Unit: III
Lecture: 7
Data Structure and Information Flow Metrics/ Measure

There are some data structure metrics to compute the effort and tie required to complete the project. These metrics are:

- The Amount of Data
- The Usage of data within a module
- Program weakness
- The sharing of Data among modules

The Amount of Data: To measure the amount of Data, there are further any different metrics and these are:

- Number of variable(VARS): In this metric, the number of variables used - in the program is counted.
- Number of Operands: In this metric, the number of operands used in the program is counted.

  Number of operands= VARS + Constants + Labels

- Total number of occurrence of the variable (N2): In this metric, the total number of occurrence of the variables are computed.

The usage of data within a module: To measure this metric, the average number of live variables are computed. A variable is live from its first to its last reference within the procedure.

$$\text{Average no of Live variables (LV)} = \frac{\text{Sum of count live variables}}{\text{Sum of count of executable statements}}$$

Program Weakness: Program weakness depends on its Modules weakness. If modules are weak (less Cohesive), then it increases the effort and time metrics required to complete the project.

$$\text{Average life of variables } (\gamma) = \frac{\text{Sum of count live variables}}{\text{Sum of count of executable statements}}$$

Module Weakness (WM) = $\overline{LV} * \gamma$

A program is normally a combination of various modules; hence, program weakness can be a useful measure and is defined as:

$$\mathbf{WP} = \frac{(\sum_{i=1}^{m} WM_i)}{m}$$

where, WMi : Weakness of the ith module

WP: weakness of the program

m : No of modules in the program

Sharing of data among Module: As the data sharing between the modules increases (higher coupling), number of parameter passing between modules also increases. As a result, more effort and time are required to complete the project. So sharing data among module is an important metrics to calculate effort and time.

Information Flow Metrics (IF-Metrics):

The IF- metrics is based on a premise that all systems (software) consists of components only that work together to influence the complexity of a system. System theory tell us that components that are highly coupled and that lack cohesion tend to be less reliable and less maintainable than those that are loosely coupled and are cohesive.

IF-metrics model the degree of cohesion and coupling for a particular system component.

This metrics is based on the measurement of the information flow among system modules. It is sensitive to the complexity due to interconnection among system component.

IF-metrics are applied to the components of a system design.

e.g. for a component/ module –A; we defined following measures:

- FAN-IN: a count of number of other components that can call or pass control to component A.
- FAN-OUT: is the number of components that are called by component A.

IF-metrics of component A is:

IF(A) =[(FAN-IN(A) * FAN-OUT(A)]