МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Чувашский государственный университет имени И.Н. Ульянова»

Факультет информатики и вычислительной техники

Кафедра вычислительной техники

Отчет по лабораторной работе №3

по дисциплине: «Системное программирование»

«Обработка ошибок»

Выполнил: студент гр. ИВТ-41-20Александров А.Ю.

Проверил: ассистент  
Яковлев С.В.

Чебоксары 2024

**Цель работы:** изучить служебные функции для вывода сообщений об ошибках и проанализировать информацию, выдаваемую функциями perror и ReportError для обычных ошибок.

**Задание:**

1. Добавьте в новый пустой проект служебную функцию **ReрortError,** которая выводит сообщение (первый параметр) и завершается с указанным кодом завершения или по return в зависимости от значения второго параметра. Третий параметр определяет, нужно ли отображать системное сообщение об ошибке:

VOID ReрortError (LРCTSTR UserMessage,

DWORD ExitCode, BOOL РrintErrorMsg)

2. Проверьте работоспособность функции **ReрortError** c помощью тестовой программы (проекта). Измените текст функции так, чтобы она выдавала сообщение об ошибке на русском языке. Для этого используйте функцию **CharToOem.** Проверьте для случаев, когда имя UNICODE определено и не определено. Возможно, потребуется изменить и функцию PrintStrings.

3. Сравните информацию, выдаваемую функциями **рerror** (библиотека С) и ReрortError для обычных ошибок: 1) открытие несуществующего файла, 2) открытие файла, используемого другой программой на чтение, для записи, 3) открытие файла с атрибутом "только для чтения" на запись, 4) запись в файл, открытый только для чтения, 5) запись в файл, редактируемой другим приложением, 6) чтение из файла, редактируемого другой программой, 7) закрытие файла, используемого другой программой.

**Код программы:**

int main(int argc, char \*argv[]) {

FILE \*file1, \*file2, \*file3;

TCHAR buffer[256] = TEXT("Message");

size\_t a = 1;

char errorMessage[7] = "perror";

file1 = fopen("test1.txt", "r");

perror(errorMessage);

ReportError(buffer, 10, TRUE);

file1 = fopen("onlyRead.txt", "r");

file2 = fopen("onlyRead.txt", "w");

perror(errorMessage);

ReportError(buffer, 10, TRUE);

fopen("onlyRead.txt", "w");

ReportError(buffer, 10, TRUE);

file1 = fopen("onlyRead.txt", "r");

const char \*data = "Trying to write to a file opened for reading only.";

fwrite(data, sizeof(char), strlen(data), file1);

perror(errorMessage);

ReportError(buffer, 10, TRUE);

HANDLE hIn = CreateFile(TEXT("test2.txt"), GENERIC\_READ, 0, NULL,

OPEN\_EXISTING, 0, NULL);

file3 = fopen("test2.txt", "wt");

fwrite("test", 1, a, file3);

perror(errorMessage);

ReportError(buffer, 10, TRUE);

HANDLE hIn = CreateFile(TEXT("test2.txt"), GENERIC\_READ, 0, NULL,

OPEN\_EXISTING, 0, NULL);

file3 = fopen("test2.txt", "rt");

char readBuffer[256];

fread(readBuffer, 1, 256, file3);

perror(errorMessage);

ReportError(buffer, 10, TRUE);

file1 = fopen("test2.txt", "rt");

file2 = fopen("test2.txt", "rt");

fclose(file2);

fwrite("test", 1, 3, file1);

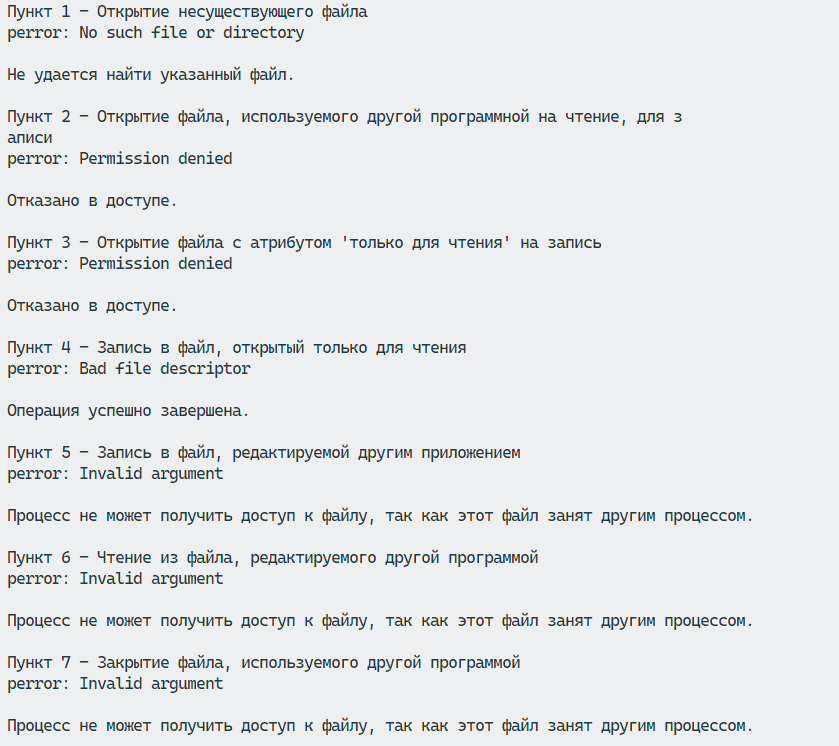
perror(errorMessage);

ReportError(buffer, 10, TRUE);

return 0;

}

**Результат работы программы:**



**Вывод:** в ходе выполнения лабораторная работы мы изучили служебные функции для вывода сообщений об ошибках и проанализировали информацию, выдаваемую функциями perror и ReportError для обычных ошибок.