МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

# Кафедра электронных вычислительных машин

Отчет по предмету

Проектирование на языке ассемблера

Вариант 1

Лабораторная работа №5

**«Работа с файлами»**

|  |  |
| --- | --- |
| **Выполнил:**  Студент группы 150501  Барило К.С. | **Проверил:**  Туровец Н.О. |

Минск 2022

**Цель работы:** Ознакомиться с основными операциями обработки файлов, получить понятие о работе с параметрами командной строки.

**Теоретические сведения:**

1. Работа с файлами.

Для работы с файлами в данной лабораторной работе лучше всего ис-

пользовать функции DOS, которые обращаются к файлу через 16-битный идентификатор (дескриптор) файла.

Первые пять значений идентификаторов такого формата инициализиру-

ются системой следующим образом:

-- 0 – STDIN – стандартное устройство ввода (клавиатура),

-- 1 – STDOUT – стандартное устройство вывода (экран),

-- 2 – STDERR – устройство вывода сообщений об ошибках (всегда экран),

-- 3 – AUX – последовательный порт (СОМ1),

-- 4 – PRN – параллельный порт (LPT1).

Работа с файлами выполняется через функции DOS в стандартном поряд-

ке**:**

1) создание или открытие существующего файла;

2) выполнение файловых операций чтения или записи данных;

3) закрытие файла.

Дополнительно доступны операции: удаление, поиск и управление.

Для создания или открытия существующего файла рекомендуется ис-

пользовать функции:

-- Функция DOS 3Ch (INT 21h) – создать файл:

Ввод:AH = 3Ch,

DS:DX = адрес ASCIZ-строки с полным именем файла.

Особенности: Если файл уже существует, функция все равно открывает его,

присваивая ему нулевую длину.

-- Функция DOS 5Bh (INT 21h) – создать и открыть файл:

Ввод: AH = 5Bh,

СХ = атрибут файла,

DS:DX = адрес ASCIZ-строки с полным именем файла.

-- Функция DOS 5Ah (INT 21h) – создать и открыть временный файл:

Ввод: AH = 5Ah,

СХ = атрибут файла,

DS:DX = адрес ASCIZ-строки с путем, оканчивающимся симво-

лом «\», и тринадцатью нулевыми байтами в конце.

Особенности: Функция создает файл с уникальным именем, который не является на самом деле временным, его следует специально удалять, для

чего его имя и записывается в строку в DS:DX.

-- Функция DOS 3Dh (INT 21h) – открыть существующий файл:

Ввод: AH = 3Dh,

AL = режим доступа (биты):

DS:DX = адрес ASCIZ-строки с полным именем файла.

CL = маска атрибутов файла.

АХ = идентификатор файла.

Во всех случаях имя файла (если диск или путь отсутствуют в описании,

то системой используются их текущие значения) описывается ASCIZ-строкой(строкой ASCII-символов, оканчивающейся нулем), которая, например, имеет следующий вид:

filename db 'с:\data\filename.txt',0

Для выполнения файловых операций чтения или записи данных рекомен-

дуется использовать функции:

-- Функция DOS 3Fh (INT 21h) – чтение из файла или устройства:

Ввод: АН = 3Fh,

ВХ = идентификатор файла,

СХ = число байт для чтения,

DS:DX = адрес буфера для приема данных.

АХ = число считанных байт.

АХ меньше, чем заказанное число в СХ, то при чтении был до-

стигнут конец файла.

-- Функция DOS 42h (INT 21h) – переместить указатель чтения/записи:

Ввод: АН = 42h,

ВХ = идентификатор файла,

CX:DX = расстояние, на которое надо переместить указатель

CX:DX = новое значение указателя (в байтах от начала файла).

Эта функция также часто используется для определения длины

файла: достаточно вызвать ее с СХ = 0, DX = 0, AL = 2, и в

CX:DX будет возвращена длина файла в байтах.

-- Функция DOS 40h (INT 21h) – запись в файл или устройство:

Ввод: АН = 40h,

ВХ = идентификатор файла,

СХ = число байт для записи,

DS:DX = адрес буфера с данными.

АХ = число записанных байт.

-- Функция DOS 68h (INT 21h) – сброс файловых буферов DOS на диск:

Ввод: АН = 68h,

ВХ = идентификатор файла.

-- Функция DOS 0Dh (INT 21h) – сброс всех файловых буферов на диск:

Ввод: АН = 0Dh

Для закрытия файла рекомендуется использовать следующую функцию:

-- Функция DOS 3Eh (INT 21h) – закрыть файл:

Ввод: АН = 3Eh,

ВХ = идентификатор файла.

**Код программы**

.model small

.data

Filename db "C:\emu8086\vdrive\C\File.txt", 0;filename

errmes db "Error",'$' ;error message

Buffer db "$$$$$" ;buffer for reading

str\_number\_mes db "Number : " , '$' ;output message

FHndl dw 0 ;descriptor

str\_number dw 0

.code

start:

mov ax, @data

mov ds, ax

mov es, ax

mov ah, 3dh ;open existing file

mov al, 0 ;for reading

lea dx, Filename ;filename to dx

int 21h

jc error ;CF == 0 - error

mov FHndl, ax ;descriptor

LP:

mov ah, 3Fh ;read from file

lea dx, Buffer ;buffer for reading

mov cx, 1

mov bx, FHndl ;descriptor to bx

int 21h

jc error ;CF == 0 - error

cmp ax, cx ;if num of readed bytes != 1 - end of file

jne EOF

push dx ;save dx

mov dl, Buffer

cmp dl , 10 ;if Buffer == '\n' - one more string

je one\_more\_str

LP2:

mov ah, 02h ;output symbol

int 21h

pop dx ;restore dx

jmp LP

EOF: ;end of file

mov bx, FHndl ;descriptor

mov ah, 3Eh ;close file

int 21h

inc str\_number ;if end of file - one more string

jnc quit

one\_more\_str:

inc str\_number

jmp LP2

error: ;out error message

mov ah, 9h

mov dx, OFFSET errmes

int 21h

quit:

mov ah , 2 ;translation to a new line

mov dx , 10

int 21h

mov ah , 09h ;output string message

mov dx , offset str\_number\_mes

int 21h

output:

lea di, str\_number ;result for writing

mov bx, 10 ;divisor

xor cx,cx ;cx = 0

mov ax , str\_number

ASCII:

xor dx,dx ;dx = 0

div bx ;divide on 10

add dl,'0' ;number to symbol

push dx ;save dx to stack

inc cx

test ax,ax ;(ax == 0)?

jnz ASCII

recordRezult: ;pop symbols from stack

pop dx ;pop symbol to dx

mov ah , 2 ;output symbol

int 21h

loop recordRezult

mov ax, 4c00h ;end of program

int 21h

end start

**Результат работы программы:**

