POSIX (Portable Operating System Interface) - это набор стандартов, определенных IEEE, который определяет интерфейс между операционной системой и прикладным программным обеспечением. Он обеспечивает переносимость программного обеспечения между различными операционными системами, которые поддерживают стандарт POSIX.

Системный вызов - это механизм, предоставляемый операционной системой, который позволяет прикладным программам взаимодействовать с ядром операционной системы. Системные вызовы предоставляют доступ к функциям операционной системы, таким как создание процессов, управление файлами, сетевое взаимодействие и т. д.

Аппаратное прерывание - это сигнал, отправляемый устройством в процессор для прерывания текущей работы и обработки события, связанного с устройством. Программное прерывание - это сигнал, отправляемый программой или операционной системой для прерывания текущей работы и выполнения определенного кода.

Процесс - это экземпляр выполняющейся программы. Он содержит код программы, данные, регистры процессора, стек вызовов и другую информацию, необходимую для выполнения программы.

Контекст процесса - это состояние процесса в определенный момент времени. Он включает в себя значения регистров процессора, указатель на текущую инструкцию, таблицы страниц памяти, открытые файлы и другую информацию, необходимую для восстановления выполнения процесса после прерывания.

Родительский процесс - это процесс, который создает другой процесс, называемый дочерним процессом.

Дочерний процесс - это процесс, созданный родительским процессом. Дочерний процесс наследует некоторые характеристики от родительского процесса, такие как идентификатор родительского процесса и текущая рабочая директория.

Процесс инициализации ОС - это первый процесс, который запускается при загрузке операционной системы. Он имеет идентификатор процесса (PID) 1 и является родительским процессом для всех остальных процессов в системе. Процесс инициализации выполняет инициализацию системы, запускает необходимые службы и обеспечивает стабильную работу операционной системы.

Области памяти процесса включают: Код: область памяти, содержащая исполняемый код программы. Данные: область памяти, содержащая глобальные переменные и статические данные программы. Стек: область памяти, используемая для хранения локальных переменных и вызовов функций. Куча: область памяти, используемая для динамического выделения памяти во время выполнения программы.

Системные процессы - это процессы, которые выполняются в контексте операционной системы и обеспечивают ее функционирование. Они имеют повышенные привилегии и могут выполнять задачи, связанные с управлением ресурсами системы.

Пользовательские процессы - это процессы, созданные пользователями для выполнения своих задач. Они выполняются в контексте пользователя и имеют ограниченные привилегии.

Windows-сервисы и Linux-демоны - это фоновые процессы, которые выполняются без прямого взаимодействия с пользователем. Они обеспечивают выполнение определенных задач, таких как обслуживание сети, планирование задач, мониторинг и т. д. Windows-сервисы работают в операционной системе Windows, а Linux-демоны (также известные как службы) работают в операционной системе Linux.

В Windows для создания дочернего процесса можно использовать системные вызовы CreateProcess или CreateThread. CreateProcess создает новый процесс, а CreateThread создает новый поток внутри текущего процесса. Разница между ними заключается в том, что процесс имеет собственное адресное пространство памяти, а поток разделяет адресное пространство с другими потоками внутри процесса. В Linux для создания дочернего процесса можно использовать системный вызов fork. Fork создает точную копию текущего процесса, включая его память, файловые дескрипторы и другие атрибуты. После создания дочернего процесса, он может использовать системный вызов exec для загрузки новой программы в свое адресное пространство памяти. Разница между fork и exec заключается в том, что fork создает новый процесс, а exec загружает новую программу в адресное пространство процесса.