

1) Структура ОС.

Если программа меняет информацию, с которой она работает (помимо своей собственной), то это *обрабатывающая* программа (компиляторы, загрузчики, редакторы связей), если не меняет, – то *управляющая*.

Управляющие программы выполняют следующие действия:

1. **Управление** – на уровне ядра, в режиме супервизора осуществляется:
 - отслеживание входа и выхода задания в системе;
 - отслеживание выполнения собственной задачи и управление ее выполнением;
 - обеспечение работы систем управления файлами, памятью, внешними устройствами.
 2. **Управление задачами.** Следит за выполнением задачи на ресурсах системы
 3. **Система управления файлами.**
 4. **Управление памятью.**
 - программное (обеспечивает эффективное функционирование памяти в соответствии с ее организацией; программа решает свои задачи на уровне процесса, то есть память выделяется процессу);
 - микропрограммное (программа запрашивает у админа, какие разделы необходимо выполнять; у каждого раздела – своя очередь);
 5. **Управление внешними устройствами.**
 6. Управление заданиями отслеживает прохождение задания от входа до выхода
- Современные ОС имеют сложную структуру, каждый элемент которой выполняет определенные функции по управлению компьютером.
- Управление файловой системой. Процесс работы компьютера сводится к обмену файлами между устройствами. В ОС имеются программные модули, управляющие файловой системой.
 - Командный процессор. В состав ОС входит специальная программа – командный процессор, – которая запрашивает у пользователя команды (запуск программы, копирование, удаление и т.д.) и выполняет их.
 - Драйверы устройств. В состав ОС входят драйверы устройств, специальные программы, которые обеспечивают управление работой устройств и согласование информационного обмена с другими устройствами, позволяют производить настройку параметров устройств. Каждому устройству соответствует свой драйвер. Программы (драйверы) для поддержки наиболее распространенных устройств входят в Windows, а для остальных устройств поставляются вместе с этими устройствами или контроллерами. Можно самому установить или переустановить драйвер.
 - Графический интерфейс. Для упрощения работы пользователя в состав ОС входят программные модули, создающие графический пользовательский интерфейс. Пользователь может вводить команды с помощью мыши, а не с клавиатуры.
 - Сервисные программы. (утилиты). Позволяют обслуживать диски (проверять, сжимать, дефрагментировать), выполнять операции с файлами (архивировать и т.д.), работать в сетях и т.д.
 - Справочная система. Позволяет оперативно получить необходимую информацию о функционировании ОС.
- 1 Виды модулей. Виды: 1) исходный (текст программы); 2) объектный; 3) загрузочный; 4) исполняемый (абсолютный). Системные программы (компилятор, редактор связей, загрузчик) используются для преобразования от 1) к 4).
 - 2 Отличие загрузочного модуля от объектного. Загрузочный модуль имеет все для своего выполнения. Написан на машинно-ориентированном языке, но быть выполненным не может. При использовании компилятора из исходного модуля получается объектный модуль, при использовании редактора связи из объектного получается загрузочный, далее из него при использовании загрузчика получается исполняемый модуль.
 - 3 Отличие загрузочного модуля от абсолютного. Загрузочный модуль имеет все для своего выполнения. Написан на машинно-ориентированном языке, но быть выполненным не может. Нужно настроить адресные константы. Абсолютный – все адресные константы уже настроены.
 - 4 Дать определение – резидентный модуль. модуль, который резидентно находится в памяти. (постоянно) Резидентные проги. – находятся в ОП, поддерживают быстрый отклик системы, Проги обработки прерываний, Процессы 4/ работы с ВУ, управление памятью, управление процессами, начальная прога. Управления заданиями.
 - 5 Дать определение – транзитный модуль. Программы, связанные с выполнением функций ОС, но не находящиеся постоянно в ОП называются транзитными. Эти программы вызываются в ОП по мере необходимости. – проги., которые могут понадобиться 4/ выполнения ф-ций ОС (но не резидентные) – могут вызываться по оверлейной || динамически-последовательной схеме в транзитную зону.
 - 6 Связь модулей по управлению. Какие операции, какими программами выполняются. Связь по управлению предусматривает сохранение состояния модуля на момент перехода на вызываемый модуль, причем действия по этому сохранению выполняет вызываемый модуль. Операции: возврат из i-1 модуля только в i+1; возврат из i+1 в i.

Вызывающий модуль после передачи управления стирается. Виды программ могут быть: повторно не исполняемые, повторно исполняемые, чистые процедуры. (via common regs) Связь по управлению – связь для организации возврата. Возврат модуль В должен иметь команду, делающую так, что в области сохранения модуля А сохраняется состояние регистров. В модуле В есть адрес области сохранения в модуле А. Сохраняется адрес возврата, для возврата адреса следующей команды.

7 Связь модулей по данным. Виды связей Виды: по данным и по управлению. По данным может быть через общие регистры или через адрес списка параметром. Связь по данным – данные через общий регистр || через системный регистр передается адрес списка параметров.

Свойства модуля. Стандартность внутренней структуры, функциональная завершенность, параметрическая универсальность, взаимная независимость.

8 Виды загрузчиков Отличаются выполнением основных 4-х функций: распределения памяти, настройки, редактирования и загрузки. Загрузчики делятся: Абсолютный загрузчик, настраивающий загрузчик, непосредственно - связывающий загрузчик.

9 Функции абсолютного загрузчика Распределение памяти, настройка, редактирование;

10 Функции настраиваемого загрузчика Распределение памяти, настройка, редактирование, загрузка. Инфу для него готовит компилятор.

11 Какие топы загрузчика работают с модулями COM и EXE. С COM - абсолютный; с EXE - настраивающий.

12 Какую информацию и как компилятор передает настраивающему загрузчику Настраивающий – работает в загр. Модуле которому больше ничего не надо. Постепенно его ф-ции взял на себя редактор связей. Непосредственно в настраивающем загрузчике каждый модуль может транслироваться отдельно. Чтобы передать сообщение редактору связей надо ему непосредственно указать, что надо транслировать. В каждом модуле в начале трансляции выделяются вектора перехода, внешние и внутренние.(Экспорт и Импорт процедур и ф-ций). Кроме того для выполнения настройки каждая команда отмечается битом переместимости. ОС выделяет и пользуется глобально выделенной памятью, а загрузчик с локальной.

13 Функции редактора связей Редактирование связей.(Выполнение связывания подпрограмм являющихся внешними по отношению к загружаемому модулю). (С помощью редактора связи мы получаем из объектного модуля загрузочный модуль имеющий всё для своего исполнения).

14 Выходная и входная информация редактора связей Входная - объектный модуль, выходная - загрузочный модуль.

15 Дать определение – обрабатывающие программы ОС Программы выполнения стандартных (в рамках ОС) функций, обработки исключительных ситуаций. Обработка – изменение ин-фы с которой работает прога в составе ОС (дрова, загрузчик)

16 Дать определение – управляющие программы ОС Программы постоянно находящиеся в памяти (резидентные) организующие корректное выполнение процессов и функционирование всех устройств системы при решении задач. Составляют ядро ОС. Управление : Заданиями – слежение за прохождением заданий от входа до выхода на всех этапах его выполнения. Задачами – (Процессами) – слежение за всеми задачами активизированными в системе и процессами их выполнения на ресурсах. Памятью – решение задач эффективного и/ mem (internal) в соответствии с ее организацией.(Защита – согласование ин-фы в кешах) Данными – эффективное размещение и и/ данных на внешних носителях (проблема эффективности и/ процессора) Внешними ус-вами ...

17 Основное отличие компилятора от интерпретатора Интерпретатор выполняет перевод части проги в машинные команды и тут же ее выполняет, выполняя все необходимые настройки адресных констант. Компилятор создает объектный модуль, который затем обрабатывается редактором связей. Во время работы интерпретатора исполняемый код программы записывается в фиксированное место и управление передается на стартовый адрес программы.

2) Особенности доставки сообщения заблокированным процессам.

Доставка сообщений заблокированному процессу.

Во время работы процесса может возникнуть ситуация, когда процесс не может продолжать свою работу, пока не будет достигнуто определенное условие. Например, процессу необходимо обратиться к внешнему устройству. Для этого процесс

использует системный вызов и активизируется новый процесс, выполняющий данную работы. Предыдущий процесс переходит в заблокированное состояние. После окончания ввода, процесс, работающий с ВУ используя системный вызов инициирует сообщение. Операционная система с помощью планировщика (кажется) доставит сообщение заблокированному процессу и переведет его в готовое состояние.

3) Реализация когерентности в многопроцессорной ОС.

В современных микропроцессорах, используемых для построения мультипроцессорных систем, идентичность данных в кэшах ВМ (когерентность кэш-кэшей) поддерживается с помощью межмодульных пересылок. Существует несколько способов реализации когерентности, применяемых в зависимости от типа используемой коммуникационной среды и сосредоточенности или физической распределенности памяти между процессорными модулями.

Рассмотрим реализацию одного из алгоритмов поддержки когерентности кэш-кэшей, известного как MESI (Modified, Exclusive, Shared, Invalid).

Каждая строка кэш-памяти ВМ может находиться в одном из следующих состояний:

M - строка модифицирована (доступна по чтению и записи только в этом ВМ, потому что модифицирована командой записи по сравнению со строкой основной памяти);

E - строка монополично копированная (доступна по чтению и записи в этом ВМ и в основной памяти);

S - строка множественно копированная или разделяемая (доступна по чтению и записи в этом ВМ, в основной памяти и в кэш-памяти других ВМ, в которых содержится ее копия);

I - строка, невозможная к использованию (строка не доступна ни по чтению, ни по записи).

Состояние строки используется, во-первых, для определения процессором ВМ возможности локального, без выхода на шину, доступа к данным в кэш-памяти, а, во-вторых, - для управления механизмом когерентности.

Для управления режимом работы механизма поддержки когерентности используется бит WT, состояние 1 которого задает режим сквозной (write-through) записи, а состояние 0 - режим обратной (write-back) записи в кэш-память.

Пропуск чтения в кэш-памяти заставляет вызвать строку из основной памяти и сопоставить ей состояние E или S. Кэш-память заполняется только при промах чтения. При промахе записи транзакция записи помещается в буфер и посылается в основную память при предоставлении шины.

Для поддержки когерентности строк кэш-памяти при операциях ввода/вывода и обращениях в основную память других процессоров на шине генерируются специальные циклы опроса состояния кэш-памятей. Эти циклы опрашивают кэш-памяти на предмет хранения в них строки, которой принадлежит адрес, используемый в операции, инициировавшей циклы опроса состояния. Возможен режим принудительного перевода строки в состояние I, который задается сигналом INV. Доп. Инф.

Когерентность

Для программиста различают явный и неявный доступ к данным. Тогда:

- 1) Явное размещение данных – явный доступ (все операции send/receive заданы явно).
- 2) Неявное размещение – неявный доступ (доступ к данным прозрачный для программиста).
- 3) Неявное размещение страниц в памяти – явное указание доступа (используется разделяемое множество страниц на разных устройствах).
- 4) Явное размещение – неявное указание доступа, используется команда Load-store.

Реализация когерентности в однопроцессорных машинах:

Три способа организации памяти:

- с прямым отображением;
- частично ассоциативная;
- ассоциативная.

Для обеспечения когерентности необходимо выбрать правила согласования:

1. Со сквозной записью (самая быстрая).
2. С обратной записью (при кэш промахе или при смене страниц).
3. С буферизацией (происходит накопление в буфере).

Сквозная – изменения переносятся вниз, как только они возникают в КЭШ-ах.