Отчёта по лабораторной работе №6

Арифметические операции в NASM.

Сидорова Арина Валерьевна

Содержание

# 1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM.

# 2 Выполнение лабораторной работы

## 2.1 1.Cимвольные и численные данные в NASM

Создаем каталог для программ лабораторной работы № 6, переходим в него и создаем файл lab6-1.asm: (рис. fig. 1).

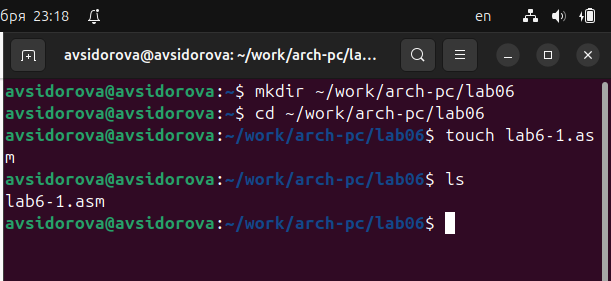


Рис. 1: Создаем каталог и файл, проверяем

Введем в файл lab6-1.asm текст программы из листинга 6.1.(рис. fig. 2).

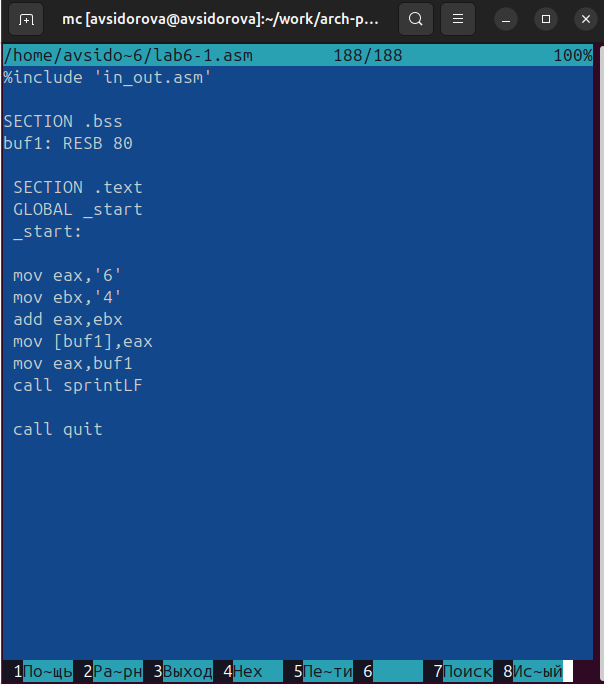


Рис. 2: Вводим программу

Создаем исполняемый файл и запускаем его (рис. fig. 3).

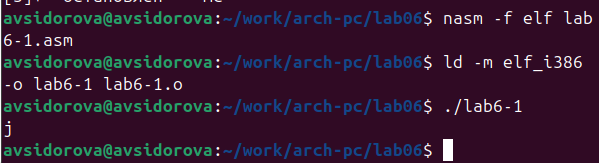


Рис. 3: Выводим результат

Далее изменим текст программы и вместо символов, запишем в регистры числа. Исправим текст программы ((рис. fig. 4).

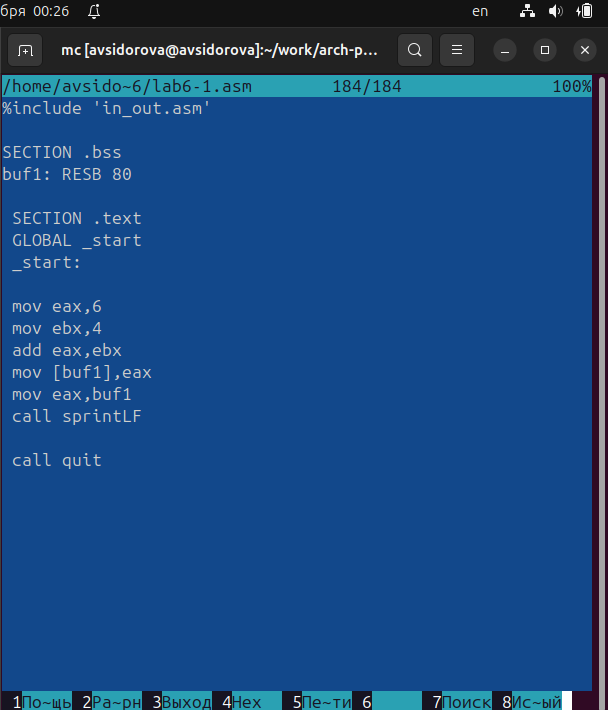


Рис. 4: Изменяем файл

Создаем исполняемый файл и запускаем его (рис. fig. 5).

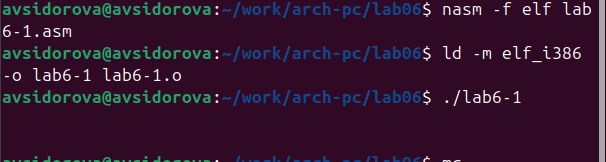


Рис. 5: Выводим результат

Создаем файл lab6-2.asm в каталоге ~/work/arch-pc/lab06(рис. fig. 6).

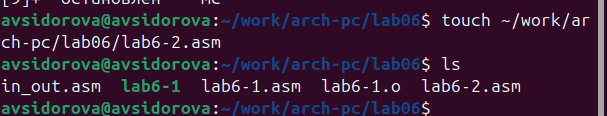


Рис. 6: Создаем файл lab6-2.asm

Введим в него текст программы из листинга 6.2. (рис. fig. 7).

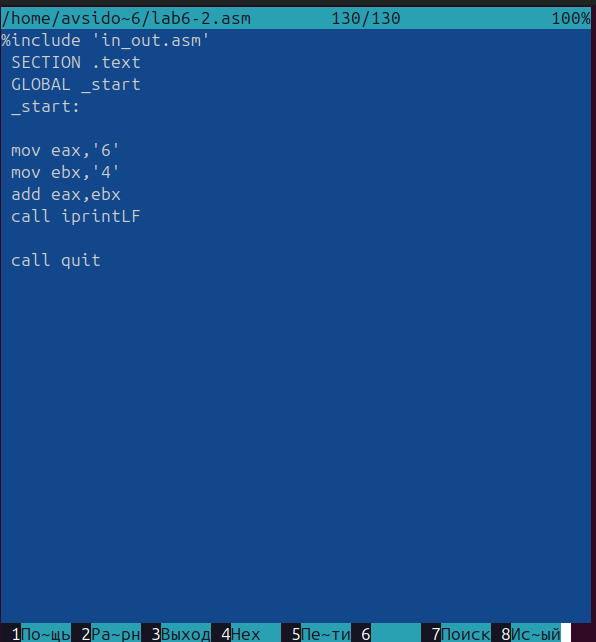


Рис. 7: Вводим текст в файл

Создаем исполняемый файл и запускаем его (рис. fig. 8).

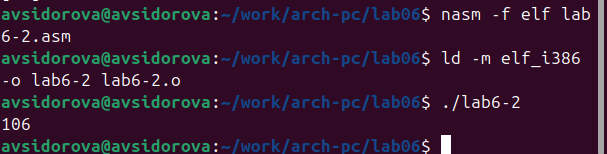


Рис. 8: Смотрим на работу программы

Аналогично предыдущему примеру изменим символы на числа. (рис. fig. 9).

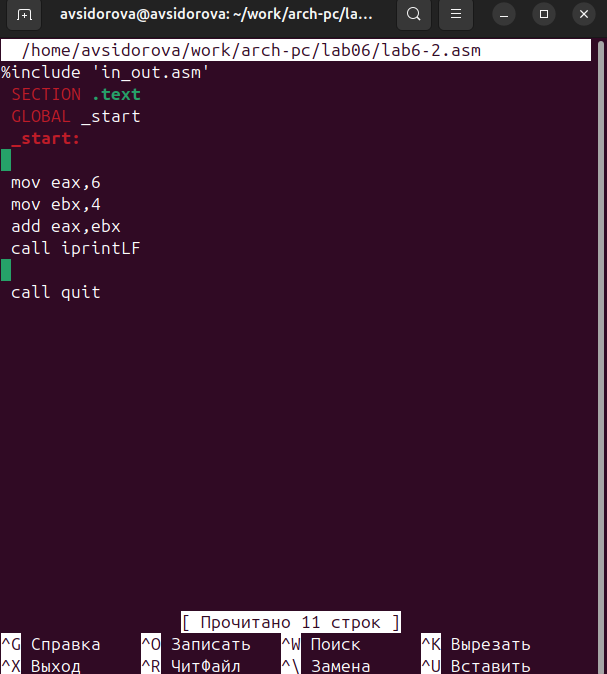


Рис. 9: Изменяем файл

Создаем исполняемый файл и запускаем его (рис. fig. 10).

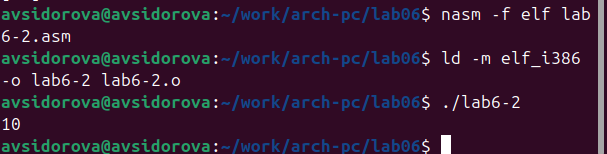


Рис. 10: Смотрим на работу программы

В результате программы получаем число 10

Заменим функцию iprintLF на iprint.(рис. fig. 11).

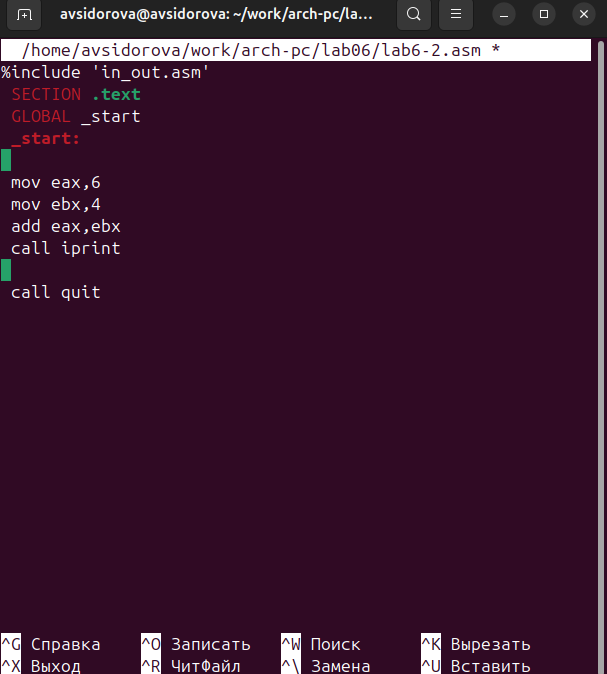


Рис. 11: Изменяем файл

Создаем исполняемый файл и запускаем его (рис. fig. 12).

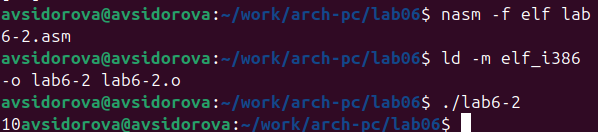


Рис. 12: Смотрим на работу программы

Вывод функций iprintLF и iprint отличаются тем, что iprintLF делает перенос на новую строку.

## 2.2 2.Выполнение арифметических операций в NASM

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

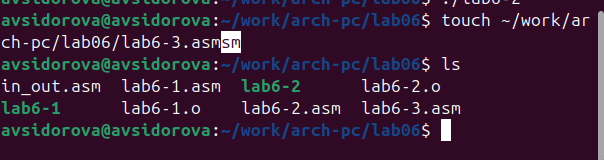


Рис. 13: Создаем файл и проверяем

Внимательно изучим текст программы из листинга 6.3 и введем в lab6-3.asm.(рис. fig. 14).

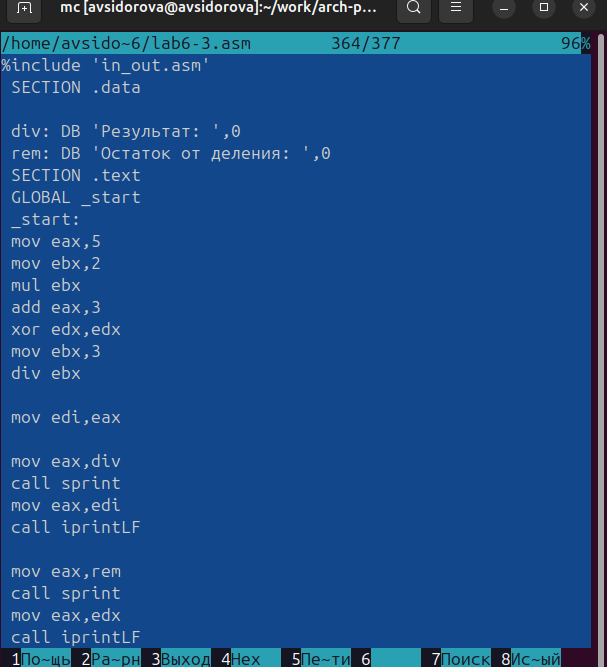


Рис. 14: Вводим программу

Создаем исполняемый файл и запускаем его (рис. fig. 15).

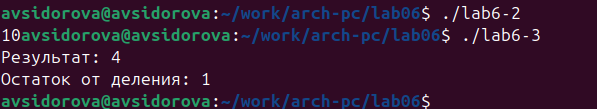


Рис. 15: Смотрим на результат работы программы

Изменим текст программы для вычисления выражения f(x) = (4\*6+2)/5 (рис. fig. 16).

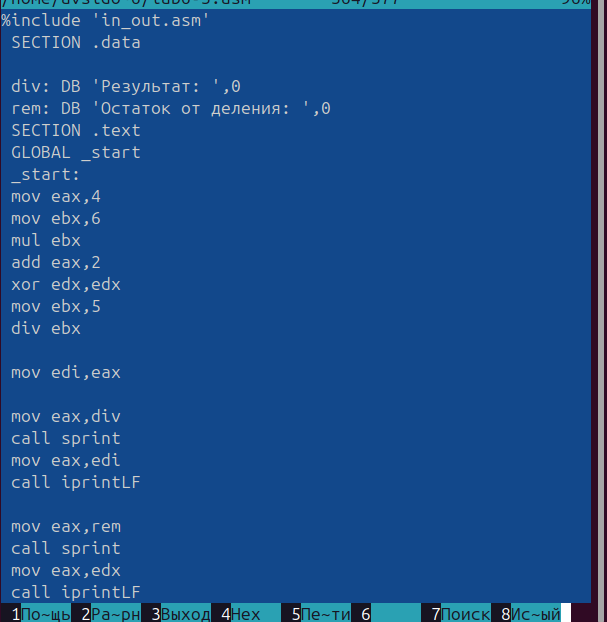


Рис. 16: Изменяем файл

Создадим исполняемый файл и проверим его работу. (рис. fig. 17).

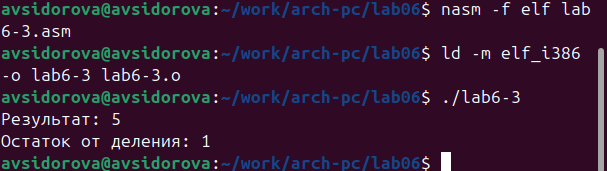


Рис. 17: Смотрим на результат работы программы

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

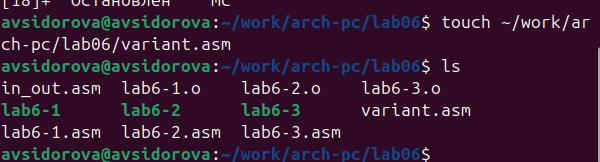


Рис. 18: Создаем файл

Внимательно изучим текст программы из листинга 6.4 и введем в файл variant.asm.(рис. fig. 19).

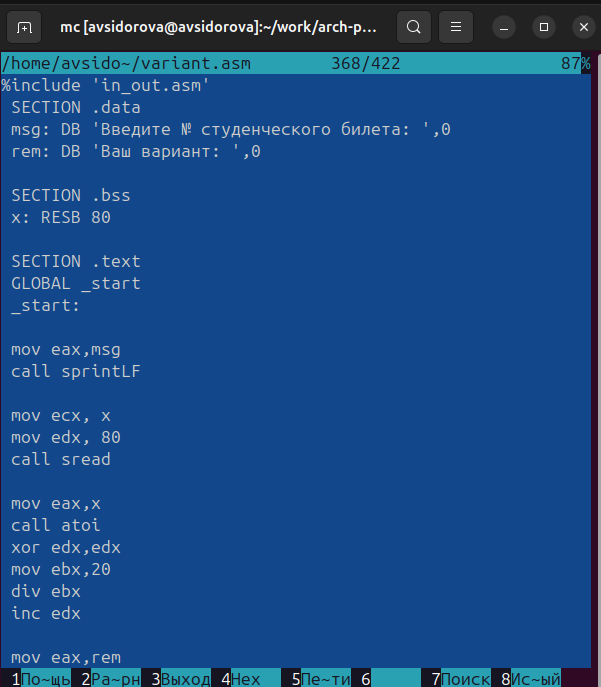


Рис. 19: Вводим програму

Создадим исполняемый файл и запустим его. (рис. fig. 20).

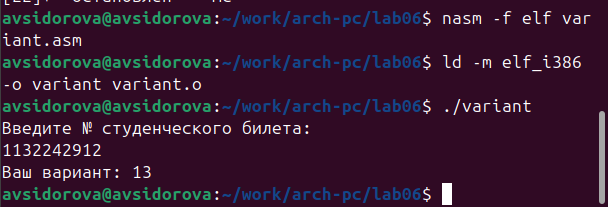


Рис. 20: Проверим результат работы программы

1. Какие строки листинга 6.4 отвечают за вывод на экран сообщения ‘Ваш вариант:’? Строки “mov eax,rem”, “call sprint”
2. Для чего используется следующие инструкции? mov ecx, x mov edx, 80 call sread

Эти инструкции вместе подготавливают необходимые параметры для вызова функции sread, которая затем считывает строку, введенную пользователем, и сохраняет ее в переменной x.

1. Для чего используется инструкция “call atoi”? call atoi используется для преобразования строки, введенной пользователем, в целое число
2. Какие строки листинга 6.4 отвечают за вычисления варианта? Строка “xor edx,edx” обнуляет регистр edx перед выполнением деления. Строка “mov ebx,20” загружает значение 20 в регистр ebx. Строка “div ebx” выполняет деление регистра eax на значение регистра ebx с сохранением частного в регистре eax и остатка в регистре edx.
3. В какой регистр записывается остаток от деления при выполнении инструкции “div ebx”? В регистр edx.
4. Для чего используется инструкция “inc edx”? Для увеличения значения в регистре edx на 1. В нашем коде она увеличивает остаток от деления на 1
5. Какие строки листинга 6.4 отвечают за вывод на экран результата вычислений? Строка “mov eax,edx” передает значение остатка от деления в регистр eax. Строка “call iprintLF” вызывает процедуру iprintLF для вывода значения на экран вместе с переводом строки.

## 2.3 Задание для самостоятельной работы

Создаем файл samostoyatelnaya.asm Открываем его и пишем программу для решения выражения f(x) = (8x+6)\*10 (рис. fig. 21 fig. 22)

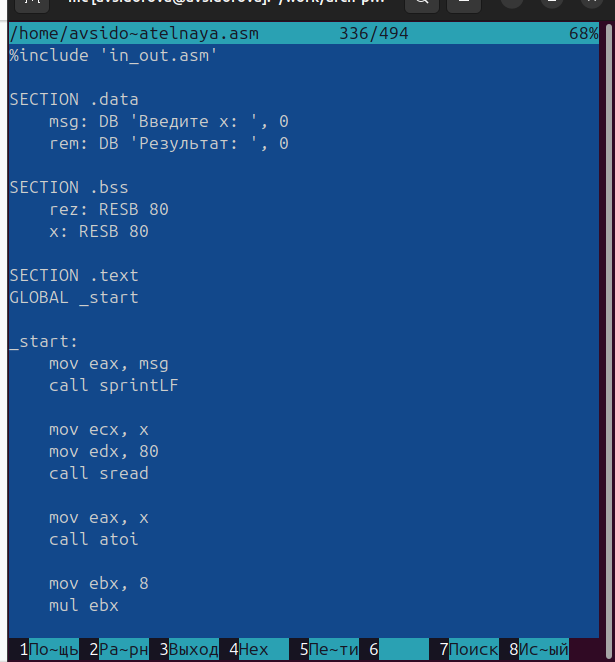


Рис. 21: Пишем программу

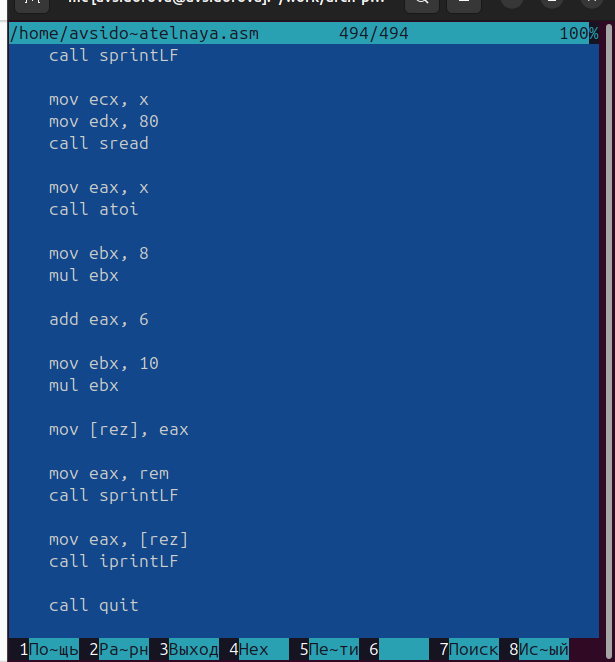


Рис. 22: Пишем программу

Компилируем программу и проверяем работу(рис. fig. 23).

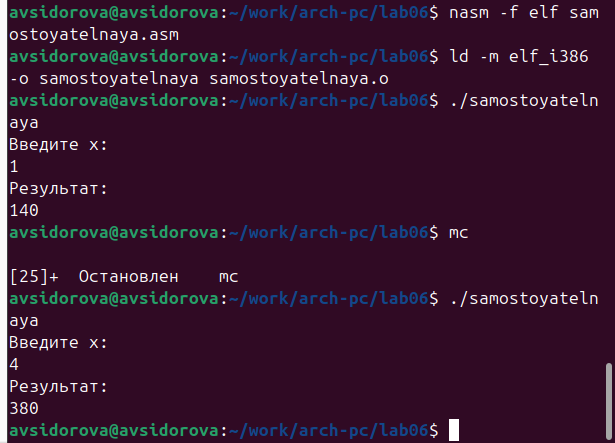


Рис. 23: Проверяем работу программы

# 3 Выводы

Освоили арифметические инструкции языка ассемблера NASM.