

Планирование и диспетчеризация процессов.

Доступ любой задачи к центральному процессору осуществляется через системные программы планировщика-диспетчера.

Планирование – это организация процессов в некоторую последовательность согласно определенной стратегии.

Планирование и диспетчеризация процессов.

Планировщик – это программа, ответственная за постановку процессов в очередь на выполнение и управляющая структурой этой очереди.

Диспетчер – программа, которая выбирает процессы из очереди-на-выполнение, переводит их в активное состояние и передает им контроль над ЦП.

Планирование и диспетчеризация процессов.

Как правило в большинстве систем не делается различий между планированием и диспетчеризацией, и обе эти операции обычно объединяют в один программный модуль.

Планирование и диспетчеризация процессов.

Квантование времени - используется для обеспечения эффективной интерактивной работы с компьютером нескольких пользователей.

Планирование и диспетчеризация процессов

Устанавливается интервал времени - квант, в течении которого процесс может исполняться на процессоре. По истечении кванта следующий процесс назначается на процессор. Готовые к исполнению процессы по очереди циклически получают свои кванты. Таким образом, создается эффект одновременного параллельного исполнения процессов.

Планирование и диспетчеризация процессов.

Приоритет - параметр, определяющий преимущества процесса в обладании ресурсами по сравнению с другими процессами.

Статический приоритет - приоритет, который устанавливается и изменяется пользователем.

Планирование и диспетчеризация процессов.

Динамический приоритет - приоритет, устанавливаемый и изменяемый операционной системой для обеспечения эффективной работы компьютера.

Планирование и диспетчеризация процессов.

Таким образом, основное отличие статических и динамических приоритетов в том, что статические приоритеты система только учитывает в своей работе, а динамические она формирует и модифицирует, исходя из общих критериев эффективности.

Планирование и диспетчеризация процессов.

Абсолютные и относительные приоритеты различаются степенью срочности предоставления привилегий их обладателям.

Относительные приоритеты учитываются при выделении свободных (освобожденных) ресурсов.

Планирование и диспетчеризация процессов.

Появление процесса с более высоким абсолютным приоритетом при отсутствии свободных ресурсов приводит к захвату ресурсов у менее приоритетных процессов.

Например, замена менее приоритетного процесса на процессоре до истечения кванта либо вытеснение из оперативной памяти во внешнюю.

Планирование и диспетчеризация процессов.

Свопинг процессов - вытеснение готового к исполнению процесса из оперативной памяти во внешнюю. При этом оперативная память, заполненная процессом, полностью освобождается.

Планирование и диспетчеризация процессов.

Свопинг процессов может использоваться как средство реализации абсолютных приоритетов при отсутствии виртуальной памяти либо инициироваться операцией на внешнем устройстве, требующей значительных временных затрат. Например, в некоторых системах выполняется свопинг процессов на время терминального ввода.

Планировщик процессов

Планировщик процессов (и регулировщик) определяет, какой из процессов, когда и на сколько времени получит управление процессором.

Ими должны выполняться следующие функции:

Планировщик процессов

1. Регистрация информации о состоянии всех процессов (процессы могут находиться в одном из трёх состояний - выполнение, готовность или ожидание). Компонент, выполняющий эти функции, был назван регулятором

Планировщик процессов

2. Принятие решения о том, какой из готовых к выполнению процессов должен получить управление, какой из имеющихся в системе процессоров ему предоставить и на сколько времени. Эти действия производятся *планировщиком процессов*.

Планировщик процессов

3. Подключение процессора к процессу. При этом необходимо установить значения регистров процессора в соответствии с текущим состоянием процесса. Эти действия производятся регулятором.

Планировщик процессов

4. Отключение процессора от процесса, например, при истечении выделенного интервала времени или при переходе процесса в состояние ожидания ввода-вывода. При этом должны быть сохранены значения отражающих состояние процесса управляющих регистров, что необходимо для последующего его восстановления. Эта функция также выполняется регулятором.

Планировщик процессов

Планирование использования процессора выступает в качестве краткосрочного планирования процессов.

Оно проводится, к примеру, при обращении исполняющегося процесса к устройствам ввода-вывода или просто по завершении определенного интервала времени. Поэтому краткосрочное планирование осуществляется весьма часто, как правило, не реже одного раза в 100 миллисекунд.

Планировщик процессов

Когда и какой из процессов нужно перекачать на диск и вернуть обратно, решается дополнительным промежуточным уровнем планирования процессов — среднесрочным.

Планировщик процессов

Планирование заданий выступает в качестве долгосрочного планирования процессов. Оно отвечает за порождение новых процессов в системе, определяя ее *степень мультипрограммирования*, т. е. количество процессов, одновременно **находящихся в ней**. Долгосрочное планирование осуществляется достаточно редко, между появлением новых процессов могут проходить минуты и даже десятки минут.

Дисциплины обслуживания

Дисциплина обслуживания определяет порядок, в котором процессор обслуживает процессы и длительность квантов времени обслуживания для каждого из них.

Дисциплины обслуживания

Длительность интервала времени, в течение которого процессор обслуживает данный процесс, может зависеть от одного из следующих событий или их комбинации:

- Выполнение процесса завершено.*
- Процесс перешел в состояние ожидания.*
- Процессор потребовался для обслуживания процесса, имеющего более высокий приоритет.*
- Завершился заранее выделенный процессу квант времени.*
- Произошла ошибка.*

Дисциплины обслуживания

Решение о выборе процесса на обслуживание принимается путём просмотра очереди готовых РСВ в соответствии с принятой дисциплиной обслуживания.

Очередь готовых РСВ может быть организована одним из следующих способов:

Дисциплины обслуживания

1. Всякий раз, когда процесс переводится в состояние готовности (из состояний хранения, выполнения или ожидания), его РСВ помещается в соответствующую приоритету процесса позицию очереди(списка). Когда процессор освобождается, он переключается на обслуживание процесса, стоящего в очереди первым.

Дисциплины обслуживания

2. Процессы располагаются в очереди (списке) произвольно. Когда необходимо выбрать процесс для обслуживания, производится полный просмотр списка.

Дисциплины обслуживания

Можно перечислить следующие широко распространённые дисциплины планирования:

Дисциплины обслуживания

1. *Равномерное циклическое квантование.*

Длительность обслуживания каждого процесса ограничена величиной стандартного кванта времени, равного, например, 100 мс.

Дисциплины обслуживания

2. Приоритет по характеру использования предшествующего кванта. Если процесс полностью использовал свой предшествующий квант времени, он помещается в конец очереди. Если он использовал только половину (вследствие блокирования по вводу-выводу), он попадает в середину очереди. Помимо "справедливости", эта дисциплина хорошо обслуживает процессы с интенсивным вводом-выводом.

Дисциплины обслуживания

3. Вариант равномерного циклического квантования с многоуровневой обратной связью. В момент введения нового процесса в систему ему выделяется столько квантов, сколько в этот момент существует других параллельных процессов. После этого продолжается обычное равномерное квантирование.

Дисциплины обслуживания

4. *Принудительный приоритет.* Выбирается процесс с наивысшим приоритетом из всех готовых. Приоритет назначается, исходя из важности соответствующей работы или с учётом дополнительной оплаты (например, задачам, обслуживающим ядерный реактор, назначается наивысший приоритет).

Дисциплины обслуживания

5. Ограниченное равномерное циклическое квантирование. Процессы обслуживаются способом равномерного квантирования некоторое определённое число раз, после чего они обслуживаются только при условии отсутствия в системе других процессов.

Дисциплины обслуживания

6. Сбалансированная загрузка. Для предотвращения простоя внешних устройств предпочтение отдаётся процессам, интенсивно использующих ввод-вывод.

Дисциплины обслуживания

7. Предпочтительное обслуживание интерактивных процессов. Если при решении задачи осуществляется оперативное взаимодействие с пользователем (интерактивные задачи), обслуживание соответствующего процесса начинается сразу после запроса пользователя. **с помощью такой дисциплины удаётся обеспечить малое время реакции системы.**

Дисциплины обслуживания

8. Автоматические приоритеты. В некоторых случаях ответственность за назначение процессам приоритетов возлагают на саму ОС. Например, она сама может повышать приоритеты коротких заданий. В других системах производится автоматическое назначение приоритетов так, что достигается сбалансированная загрузка оборудования, как в случае совместного выполнения двух процессов, один из которых интенсивно использует ввод-вывод (*задача с преимущественным вводом-выводом*), а другой, наоборот, использует, его в очень малой степени (*задача с преимущественным счётом*).

Дисциплины обслуживания

Если выполнять несколько процессов, имеющих сходные характеристики, близкие к одному из двух вышеназванных пределов, может возникнуть ситуация, когда каналы ввода-вывода окажутся перегруженными, а процессор будет простаивать или наоборот. Для достижения максимальной производительности система производит назначение приоритетов таким образом, что постоянно поддерживается состав смеси процессов, гарантирующий сбалансированную загрузку системы.

Дисциплины обслуживания

Подобно этому, система может увеличивать приоритеты процессов, монопольно владеющих критическими ресурсами (например, накопителем на магнитной ленте), благодаря чему ресурсы освобождаются для использования другими процессами быстрее.

Общий алгоритм планировщика процессов

алгоритм `schedule_process`

входная информация: отсутствует

выходная информация: отсутствует

```
{  
    выполнять пока (для запуска не будет выбран один из про-  
        цессов)  
    {  
        для (каждого процесса в очереди готовых к выполнению)  
            выбрать процесс с наивысшим приоритетом из загру-  
                женных в память;  
        если (ни один из процессов не может быть избран для  
            выполнения)  
            приостановить машину;  
            /* машина выходит из состояния простоя по преры-  
            /* ванию  
            */  
    }  
    удалить выбранный процесс из очереди готовых к выполне-  
        нию;  
    переключиться на контекст выбранного процесса, возобно-  
        вить его выполнение;  
}
```