1. Структурное программирование (процедурное, функциональное): «код изменяет данные».   
   Задача 🡪 Алгоритм 🡪 Действия 🡪 Функции/Процедуры.
2. Цель создания ООП методологии: сложность проектов, сильные межкомпонентные связи. Макаронный код. ООП 🡪 удобная логическая структура программы.
3. ООП программирование: «данные контролируют доступ к коду».   
   В основе подхода – модель данных: аналогия с реальным миром – берем объект, среди бесконечного множества свойств и характеристик те, которые представляют интерес в контексте решения поставленной задачи. Пример: объект – автомобиль. Свойства в контекстах задач построения автосервиса и тюнинг-ателье будут разными.   
   Таким образом: «в решении описывается модель данных, а затем код, их обрабатывающий».
4. **Принцип проектирования** – описание модели данных. Выбор свойств и характеристик объектов. Далее написание кода.
5. **Принцип проектирования** – код должен быть закрыт для изменения, но открыт для расширения. Программисты должны изолировать хорошо работающий код, выделить изменяющиеся части. Программисты должны быть экстрасенсами.
6. Из этого принципа вытекает необходимость обеспечения минимальной связанности, принцип инверсии зависимостей, паттерн «фабричный метод».