Отчёт по лабораторной работе №6

Арифметические операции в NASM

Малкина Дарья Александровна

Содержание

# 1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

# 2 Выполнение лабораторной работы

## 2.1 Символьные и численные данные в NASM

1. Создадим каталог для программ лабораторной работы № 6, в нем создадим файл lab6-1.asm, в который введём текст программы из листинга 6.1.:

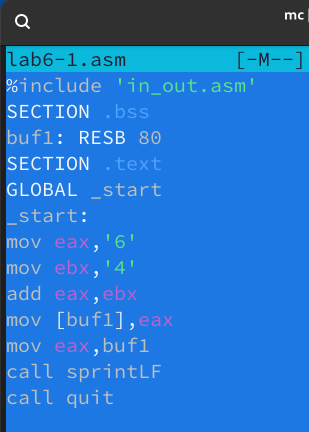


Рис. 1: Вводим текст программы lab6-1

Создаём исполняемый файл и запускаем его, программа выводит символ j:

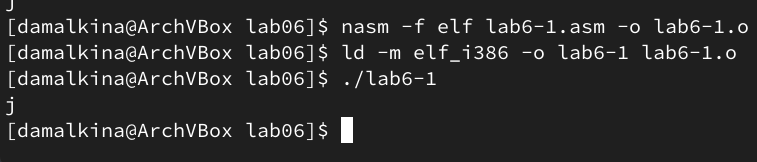


Рис. 2: Результат программы - j

1. Далее изменим текст программы и вместо символов, запишем в регистры числа:

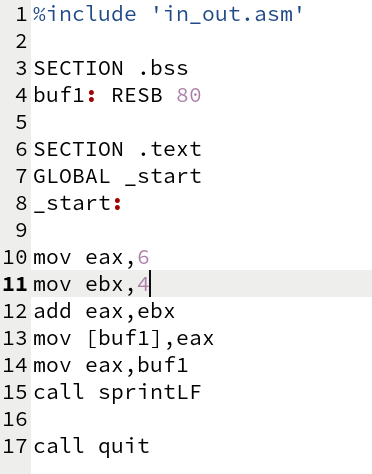


Рис. 3: Редактируем текст программы lab6-1

Создаём исполняемый файл и запускаем его, программа выводит символ с кодом 10. Пользуясь таблицей ASCII определяем, что коду 10 соответствует символу перевода строки, что мы и наблюдаем в результате:

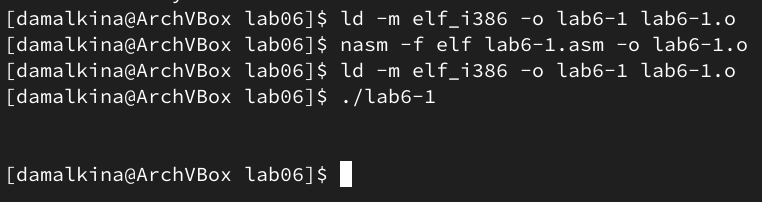


Рис. 4: Результат программы - перевод строки

1. Создаём файл lab6-2.asm в каталоге ~/work/arch-pc/lab06 и вводим в него текст программы, преобразованный с использованием функций из внешнего файла in\_out.asm:

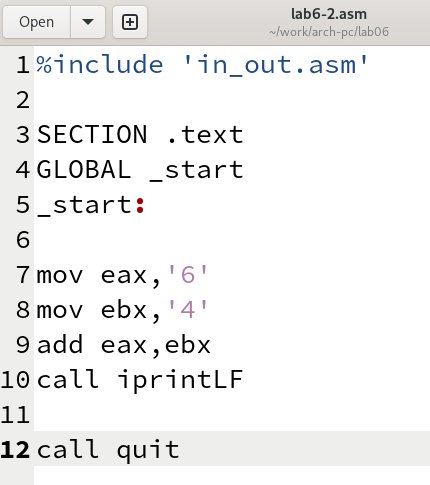


Рис. 5: Программа lab6-2 с функциями внешнего файла

В результате работы программы получаем число 106:

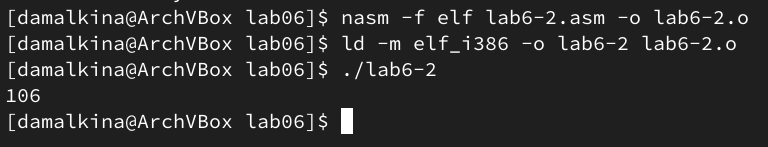


Рис. 6: Результат программы - 106

1. Изменим текст программы - вместо символов, запишем в регистры числа:

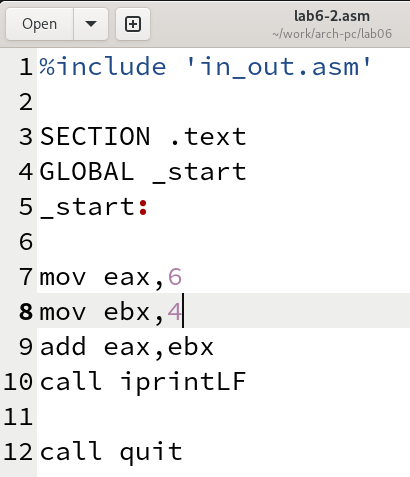


Рис. 7: Редактируем текст программы lab6-2

Создаём исполнительный файл и запускаем его, при исполнении программы получили результат 10:

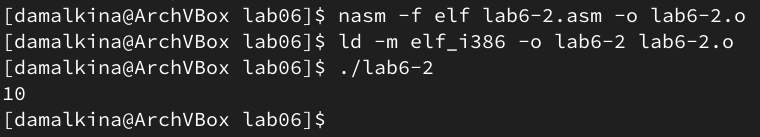


Рис. 8: Результат программы - 10

Заменим функцию iprintLF на iprint, тогда после запуска программа не переводит строку по завершении работы:

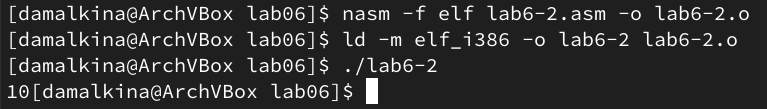


Рис. 9: Результат программы

## 2.2 Выполнение арифметических операций в NASM

1. Создадим файл lab6-3.asm и введем в него программу вычисления арифметического выражения f(x) = (5 ∗ 2 + 3)/3:

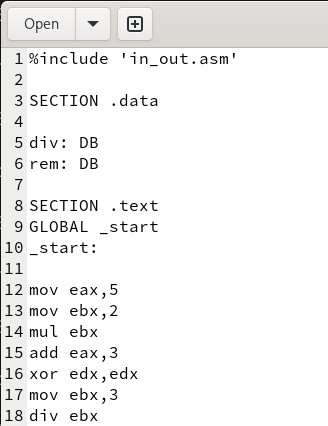


Рис. 10: Вводим текст программы lab6-3

Создаём исполнительный файл и запускаем его:

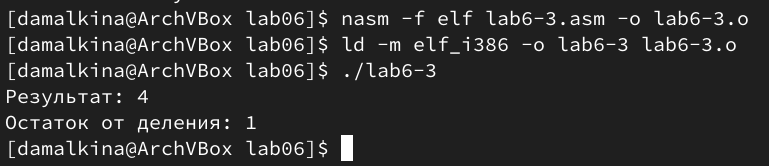


Рис. 11: Результат вычисления f(x) = (5 ∗ 2 + 3)/3

Теперь изменим текст программы, чтобы получить результат вычисления выражения f(x) = (4 ∗ 6 + 2)/5:

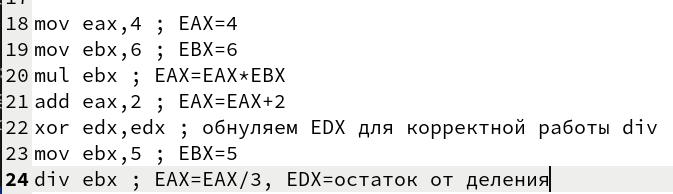


Рис. 12: Редактируем текст программы lab6-3

Создаём исполнительный файл и запускаем его:

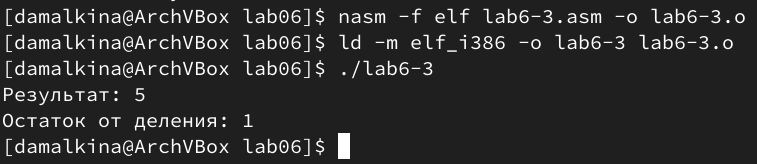


Рис. 13: Результат вычисления f(x) = (4 ∗ 6 + 2)/5

1. Создадим файл variant.asm, введем в файл текст программы, которая будет вычислять вариант задания по номеру студенческого билета. Создаем исполнительный файл var и запускаем его:

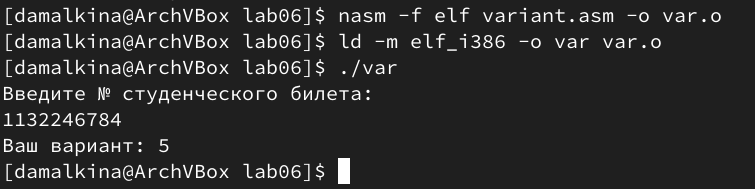


Рис. 14: Результат программы var

Проверим результат аналитически: 56612339 \* 20 = 1132246780 1132246784 - 1132246780 = 4 (остаток от деления) Получаем 4 + 1 = 5 выриант задания.

## 2.3 Оветы на ворпросы

1. За вывод на экран сообщения отвечают следующие строки:

rem: DB ‘Ваш вариант:’,0 … mov eax,rem call sprint

1. Строки кода ниже отвечают за считывание строки с клавиатуры и сохранение ее в буфер для дальнейшей обработки:

mov ecx, x mov edx, 80 call sread

1. Инструкция “call atoi” используется для вызова функции “atoi”, которая реобразует ascii-код символа в целое число и записает результат в регистр eax.
2. За вычисления варианта отвечают следующие строки кода:

mov eax,x call atoi … 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” увеличивает значение в регистре edx на 1. То есть, после деления значение в edx будет остаток от деления, к этому остатку добавляется 1, чтобы получить вариант.
3. За вывод на экран результата вычислений отвечают строки:

mov eax,rem call sprint mov eax,edx call iprintLF

# 3 Выполнение задания для самостоятельной работы

Напишем программу вычисления выражения y = f(x) = (9\*x - 8)/8. Возмём за основу текст программы var. Создадим копию файла var.asm файл var5.asm. Внесём изменения:

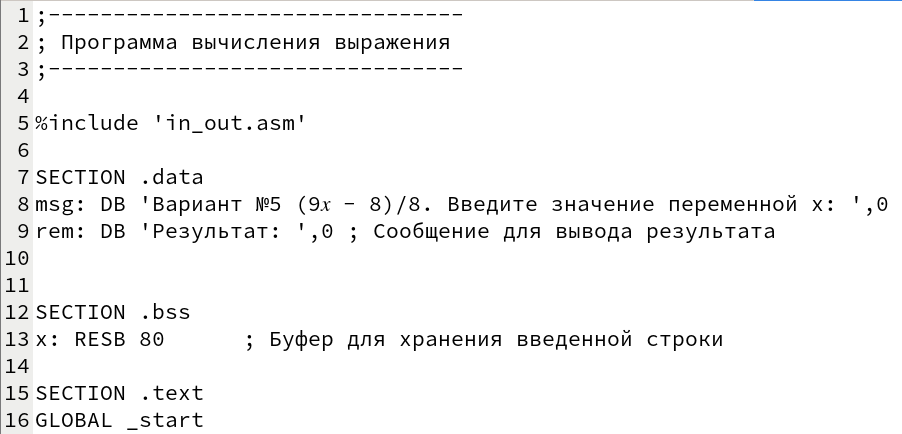


Рис. 15: Вводим текст сообщений

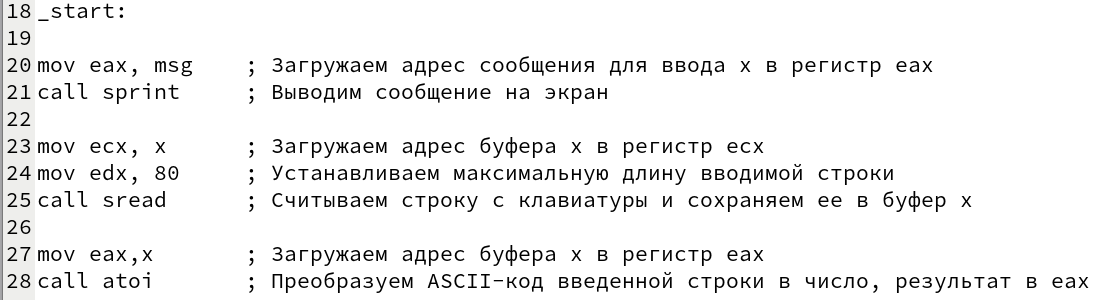


Рис. 16: Старт программы, запись и преобразование строки с клавиатуры

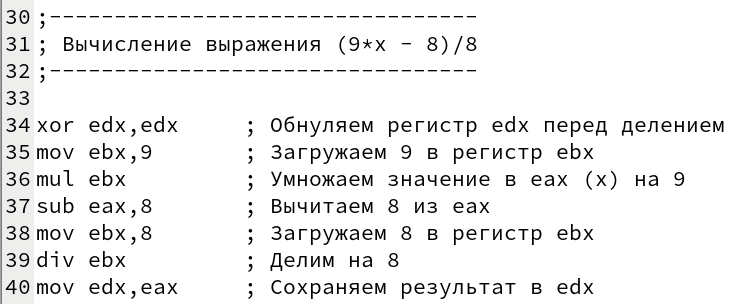


Рис. 17: Вычисление выражения

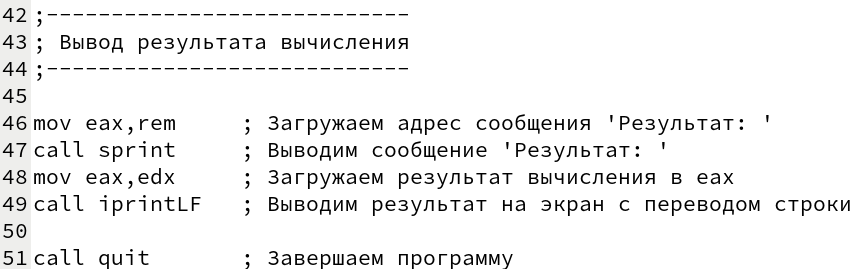


Рис. 18: Вывод результата и завершение программы

Создаём исполняемый файл и проверяем его работу для значений x1=8 и x2=64:

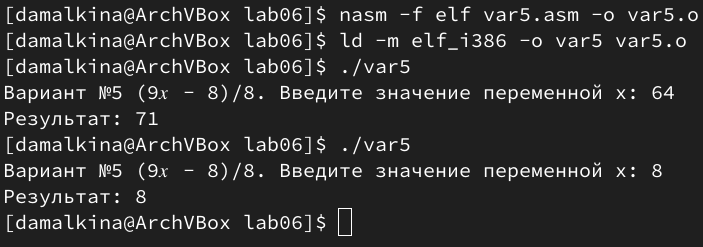


Рис. 19: Результат программы var5

Проверим работу программы посчитав аналитически значения выражения при x1=8 и x2=64:

x1=8: (9*8 - 8)/8 = 64/8 = 8 x2=64: (9*64 - 8)/8 = 8\*(72 - 1)/8 = 71

Программа работает верно.

# 4 Выводы

В ходе выполнения лабораторной работы мы узнали об арифметических инструкциях языка ассемблера NASM и научились их использовать при написании текста программы. Также научились создавать ассемблерную программу, которая считывает данные от пользователя, выполняет арифметические операции и выводит результат на экран. Мы узнали, как выполнять операции умножения и деления, а также как использовать подпрограммы для ввода-вывода. Кроме того, мы научились правильно обрабатывать данные, которые пользователь вводит с клавиатуры, преобразуя их из ASCII-кода в числовое значение.

# Список литературы