**SOEN 6841 Software Project Management**

**Winter 2025**

**Chapter 3 Exercise**

**Amanpreet Kaur (40301892)**

**3.2 Describe the COCOMO technique for deriving effort and cost estimates for software**

**projects?**

**Answer:**

**COCOMO (COnstructive COst MOdel)** is a widely used technique for estimating the effort and cost of software development projects. It provides a realistic way to estimate the resources required for software development based on project characteristics. COCOMO was introduced by Barry Boehm in 1981 and has three variations: Basic, Intermediate, and Detailed.

* **Basic COCOMO:**

Basic COCOMO is the simplest version, used for rough estimates in the early stages of project planning. It provides a high-level estimation of effort and cost without considering detailed project attributes. The estimation is based on the size of the software, measured in **Kilo Lines of Code (KLOC)**, and categorized into three modes of development:

* **Organic**: Small teams working on well-understood projects (e.g., payroll system).
* **Semi-Detached**: Projects with a mix of experienced and inexperienced developers (e.g., a database management system).
* **Embedded**: Complex, real-time, hardware-software integrated projects (e.g., aircraft control systems).

**Formula for Basic COCOMO:**

Effort = 2.94 × EAF × KLOCE

Duration = EffortSE

where:

**EAF** = Effort Adjustment Factor (Default is 1 in Basic COCOMO).

**KLOC** = Size of the project in Kilo Lines of Code.

**E** and **SE** are exponents derived from scale drivers.

**Example**

Consider a payroll management system estimated to be **50 KLOC**. Assuming it falls under the **organic** category:

Effort = 2.94×1×501.05 = 2.94×57.87 = 170.15 Person-Months

Duration = 170.150.38 = 10.3 Months

This means that about **170 person-months** of effort will be required, and the project may take around **10 months** to complete.

* **Intermediate COCOMO:**

Intermediate COCOMO refines the Basic COCOMO model by considering **cost drivers** that impact software development. These cost drivers are divided into **four categories**:

* **Product Attributes**: Complexity, required reliability, database size, etc.
* **Hardware Attributes**: Execution time constraints, memory requirements, etc.
* **Personnel Attributes**: Experience level, programming capability, etc.
* **Project Attributes**: Development tools, schedule constraints, etc.

Each attribute is assigned a weight (0.7 to 1.7), which affects the estimation.

**Formula for Intermediate COCOMO:**

Effort = a × EAF × KLOCE

Where:

**a** and **E** depend on the type of software project (Organic, Semi-Detached, Embedded).

**EAF** is the calculated effort adjustment factor based on cost drivers.

**Example**

For a database management system (semi-detached), assume:

KLOC = 100

Cost drivers (EAF) = 1.2 (after adjusting for complexity and team experience)

a = 3.0, E = 1.12

Effort = 3.0 × 1.2 × 1001.12 = 3.0 × 1.2 × 132 = 475.2 Person-Months

This shows that the effort required will be **475 person-months**, considering project-specific cost drivers.

* **Detailed COCOMO**

Detailed COCOMO extends Intermediate COCOMO by applying cost driver analysis to **each phase** of software development (e.g., requirements, design, coding, testing). This results in more precise effort estimation per phase.

* Used during project development, unlike Basic and Intermediate COCOMO, which estimate the entire project.
* Breaks down effort estimation phase-wise, ensuring better resource allocation.
* More suited for large-scale or long-duration projects.

**Example:**

For an automobile control system (embedded project, 200 KLOC):

Effort for Design Phase = 35% of total effort

Effort for Coding Phase = 40% of total effort

Effort for Testing Phase = 25% of total effort

If the total estimated effort from Intermediate COCOMO is **800 person-months**, we can distribute effort per phase:

Design Phase Effort = 280 person-months

Coding Phase Effort = 320 person-months

Testing Phase Effort = 200 person-months

By selecting the appropriate **COCOMO model**, project managers can make realistic software effort and cost estimations, ensuring smooth project execution.