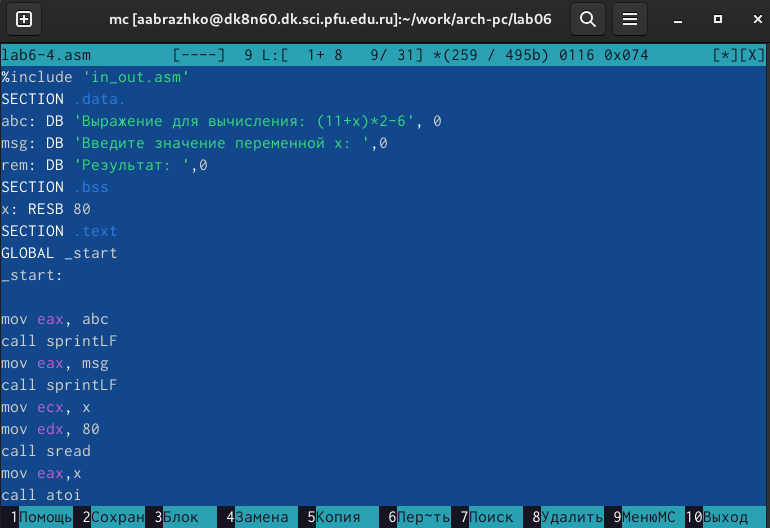


Рис. 22: Открытие файла и написание программы

Создаём и запускаем исполняемый файл для х=1 и x=9 (рис. 23, рис. 24).

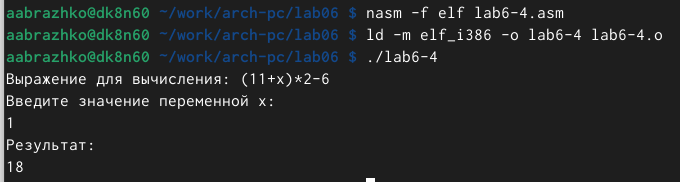


Рис. 23: Работа команды для х=1

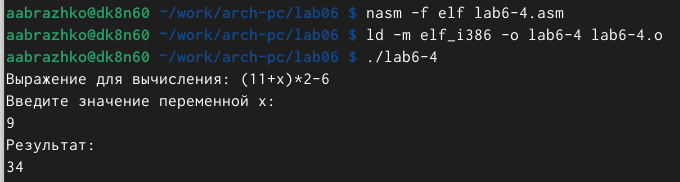


Рис. 24: Работа команды для x=9

# 8 Выводы

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

# Список литературы

1. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.

2. Robbins A. Bash Pocket Reference. O’Reilly Media, 2016. 156 с.

3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.

4. Newham C. [Learning the bash Shell: Unix Shell Programming](http://www.amazon.com/Learning-bash-Shell-Programming-Nutshell/dp/0596009658). O’Reilly Media, 2005. 354 с.