

ОС – комплект программ, которые совместно управляют ресурсами вычислительной системы и процессами, которые используют эти ресурсы в вычислениях.

Ресурс – любой компонент выч.системы и предоставляемые ею возможности, напр. Время ЦП, объем ОП,.

Тупиковая ситуация (тупик) - это ситуация, возникающая в результате монопольного использования разделяемых ресурсов, когда процесс, владея ресурсом, запрашивает другой ресурс, занятый непосредственно или через цепочку запросов другим процессом, ожидающим освобождения ресурса занятого первым процессом.

Резидентная программа – программа, которая должна постоянно находиться в памяти и реагировать на запросы пользователя быстро или на какие-либо события, проходящие в выч.системе.

Прерывание – асинхронное событие,возникающее в вычислительной системе.

Процесс - программа в стадии выполнения. Единица декомпозиции ОС (именно ему выделяются ресурсы ОС).

М. делиться на потоки, программист созд. в своей программе потоки, которые выполняются квазипараллельно.

Поток (thread, light-weight process, нить) - часть последовательного кода процесса, которая м. выполняться параллельно с другими частями кода. Не имеет своего адресного пространства.

Аппаратный контекст процесса – сост. регистров.

Полный контекст процесса - сост. регистров + сост. памяти.

Планирование – управление распределением ресурсов ЦП между разл. конкурирующими процессами

путем передачи им управления согласно некот. стратегии планирования.

Диспетчеризация – выделение процессу процессорного времени.

Программа-планировщик

программа, отвечающая за управление использованием совместного ресурса.

Монитор –

языковая конструкция, состоящая из структур данных и подпрограмм, использующих данные структуры. Монитор защищает данные. Доступ к данным монитора могут получить только п/п монитора. В каждый момент времени в мониторе м. находиться только 1 процесс. Монитор является ресурсом. Процесс, захвативший монитор, -

процесс в мониторе, процесс, ожидающий в очереди, – процесс на мониторе.

Простой монитор – предназначен для разделения 1 единственного ресурса произвольному числу процессов. Монитор, обслуживающий произвольное количество процессов, ограничен длиной своей очереди.

Монитор «кольцевой буфер» – решает задачу «производство-потребление», то есть существуют процессы производители и процессы-потребители, а также буфер – массив заданного размера, куда производитель помещают данные, а потребители считывают оттуда данные в том порядке, в котором они помещались (FIFO “первым вошел, первым вышел”).

Семафор - неотрицательная защищённая переменная S, над которой определено 2 неделимые операции: P (пропустить) и V (освободить). Защищённость семафора означает, что значение семафора м. изменяться только операциями P и V.

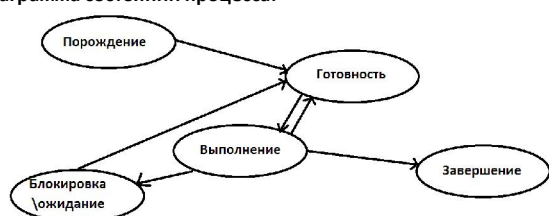
1. Операция P(S): $S = S - 1$.
Декремент семафора если он возможен). Если $S = 0$, то процесс, пытающийся выполнить операцию P, будет заблокирован на семафоре в ожидании, пока S не станет больше 0. Его освобождает другой процесс, выполняющий операцию V(S).

2.Операция V(S): $S = S + 1$.Инкремент S одним неделимым действием (последовательность непрерывных действий: инкремент, выборка и запоминание). Во время операции к семафору нет доступа для других процессов. Если $S = 0$,то V(S) приведёт к $S = 1$.Это приведёт к активизации процесса, ожидающего на семафоре.

Монопольное использование -

если процесс получил доступ к разделяемому ресурсу, то др. процесс не м. получить доступ. Необходимо обеспечить монопольный доступ процесса к разделяемому ресурсу до тех пор, пока процесс его не освободит. Все алгоритмы программной реализации обобщил Дейкстра, введя понятие семафора.

Диаграмма состояний процесса:



Защищенный режим – многопроцессный

режим. В памяти компьютера одновременно находится большое число программ с квантованием процессорного времени с виртуальной памятью. Управляет защищенным режимом ОС с разделением времени (Windows, Linux). В защищенном режиме 4 уровня привилегий, ядро ОС – на 0-м. Создан для работы нескольких независ. программ. Для обеспечения совместной работы нескольких задач необходимо защитить их от взаимного влияния, взаимодей. задач д. регулироваться. Разработан для 32-разрядных компьютеров. Формирование таблиц описания памяти, которые определяют состояние её отдельных сегментов/страниц и т. п.

Расширенная память XMS (eXtended Memory Specification) - программная спецификация использования дополнительной памяти DOS-программами, разработанная компаниями Lotus, Intel, Microsoft и AST для компьютеров на процессорах 286 и выше. Эта спецификация позволяет программе получить в распоряжение 1 или неск. областей доп.памяти, а также использовать область HMA. Распределением областей ведаёт диспетчер расширенной памяти - драйвер HIMEM.SYS. Диспетчер позволяет захватить или освободить область HMA (65 520 байт, начиная с 100000h), а также управлять вентилем линии адреса A20.

Программный канал –

это специальный буфер, который создается в системной области памяти. Они описываются в соответствующей системной таблице. Информация в канал записывается по принципу FIFO и не модифицируется. В канал нельзя писать, если его читают и наоборот. 2 типа программных каналов: имеющиеся, неименованные.

Разделяемые память (сегменты) –

средство взаимодействия процессов через разделяемое адресное пространство. Т.к.

адресное пространство защищено, процессы могут взаимодействовать только через ядро. Подключается к адресному пространству процесса (виртуальный).

Виртуальная память – память размер которой превосходит размер физического адресного пространства. Используется адресное пространство диска как область свопинга или педжинга, т.е. для временного хранения областей памяти. ИЛИ система, при которой рабочее пространство процесса частично располагается в основной памяти, а частично во вторичной. При обращении по к-л адресу система аппаратными средствами определяет: присутствует ли искомая область физической памяти, если отсутствует, то генерируется прерывание, это позволяет супервизору передать необходимые данные из вторичной области в основную.

Супервизор – управляющая программа (или комплекс) предназначенная для организации многопрограммного режима работы ЭВМ.

«сирота» – возникает в том случае, если процесс предок завершился раньше своих потомков. При завершении процесса система проверяет не осталось ли у этого процесса незавершенных потомков. Если остались – система принимает действия по их усыновлению. Процесс потомок усыновляется терминальным процессом. Система переписывает идентификатор такого процесса на 1

зомби – если процесс потомок завершился до того как предок вызвал wait (возможно при аварийном завершении exes), то для того чтобы предок не завис в ожидании несуществующего процесса, система отбирает у него все ресурсы и помещает строку в таблице процессов, помечая такой процесс как зомби. Сделано это для того, чтобы процесс получил статус завершения всех своих потомков.

Чтобы родитель не завис система отбирает у потомка все ресурсы, но оставляет строку в таблице процессов.

демон – процесс, который не имеет предков (сервисные функции)

Исключения являются синхронным событием, возникают в процессе выполнения программы, возникают в таких случаях, как:

- Арифметическое переполнение
- деление на ноль
- попытка выполнить некорректную команду,
- Ссылка на запрещенную область памяти
- При образовании адреса, при обращении к физическому адресу, которого нет

Исправимые Исключения – например, обращение к некорректному адресу, но он прошел проверку адреса, обращение к отсутствующему сегменту. Процесс может продолжаться с той же команды, в которой произошло исключение.

Неисправимые Исключения - заканчиваются завершение программы

Нарушения – fault – это исключение, фиксируемое до выполнения команды или в процессе её выполнения.

Ловушка – Trap – процессором обрабатывается после команды, вызвавшей это исключение.

Авария – abort – данный тип исключения является следствием невосстановимых, неисправимых ошибок, например, деление на ноль.

Системные вызовы Также являются исключениями, но с точки зрения реализации – это системные ловушки.