Отчет по лабораторной работе №7

Смирнов Дмитрий Романович

Содержание

# 1 Цель работы

Освоение арифметических инструкций языка ассемблера NASM

# 2 Выполнение лабораторной работы

Создам каталог для программам лабораторной работы № 7, перейду в него и создам файл lab7-1.asm: Введу в файл lab7-1.asm текст программы из листинга 7.1.

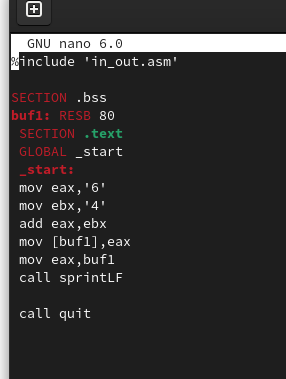


Рис. 1: Рис1

Создам исполняемый файл и запущу его.

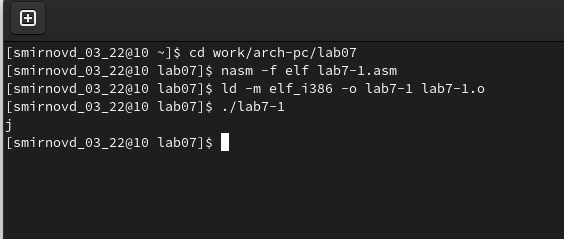


Рис. 2: Рис2

Далее изменю текст программы и вместо символов, запишу в регистры числа

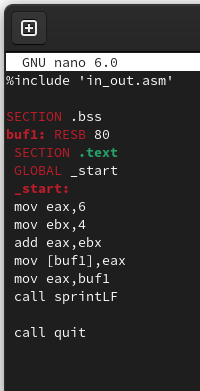


Рис. 3: Рис3

Запущу файл.

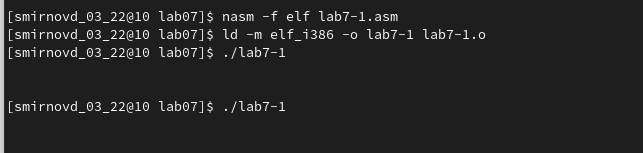


Рис. 4: Рис4

На выводится символ с кодом 10, обозначающий спец. символ LF, , который не отображается при выводе на экран. Создам файл lab7-2.asm в каталоге /work/arch-pc/lab07 и введу в него текст программы из листинга 7.2. и запущу файл.

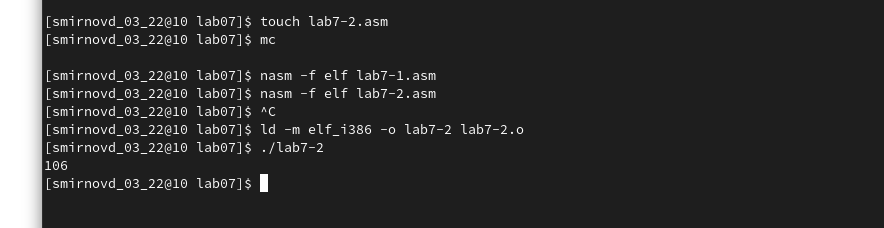


Рис. 5: Рис5

Заменю символы на числа аналогично предыдущему примеру и запущу файл.

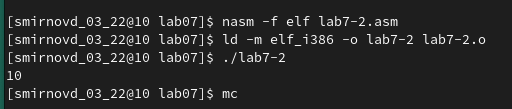


Рис. 6: Рис6

Заменю функцию iprintLF на iprint

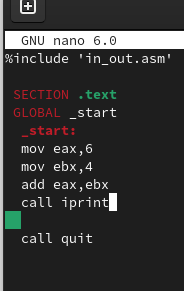


Рис. 7: Рис7

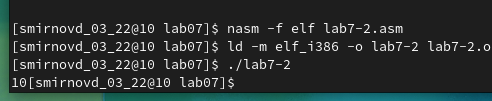


Рис. 8: Рис8

При использовании iprintLF следующий ввод пользователь переносится на новую строку в отличии от iprint Создам файл lab7-3.asm в каталоге /work/arch-pc/lab07: Запишу туда текст из листинга 7.3 и запущу файл.

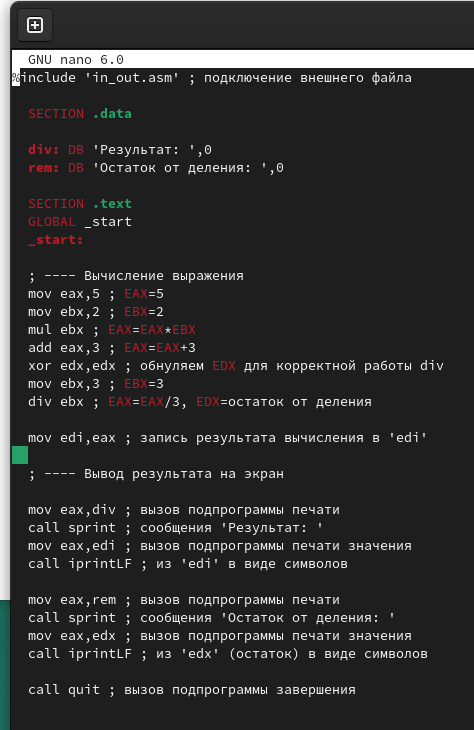


Рис. 9: Рис9

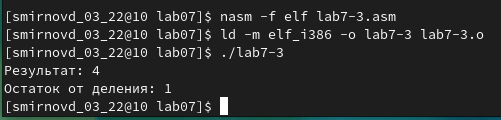


Рис. 10: Рис10

Изменю текст программы для вычисления выражения 𝑓(𝑥) = (4 ∗ 6 + 2)/5

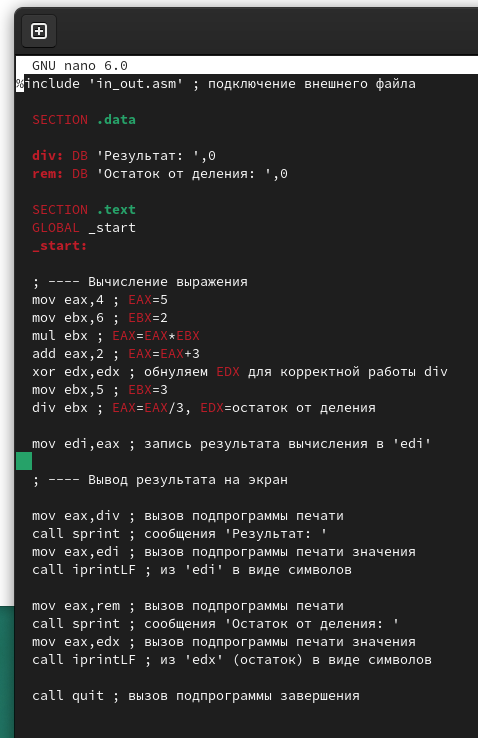


Рис. 11: Рис11

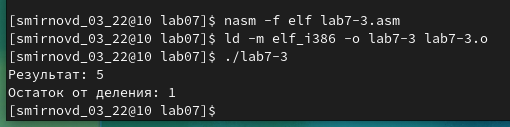


Рис. 12: Рис12

Далее запишу программу, рассчитывающую вариант в зависимости от номера студенческого

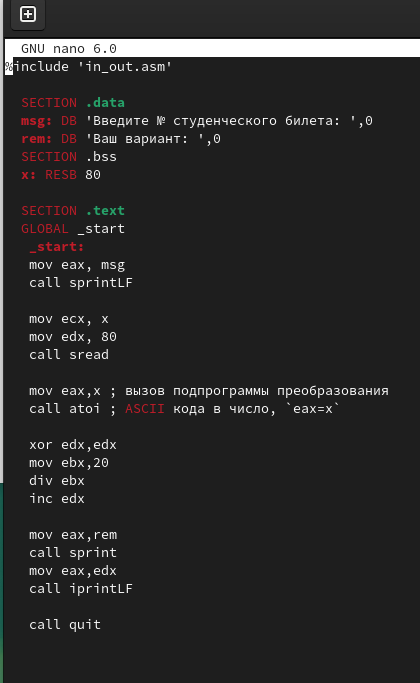


Рис. 13: Рис13

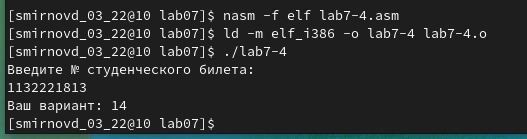


Рис. 14: Рис14

X=1

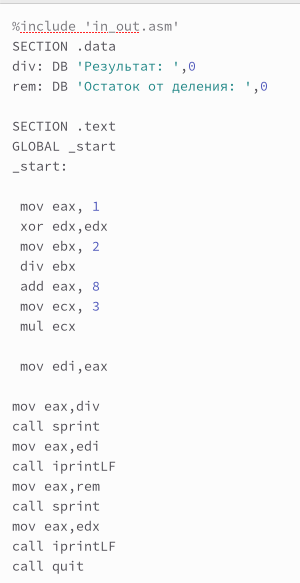


Рис. 15: Рис15

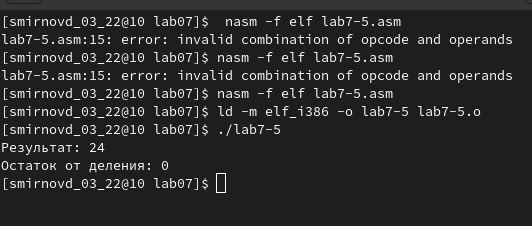


Рис. 16: Рис16

X=4

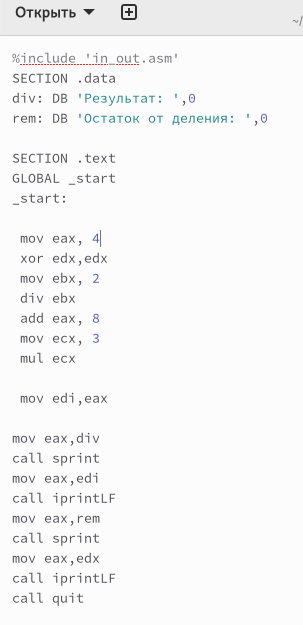


Рис. 17: Рис17

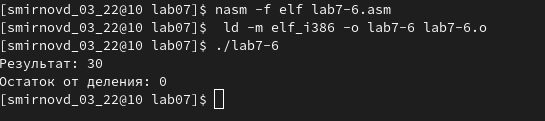


Рис. 18: Рис18

# 3 Вопросы для самопроверки

1. Какие строки листинга 7.4 отвечают за вывод на экран сообщения ‘Ваш вариант:’? mov eax,rem call sprint
2. Для чего используется следующие инструкции nasm? mov ecx, x ; mov edx, 80 ; call sread mov ecx – ecx принимает значение x (строки с клавиатуры) mov edx, 80 – edx принимает значение 80 (длина допустимого сообщения) call sread – выводит строки с клавиатуры в ecx-адрес длины edx
3. Для чего используется инструкция “call atoi”? Переводит ASCII кода в число
4. Какие строки листинга 7.4 отвечают за вычисления варианта? xor edx,edx ; mov ebx,20 ; div ebx ; inc edx
5. Для чего используется инструкция “inc edx”? Увеличивает операнда на единицу
6. Какие строки листинга 7.4 отвечают за вывод на экран результата вычислений? mov eax,rem ; call sprint ; mov eax,edx ; call iprintLF

# 4 Вывод

Я освоил арифметические инструкций языка ассемблер NASM