

Лабораторная работа №11. Работа с файлами средствами Nasm

Дисциплина: Архитектура ЭВМ

Осокин Георгий Иванович НММбд-02-22

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Запретим исполнение файлу ./lab11-1	8
2.2	Разрешим исполнение файлу ./lab11-1.asm	8
2.3	Изменение доступа к readme.txt	9
3	Задание для самостоятельной работы	10
4	Выводы	14

Список иллюстраций

2.1	Создание каталога и файлов	6
2.2	Содержимое readme.txt после исполнения lab11-1	8
2.3	Изменение разрешений файла lab11-1	8
2.4	Попытка исполнить файл ./lab11-1	9
2.5	Изменение разрешения файла	9
3.1	Вывод содержимого файла name.txt	13

Список таблиц

1 Цель работы

Приобретение навыков написания программ для работы с файлами.

2 Выполнение лабораторной работы

Создадим каталог с файлами lab11.asm и readme.md

```
* Code cd work
→ work cd ~/Code/work/arch-pc
→ arch-pc mkdir lab11
→ arch-pc touch lab11/{lab11.asm,readme.txt}
→ arch-pc ls lab11
lab11.asm  readme.txt
→ arch-pc |
```

Рис. 2.1: Создание каталога и файлов

Введем исходный текст из листинга 11.1

```
;-----
; Запись в файл строки введенной на запрос
;-----

%include 'in_out.asm'

SECTION .data
filename db 'readme.txt', 0h ; Имя файла
msg db 'Введите строку для записи в файл: ', 0h ; Сообщение

SECTION .bss
contents resb 255 ; переменная для вводимой строки

SECTION .text
global _start
_start:
```

```

; --- Печать сообщения `msg`
    mov eax,msg
    call sprint

; ---- Запись введенной с клавиатуры строки в `contents`
    mov ecx, contents
    mov edx, 255
    call sread

; --- Открытие существующего файла (`sys_open`)
    mov ecx, 2 ; открываем для записи (2)
    mov ebx, filename
    mov eax, 5
    int 80h

; --- Запись дескриптора файла в `esi`

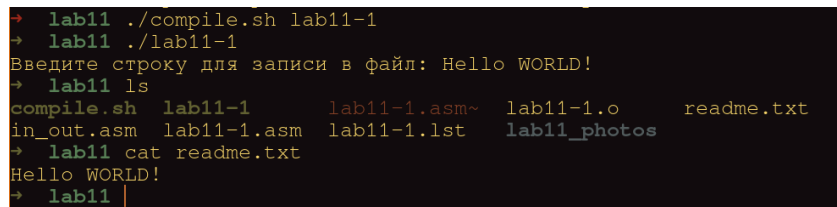
    mov esi, eax ; --- Расчет длины введенной строки
    mov eax, contents ; в `eax` запишется количество
    call slen ; введенных байтов
; --- Записываем в файл `contents` (`sys_write`)
    mov edx, eax
    mov ecx, contents
    mov ebx, esi
    mov eax, 4
    int 80h

; --- Закрываем файл (`sys_close`)

```

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

Создадим исполняемый файл и запустим его. Выведем содержимое `readme.txt` на экран.

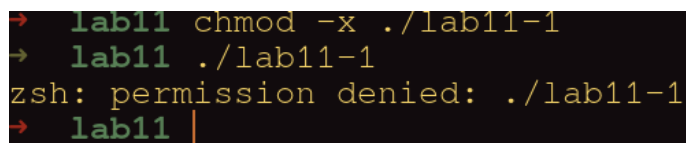


```
→ lab11 ./compile.sh lab11-1  
→ lab11 ./lab11-1  
Введите строку для записи в файл: Hello WORLD!  
→ lab11 ls  
compile.sh lab11-1 lab11-1.asm~ lab11-1.o readme.txt  
in_out.asm lab11-1.asm lab11-1.lst lab11_photos  
→ lab11 cat readme.txt  
Hello WORLD!  
→ lab11 |
```

Рис. 2.2: Содержимое `readme.txt` после исполнения `lab11-1`

2.1 Запретим исполнение файлу `./lab11-1`

Сделаем это с помощью команды `chmod -x ./lab11-1`



```
→ lab11 chmod -x ./lab11-1  
→ lab11 ./lab11-1  
zsh: permission denied: ./lab11-1  
→ lab11 |
```

Рис. 2.3: Изменение разрешений файла `lab11-1`

Как видим, при попытке исполнить файл, на экран выводится “permission denied”

2.2 Разрешим исполнение файлу `./lab11-1.asm`

Сделаем это командой `chmod +x ./lab11-1.asm`

Попробуем исполнить этот файл.


```

→ lab11 chmod +x ./lab11-1.asm
→ lab11 ./lab11-1.asm
./lab11-1.asm: line 1: syntax error near unexpected token `;'
./lab11-1.asm: line 1: `;-----'
→ lab11 |

```

Рис. 2.4: Попытка исполнить файл ./lab11-1

Программная оболочка “bash” пытается его исполнить, но воспринимает точку с запятой как команду и не понимает ее.

2.3 Изменение доступа к readme.txt

Так как наш вариант - 18, назначим файлу readme.txt права

`-wx -r-x -wx = 353`

В восьмеричной системе, это выглядит как 353

```

→ lab11 chmod 353 readme.txt
→ lab11 ls -l readme.txt
--wxr-x-wx 1 horhik horhik 13 Dec 20 19:44 readme.txt
→ lab11 |

```

Рис. 2.5: Изменение разрешения файла

Проверим командной `ls -l`

3 Задание для самостоятельной работы

Напишем программу, которая будет принимать наше имя из командной строки и записывать результат в файл `name.txt`

```
%include 'in_out.asm'
```

```
SECTION .data
```

```
filename db 'name.txt', 0h
```

```
msg db 'Как Вас зовут? ', 0h
```

```
initial_text db 'Меня зовут: ', 0h
```

```
SECTION .bss
```

```
;;; Note that content follows by `initial_text` that allows us not to concatenat
```

```
content resb 512
```

```
SECTION .text
```

```
global _start
```

```
_start:
```

```
;;; PRINT msg
```

```
    mov eax, msg
```

```
    call sprint
```

```
;;; TAKING INPUT
```

```
    mov ecx, content
```

```
    mov edx, 512
```

```

    call sread

;;; CREATING FILE

    mov ebx, filename
    mov ecx, 0777o      ; read and write for all, rwx for owner
    mov eax, 8
    int 80h

;;; OPENING FILE

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

;;; WRITING INTO FILE

    mov esi, eax

;; Counting len of initial_len
    mov eax, initial_text    ; Len of initial_text to eax
    call slen
    mov edx, eax           ; edx = len(initial_text)
;;; Counting len of content
    mov eax, content        ; Len of content to eax
    call slen
;;; Summing lens
    add edx, eax           ; edx = len(initial_text) + len(content)

```

```

add edx, 3      ; to get terminate symbol

mov ecx, initial_text    ; adding text to print
mov ebx, esi             ; Moving descriptor of file above to EBX
mov eax, 4
int 80h

;;; Quit

mov ebx, esi
mov eax, 6
int 80h

call quit

```

Вместо использования `sys_leek` мы можем инициализировать область памяти для введенного сообщения после инициализации переменной `initial_text` таким образом, если мы вначале введем текст, а потом введем в файл значение из `initial_text` указав его длину большей на длину введенной строки. Тогда мы зайдем на область памяти `content` и выведем и ее на экран. Благодаря этому мы можем вывести изначальный текст и введенный за одно действие.

Запустим программу и проверим ее результат.

```
ские, веками обрабатываемые водой камни
→ lab11 rm name.txt ; ./compile.sh lab11-2;
→ lab11 ./lab11-2
Как Вас зовут? Осокин Георгий
→ lab11 ls
compile.sh lab11-1 lab11-1.asm~ lab11-1.o lab11
in_out.asm lab11-1.asm lab11-1.lst lab11-2 lab11
→ lab11 cat name.txt
Меня зовут: Осокин Георгий
→ lab11 |
```

Рис. 3.1: Вывод содержимого файла name.txt

Как видим, программа работает корректно и добавляет в файл введенную строку.

4 Выводы

Мы приобрели навыки написания программ для работы с файлами и написали программу коорая добавляет в файл введенный текст.