# Операционные системы

ТЕМЫ НА СЕМЕСТР

I. Архитектура ОС

1. Подсистема управления процессом
2. Подсистема управления памятью
3. Подсистема управления файлами
4. Подсистема ввода-вывода

II. Unix,Linux

III. Безопасность

Тест по лекциям за месяц до окончания семестра

## Лекция 1: Архитектура операционных систем

### Функциональные компоненты ОС

ОС – это комплекс взаимодействующих программ, предназначенных для повышения эффективности аппаратуры и управления ресурсами ПК.

Управление ресурсами включает в себя следующие задачи:

1. Планирование ресурсов, т.е. к какому процессу (запущенных процесс) выделить ресурс
2. Удовлетворение запроса на ресурс
3. Отслеживание состояние ресурса
4. Разрешение конфликтов между процессами

#### Основные ресурсы ОС

1. Процессы – самый главный ресурс
2. Память
3. Таймеры
4. Наборы данных
5. Сетевые устройства
6. Устройства ввода-вывода

Все эти ресурсы распределяются между процессами.

Процесс – динамический объект, который возникает после запуска программы на выполнение. Задача любой ОС – эффективно управлять процессами

Все функции ОС группируются в соответствии с типами ресурсов. Это 4 группы, которые называются подсистемами ОС.

1. Подсистема управления процессами.
2. Подсистема управления памятью
3. Файловая система
4. Подсистема ввода-вывода

#### Подсистема управления процессами

Для выполнения процессов ОС должна выделить область памяти в которой разместятся данные процессы и выделить процессу нужное ему кол-во процессорного времени (время в ЦП). Одновременно может быть запущено несколько процессов. Процессы бывают пользовательскими, т.. запущенными пользователем и системными, запускается самой ОС для своего функционирования. Поскольку процессы претендуют на одни и те же ресурсы ОС создает очереди заявок процессов на ресурсы. Одна из задач ОС – защита ресурсов, выделенных процессу от других процессов. Например, защищать память текущего процесса и не давать ее другим. В итоге подсистема:

1. Планирует выполнение процессов, распределяя между ними процессорное время
2. Занимается созданием и уничтожением процессов
3. Обеспечивает процессы ресурсами
4. Обеспечивает взаимодействие процессов

#### Подсистема управления памятью

Функции:

1. Отслеживание свободной и занятой памяти
2. Выделение памяти процессу
3. Освобождение памяти при завершении процесса
4. Защита памяти
5. Использование виртуальной памяти – это вытеснение процессов из оперативной памяти на жесткий диск, когда размеры основной памяти недостаточны для выполнения всех работающих программ и возвращение процессов в оперативную память, когда место освобождается.

#### Подсистема управления файлами и внешними устройствами

ОС виртуализирует набор данных, хранящихся на внешнем носителе в виде последовательности байтов, которым дают символьное имя конкретного файла. Основное назначение файловой системы:

1. Преобразовать символьное имя в физический адрес
2. Организовать совместный доступ к файлом
3. Их защита

При выполнении своей функции файловая система взаимодействует с устройствами ввода-вывода через специальные программы – драйвера, которые управляют конкретной моделью устройства. Драйвер также может управлять группой устройств. От наличия драйверов в составе ОС зависит успех этой ОС на рынке.

#### Требования к современным ОС

1. Расширяемость. Достигается за счет модульности ОС, когда программа строится из отдельных модулей (Unix/Linux – открытая архитектура)
2. Переносимость – это способность ОС работать на разных платформах, которые отличаются и процессором и аппаратной частью.
3. Совместимость. Если у ОС имеются средства к выполнению прикладных программ, написанной для другой ОС, то она обладает совместимостью.
4. Отказоустойчивость и надёжность. Система должна иметь защиту от внутренних и внешних ошибок: сбоев, отказов, а ее действия в этих ситуациях должны быть предсказуемы. Приложения не должны наносить вред ОС.
5. Безопасность. ОС должна нести защиту от несанкционированного доступа. Это реализуется тремя параметрами:

* Аутентификация – определение пользователя через его учетную запись
* Авторизация – это предоставление прав доступа
* Аудит – это фиксация всех событий ОС

1. Производительность. Определяется архитектурой, качеством ядра ОС и многопроцессорной платформой, размером кэша
2. Сетевая поддержка – это совокупность серверной и клиентской части, дающий доступ к ресурсу через сеть и называемая *сетевой службой.* Все сетевые службы ОС формируют сетевой сервис ОС, а в зависимости типа ОС службы могут быть встроены или поставляться.

### Подсистема управления процессами

Процесс – динамический объект, которому предоставлен ЦП. Когда ОС переключает ЦП с процесса на процесс, она использует регистры ЦП, в котором хранится контекст процессов

#### Контекст процессов

Планировщик ОС отвечает за очередность выполнения процессов и за выдачи им процессорного время. Время процессам выделяется квантами. Каждый процесс ставится в ЦП на квант времени.

Размер Кванта

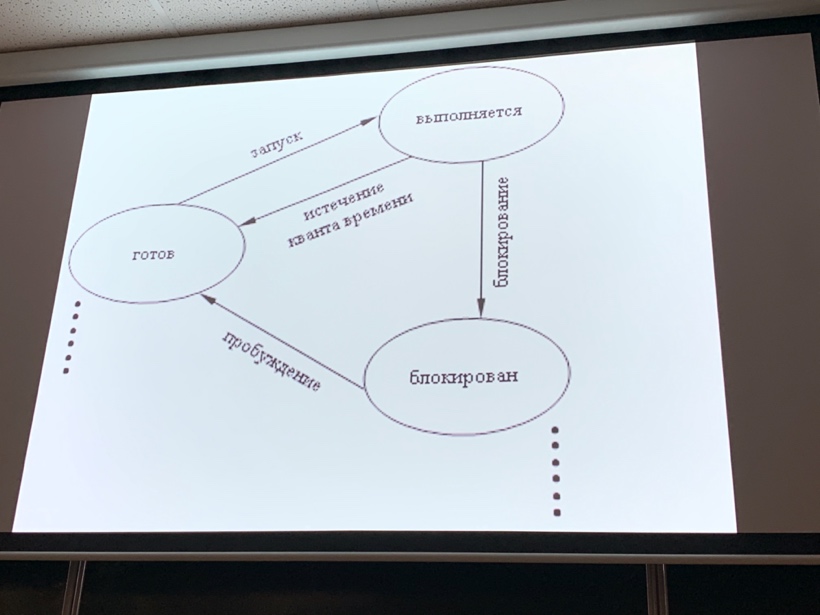
Unix – 20-40 миллисекунд

Linux – 200 миллисекунд

Windows – 15-20 миллисекунд

Размер пользователь может изменить

Малый квант означает, что ОС часто снимает процессы, в результате переключения – много ненужной работы, но в целом программы работают процессы. По окончанию кванты процесс снимается с ЦП и запускается следующий процесс. Эта ситуация называется переключением контекста процесса. каждый процесс описывается своим контекстом. Контекст включает:

1. Адресное пространство
2. Сегменты программного кода
3. Данные для его выполнения
4. Параметры виртуальной памяти
5. Содержимое аппаратных регистров
6. Данные ядра
7. Идентификатор процесса, каждый процесс имеет свой уникальный идентификатор PID
8. Идентификатор родительского процесса
9. Состояние процесса. Процесс может находится в 3 состояния: готов, выполняется, блокирован. Если готов, то может сразу поставить на ЦП. Выполняется: процесс находится на ЦП, на квант времени. Если при этом кванта не хватает, а данные есть, процесс переходит в состояние готов. Если во время выполнения кончились данные, а квант не израсходован, процесс снимается с ЦП и блокируется до момента пока кэш не получит данные и не передаст в регистр процесса.
10. Приоритет процесса. Все программы выполняются согласно своему приоритету. В Windows 32 уровня приоритета:

* Real time class (24) – программа выполняется так, будто она одна, все ресурсы этой программе
* High class (13) - программа выполняется так, что чуть-чуть оставляет ресурсы другим
* Above normal class 10
* Normal class 8 – обычное выполнение пользовательской программы
* Below normal class 6
* Idle class (4) – программа выполняется, когда ОС нечего делать

### Планирование выполнения процессов