**Software Project Estimation**:-

Software project estimation is necessary to achieve reliable cost and effort prediction. A project estimation method usually involves the following steps:

identify the project scope

* + perform project decomposition
  + compute the empirical metrics KLOC and/or FP
  + apply an empirical model to obtain estimates of software cost, effort and duration.
  + use for comparison automated estimation tools (optional)

The contemporary software projects are usually extremely large, and require decomposition and re-characterization as a set of smaller, more manageable sub-problems.

The decomposition techniques take the "divide and conquer" approach to software project estimation. Software estimation activities can be performed in a stepwise fashion when the project is decomposed in major functions and related tasks.

The expected values for KLOC and FP can be computed as follows:

E = ( a + 4 m + b ) / 6

where: a is the optimistic value m is the most likely value b is the pessimistic value.

**2. Empirical Estimation Models**:-

An estimation model for computer software uses empirically derived formulas to predict effort as a function of LOC or FP.

**2.1. Structure of the Estimation Models**:-

E = A + B \* ( ev )C

where: E is effort in person-months A, B, and C are empirically derived constants ev is the estimated variable (LOC or FP).

**2.2. Concrete Estimation Models**:-

LOC oriented models

E = 5.2 \* ( KLOC )0.91

E = 5.5 + 0.73 \* ( KLOC )1.16

E = 3.2 \* ( KLOC )1.05

E = 5.288 \* ( KLOC )1.047

FP oriented models

E = -13.39 + 0.0545 \* FP

E = 60.62 \* 7.728 \* 10-8 \* FP3

E = 585.7 + 15.12 FP

**6.3. The COCOMO Models: Constructive Cost Models**:-

The COCOMO model is an hierarchy of models as follows:

**Model 1:** The Basic COCOMO model- computes software development effort and cost as a function of program   
size expressed in estimated lines of code.

**Model 2:**Th e Intermediate COCOMO -computes software development effort and cost as a function of program   
size and a set of cost drivers that include subjective assessments of product, hardware, personel, and project attributes.

**Model 3:** The Advanced COCOMO model- incorporates all characteristics of the intermediate model with an assessment of the impact on each step of the software development process.

The COCOMO can be applied to three different kinds of software project classes:

* organic mode projects: small, simple projects developed by small teams of software engineers, that work with less than rigid requirements;
* semi-detached mode projects: medium in size and complexity projects, developed by teams with mixed experience, that work with mixed requirements;
* embedded mode projects: projects that have to be developed under tight hardware and software constraints.