**27. Что такое процесс?**

Процесс – управляющий объект, который обеспечивает изоляцию адресных пространств и представляет работающий экземпляр программы

**28. Что такое контекст процесса?**

Каждому процессу соответствует контекст, в котором он выполняется. Этот контекст включает содержимое пользовательского адресного пространства - пользовательский контекст (т. е. содержимое сегментов программного кода, данных, стека, разделяемых сегментов и сегментов файлов, отображаемых в виртуальную память) , содержимое аппаратных регистров - регистровый контекст (таких, как регистр счетчика команд, регистр состояния процессора, регистр указателя стека и регистров общего назначения) , а также структуры данных ядра (контекст системного уровня) , связанные с этим процессом.

**29. Что такое родительский и дочерний процесс?**

**Родительский процесс:** Все процессы создаются, когда процесс выполняет системный вызов fork(), за исключением процесса запуска. Процесс, который выполняет[системный вызов fork(),](https://www.geeksforgeeks.org/fork-system-call/) является родительским процессом. Родительский процесс - это процесс, который создает дочерний процесс с помощью системного вызова fork(). Родительский процесс может иметь несколько дочерних процессов, но дочерний процесс - только один родительский процесс.

**Дочерний процесс:**Дочерний процесс создается родительским процессом в операционной системе с помощью системного вызова fork(). Дочерний процесс также может быть известен как подпроцесс или подзадача.

**30. Что такое процесс инициализации OS?**

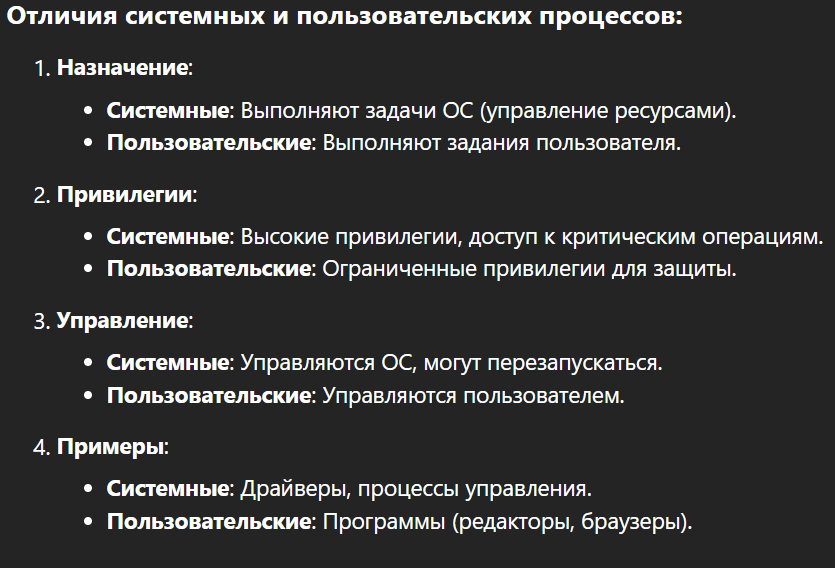
Процесс инициализации операционной системы (OS) — это начальный этап загрузки системы, который выполняется после включения компьютера. Этот процесс включает в себя несколько ключевых шагов: POST (Power-On Self-Test) На этом этапе выполняется проверка оборудования, чтобы убедиться, что все компоненты, такие как оперативная память, процессор и устройства хранения, функционируют корректно.: Загрузка загрузчика: Загрузка ядра ОС: Инициализация системы: Запуск пользовательских процессов: Готовность к работе:

**31. Перечислите области памяти процесса и поясните их назначение.**

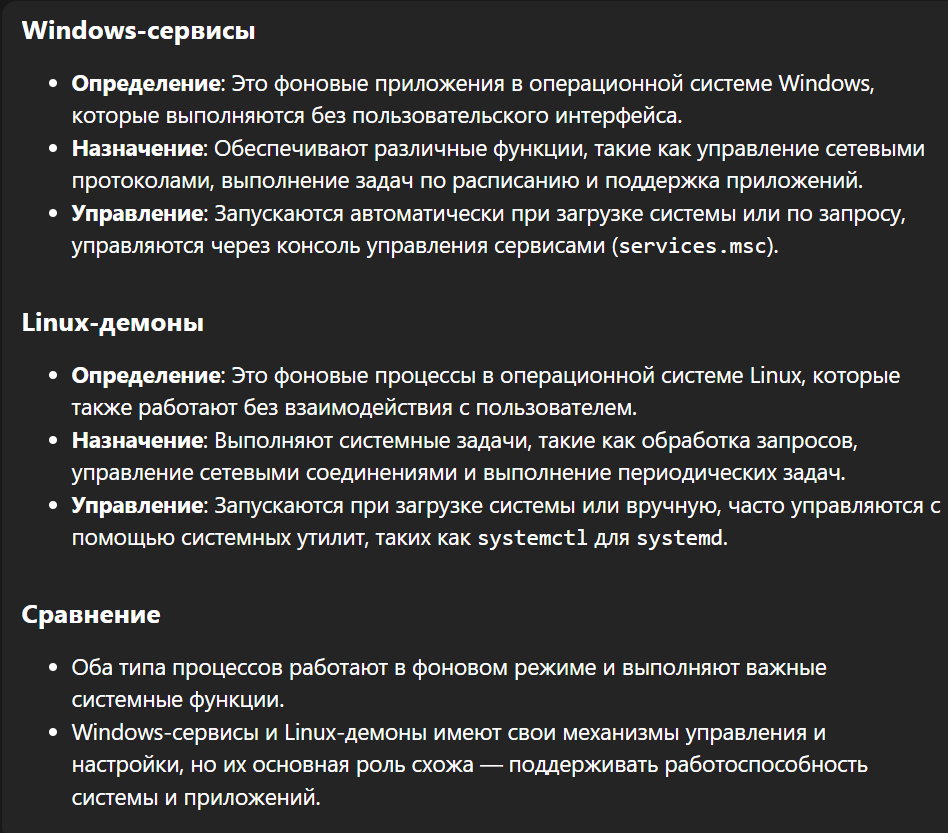
В процессе выполнения приложения память делится на несколько областей, каждая из которых имеет свое назначение. Основные области памяти процесса:

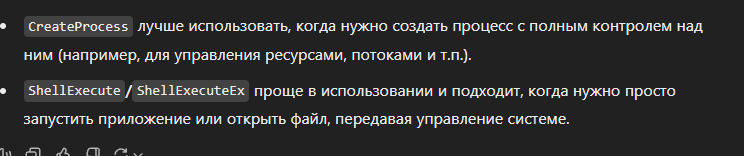
1. Текстовая область (Text Segment):
   * Назначение: Содержит исполняемый код программы.
2. Область данных (Data Segment):
   * Назначение: Содержит глобальные и статические переменные, которые инициализированы и доступны на протяжении всего времени выполнения программы.
3. Стек (Stack):
   * Назначение: Используется для хранения локальных переменных, параметров функции и информации о возврате.
4. Куча (Heap):
   * Назначение: Используется для динамического распределения памяти во время выполнения программы.
5. Область памяти, выделяемая для окружения (Environment Segment):
   * Назначение: Содержит информацию о среде выполнения, такую как переменные окружения.

**32. Чем отличаются системные процессы от пользовательских?**

****

**33. Что такое Windows-сервисы, Linux-демоны?**

****

**34. С помощью каких системных вызовов можно создать дочерний процесс в Windows? Поясните разницу. **

**35. С помощью каких системных вызовов можно создать дочерний процесс в Linux? Поясните разницу.**

Функция fork – предназначена для создания нового дочернего процесса, который будет являться полной (насколько это возможно) копией родительского процесса

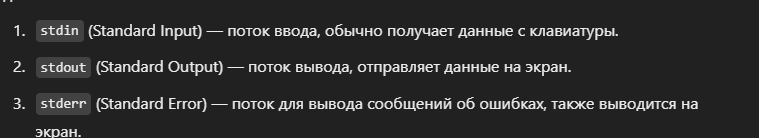
Функция exec – предназначена для замены образа исполняемого файла в рамках существующего процесса

Важно! Никаких новых процессов при вызове exec не создаётся. Загружается новая программа!

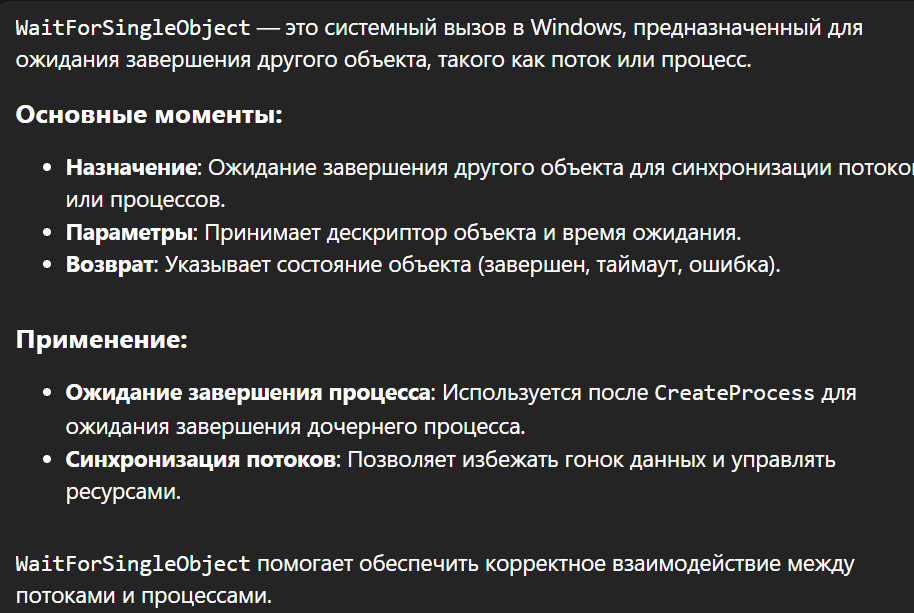
По сути сбрасывается контекст процесса и загружается новый

* fork(): Создает новый процесс, который является копией родительского процесса. После вызова fork() оба процесса продолжают выполняться, и дочерний процесс может выполнять тот же код.
* exec(): Замещает текущий процесс новым. То есть, если дочерний процесс вызовет exec(), он перестанет существовать как копия родительского и начнет выполнение новой программы.

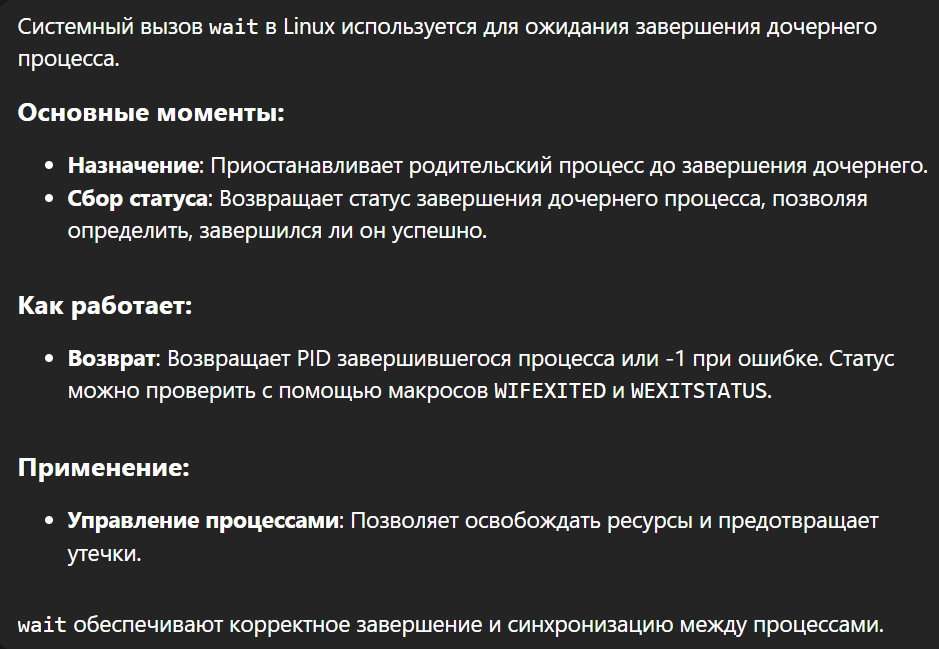
**36. Какие потоки данных доступны любому процессу автоматически?**

****

**37. Поясните данных назначение доступны любому системного WaitForSingleObject в Windows-приложении.**

****

**38. Поясните назначение системного вызова wait в Linux приложении.**

****

**39. Дайте развернутое определение процесса OS.**

В операционных системах (OS) **процесс** — это базовая единица выполнения программы, представляющая собой экземпляр программы, который выполняется на компьютере. Процесс включает в себя не только сам код программы, но и все ресурсы, необходимые для его выполнения.