**Назначение и основные функции операционных систем. Состав операционной системы..**

Доброго времени суток уважаемый пользователь. На этой страничке мы поговорим на такие темы, как: Назначение и основные функции операционных систем. Состав операционной системы.

Операционная система (ОС) – это комплекс взаимосвязанных системных программ для организации взаимодействия пользователя с компьютером и выполнения всех других программ. ОС относятся к составу системного программного обеспечения и являются основной его частью. Операционные системы: MS DOS 7.0, Windows Vista Business, Windows 2008 Server, OS/2, UNIX, Linux.

**Основные функции ОС:**

* управление устройствами компьютера (ресурсами), т.е. согласованная работа всех аппаратных средств ПК: стандартизованный доступ к периферийным устройствам, управление оперативной памятью и др.
* управление процессами, т.е. выполнение программ и их взаимодействие с устройствами компьютера.
* управление доступом к данным на энергонезависимых носителях (таких как жесткий диск, компакт-диск и т.д.), как правило, с помощью файловой системы.
* ведение файловой структуры.
* пользовательский интерфейс, т.е. диалог с пользователем.

**Дополнительные функции:**

* параллельное или псевдопараллельное выполнение задач (многозадачность).
* взаимодействие между процессами: обмен данными, взаимная синхронизация.
* защита самой системы, а также пользовательских данных и программ от злонамеренных действий пользователей или приложений.
* разграничение прав доступа и многопользовательский режим работы (аутентификация, авторизация).

**Состав операционной системы**

В общем случае в состав ОС входят следующие модули:

* Программный модуль, управляющий файловой системой.
* Командный процессор, выполняющий команды пользователя.
* Драйверы устройств.
* Программные модули, обеспечивающие графический пользовательский интерфейс.
* Сервисные программы.
* Справочная система.

Драйвер устройства (device driver) – специальная программа, обеспечивающая управление работой устройств и согласование информационного обмена с другими устройствами.

Командный процессор (command processor) – специальная программа, которая запрашивает у пользователя команды и выполняет их (интерпретатор программ).

Интерпретатор  команд отвечает за загрузку приложений и управление информационным потоком между приложениями.

Для упрощения работы пользователя в состав современных ОС входят программные модули, обеспечивающие графический пользовательский интерфейс.    
Процесс работы компьютера в определенном смысле сводится к обмену файлами между устройствами. В ОС имеется программный модуль, управляющий файловой системой.

Сервисные программы позволяют обслуживать диски (проверять, сжимать, дефрагментировать и др.), выполнять операции с файлами (копирование, переименование и др.), работать в компьютерных сетях.

Для удобства пользователя в состав ОС входит справочная система, позволяющая оперативно получить необходимую информацию о функционировании как ОС в целом, так и о работе ее отдельных модулей.

*Примечание*

*Состав модулей ОС, а также их количество зависит от семейства и вида ОС. Так, например, в ОС MS DOS отсутствует модуль, обеспечивающий графический пользовательский интерфейс.*

Наиболее общим подходом к структуризации операционной системы является разделение всех ее модулей на две группы:

Ядро – это модули, выполняющие основные функции ОС.

Вспомогательные модули, выполняющие вспомогательные функции ОС. Одним из определяющих свойств ядра является работа в привилегированном режиме.

Модули ядра выполняют следующие базовые функции ОС: Управление процессами, Управление системой прерываний, Управление памятью, управление устройствами ввода-вывода, Функции, решающие внутрисистемные задачи организации вычислительного процесса: переключение контекстов, загрузка/вы­грузка страниц, обработка прерываний. Эти функции недоступны для приложе­ний. Функции, служащие для поддержки приложений, создавая для них так называемую прикладную программную среду.

Приложения могут обращаться к ядру с запросами – системными вызовами – для выполнения тех или иных действий: для открытия и чтения файла, вывода графической информации на дисплей, получения системного времени и т.д. Функции ядра, которые могут вызываться приложениями, образуют интерфейс прикладного программирования – API (Application programming interface).

*Пример.   
Базовый код API Win32 содержится в трех библиотеках динамической загрузки (Dynamic Link Library, DLL): USER32, GDI32 и KERNEL32.*

Kernel - модуль Windows, который поддерживает низкоуровневые функции по работе с файлами и управлению памятью и процессами. Этот модуль обеспечивает сервис для 16- и 32-разрядных приложений.

GDI (Graphics Device Interface) - модуль Windows, обеспечивающий реализацию графических функций по работе с цветом, шрифтами и графическими примитивами для дисплея и принтеров.

User - модуль Windows, который является диспетчером окон и занимается созданием и управлением отображаемыми на экране окнами, диалоговыми окнами, кнопками и другими элементами пользовательского интерфейса.

Ядро является движущей силой всех вычислительных процессов в компьютерной системе, и крах ядра равносилен краху всей системы, без него ОС является полностью неработоспособной и не сможет выполнить ни одну из своих функций. Поэтому разработчики операционной системы уделяют особое внимание надежности кодов ядра, в результате процесс их отладки может растягиваться на многие месяцы.

Обычно ядро оформляется в виде программного модуля некоторого специального формата, отличающегося от формата пользовательских приложений.  
Вспомогательные модули ОС выполняют вспомогательные функции ОС (полезные, но менее обязательные чем функции ядра).

Примеры вспомогательных модулей:

* Программа архивирования данных.
* Программа дефрагментации диска.
* Текстовый редактор.

Вспомогательные модули ОС оформляются либо в виде приложений, либо в виде библиотек процедур. Вспомогательные модули ОС подразделяются на следующие группы:

* утилиты – программы, решающие задачи управления и сопровождения компьютерной системы: обслуживание дисков и файлов.
* системные обрабатывающие программы – текстовые или графические редакторы, компиляторы, компоновщики, отладчики.
* программы предоставления пользователю дополнительных услуг пользовательского интерфейса (калькулятор, игры).
* библиотеки процедур различного назначения, упрощающие разработку при­ложений (библиотека математических функций, функций ввода-вывода).

Как и обычные приложения, для выполнения своих задач утилиты, обрабатывающие программы и библиотеки ОС, обращаются к функциям ядра посредством системных вызовов.   
Функции, выполняемые модулями ядра, являются наиболее часто используемыми функциями операционной системы, поэтому скорость их выполнения определяет производительность всей системы в целом. Для обеспечения высокой скорости работы ОС все модули ядра или большая их часть постоянно находятся в оперативной памяти, то есть являются резидентными.

Вспомогательные модули обычно загружаются в оперативную память только на время выполнения своих функций, то есть являются транзитными. Такая организация ОС экономит оперативную память компьютера.

*Примечание*

*Разделение операционной системы на ядро и вспомогательные модули обеспечивает легкую расширяемость ОС. Чтобы добавить новую высокоуровневую функцию, достаточно разработать новое приложение, и при этом не требуется модифицировать основные функции, образующие ядро системы.*

Объектами ядра ОС являются:

* Процессы (рассмотрено в теме 2.3).
* Файлы.
* События.
* Потоки (рассмотрено в теме 2.3).
* Семафоры – объекты, позволяющие войти в заданный участок кода не более чем n потокам.
* Мьютексы – одноместные семафоры, служащие в программировании для синхронизации одновременно выполняющихся потоков.
* Файлы, проецируемые в память.