



FACULTY OF ENGINEERING & TECHNOLOGY

Brajesh Mishra

Assistant Professor

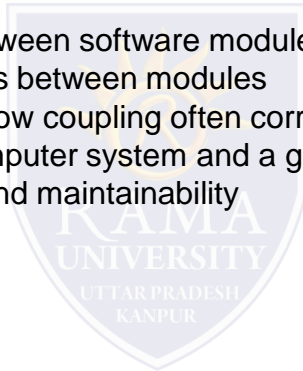
Department of Computer Science & Engineering

Topics Covered

Coupling
Types of coupling



- **coupling** is the degree of interdependence between software modules; a measure of how closely connected two routines or modules are; the strength of the relationships between modules
- Coupling is usually contrasted with cohesion. Low coupling often correlates with high cohesion, and vice versa. Low coupling is often a sign of a well-structured computer system and a good design, and when combined with high cohesion, supports the general goals of high readability and maintainability



Types of coupling

- Coupling can be "low" (also "loose" and "weak") or "high" (also "tight" and "strong"). Some types of coupling, in order of highest to lowest coupling, are as follows:

- **Procedural programming**

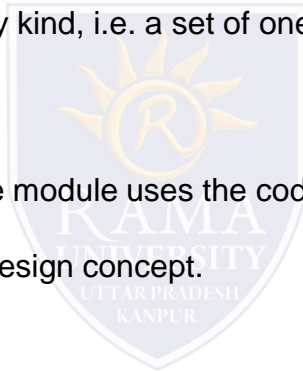
- A module here refers to a subroutine of any kind, i.e. a set of one or more statements having a name and preferably its own set of variable names.

- **Content coupling (high)**

- Content coupling is said to occur when one module uses the code of another module, for instance a branch.
- This violates information hiding – a basic design concept.

- **Common coupling**

- Common coupling is said to occur when several modules have access to the same global data.
- It can lead to uncontrolled error propagation and unforeseen side-effects when changes are made.



Types of coupling

•External coupling

•External coupling occurs when two modules share an externally imposed data format, communication protocol, or device interface. This is basically related to the communication to external tools and devices.

•Control coupling

•Control coupling is one module controlling the flow of another, by passing it information on what to do (e.g., passing a what-to-do flag). Stamp coupling (data-structured coupling) Stamp coupling occurs when modules share a composite data structure and use only parts of it, possibly different parts (e.g., passing a whole record to a function that needs only one field of it). In this situation, a modification in a field that a module does not need may lead to changing the way the module reads the record.

•Data coupling

•Data coupling occurs when modules share data through, for example, parameters. Each datum is an elementary piece, and these are the only data shared (e.g., passing an integer to a function that computes a square root).

