Отчёта по лабораторной работе №6

НПИоб-03-23

Махмудов Суннатилло Баходир угли

Содержание

# 1 Цель работы

Цель данного шаблона — максимально упростить подготовку отчётов по лабораторным работам. Модифицируя данный шаблон, студенты смогут без труда подготовить отчёт по лабораторным работам, а также познакомиться с основными возможностями разметки Markdown.

# 2 Задание

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

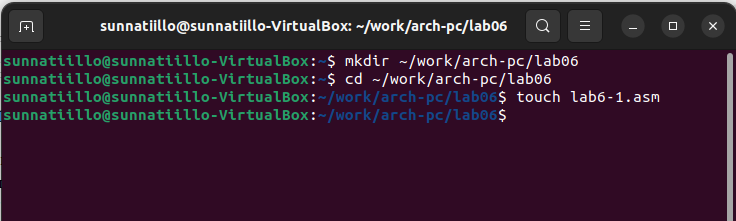
# 3 Теоретическое введение

Большинство инструкций на языке ассемблера требуют обработки операндов. Адрес опе- ранда предоставляет место, где хранятся данные, подлежащие обработке. Это могут быть данные хранящиеся в регистре или в ячейке памяти. Далее рассмотрены все существующие способы задания адреса хранения операндов – способы адресации.

# 4 Выполнение лабораторной работы

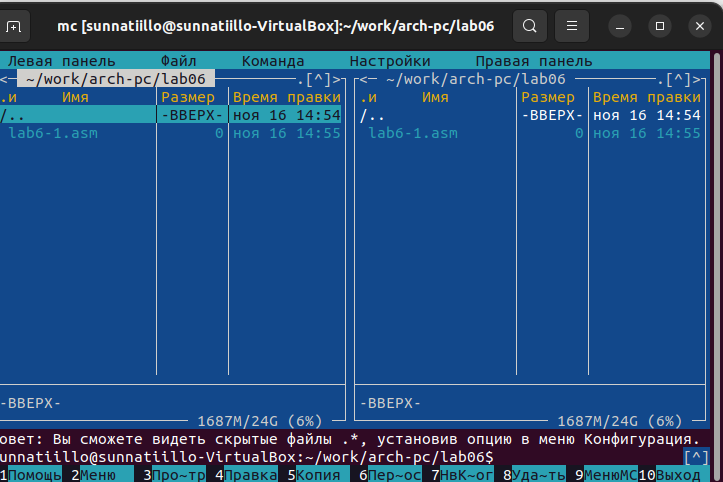
#Символьные и численные данные в NASM

Создаю каталог для программ лабораторной работы № 6, перейду в него и создаю файл lab6-1.asm:(рис. ??).



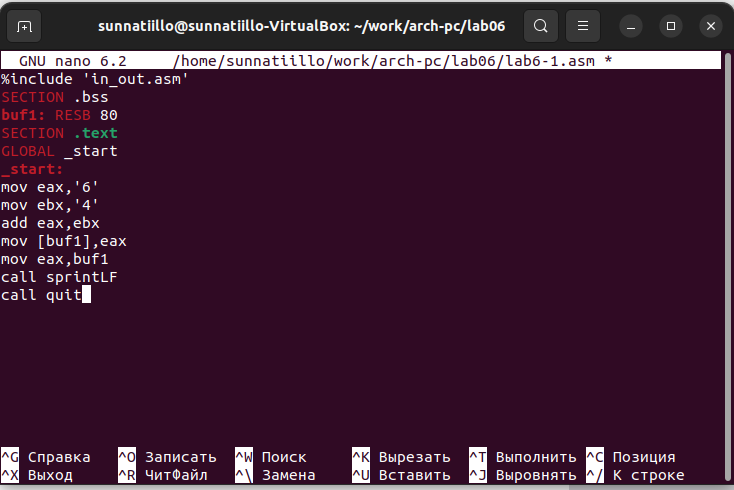
Создание деректории

Рассмотрим примеры программ вывода символьных и численных значений. Програм- мы будут выводить значения записанные в регистр eax. (рис. ??).



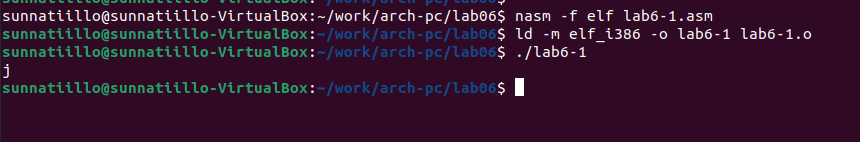
файл lab6-1.asm

Введу в файл lab6-1.asm текст программы из листинга 6.1.(рис. ??).



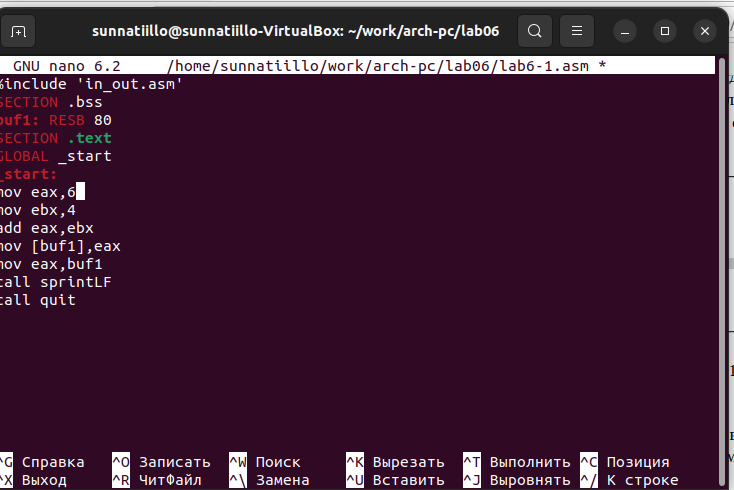
текст программы из листинга 6.1

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



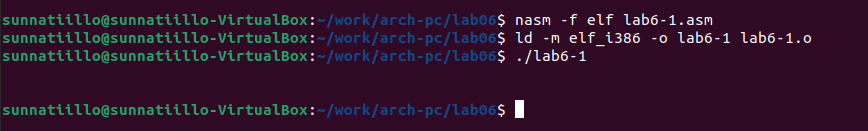
исполняемый файл

Далее изменяю текст программы и вместо символов, запишу в регистры числа. Ис- правляю текст программы (Листинг 6.1) следующим образом: заменяю строки (рис. ??).



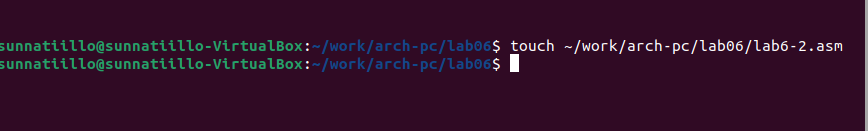
изменил текст

Создаю исполняю файл и запускаю его (рис. ??).



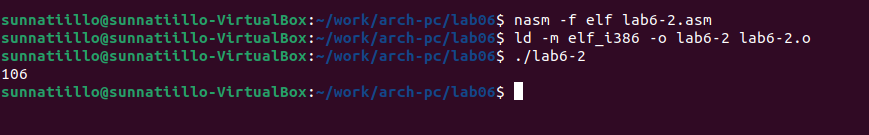
исполняемый файл

Создаю файл lab6-2.asm в каталоге ~/work/arch-pc/lab06 и ввожу в него текст про- граммы из листинга 6.2.(рис. ??).



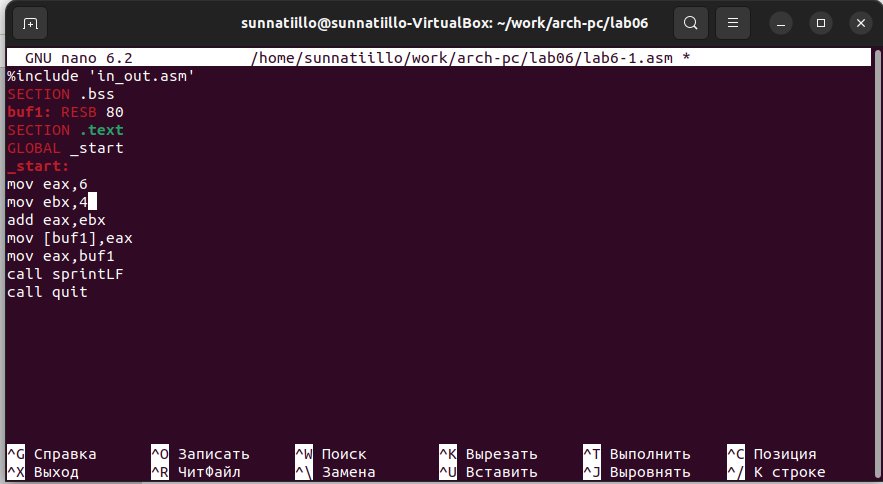
файл lab6-2.asm

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



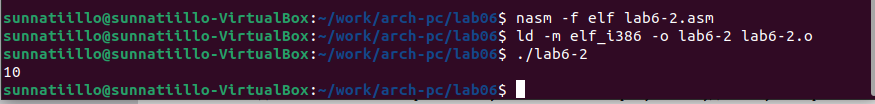
файл lab6-2.asm 106

Аналогично предыдущему примеру изменяю символы на числа. Заменяю строки (рис. ??).



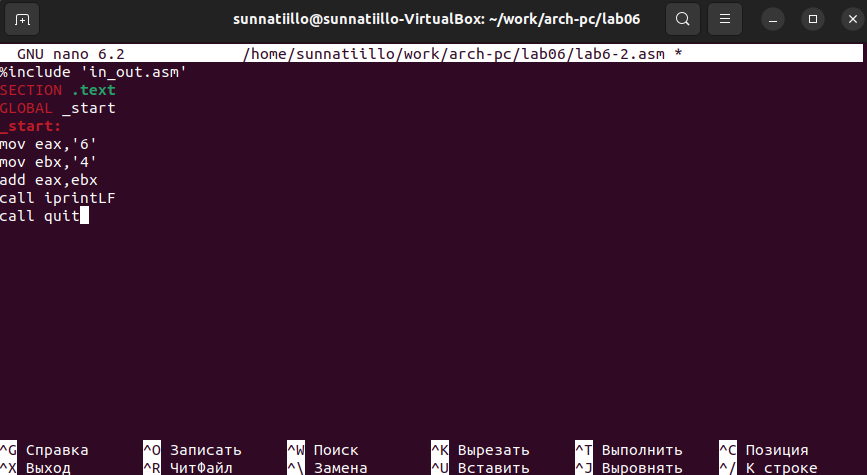
изменим символы на числа

Создаю исполняемый файл и запускаю его. Какой результат будет получен при исполне- нии программы?(рис. ??).



Создал исполняемый файл

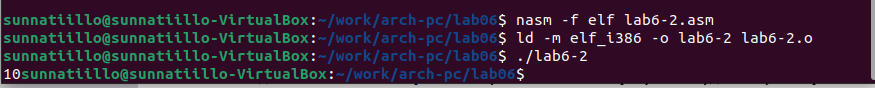
Заменяю в тексте программы функцию iprintLF на iprint (рис. ??).



Редактирование файла

Создаю и запускаю новый исполняемый файл (рис. ??).

Вывод изменился,потому что симбол переноса строки отображался когда программа исполнялась с функцией iprintLF,а iprint добавляет к выводу симбол переноса строку в отличие от iprintLF.



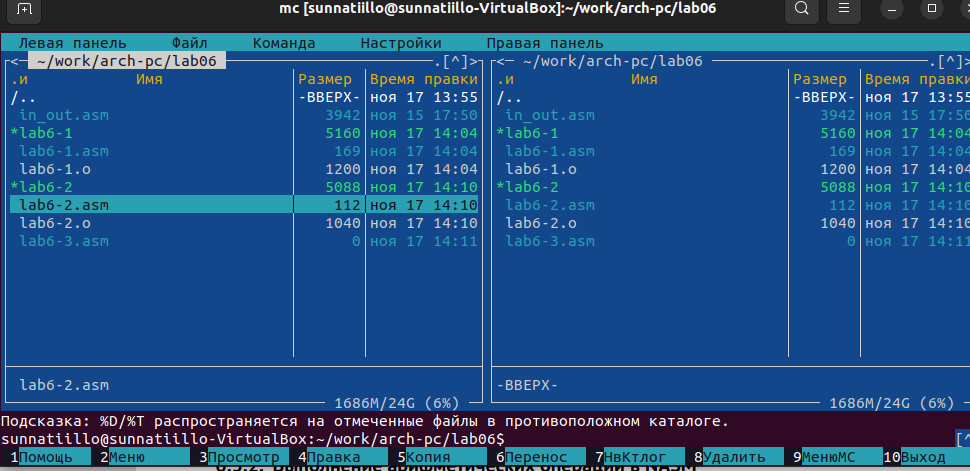
Запуск исполняемого файла

#Выполнение арифметических операций в NASM

Создаю файл lab6-3.asm в каталоге ~/work/arch-pc/lab06:(рис. ??).

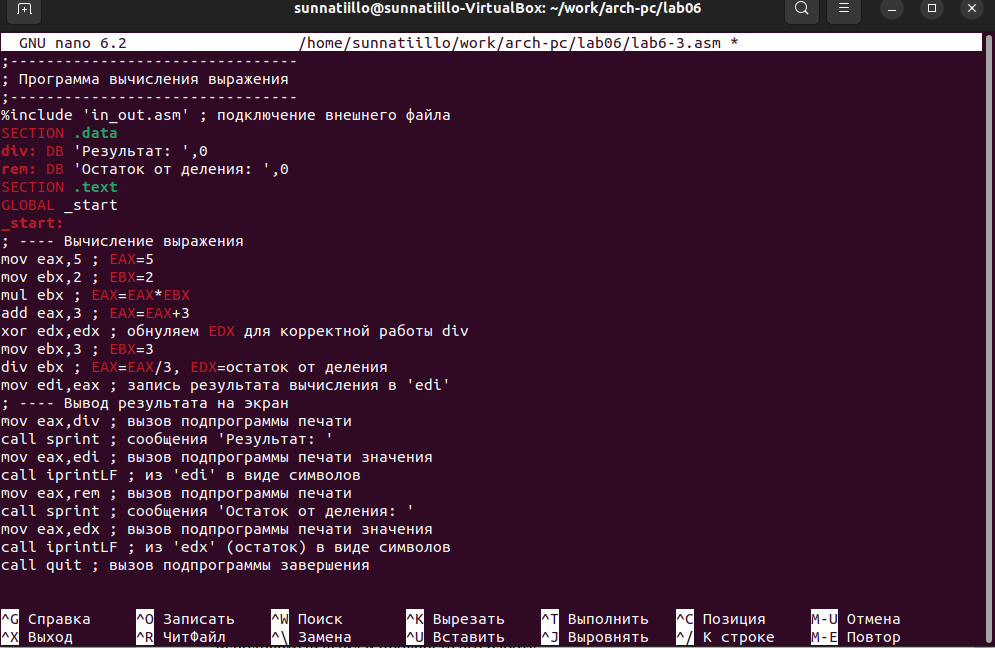
lab6-3.asm

lab6-3.asm



lab6-3.asm

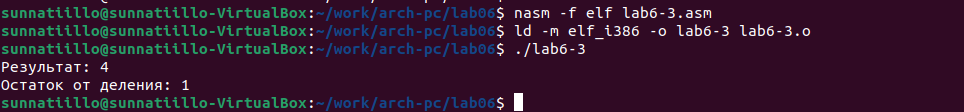
из листинга 6.3 и введите в lab6-3.asm.(рис. ??).



Я ввел текст

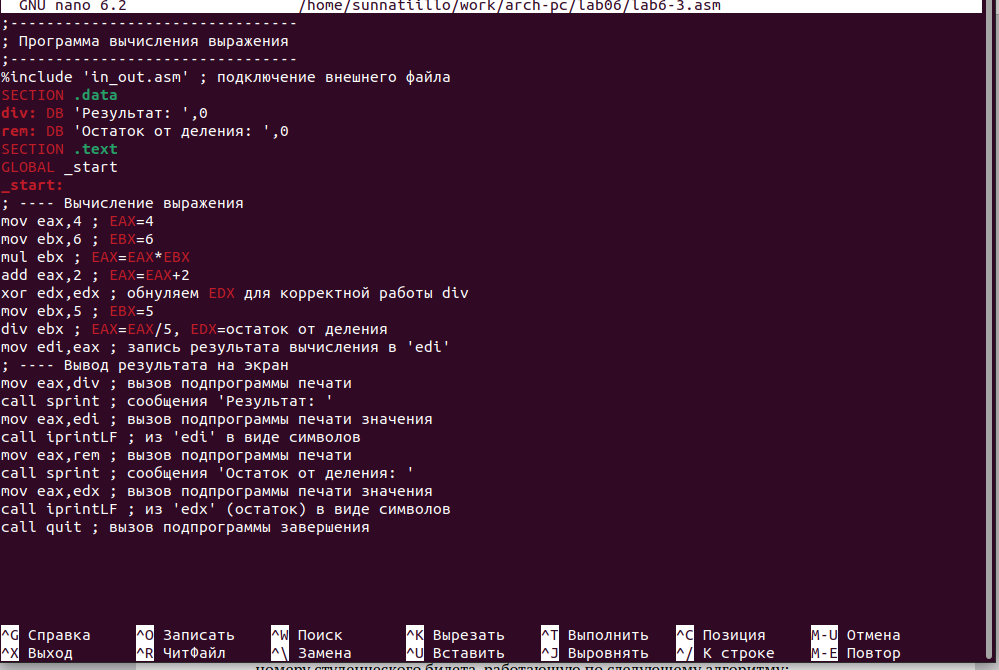
Листинг 6.3. Программа вычисления выражения

Создаю исполняемый файл и запускаю его. Результат работы программы должен быть следующим:(рис. ??).



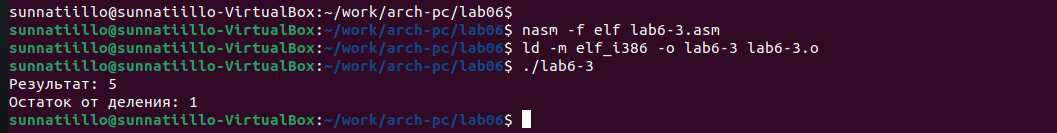
Результат работы

Изменяю текст программы для вычисления выражения 𝑓(𝑥) = (4 ∗ 6 + 2)/5. .(рис. ??)



Изменил текст

Создаю исполняемый файл и проверяю его работу (рис. ??)



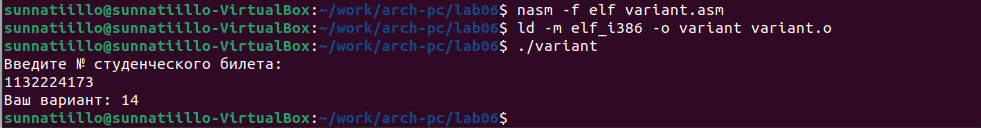
Проверял его работу

Создаю файл variant.asm в каталоге ~/work/arch-pc/lab06:(рис. ??) Листинг 6.4. Программа вычисления вычисления варианта задания по номеру студенческого билета

Проверял его работу

Проверял его работу

Листинг 6.4. Программа вычисления варианта задания по номеру студенческого билета .(рис. ??) При выполнении данной лабораторной работы я освоил арифметические инструкции языка ассемблера NASM



Проверял его работу

#Ответы на вопросы по программе

1. За вывод сообщения “Ваш вариант” отвечают строки кода:

mov eax.rem

call sprint

1. Инструкция mov еск, к используется, чтобы положить адрес вводимой стро ких в регистр есx mov edx, 80 запись в регистр ейх длины вводимой строки call sread вызов подпрограммы из внешнего файла, обеспечивающей ввод сообщения с клавиатуры 2022-252
2. call atol используется для вызова подпрограммы из внешнего файла, кото рая преобразует всії-код символа в целое число и записывает результат в регистр сах
3. За вычисления варианта отвечают строки:

ко ехех обнуление ех для корректной работы там «ок.20 сок - 20 div тах/20, dx остаток от деления ine edx edx estx 1

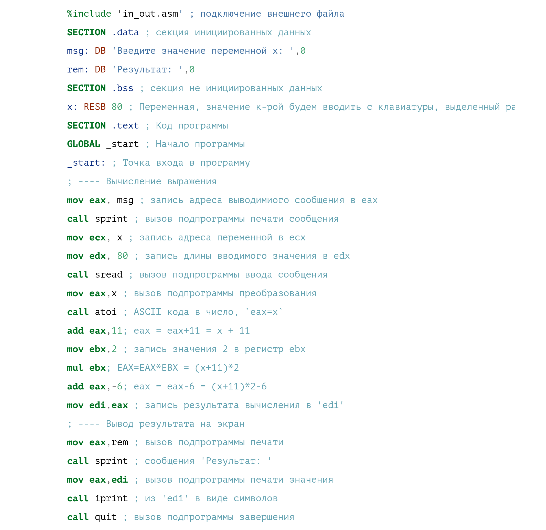
1. При выполнении инструкции div ebx остаток от деления записывается регистр ех
2. Инструкция ine edx увеличивает значение регистра eх на 1
3. За вывод на экран результатов вычислений отвечают строки:

nov eax,

eds call iprintLF

#Задание для самостоятельной работы

Написать программу вычисления выражения 𝑦 = 𝑓(𝑥). Программа должна выводить выражение для вычисления, выводить запрос на ввод значения 𝑥, вычислять задан- ное выражение в зависимости от введенного 𝑥, выводить результат вычислений. Вид функции 𝑓(𝑥) выбрать из таблицы 6.3 вариантов заданий в соответствии с номером полученным при выполнении лабораторной работы. Создайте исполняемый файл и проверьте его работу для значений 𝑥1 и 𝑥2 из 6.3.(рис. ??)



Проверял его работу

Winclude ’ as \* m’ ; подключение внешнего файла

SECTION data; секция инициированных данных

msg: DB ’Введите значение переменной х:,0

rem: DB ‘Результат:’,0

SECTION bss; секция не инициированных данных

х: RЕЅB 80 ; Переменная, значение к-рой будем вводить с клавиатуры, выделенный ре

SECTION text; Код программы

GLOBAL \_start; Начало программы

\_start: Точка входа в программу

Вычисление выражения

mov eax, ns; запись адреса выводимного сообщения в сах

call sprint; вызов подпрограммы печати сообщения

mov есх, х; запись адреса переменной в есх mov edx, 80; запись длины вводимого значения в

call sread; вызов подпрограммы ввода сообщения

mov eax,x ; вызов подпрограммы преобразования

call atoi; ASCII кода в число, сахих

add eax,11; eax = eax + 11 = x + 11

mov ebx,2; запись значения 2 в регистр сх

mul ebx; EAX=EAX^ \* EBX = (x + 11) + 2

add eax,-6;

eax = eax - 6 = (x + 11) \* 2 \* 6

mov edi, eax; запись результата вычисления в ‘edi’

Вывод результата на экран

mov eax,rem, вызов подпрограммы печати

call sprint: сообщения Результат:

mov eax, edi ; вызов подпрограммы печати значения call iprint; из ’edi в виде символов

call quit, вызов подпрограммы завершения

# 5 Выводы

При выполнении данной лабораторной работы я освоил арифметические инструкции языка ассемблера NASM