В основе того или иного языка программирования лежит некая руководящая идея, вызванная потребностями или, чаще всего, кризисом в области программирования и создания программного обеспечения, которая оказывает существенное влияние на стиль программирования и помогает преодолеть указанный кризис. Рассмотрим вкратце историю появления и развития основных стилей программирования и процедурных алгоритмических языков.

Виды программирования:

\* Машинно-ориентированное программирование.
\*Процедурное программирование.
\*Структурное программирование.
\* Объектно-ориентированное программирование.
\*Обобщенные  технологии разработки приложений.
\*Модульное программирование.

***Модульное программирование.*** Здесь основная идея заключалась в том, чтобы "спрятать" данные и процедуры внутри независимых программных единиц - модулей*.* Эту идею впервые реализовал Н. Вирт в алгоритмическом языке Modula (1975-1979 годы), а затем "подхватили" и остальные, распространенные в то время языки программирования. Например, известные системы программирования Turbo Pascal и Turbo С.
    «На  этом можно было бы остановиться, т.к. я дошел темы моей работы, но я не могу обойти без внимания дальнейшее развитие технологий программирования, поэтому продолжу хронологию развития программирования.»- говорил Н. Вирт.

Цель  модульного программирования:

     Приступая к разработке каждой программы, следует иметь в виду, что она, как правило, является большой системой, поэтому надо принять меры для ее упрощения. Для этого такую программу разрабатывают по частям, которые называются программными модулями. А сам такой метод разработки программ называют модульным программированием. Программный модуль - это любой фрагмент описания процесса, оформляемый как самостоятельный программный продукт, пригодный для использования в описаниях процесса.  Модульное программирование является воплощением в процессе разработки программ обоих общих методов борьбы со сложностью: и обеспечение независимости компонент системы и использование иерархических структур. Для воплощения первого метода формулируются определенные требования, которым должен удовлетворять программный модуль, т.е. выявляются основные характеристики «хорошего» программного модуля. Для воплощения второго метода используют древовидные модульные структуры программ (включая деревья со сросшимися ветвями).