Шаблон отчёта по лабораторной работе

6

Разанацуа Сара Естэлл

Содержание

# 1 Цель работы

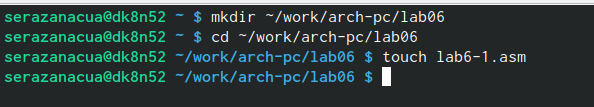
* Цель данной лабораторной работы - освоение арифметческих инструкций языка ассемблера NASM..

# 2 Задание

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

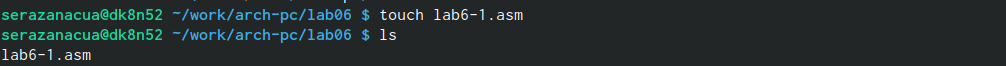
# 3 Выполнение лабораторной работы

* 1. Символьные и численные данные в NASM
* С помощью утилиты mkdir создаю директорию, в которой буду создавать файлы с программами для лабораторной работы №6 (рис. [??]). Перехожу в созданный каталог с помощью утилиты cd.(рис. ??).



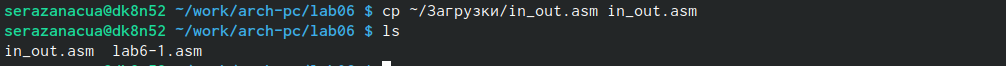
создание директории

* С помощью утилиты touch создаю файл lab6-1.asm.(рис. ??).



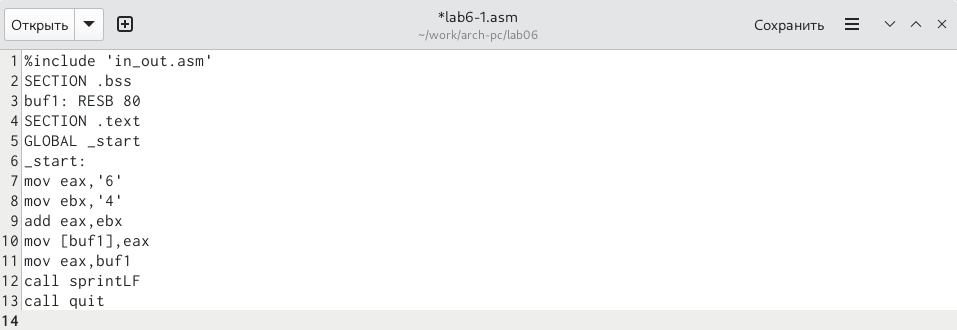
создание файла

* Копирую в текущий каталог файл in\_out.asm с помощью утилиты cp, т.к. он будет использоваться в других программах (рис. ??).



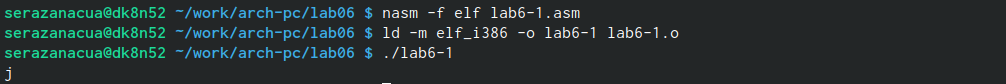
создание копии файла

* Открываю созданный файл lab6-1.asm, вставляю в него программу вывода значения регистра eax (рис. ??).



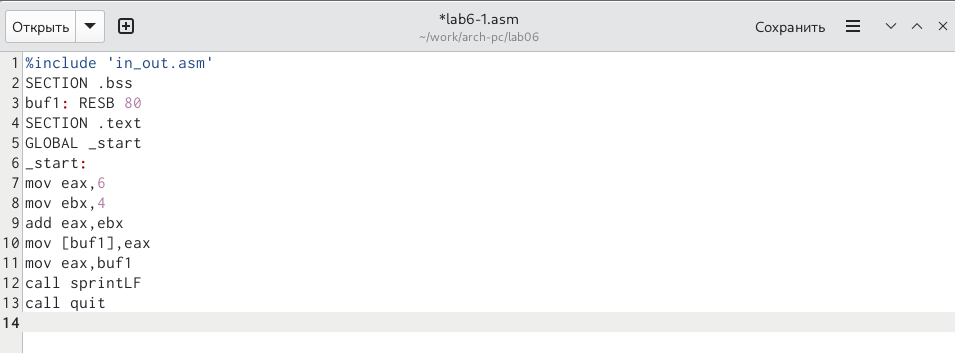
редактирование файла

* Создаю исполняемый файл программы и запускаю его (рис. [??]). Вывод программы: символ j, потому что программа вывела символ, соответствующий по системе ASCII сумме двоичных кодов символов 4 и 6. (рис. ??).



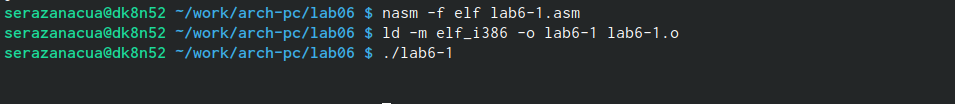
запуск исполняемого файла

* Изменяю в тексте программы символы “6” и “4” на цифры 6 и 4 (рис. ??).



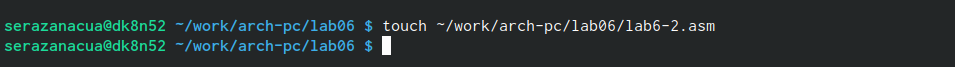
редактирование файла

* Создаю новый исполняемый файл программы и запускаю его (рис. [??]). Теперь вывелся символ с кодом 10, это символ перевода строки, этот символ не отображается при выводе на экран.(рис. ??).



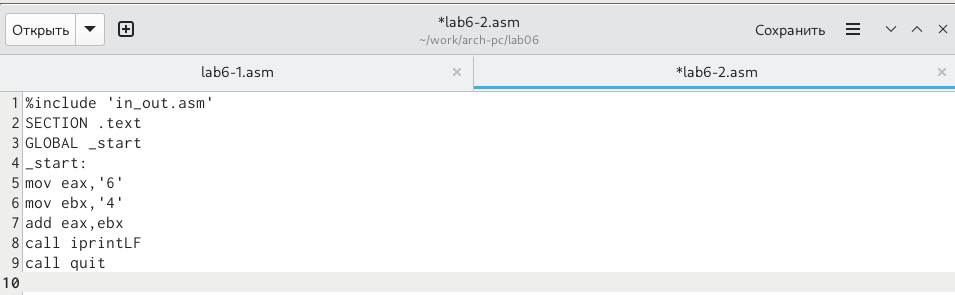
запуск исполняемого файла

* Создаю новый файл lab6-2.asm с помощью утилиты touch (рис. ??).



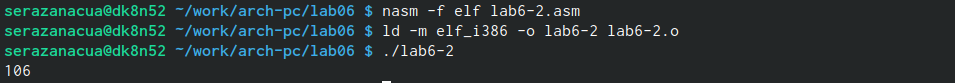
создание копии файла

* Ввожу в файл текст другойпрограммы для вывода значения регистра eax (рис. ??).



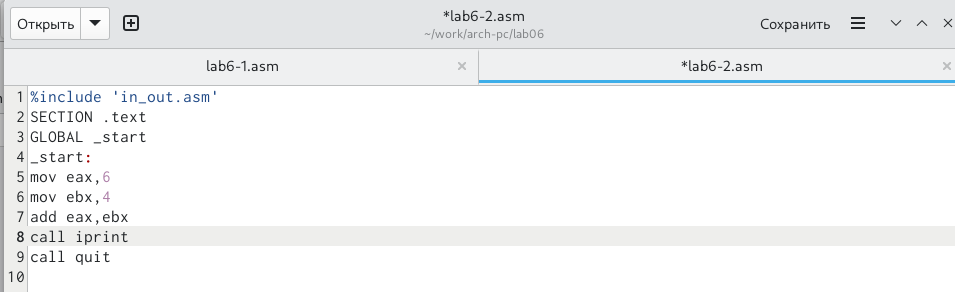
редактирование файла

* Создаю и запускаю исполняемый файл lab6-2 (рис. [??]). Теперь вывод число 106, потому что программа позволяет вывести именно число, а не символ, хотя все еще происходит именно сложение кодов символов “6” и “4”. (рис. ??).



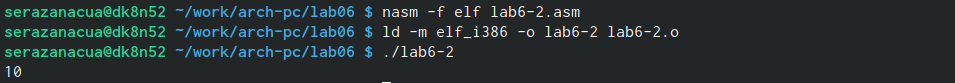
запуск исполняемого файла

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



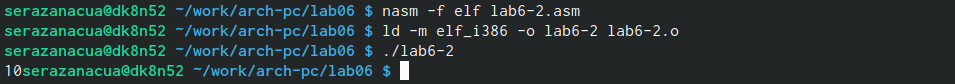
редактирование файла

-Создаю и запускаю новый исполняемый файл (рис. [??]).. Теперь программа складывает не соответствующие символам коды в системе ASCII, а сами числа, поэтому вывод 10. (рис. ??).



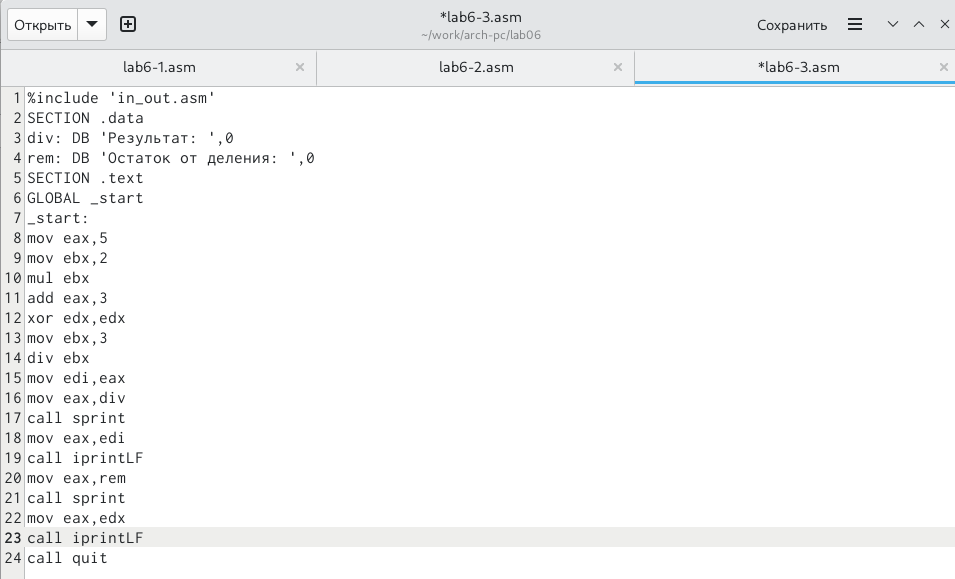
запуск исполняемого файла

* Заменяю в тексте программы функцию iprintLF на iprint
* Создаю и запускаю новый исполняемый файл (рис. [??]). Вывод не изменился, потому что символ переноса строки не отображался, когда программа исполнялась с функцией iprintLF, а iprint не добавляет к выводу символ переноса строки, в отличие от iprintLF (рис. ??).



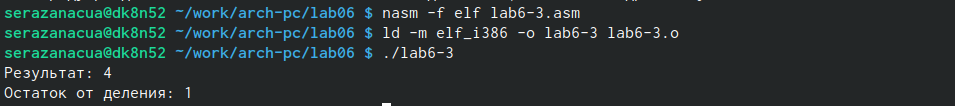
запуск исполняемого файла

* 1. Выполнение арифметических операций в NASM
* Создаю файл lab6-3.asm с помощью утилиты touch . Ввожу в созданный файл текст программы для вычисления значения выражения f(x) = (5 \* 2 + 3)/3 (рис. ??).



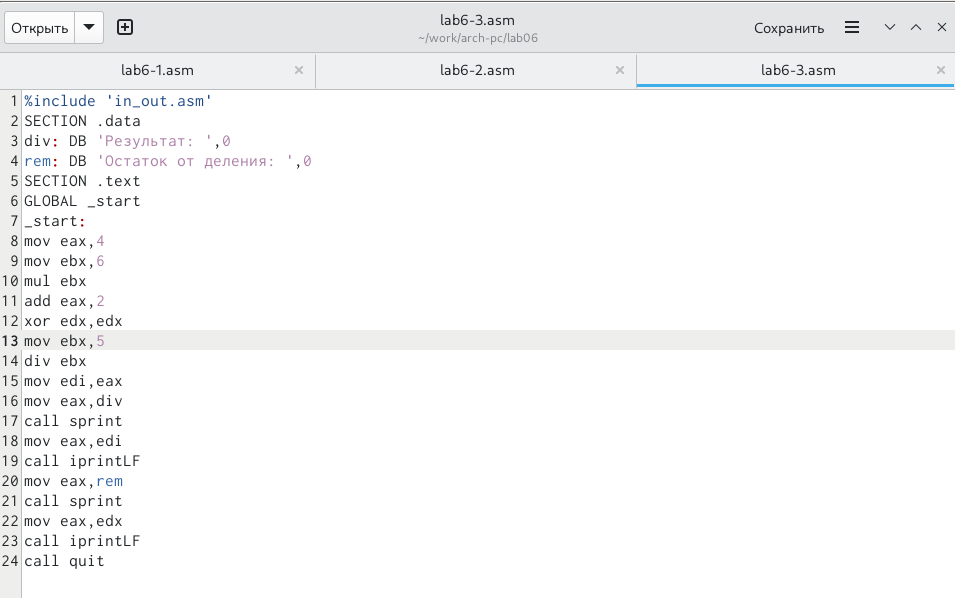
редактирование файла

* Создаю исполняемый файл и запускаю его (рис. ??).



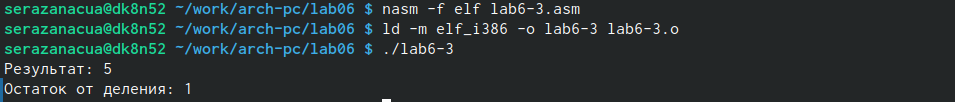
запуск исполняемого файла

* Изменяю программу так, чтобы она вычисляла значение выражения f(x) = (4 \* 6 + 2)/5 (рис. ??).



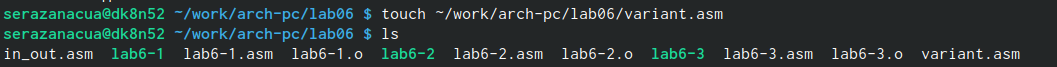
изменение программы

* Создаю и запускаю новый исполняемый файл (рис. [??]). Я посчитала для проверки правильности работы программы значение выражения самостоятельно, программа отработала верно. (рис. ??).



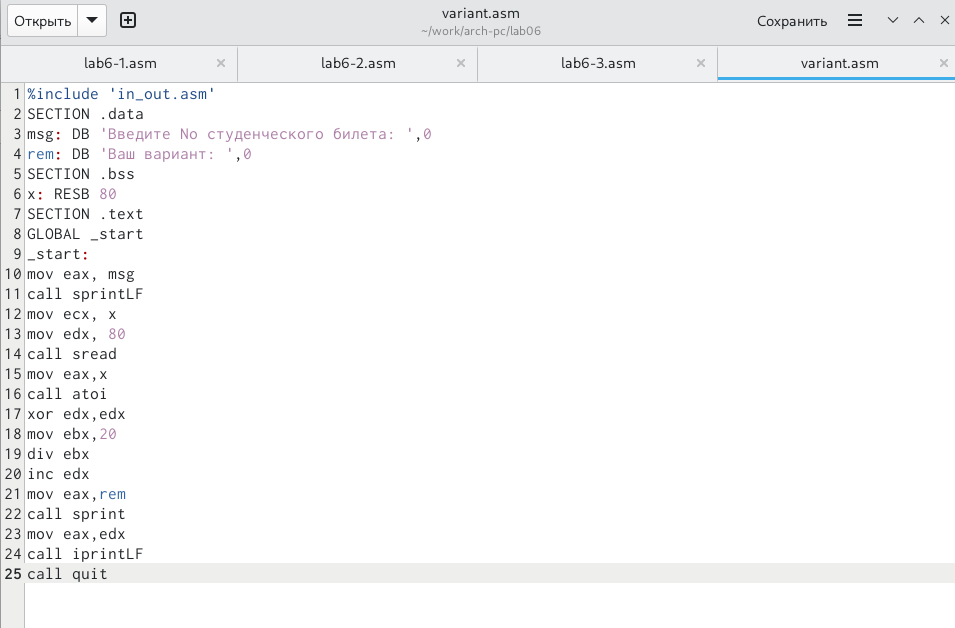
запуск исполняемого файла

* Создаю файл variant.asm с помощью утилиты touch. (рис. ??).



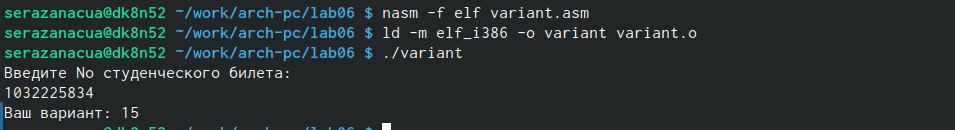
создание копии файла

* Ввожу в файл текст программы для вычисления варианта задания по номеру студенческого билета.(рис. ??).



редактирование файла

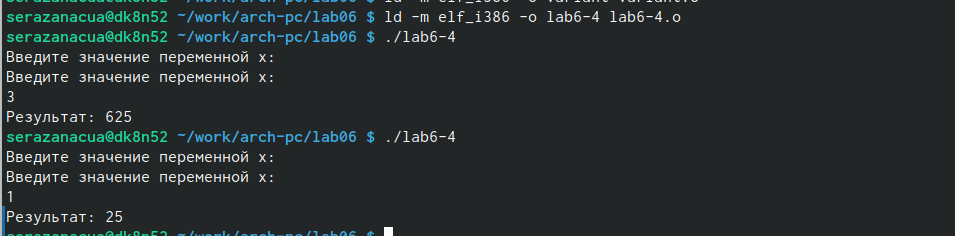
* Создаю и запускаю исполняемый файл (рис. [??]). Ввожу номер своего студ. билета с клавиатуры, программа вывела, что мой вариант - 15. рис. ??).



запуск исполняемого файла

## 3.1 Ответы на вопросы по программе

* 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
* 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
  1. Выполнение заданий для самостоятельной работы
* Создаю и запускаю исполняемый файл. рис. ??).



запуск исполняемого файла

# 4 Выводы

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

# Список литературы