1. IA-32 (также известный как x86) может работать в нескольких режимах:

* Реальный режим: Это режим совместимости, который поддерживает программы, написанные для предшествующих процессоров семейства x86. В этом режиме процессор использует сегментную адресацию и работает в 16-битном режиме.
* Защищенный режим: Это режим современных операционных систем, таких как Windows и Linux. В защищенном режиме доступны расширенные функции, такие как виртуальная память, механизмы защиты и многозадачность. Процессор работает в 32-битном или 64-битном режиме, в зависимости от конкретной версии процессора и операционной системы.
* Режим системного управления: Этот режим предназначен для привилегированного системного программного обеспечения, такого как операционные системы. Он обеспечивает полный контроль над аппаратными средствами и привилегированными инструкциями.

2. В сегментной адресации физический адрес формируется путем комбинации базового адреса сегмента и смещения. Сегментный регистр содержит базовый адрес сегмента, а смещение указывает на конкретный адрес внутри сегмента. Физический адрес получается путем сложения базового адреса сегмента и смещения.

3. В страничной адресации физический адрес формируется в два этапа. Сначала виртуальный адрес разбивается на две части: номер страницы и смещение внутри страницы. Затем используется таблица страниц (Page Table), которая содержит соответствия между виртуальными и физическими адресами. Таблица страниц содержит записи, в каждой из которых указан физический адрес начала страницы. Физический адрес получается путем замены номера страницы на соответствующий физический адрес из таблицы страниц и добавления смещения внутри страницы.

4. Многозадачность - это способность операционной системы выполнять несколько задач (программ) одновременно. Она позволяет разным программам работать параллельно, даже если физический процессор доступен только одному ядру.

Многозадачность поддерживается средствами операционной системы, которая обеспечивает планирование и выделение рес

урсов между различными задачами. ОС использует планировщик задач для определения, какая задача будет выполняться в данный момент времени и на какое время. Она также использует механизмы контекстного переключения для сохранения состояния одной задачи и восстановления состояния другой задачи при переключении между ними.

5. Для защиты сегментов кода, стека и данных применяются правила на основе привилегий. Привилегии определяют уровень доступа, который имеет программа к определенным сегментам памяти. Например, сегмент кода может быть отмечен как доступный только для чтения и выполнения, чтобы предотвратить запись данных в этот сегмент. Стек может быть помечен как доступный только для записи и выполнения, чтобы предотвратить чтение данных из стека другим программам. Правила на основе привилегий позволяют контролировать доступ к различным сегментам памяти и обеспечивать безопасность и защиту данных от несанкционированного доступа.