Отчет по лабораторной работе №10

Дисциплина: Архитектура компьютера

Зайцева Ульяна Владимировна

Содержание

# 1 Цель работы

Приобретение навыков написания программ для работы с файлами

# 2 Задание

1. Написание программ для работы с файлами.
2. Задание для самостоятельной работы.

# 3 Теоретическое введение

ОС GNU/Linux является многопользовательской операционной системой. И для обеспечения защиты данных одного пользователя от действий других пользователей существуют специальные механизмы разграничения доступа к файлам. Кроме ограничения доступа, данный механизм позволяет разрешить другим пользователям доступ данным для совместной работы. Права доступа определяют набор действий (чтение, запись, выполнение), разрешённых для выполнения пользователям системы над файлами. Для каждого файла пользователь может входить в одну из трех групп: владелец, член группы владельца, все остальные. Для каждой из этих групп может быть установлен свой набор прав доступа. Владельцем файла является его создатель.

Набор прав доступа задается тройками битов и состоит из прав на чтение, запись и исполнение файла. В символьном представлении он имеет вид строк rwx, где вместо любого символа может стоять дефис. Всего возможно 8 комбинаций, приведенных в таблице 10.1. Буква означает наличие права (установлен в единицу второй бит триады r — чтение, первый бит w — запись, нулевой бит х — исполнение), а дефис означает отсутствие права (нулевое значение соответствующего бита). Также права доступа могут быть представлены как восьмеричное число. Так, права доступа rw- (чтение и запись, без исполнения) понимаются как три двоичные цифры 110 или как восьмеричная цифра 6.

Свойства (атрибуты) файлов и каталогов можно вывести на терминал с помощью команды ls с ключом -l.

Обработка файлов в операционной системе Linux осуществляется за счет использования определенных системных вызовов. Для корректной работы и доступа к файлу при его открытии или создании, файлу присваивается уникальный номер (16-битное целое число) – дескриптор файла

Для записи в файл служит системный вызов sys\_write, который использует следующие аргументы: количество байтов для записи в регистре EDX, строку содержимого для записи ECX, файловый дескриптор в EBX и номер системного вызова sys\_write (4) в EAX

Для чтения данных из файла служит системный вызов sys\_read, который использует следующие аргументы: количество байтов для чтения в регистре EDX, адрес в памяти для записи прочитанных данных в ECX, файловый дескриптор в EBX и номер системного вызова sys\_read (3) в EAX.

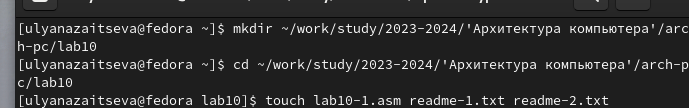
Для изменения содержимого файла служит системный вызов sys\_lseek, который исполь- зует следующие аргументы: исходная позиция для смещения EDX, значение смещения в байтах в ECX, файловый дескриптор в EBX и номер системного вызова sys\_lseek (19) в EAX.

Удаление файла осуществляется системным вызовом sys\_unlink, который использует один аргумент – имя файла в регистре EBX.

# 4 Выполнение лабораторной работы

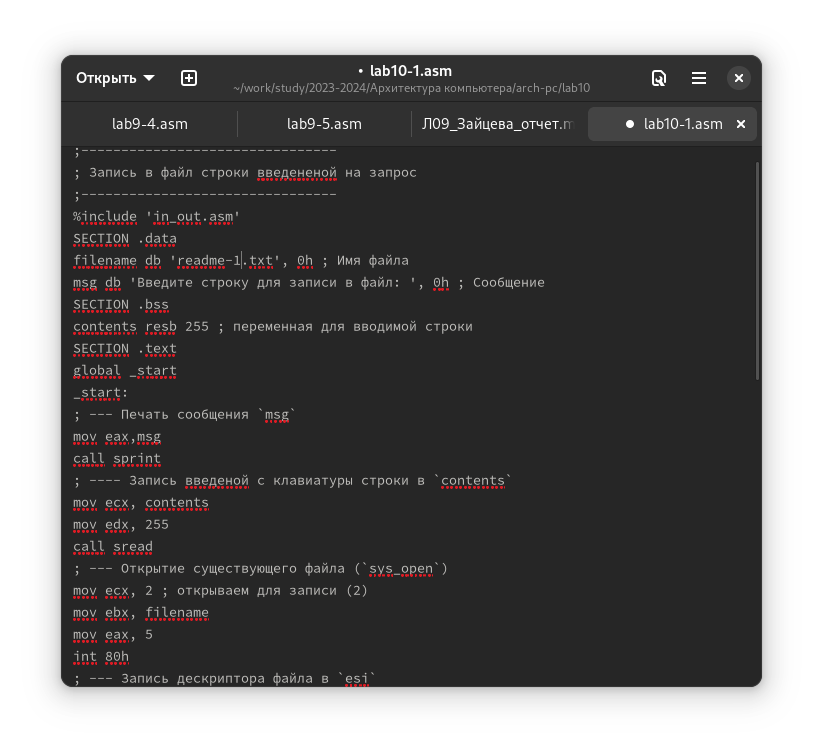
1. Написание программ для работы с файлами.

Создаю каталог для программ лабораторной работы №10, перехожу в него и создаю файлы lab10-1.asm, readme-1.txt и readme-2.txt.(рис. ??)



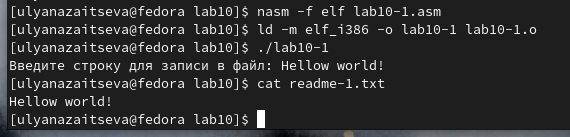
Новый каталог

Ввожу в файл lab10-1.asm текст программы из листинга 10.1(рис. ??)



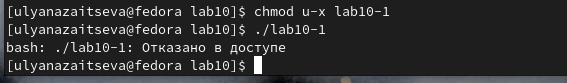
Листинг 10.1

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



Проверка работы

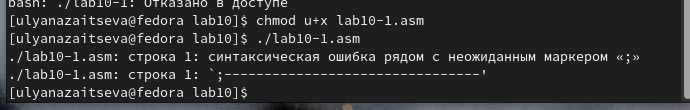
С помощью команды chmod изменяю права доступа к файлу lab10-1.asm с исходным текстом программы, добавив права на исполнение.(рис. ??)



Попытка выполнить

Файл не выполняется, потому что в команде я указала себя владельцем - ‘u’, ‘-’ - отменить набор прав, ‘х’ - право на исполнение.

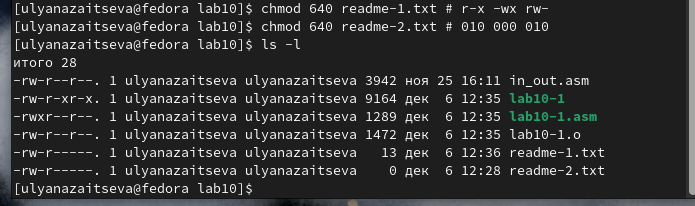
С помощью команды chmod u+х меняю права доступа к файлу lab10-1.asm с исходным текстом программы, добавив права на исполнение, и пробую выполнить его.(рис. ??)



Изменяю права доступа

Файл начинает исполнение, но не исполняется, потому что не имеет в себе команд для терминала.

Мой вариант - 17. Предоставляю права доступа к файлу readme-1.txt представленные в символьном виде, а для файла readme-2.txt – в двочном виде в соответсвии с таблицей 10.4.(рис. ??)

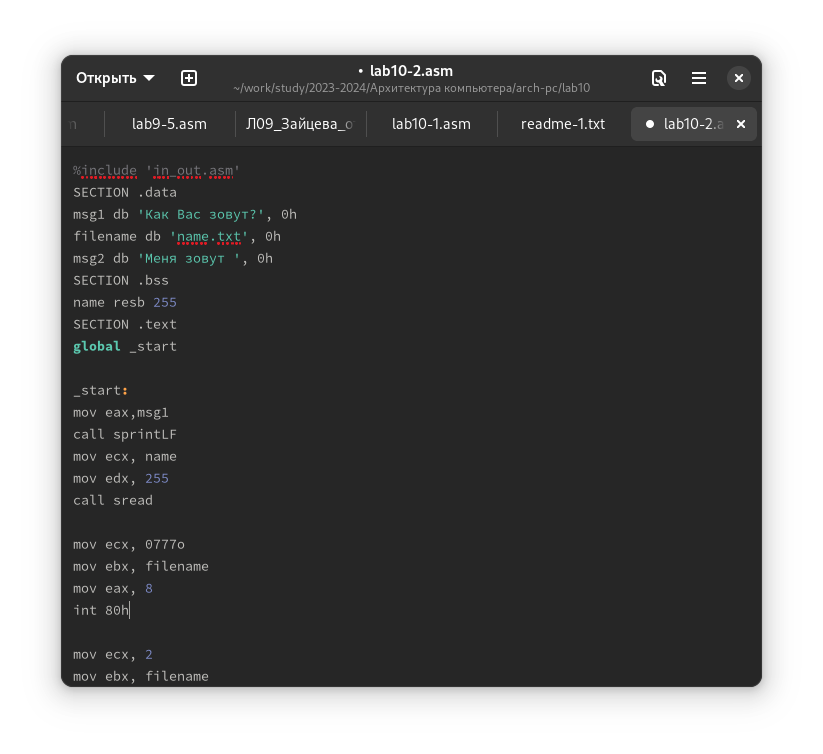


Предоставляю права доступа

1. Задание для самостоятельной работы.

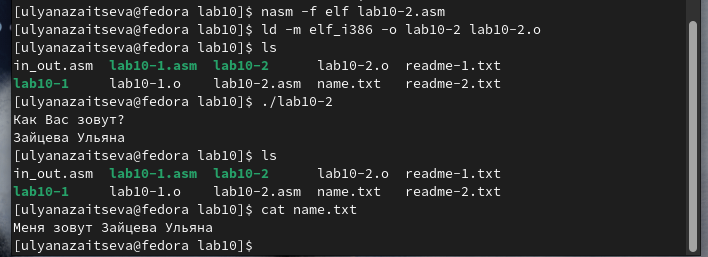
Создаю файл lab10-2.asm

Пишу программу работающую по следующему алгоритму: • Вывод приглашения “Как Вас зовут?” • ввести с клавиатуры свои фамилию и имя • создать файл с именем name.txt • записать в файл сообщение “Меня зовут” • дописать в файл строку введенную с клавиатуры • закрыть файл(рис. ??)



Написание программы для выполнения алгоритма

Создаю исполняемый файл и проверяю его работу. Проверяю наличие файла и его содержимое с помощью команд ls и cat(рис. ??)



Проверка работы программы

Код:

%include ‘in\_out.asm’ SECTION .data msg1 db ‘Как Вас зовут?’, 0h filename db ‘name.txt’, 0h msg2 db ‘Меня зовут’, 0h SECTION .bss name resb 255 SECTION .text global \_start

\_start: mov eax,msg1 call sprintLF mov ecx, name mov edx, 255 call sread

mov ecx, 0777o mov ebx, filename mov eax, 8 int 80h

mov ecx, 2 mov ebx, filename mov eax, 5 int 80h

mov esi, eax

mov eax, msg2 call slen

mov edx, eax mov ecx, msg2 mov ebx, esi mov eax, 4 int 80h

mov eax, name call slen

mov edx, eax mov ecx, name mov ebx, esi mov eax, 4 int 80h

mov ebx, esi mov eax, 6 int 80h call quit

# 5 Выводы

В ходе выполнения этой лабораторной работы я приобрела навыки написания программ для работы с файлами