**Основные характеристики программного модуля**

С момента своего возникновения программирование боролось за возможность решать все более сложные задачи, создавать все более сложные программные системы и делать это как можно быстрее. В процессе разработки программ модульное программирование явилось воплощением общих методов борьбы со сложностью, обеспечения независимости компонент системы и использования иерархических структур.

Метод разработки программ по частям называют модульным программированием.

Программный модуль - это любой фрагмент описания процесса, оформляемый как самостоятельный программный продукт, пригодный для использования в других описаниях процесса.

Критерии приемлемости выделенного модуля (по Хольту):

-хороший модуль снаружи проще, чем внутри;

-хороший модуль проще использовать, чем построить.

Критерии приемлемости выделенного модуля (по Майерсу):

-размер модуля,

-прочность модуля,

-сцепление с другими модулями,

-рутинность модуля (независимость от предыстории обращений к нему).

**Размер модуля** измеряется числом содержащихся в нем операторов или строк. Обычно рекомендуются программные модули размером от нескольких десятков до нескольких сотен операторов.

**Прочность модуля** - это мера его внутренних связей. Чем выше прочность модуля, тем больше связей он может спрятать от внешней по отношению к нему части программы и, следовательно, тем больший вклад в упрощение программы он может внести. Для оценки степени прочности модуля Майерс предлагает упорядоченный по степени прочности набор из семи классов модулей. Для использования рекомендуются только два высших по прочности класса модулей. Функционально прочный модуль - это модуль, реализующий одну какую-либо определенную функцию. При реализации этой функции такой модуль может использовать и другие модули. Информационно прочный модуль - это модуль, выполняющий (реализующий) несколько операций (функций) над одной и той же структурой данных (информационным объектом), которая считается неизвестной вне этого модуля. Для каждой из этих операций в таком модуле имеется свой вход со своей формой обращения к нему.

**Сцепление модуля** - это мера его зависимости по данным от других модулей. Характеризуется способом передачи данных. Единственным видом сцепления модулей, который рекомендуется для использования современной технологией программирования, является параметрическое сцепление (сцепление по данным по Майерсу) - данные передаются модулю либо при обращении к нему как значения его параметров, либо как результат его обращения к другому модулю для вычисления некоторой функции. Такой вид сцепления модулей реализуется на языках программирования при использовании обращений к процедурам (функциям).

**Рутинность модуля** - это его независимость от предыстории обращений к нему. Модуль будем называть рутинным, если результат (эффект) обращения к нему зависит только от значений его параметров (и не зависит от предыстории обращений к нему).

Модуль будем называть зависящим от предыстории, если результат (эффект) обращения к нему зависит от внутреннего состояния этого модуля, изменяемого в результате предыдущих обращений к нему.

Приемлема следующая рекомендация:

всегда следует использовать рутинный модуль, если это не приводит к плохим (не рекомендуемым) сцеплениям модулей;

зависящие от предыстории модули следует использовать только в случае, когда это необходимо для обеспечения параметрического сцепления;

в спецификации зависящего от предыстории модуля должна быть четко сформулирована эта зависимость таким образом, чтобы было возможно прогнозировать поведение (эффект выполнения) данного модуля при разных последующих обращениях к нему.