**Алгоритмы и планирование**

**Уровни планирования**

Процедура сохранения/восстановления контекста называется переключением контекста. Время, затраченное на переключение контекста, не используется – снижается производительность.

Планирование процессов.

Уровни планирования.

Планирование процессов возникает в мультипрограммных вычислительных системах, когда в состоянии готовности могут находиться сразу несколько процессов.

Процедура планирования используется для выбора одного из этих процессов, который получит в своё распоряжение центральный процессор.

Планирование заданий используется в качестве долгосрочного планирования процессов.

Оно отвечает за порождение новых процессов в системе и количество одновременно рабочих процессов.

Если степень мультипрограммирования системы поддерживается постоянной, то новые процессы могут появляться только после завершения ранее загруженных.

Поэтому долгосрочное планирование осуществляется достаточно редко, между появлением новых процессов могут проходить минуты и даже десятки минут.

Отсюда и название этого уровня планирования – долгосрочное.

В некоторых ОС долгосрочное планирование сведено к минимуму или отсутствует вовсе.

Так, например, во многих интерактивных системах с разделением времени рождение новых процессов происходит сразу после соответствующего запроса. А поддержание мультипрограммирования осуществляется за счёт ограниченного количества пользователей и человеческой психологией.

Планирование использования процессора применяется в качестве краткосрочного планирования.

Оно проводится при обращении исполняющегося процесса к устройствам ввода вывода или просто по заверении определённого интервала времени.

Поэтому краткосрочное планирование осуществляется не реже одного раза в 100 миллисекунд.

Выбор нового процесса для исполнения оказывает влияние на небольшой период времени.

В некоторых ВТ для повышения производительности можно временно выгрузить какой-либо частично выполнившийся процесс из ОС во внешнюю память, а позже вернуть его назад.

Среднесрочное планирование решает, что и когда выгрузить.

**Критерии планирование и требования к алгоритмам**

Для каждого уровня планирования можно разработать множество алгоритмов. Выбор конкретного алгоритма определяется классом задач, решаемых ВС, и целями, которые мы хотим достичь, используя планирование.

К числу таких целей можно отнести следующие:

1. Справедливость  
   Гарантировать каждому заданию или процессу определённую часть времени использования процессора в компьютерной системе.
2. Эффективность  
   Постараться занять процессор на 100% его рабочего времени, не позволяя ему простаивать в ожидании процессов, готовых к исполнению.
3. Сокращение полного времени выполнения  
   Обеспечить минимальное время между стартом работы процесса и его завершения.
4. Сокращение времени ожидания  
   Сократить время, которое проводят процессы в состоянии готовность и задания в очереди для загрузки.
5. Сокращение времени отклика  
   Минимизировать время, которое потребуется процессу в интерактивных системах для ответа на запрос пользователя.

**Свойства алгоритмов**

Алгоритмы должны обладать следующими свойствами:

* Быть предсказуемыми.  
  Одно и то же задание должно выполняться приблизительно за одно и то же время.
* Были связаны с минимальными накладными расходами.  
  Продолжительность работы алгоритма должна быть меньше работы процесса.
* Равномерно загружали ресурсы ВС.  
  Отдавать предпочтение тем процессам, которые будут занимать малоиспользуемые ресурсы.
* Обладали масштабируемостью.  
  Не сразу теряли работоспособность при увеличении нагрузки.

Критерии противоречат друг другу, поэтому идеального решения не существует, приходится выбирать.

**Параметры планирования**

Все параметры планирования можно разбить на две большие группы: статические и динамические.

Статические параметры не изменяются в ходе функционирования ВТ, динамические же, напротив, подвержены постоянным изменениям.

К статическим параметрам ВС можно отнести предельное значение её ресурсов (размер оперативной памяти, количество подключённых устройств ввода-вывода и т.д.)

Динамические параметры системы описывают количество свободных ресурсов на данный момент.

К статическим параметрам процессов относятся характеристики, как правило присущие заданиям уже на этапе загрузки.

Алгоритмы долгосрочного планирования используют в своей работе статические и динамические параметры ВС и статические параметры процессов.

Алгоритмы краткосрочного и среднесрочного планирования дополнительно учитывают и динамические характеристики процессов.

Для среднесрочного планирования в качестве таких характеристик может использоваться следующая информация:

* Сколько времени прошло с момента выгрузки процесса на диск или его загрузки в оперативную память
* Сколько оперативной памяти занимает процесс
* Сколько процессорного времени уже предоставлено процессу

Деятельность любого процесса можно представить как последовательность циклов использования процессора и ожидания завершения операции ввода-вывода.

Важные динамические параметры процессора:

* Промежуток времени непрерывного использования процессора носит название CPU burst
* Промежуток времени непрерывного ожидания ввода-вывода – I/O burst

**Вытесняющее и не вытесняющее планирование**

Процесс планирования осуществляется частью ОС, называемой планировщиком.

Планировщик может принимать решения о выборе для исполнения нового процесса из числа находящихся в состоянии готовность в 4-х случаях:

Процесс переводится из состояния исполнения в состояние закончил исполнение.

Процесс переводится из состояния исполнение в состояние ожидание

Процесс переводится из состояние исполнение в состояние готовность

Процесс переводится из состояния ожидание в состояние готовность

В первых двух случаях процесс, находящийся в состоянии исполнения не может исполняться дальше. ОС вынуждена осуществлять планирование.

В случаях 3 и 4 планирование может как проводиться, так и не проводиться (не обязательно запускать планировщик)

Если в ОС планирование выполняется только в вынужденных ситуациях, то это не вытесняющее планирование.

Если планирование осуществляется во всех 4 ситуациях, то это вытесняющее планирование.

Вытесняющее планирование используется в интерактивных системах и в системах с выделением времени.

**Алгоритмы планирования**

Простейший алгоритм – First-Come, First-Served (FCFS)

Для описания работы этого алгоритма можно представить, что все алгоритмы образуют очередь. На исполнение берётся процесс из начала очереди.

Процесс, получивший в своё распоряжение процессор, занимает его на всё, нужное ему время (до истечения текущего CPU burst)

**Алгоритм планирования Round Robin**

Это алгоритм FCFS, реализованный в режиме вытесняющего планирования.

Реализуется такой алгоритм так же, как и предыдущий, с помощью организации процессов, находящихся в состоянии готовность, в очередь FIFO.

Планировщик выбирает для очередного исполнения процесс, расположенный в начале очереди, и устанавливает таймер для генерации прерывания по истечении определённого кванта времени.

При выполнении процесса возможны два варианта:

1. Время непрерывного использования процессора, необходимое процессу (остаток текущего CPU burst), меньше или равно продолжительности кванта времени. Тогда процесс по своей воле освобождает процессор до истечения кванта времени, на исполнение поступает новый процесс из начала очереди, и таймер начинает отсчёт кванта заново.
2. Продолжительность остатка текущего CPU burst процесса больше, чем квант времени. Тогда по истечении этого кванта процесс прерывается таймером и помещается в конец очереди процессов, готовых к исполнению, а процессор выделяется для использования процессу, находящемуся в её начале.

При очень больших величинах кванта времени, когда каждый процесс успевает завершить свой CPU burst до возникновения прерывания по времени, алгоритм PR вырождается в алгоритм FCFS.

При очень малых величинах создаётся иллюзия того, что каждый из n процессов работает на собственном виртуальном процессоре с производительностью ~1/n от производительности реального процессора. Т.е. накладные расходы резко снижают производительность всей системы.

**Алгоритмы планирования Shortest-Job-First (SJF)**

SJF-алгоритм краткосрочного планирования может быть как вытесняющим, так и невытесняющим.

При невытесняющем SJF - планировании процессор предоставляется избранному процессу на все необходимое ему время, независимо от событий, происходящих в вычислительной системе.

При вытесняющем SJF - планировании учитывается появление новых процессов в очереди готовых к исполнению (из числа вновь родившихся или разблокированных) во время работы выбранного процесса.

Если CPU burst нового процесса меньше, чем остаток CPU burst у исполняющегося, то исполняющийся процесс вытесняется новым.

Невозможно точно определить время выполнения (точное CPU burst) для исполняющихся процессов.

При краткосрочном планировании можно сделать только прогноз длительности очередного времени исполнения, исходя из предыстории его работы.