Отчёта по лабораторной работе №6

Арифметические операции в NASM.

Камбунду Паулине

Содержание

# 1 Цель работы

Освоить арифметических инструкций языка ассемблера NASM и написать программы для вычисления арифметических выражений с неизвестной.

# 2 Задание

Написать программы для решения выражений.

# 3 Выполнение лабораторной работы

## 3.1 Cимвольные и численные данные в NASM

Создаем каталог для программ ЛБ6, и в нем создаем файл (рис. fig. 1).

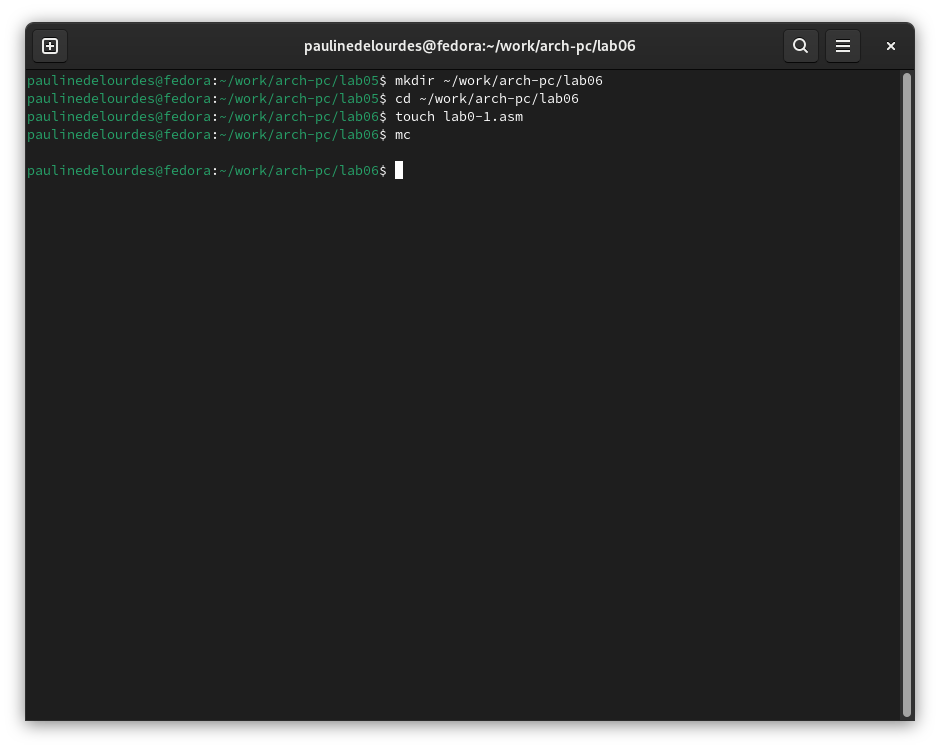


Рис. 1: Создаем каталог с помощью команды mkdir и файл с помощью команды touch

Открываем файл в Midnight Commander и заполняем его в соответствии с листингом 6.1 (рис. fig. 2).

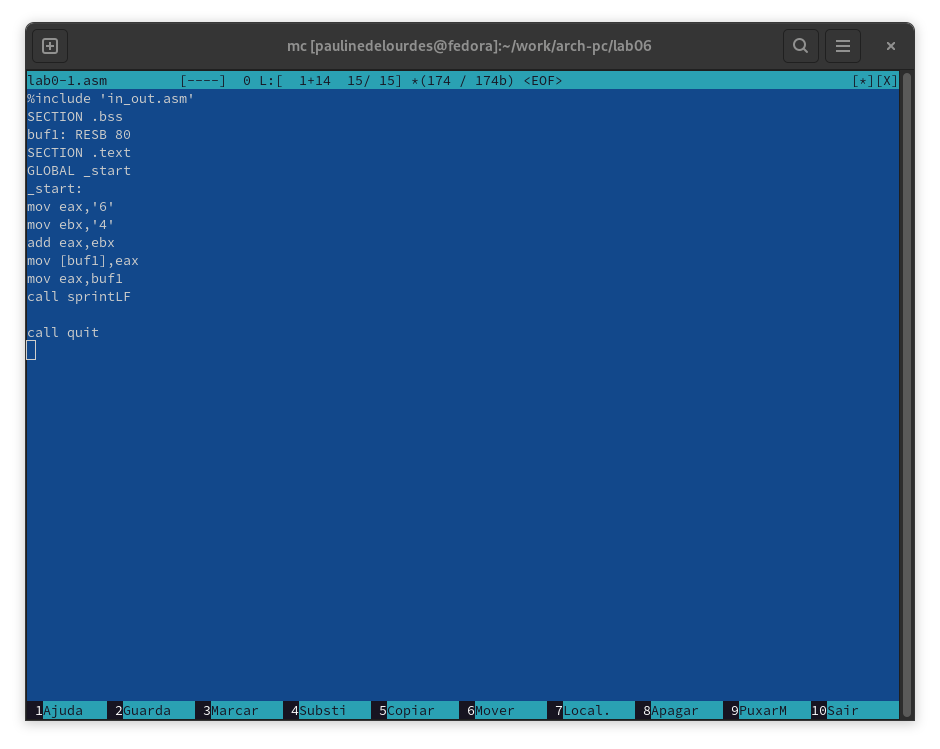


Рис. 2: Заполняем файл

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

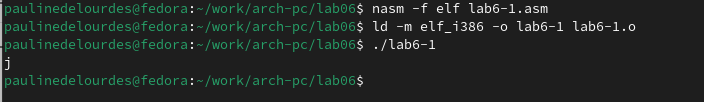


Рис. 3: Запускаем файл и смотрим на его работу

Снова открываем файл для редактирования и убиравем кавычки с числовых значений (рис. fig. 4).

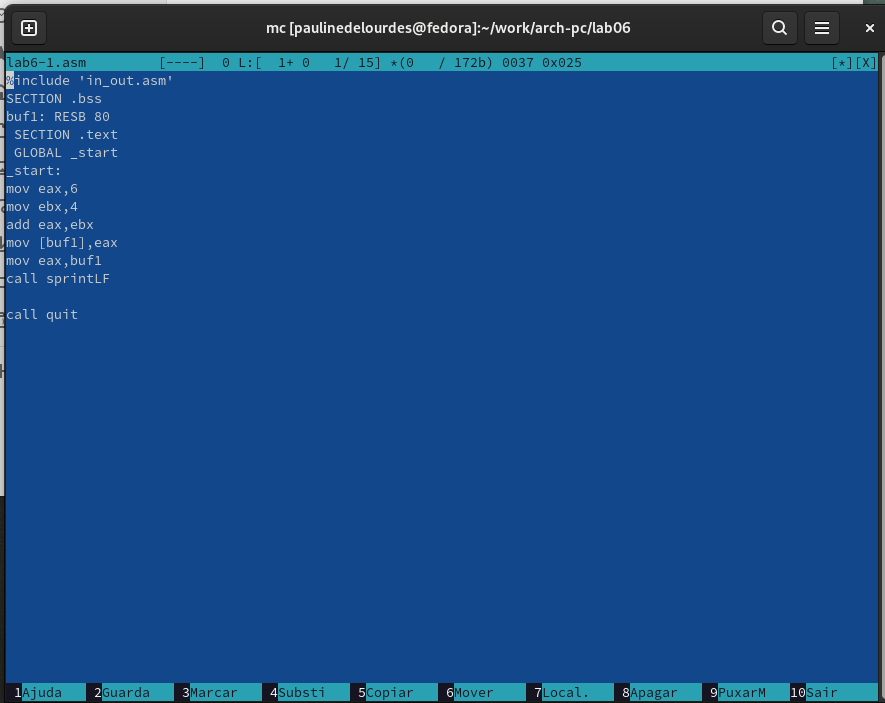


Рис. 4: Изменяем файл

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

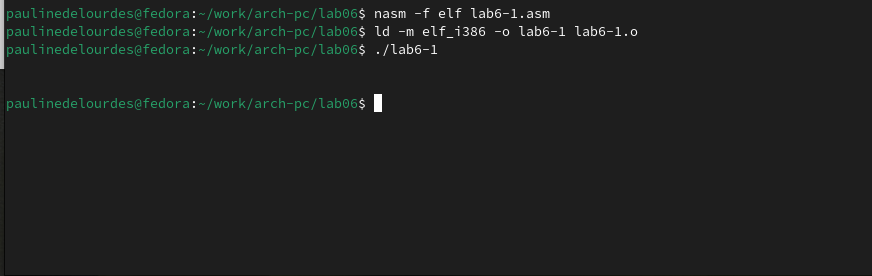


Рис. 5: Запускаем файл и смотрим на его работу

Создаем новый файл в каталоге (рис. fig. 6).

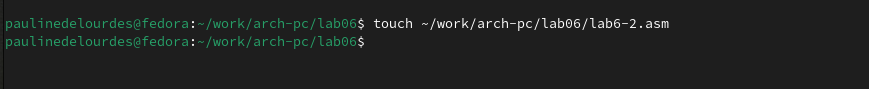


Рис. 6: Создаем файл

Заполняем файл в соответствии с листингом 6.2 (рис. fig. 7).

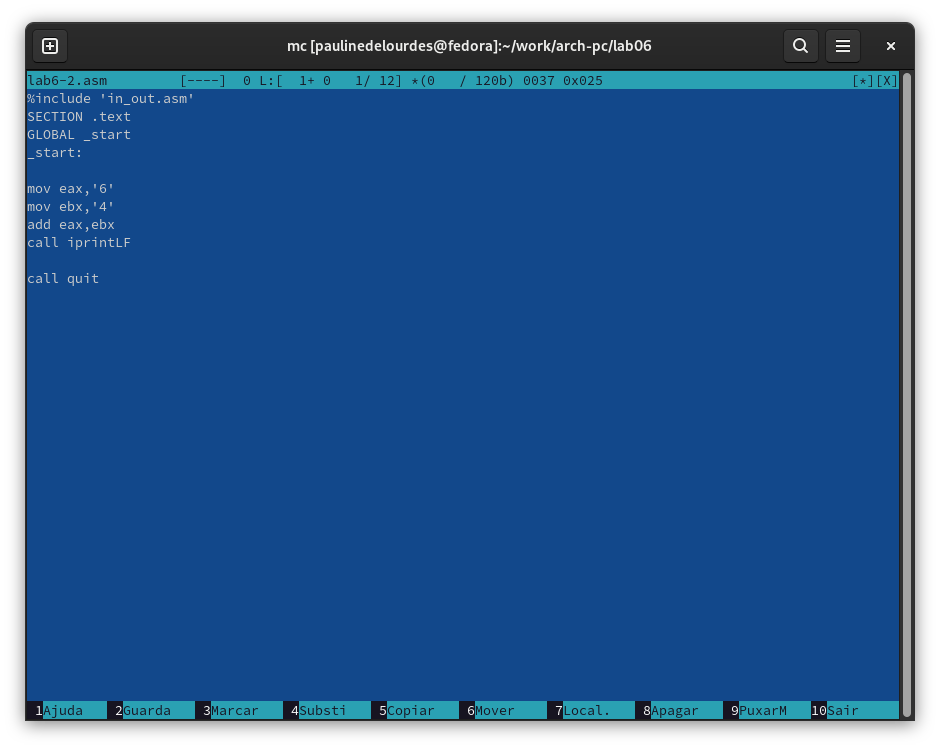


Рис. 7: Заполняем файл

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

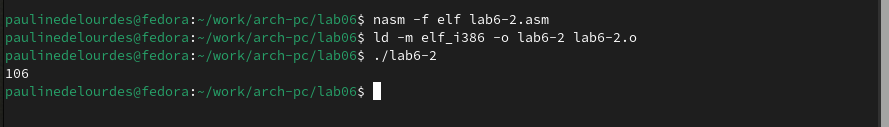


Рис. 8: Смотрим на работу программы

Снова открываем файл для редактирования и убиравем кавычки с числовых значений (рис. fig. 9).

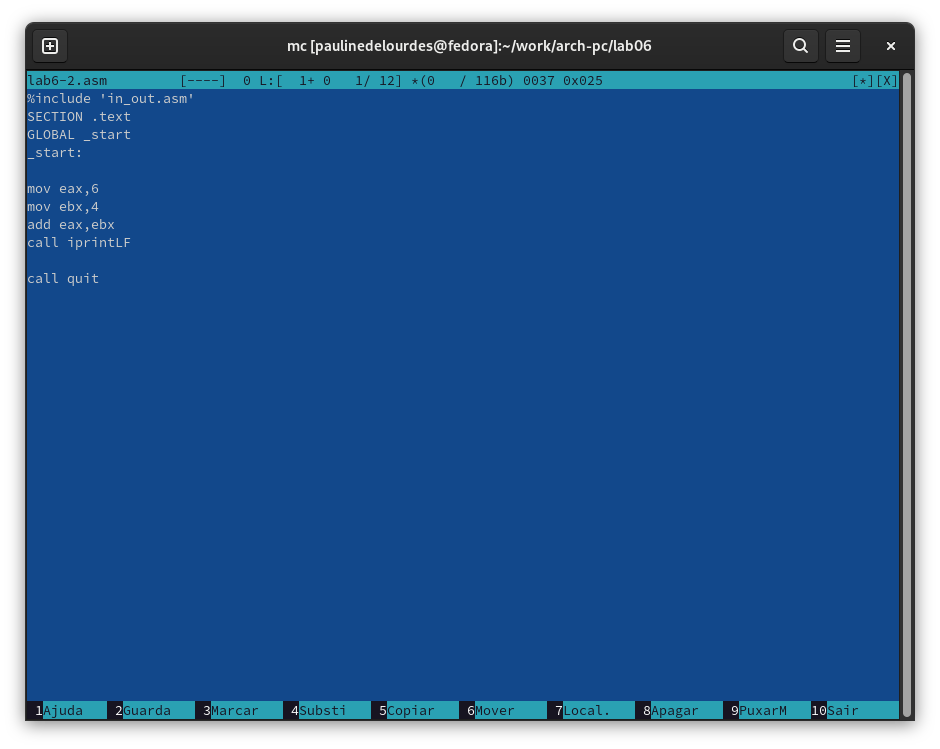


Рис. 9: Изменяем файл

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

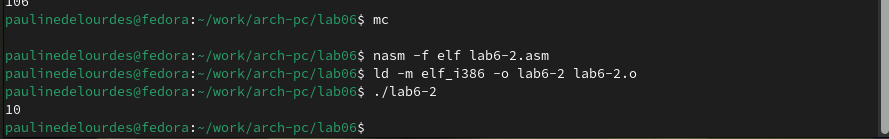


Рис. 10: Смотрим на работу программы

Снова открываем файл для редактирования и меняем iprintLF на iprint (рис. fig. 11).

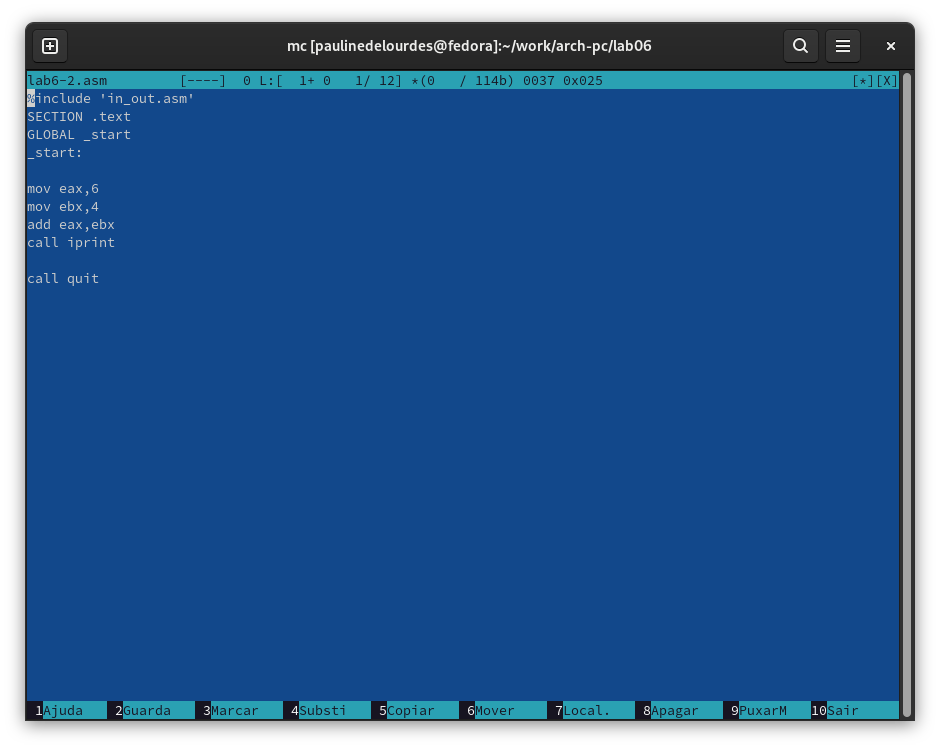


Рис. 11: Изменяем файл

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

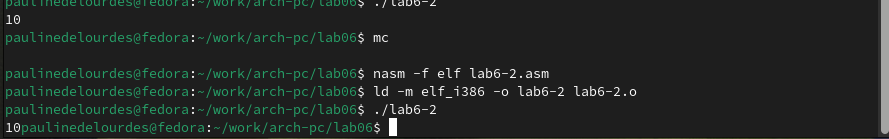


Рис. 12: Смотрим на работу программы

Вывод функций iprintLF и iprint отличаются только тем, что LF переносит на новую строку.

## 3.2 Выполнение арифметических операций в NASM

Создаем новый файл в каталоге (рис. fig. 13).

Создаем файл

Рис. 13: Создаем файл

Открываем файл и редактируем в соответствии с листингом 6.3 (рис. fig. 14).

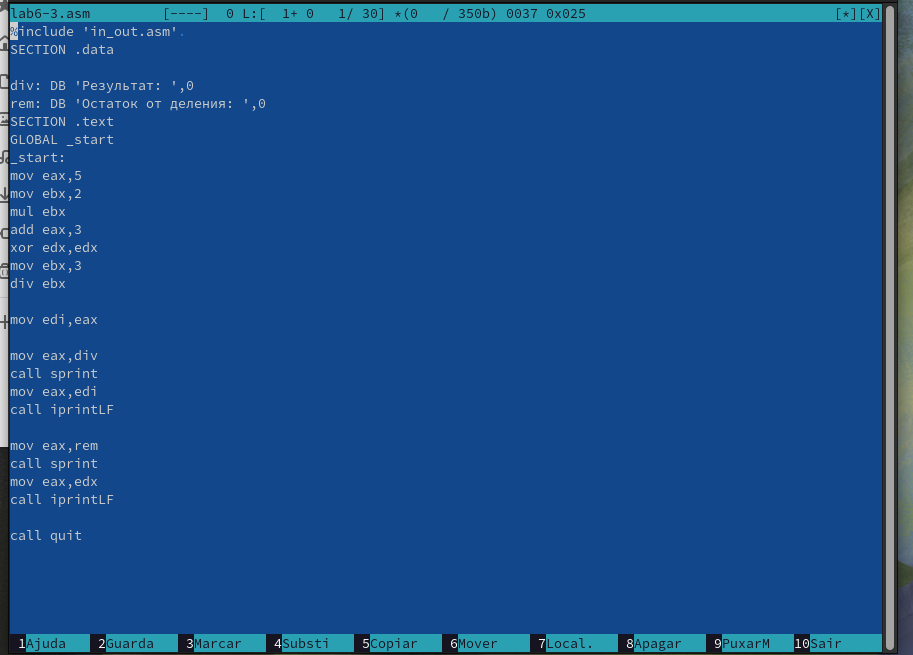


Рис. 14: Заполняем файл

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

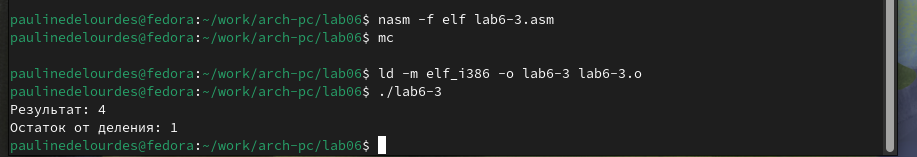


Рис. 15: Смотрим на результат работы программы

Открываем файл и редактируем его для вычисления выражения f(𝑥) = (4 ∗ 6 + 2)/5 (рис. fig. 16).

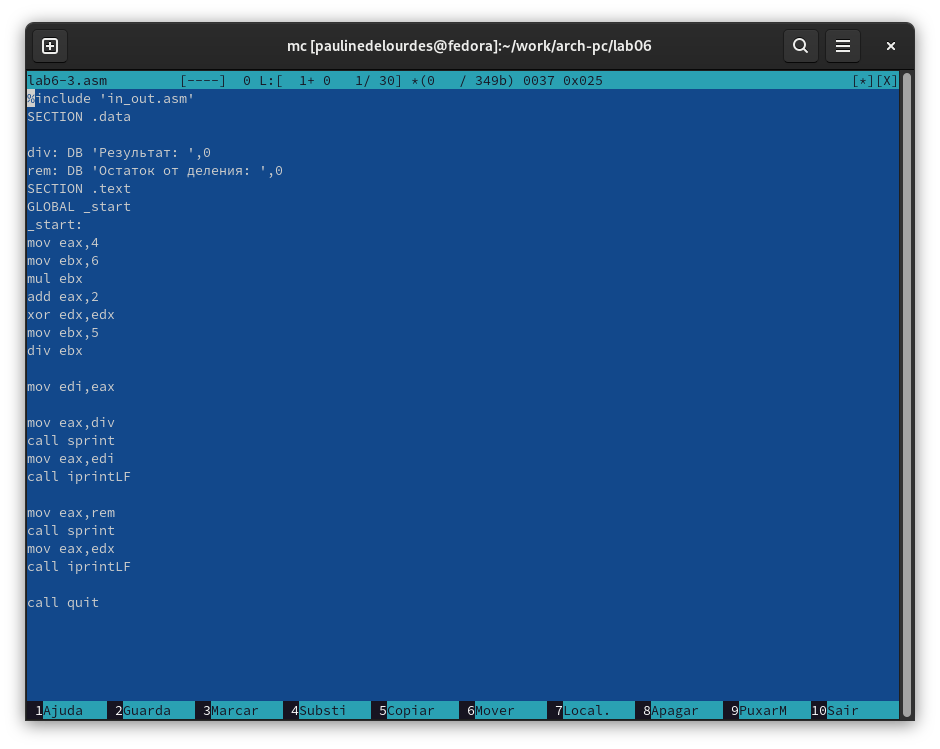


Рис. 16: Редактируем файл

Компилируем файл и запускаем программу (рис. fig. 17).

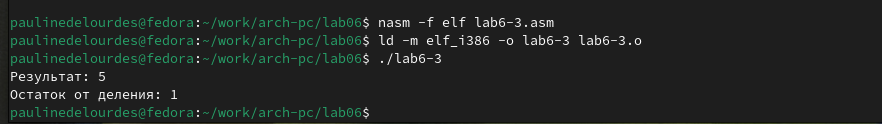


Рис. 17: Смотрим на результат работы программы

Создаем новый файл в каталоге (рис. fig. 18).

Создаем файл

Рис. 18: Создаем файл

Открываем файл и редактируем в соответствии с листингом 6.4 (рис. fig. 19).

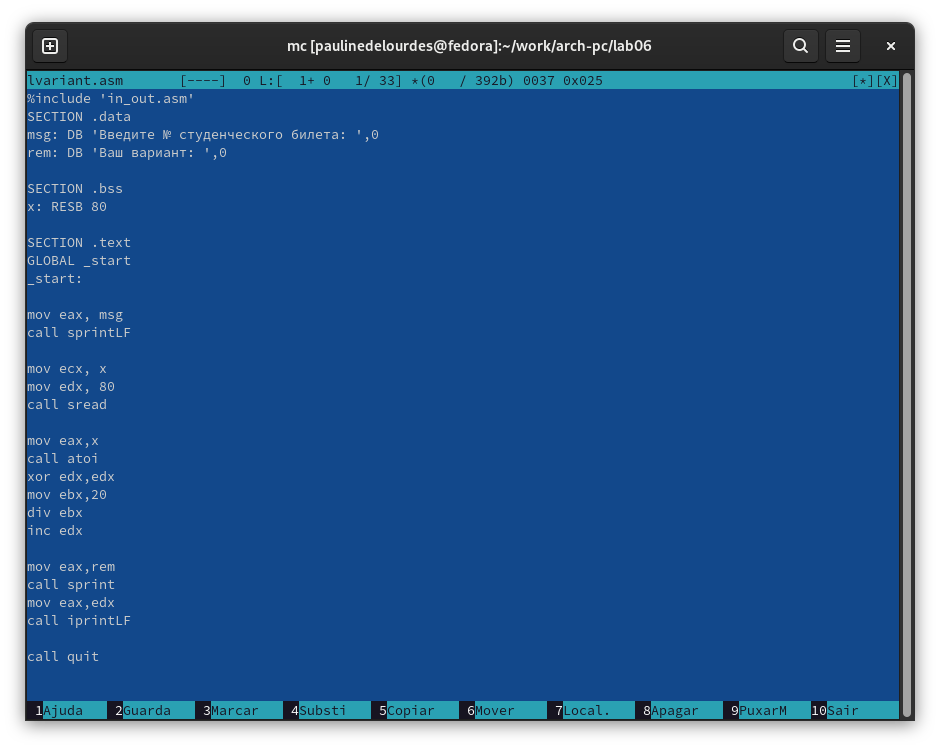


Рис. 19: Заполняем файл

Компилируем файл и запускаем его (рис. fig. 20).

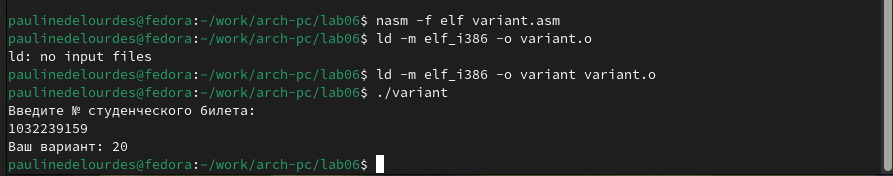


Рис. 20: Проверяемс результат работы программы

## 3.3 Ответы на вопросы по программе

1. Строка “mov eax,rem” и строка “call sprint” отвечают за вывод на экран сообщения ‘Ваш вариант:’.
2. Эти инструкции используются для чтения строки с вводом данных от пользователя. Начальный адрес строки сохраняется в регистре ecx, а количество символов в строке (максимальное количество символов, которое может быть считано) сохраняется в регистре edx. Затем вызывается процедура sread, которая выполняет чтение строки.
3. Инструкция “call atoi” используется для преобразования строки в целое число. Она принимает адрес строки в регистре eax и возвращает полученное число в регистре eax.
4. 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.
5. Остаток от деления записывается в регистр edx.
6. Инструкция “inc edx” используется для увеличения значения в регистре edx на 1. В данном случае, она увеличивает остаток от деления на 1.
7. 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.

## 3.4 Задание для самостоятельной работы

Создаем новый файл в каталоге (рис. fig. 21).

Создаем файл

Рис. 21: Создаем файл

Открываем его и заполняем, чтобы решалось выражение f(x)=𝑥^3 ⋅ 1/3 + 21(рис. fig. 22).

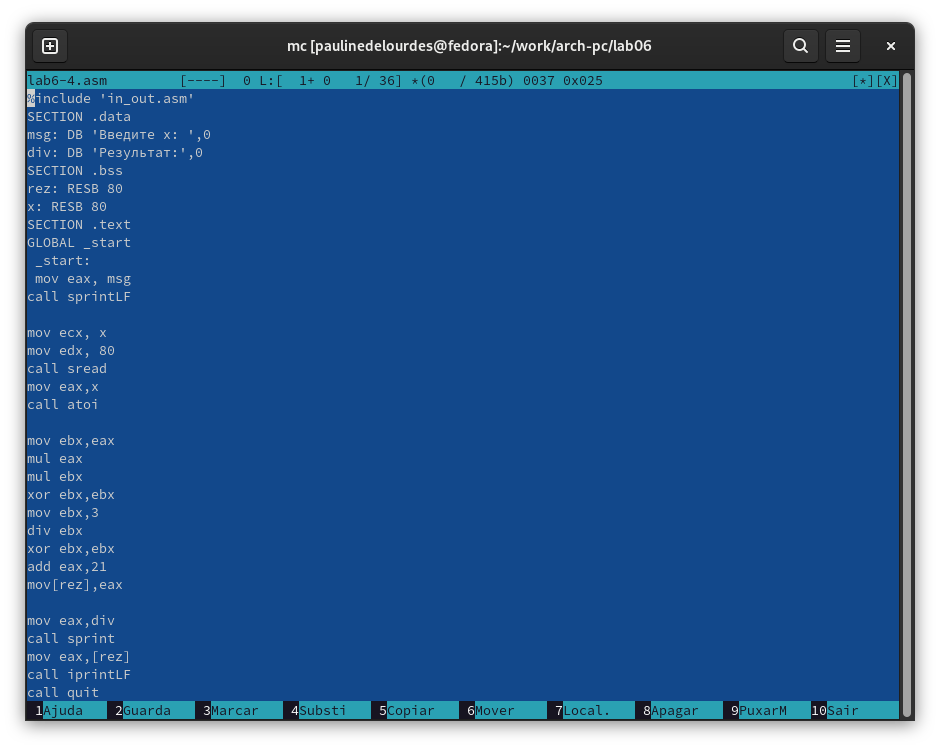


Рис. 22: Заполняем файл

Компилируем программу и проверяем для x=1 (рис. fig. 23).

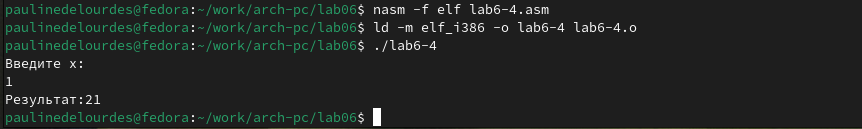


Рис. 23: Проверяем работу программы

Компилируем программу и проверяем для x=3 (рис. fig. 24).

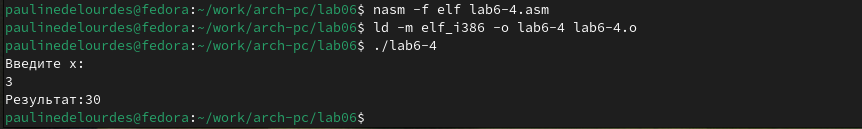


Рис. 24: Проверяем работу программы

# 4 Выводы

Мы приобрели навыки создания исполнительных файлов для решения выражений и освоили арифметические инструкции в NASM.