**Задача конструктора в объектном типе. Механизм его работы.**

1)Основная задача конструктора – назначение ТВМ (таблиц виртуальных методов) динамическому объекту. Конструктор делает это неявно для всех объектов содержащих ВМ. Нужно понимать, что сам конструктор не составляет ТВМ. Они делаются на этапе компиляции (по одной для каждого типа). Конструктор только заносит в экземпляр объекта адрес ТВМ.

2)Вторая задача – инициализация полей, однако конструктор может этого и не делать. Т.е. пустой конструктор – тоже конструктор.

-----------------

Конструктор может быть параметрический (содержать аргументы):

Constructor Init(a1, a2 : integer);

Или простой (без параметров):

Constructor init\_t;

Тела конструкторов описываются вместе с процедурами для типа. В одном типе может быть несколько конструкторов, но они должны иметь разные имена.

При построении сложной иерархии объектов с наследованием имеет смысл вызывать конструктор базового класса в конструкторе предка (причем делать это с самого начала, до выполнения других операций!).

Constructor t1.Init(a, b : integer);

Begin

Inherited t0.Init(a); {вызов конструктора предка из потомка с передачей параметра}

//другие действия

End;

При использовании Виртуальных методов:   
2)класс, реализующий ВМ, обязательно должен иметь конструктор (т.к. при вызове конструктора неявно происходит связывание ТВМ с экземпляром класса). Важная деталь: конструктор необязательно должен быть у всех классов в иерархии. В принципе он может быть только у базового., но вызов конструктора всё равно обязателен.

3)очевидно, что для использования объектов с виртуальными методами необходимо вызывать конструктор. Причем и для статических, и для динамических объектов. (существует расширенная форма записи new : New(obj, init);, сразу включающая вызов конструктора).