

Операционная система — это комплекс программ, обеспечивающий управление аппаратными средствами компьютера, организующий работу с файлами и выполнение прикладных программ, осуществляющий ввод и вывод данных. В одну из задач ОС входит управление процессами.

В привилегированном режиме, ЦПУ имеет доступ ко всему набору инструкций, а в непривилегированном режиме некоторые инструкции отключаются

Процесс — это экземпляр программы, выполняющейся на компьютере

Многозадачный режим — свойство операционной системы выполнения обеспечивать возможность параллельной обработки нескольких задач.

Единый процесс, который получает определенные ресурсы, может быть разбит на несколько параллельно работающих **потоков**, которые совместно выполняют работу процесса.

Компьютерная шина — соединение, служащее для передачи данных между функциональными блоками компьютера.

Информация об характеристиках процесса хранится в структуре данных, которая обычно называется **управляющим блоком процесса** (Идентификатор, Состояние, Приоритет, Программный счетчик, Указатели памяти, Данные контекста, Информация о состоянии ввода-вывода, Учетная информация).

**4 способа создания процессов** ( Новое пакетное задание, вход нового пользователя, сама операционна создает нужный ей процесс, другим процессом)



Поток — это часть процесса, который выполняет определенные инструкции процесса. В каждом процессе создается по крайней мере один поток. Быстро создается. Есть два основных места реализации набора потоков: в пользовательском пространстве и в ядре. Первый способ — это поместить весь набор потоков в пользовательском пространстве. И об этом наборе ядру ничего не известно. Что касается ядра, оно управляет обычными, однопотоковыми процессами.

**Многозадачность:** управление множеством процессов в однопроцессорной системе.

**Многопроцессорность:** управление множеством процессов в многопроцессорной системе.

**Распределенные вычисления:** управление множеством процессов, в системе с множеством компьютеров. Основным примером это кластер (группа компьютеров)

**Атомарная операция** — последовательность из одной или нескольких инструкций, представляющих неделимые (никакой процесс не может увидеть промежуточное состояние или прервать)

**Критическая секция** — участок кода в рамках процесса, который требует доступа к общим ресурсам и который не должен выполняться в то время, когда в этом участке кода находится другой процесс.

**Взаимоблокировка**(клинч, deadlock) — ситуация, когда два и более процессов не в состоянии работать, поскольку каждый из процессов ожидает выполнения некоторого действия другим процессом.

**Взаимоисключение** — требование, чтобы, когда один процесс находится в критическом участке, который получает доступ к общим ресурсам, никакой другой процесс не мог находиться в критическом участке, который обращается к любому из этих общих ресурсов.

**Динамическая взаимоблокировка** — ситуация, когда два и более процессов постоянно изменяют свои состояния в ответ на изменения в других процессах без выполнения полезной работы.

**Состояние гонки** возникает, когда несколько процессов или потоков читают и записывают элементы данных так, что конечный результат зависит от порядка выполнения инструкций в нескольких процессах.

**Голодание** – ситуация, когда запуск процесса пропускается планировщиком бесконечное количество раз; хотя процесс готов работать, он никогда не выбирается.

Алгоритм Деккера: у нас должна быть возможность следить за состоянием обоих процессов, что обеспечивается массивом `flag`. Мы должны навязать определенный порядок действий двум процессам, чтобы избежать проблемы "взаимной вежливости", используем переменную `turn`, которая указывает, какой из процессов имеет право на вход в критический участок.

Алгоритм Петерсона: глобальная переменная `flag` указывает положение каждого процесса по отношению к взаимоисключению, а глобальная переменная `turn` разрешает конфликты одновременности.

Условия взаимоблокировки — взаимоисключение, удержание и ожидание, отсутствие перераспределения, циклическое ожидание.

Семафор — целочисленное значение для передачи сигналов между процессами. 3 операции на нем это инициализация, инкремент и декремент.

Мьютекс — примитив синхронизации, обеспечивающий взаимоисключение критического участка кода. Процесс, блокирующий мьютекс должен его же и разблокировать в отличие от бинарного семафора.

Условная переменная - тип данных, используемый для блокировки процесса или потока до тех пор, пока не станет истинным некоторое условие.

Переменная флаг — это, как правило, переменная логического типа, значение которой сигнализирует о состоянии вычислительного процесса.

Спин-блокировки — механизм взаимоисключения, в котором процесс выполняется в бесконечном цикле, ожидая, когда значение блокирующей переменной укажет доступность критического участка

Монитор — высокоуровневым метод синхронизации, неуступающий по функционалу семафорам, но более эффективный.