

# Системное программирование

## Л.р.4. Работа с файлами

### Цель:

Расширенные/дополнительные возможности и механизмы работы с файлами и организации ввода-вывода: неблокирующие и асинхронные операции, мультиплексирование ввода-вывода, отображение файлов в память, мультиплексирование и др.

### Теоретическая и методическая часть

Подсистема ввода-вывода, разновидности ввода-вывода, логические устройства.

Файл и файловая система. Иерархия, виды файлов, файлы и директории (папки). Открытый файл (устройство) как объект, дескрипторный доступ, файловые дескрипторы (Handle).

Основной набор API для работы с файлами: `CreateFile()`, `CloseHandle()`, `ReadFile()`, `WriteFile()`. Синхронный ввод-вывод, в т.ч. неблокирующий.

Асинхронный («перекрывающийся») ввод-вывод. Структура `OVERLAPPED`, функции ожидания `WaitFor***`. Варианты реализации контроля хода транзакций.

Мультиплексированный ввод-вывод. Функции `select()` и др.

Отображение (проецирование) файлов в память. Функции `CreateFileMapping()`, `MapViewOfFile()`. Типовые применения файловых отображений.

### Практическая часть

#### Общая постановка задачи:

Приложение, демонстрирующее работу с файлами, в первую очередь с использованием асинхронных методов и отображения файла в память. Оценка эффективности по сравнению с простейшим (синхронным) вводом-выводом (и, возможно, с многопоточной реализацией приложения при выполнении аналогичных операций над данными).

Специальных требований к приложениям не предъявляется; в частности, во многих случаях они могут быть не обязательно оконными, но также и консольными.

#### Варианты заданий:

- Обработка содержимого файла, используя асинхронные операции
- Обработка содержимого файла, используя отображение в память
- Файл прямого доступа – база данных

– Мультиплексирование ввода-вывода

– ...

### **1 Обработка содержимого файла, используя асинхронные операции**

Реализация обработки содержимого файла данных, используя асинхронный ввод-вывод (чтение/запись очередных порций данных параллельно с выполнением обработки данных в памяти).

Варьирование количества параллельно инициированных операций ввода-вывода, влияние их на общую производительность. Поиск «узкого места» в программной реализации.

Оценка эффективности (производительности) по сравнению с традиционным подходом (чтение → обработка → выгрузка).

Опционально – то же по сравнению с многопоточной реализацией.

Возможные варианты обработки: сортировка, статистическая обработка, криптография и т.д. Объем данных следует выбирать достаточно большим, чтобы длительность обработки была заметна и хорошо поддавалась измерению.

### **2 Обработка содержимого файла, используя отображение в память**

Реализация обработки содержимого файла данных путем отображения его в память и обращения к этой памяти.

Варьирование количества потоков, получающих доступ к отображению (специальных мер для предотвращения коллизий не предусматривается), влияние на общую производительность.

Оценка эффективности (производительности) по сравнению с традиционным подходом (чтение → обработка → выгрузка), в т.ч. в многопоточном варианте.

Возможные варианты обработки: сортировка, статистическая обработка, криптография и т.д. Объем данных следует выбирать достаточно большим, чтобы длительность обработки была заметна и хорошо поддавалась измерению.

### **3 Файл прямого доступа – «база данных»**

Упрощенная база данных (хранение рабочих данных, конфигураций и т.п.). Данные структурированные. Прimitives (процедуры) доступа к структурам данным в файле (воспроизводить язык запросов не требуется).

Доступ осуществляется путем отображения файла в память.

Опционально – сравнение с реализацией доступа синхронными операциями ввода-вывода.

Вопросы для продумывания:

- «метаданные» – описание структуры базы (простейшее минимальное)
- удаление объектов из базы («дырки», сжатие/дефрагментация базы)
- расширение базы (увеличение размера файла)

#### **4 Мультиплексирование ввода-вывода**

Реализация обмена с множеством (ограниченным) источников и/или получателей данных. В качестве источников и получателей могут выступать файлы, устройства, каналы.

Вопросы для проработки:

– моделирование задержек и готовности/неготовности файлов, каналов, устройств.

#### **5 ...**