Unit-2

Software Project Scheduling

Software project scheduling is the process of determining when project activities will be performed and in what sequence. It involves creating a timeline that outlines the start and end dates for each task or activity within the project, as well as dependencies between tasks. The primary goal of software project scheduling is to develop a realistic and achievable plan for completing the project within the defined constraints of time, budget, and resources.

Key aspects of software project scheduling include:

1. Task Sequencing
2. Estimating Task Durations
3. Resource Allocation
4. Identifying Milestones
5. Critical Path Analysis
6. Contingency Planning
7. Updating and Monitoring

Scheduling Techniques

1. Work Breakdown Structure (WBS)
2. Network Planning
3. Critical Path Method (CPM)
4. Program Evaluation and Review Technique (PERT)
5. Precedence Diagramming Method (PDM)
6. Shortening project duration
7. Identifying critical activities
8. Forward pass and Backward pass

Objectives of activity planning

1. Resource Allocation: Activity planning helps in determining the resources required for various tasks within the project. This includes human resources, equipment, software tools, and any other resources necessary for the completion of activities.

2. Time Management: It aids in scheduling the activities in a logical sequence, estimating the time required for each activity, and allocating appropriate deadlines. This ensures timely completion of tasks and helps in meeting project deadlines.
3. Risk Management: By breaking down the project into smaller activities, potential risks and uncertainties can be identified at each stage. This allows project managers to develop contingency plans and mitigate risks effectively.
4. Task Dependencies: Activity planning helps in identifying dependencies between tasks. Understanding these dependencies is crucial for scheduling activities in the correct sequence and avoiding bottlenecks in the project workflow.
5. Progress Tracking: It provides a framework for monitoring the progress of the project. By tracking the completion of individual activities against the planned schedule, project managers can identify deviations early and take corrective actions to keep the project on track.
6. Communication and Coordination: Activity planning facilitates communication and coordination among team members by providing a clear roadmap of tasks and deadlines. It ensures that everyone is aware of their responsibilities and contributes towards achieving the project goals.
7. Budget Management: By estimating the resources required for each activity and their associated costs, activity planning helps in budget management. It enables project managers to allocate funds efficiently and avoid cost overruns.
8. Quality Assurance: Planning activities allows for the inclusion of quality assurance tasks at appropriate stages of the project. By integrating quality checks into the project schedule, it helps in ensuring that deliverables meet the required standards and specifications.

Work Breakdown Structure (WBS)

A Work Breakdown Structure (WBS) is a fundamental tool in project management used to break down large projects into smaller, more manageable components. It's like creating a roadmap for your project, visually outlining all the steps and tasks needed to achieve your goals.

Benefits/Advantages of using a WBS:

- **Clearer project scope:** Defines all the work required and ensures nothing is missed.

- **Improved planning and scheduling:** Break down larger tasks into achievable steps, making it easier to estimate effort and duration.
- **Enhanced cost control:** Allows for more accurate cost estimates and better cost tracking.
- **Better communication and collaboration:** Provide a shared understanding of the project among stakeholders.
- **Reduced risk:** Helps identify potential risks and develop mitigation strategies.

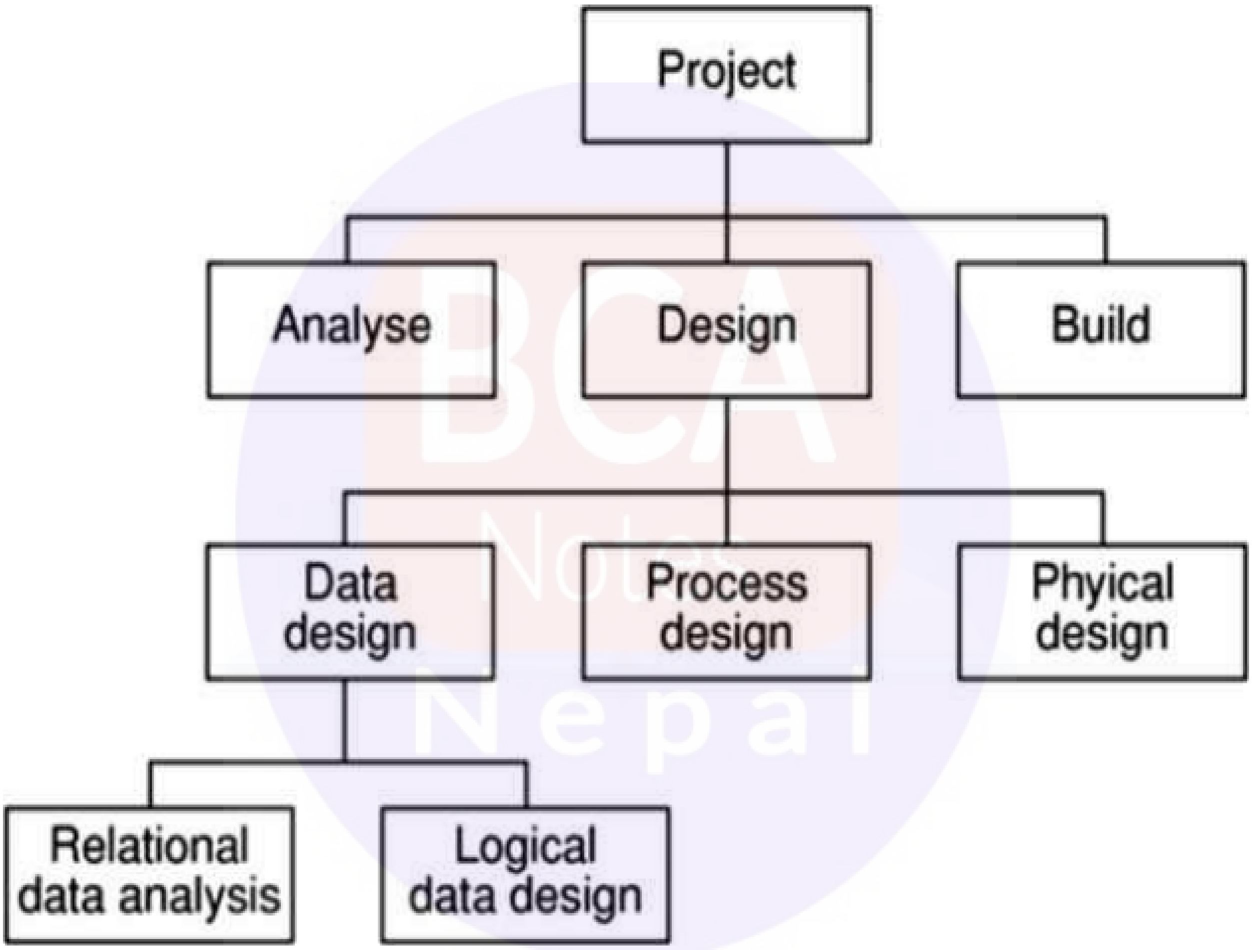


Fig: Fragment of an activity-based work breakdown structure

Network Planning Model (NPM)

A network planning model refers to a specific technique for visualizing and analyzing the tasks involved in a project. It represents the project as a network of activities connected by dependencies, showing the order in which tasks must be completed and how delays in one task can impact others. These models are vital for planning and scheduling, resource allocation, and risk management.

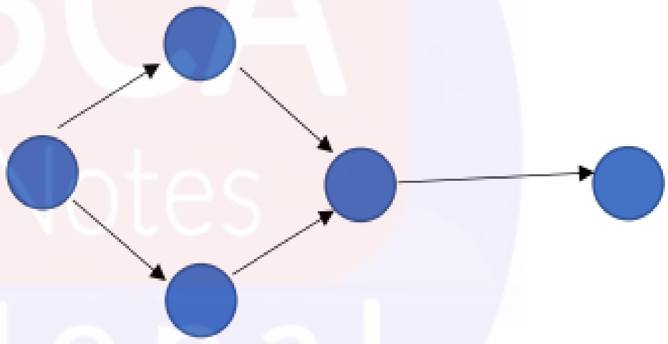
// you don't need to write it just for information

What is network?

A group of computers and other devices connected together to share resources and information are called network.

Network consists of:

1. Node
2. Arrow



Here are the two most common network planning models used in software development

1. Critical Path Method (CPM):
2. Program Evaluation and Review Technique (PERT):

Critical Path Method (CPM)

- Determines the longest sequence of activities that must be completed on time for the project to finish on schedule. This sequence is called the critical path.
- Uses deterministic estimates for task durations, meaning specific fixed values.
- Identifies critical tasks that have no slack time and any delay in which will directly impact the entire project timeline.

Used by CPM (Critical Path Method) and PERT (Program Evaluation Review Technique) to visualize the project as a network. Both of these techniques used two approaches:

1. activity-on-arrow

2. activity-on-node

activity-on-arrow

In activity-on-arrow (AOA) network diagrams, activities are represented by arrows, while events or milestones are depicted by circles (called nodes). This method emphasizes the flow and duration of tasks. Here's a breakdown:

Elements:

- **Arrows:** Represent activities, with the length often proportional to the activity's estimated duration. Labels on the arrows describe the specific tasks.
- **Nodes:** Represent the start and end points of activities, signifying events or milestones. They don't represent time, just completion of the preceding activity.
- **Dependencies:** Connections between nodes with directed arrows show the order in which tasks must be completed.

Advantages:

- **Simple and visually clear:** Easy to understand the flow of activities.
- **Focuses on durations:** Length of arrows directly communicates task duration.
- **Suitable for linear projects:** Best for projects with well-defined dependencies and limited branching.

Disadvantages:

- **Limited flexibility:** Can be challenging to show complex tasks or parallel activities.
- **Resource allocation unclear:** Doesn't explicitly show which resources are needed for each activity.
- **Less common:** Less widely used than the alternative - activity-on-node (AON) method.

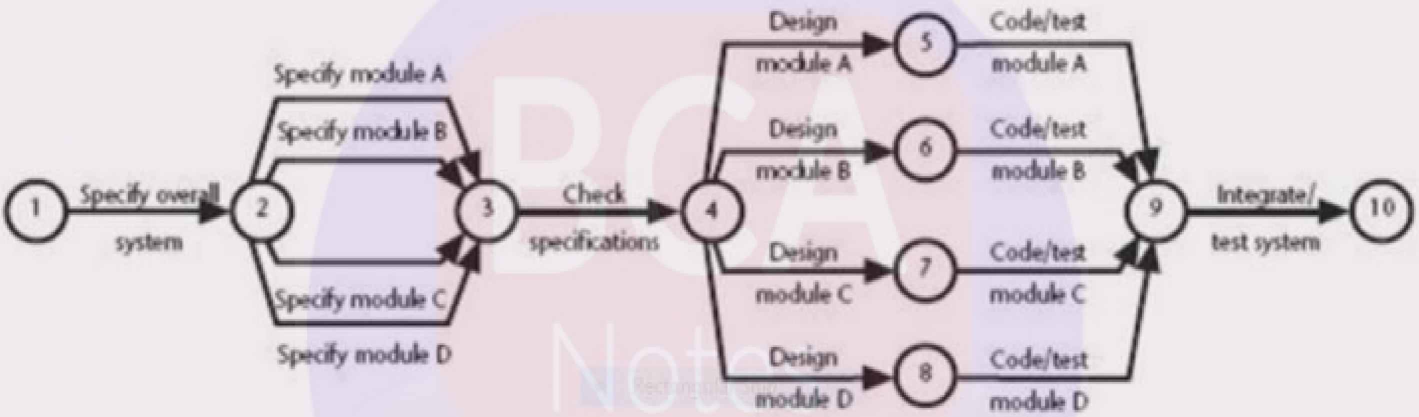


Fig: Example of activity on arrow

activity-on-node

In activity-on-node (AON) network diagrams, the focus shifts, representing activities as boxes (nodes) and dependencies as arrows connecting them. This method highlights the relationships and dependencies between tasks. Here's a look at its key features:

Elements:

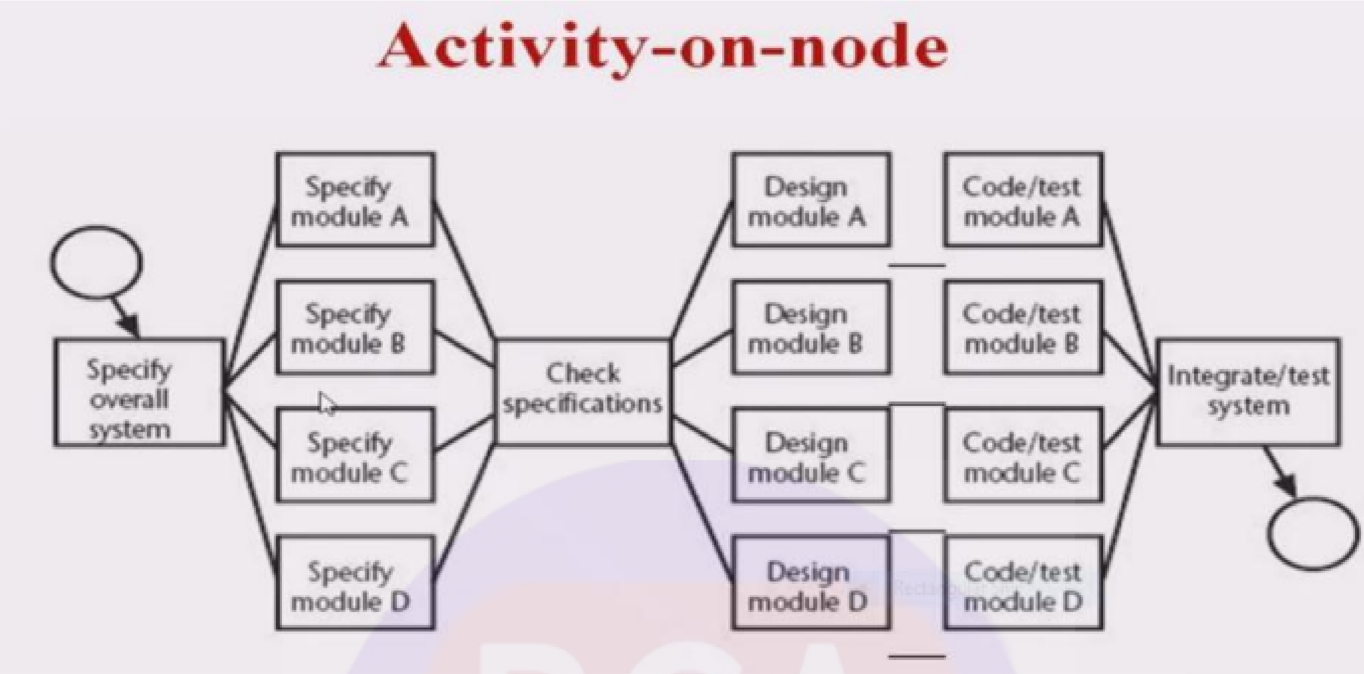
- **Nodes:** Represent activities, often shaped like boxes or squares. Labels within the nodes describe the specific tasks.
- **Arrows:** Indicate the **dependencies** between activities. Arrow direction shows the required sequence (e.g., finish-to-start, start-to-start).
- **Dummy activities:** Sometimes used to represent external dependencies or time gaps without actual work.

Advantages:

- **Greater flexibility:** Easier to show complex tasks, parallel activities, and different dependency types.
- **Resource allocation possible:** Can associate resources with activities within the nodes.
- **More widely used:** The preferred method for most project management scenarios.

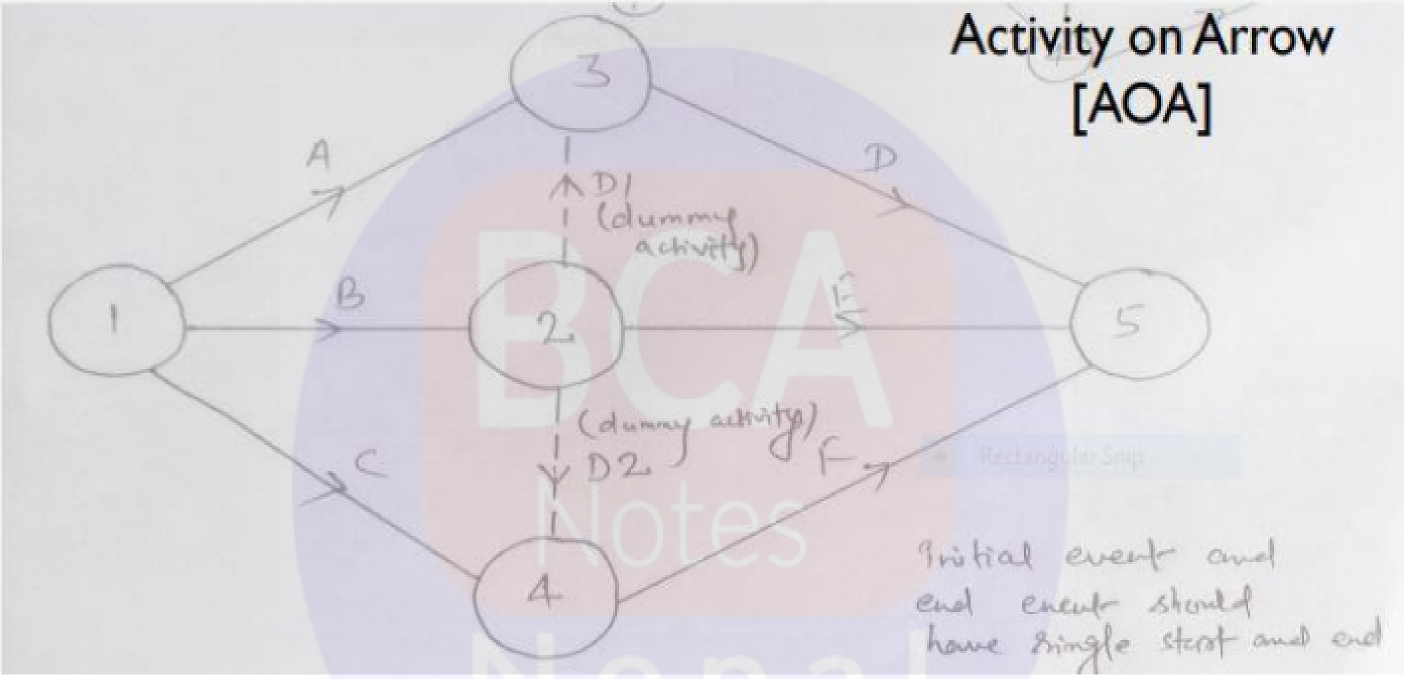
Disadvantages:

- **Visualization of duration less direct:** Activity duration isn't always readily apparent from the diagram.
- **Can appear more complex:** Initial understanding might require more effort compared to AOA.


Activity-on-node
Specify overall system
Specify module A
Specify module B
Specify module C
Specify module D
Check specifications
Design module A
Design module B
Design module C
Design module D
Code/test module A
Code/test module B
Code/test module C
Code/test module D
Integrate/test system

Example 1

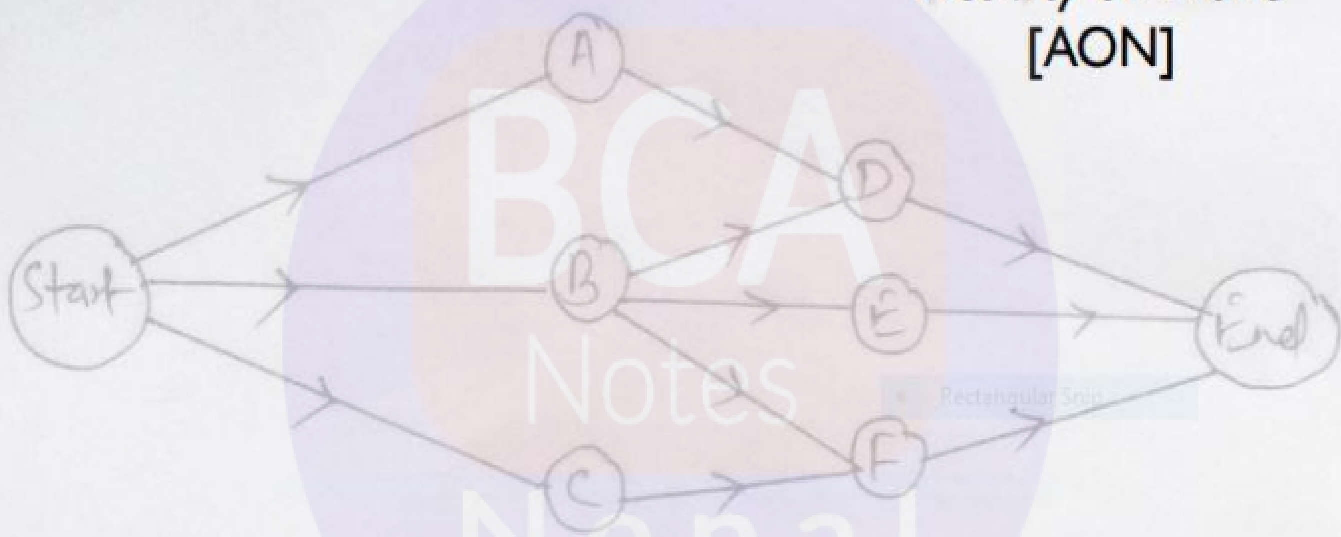
Activity	Predecessor Activity
A	-
B	-
C	-
D	A,B
E	B
F	B,C



Example 1

Activity	Predecessor Activity
A	-
B	-
C	-
D	A,B
E	B
F	B,C

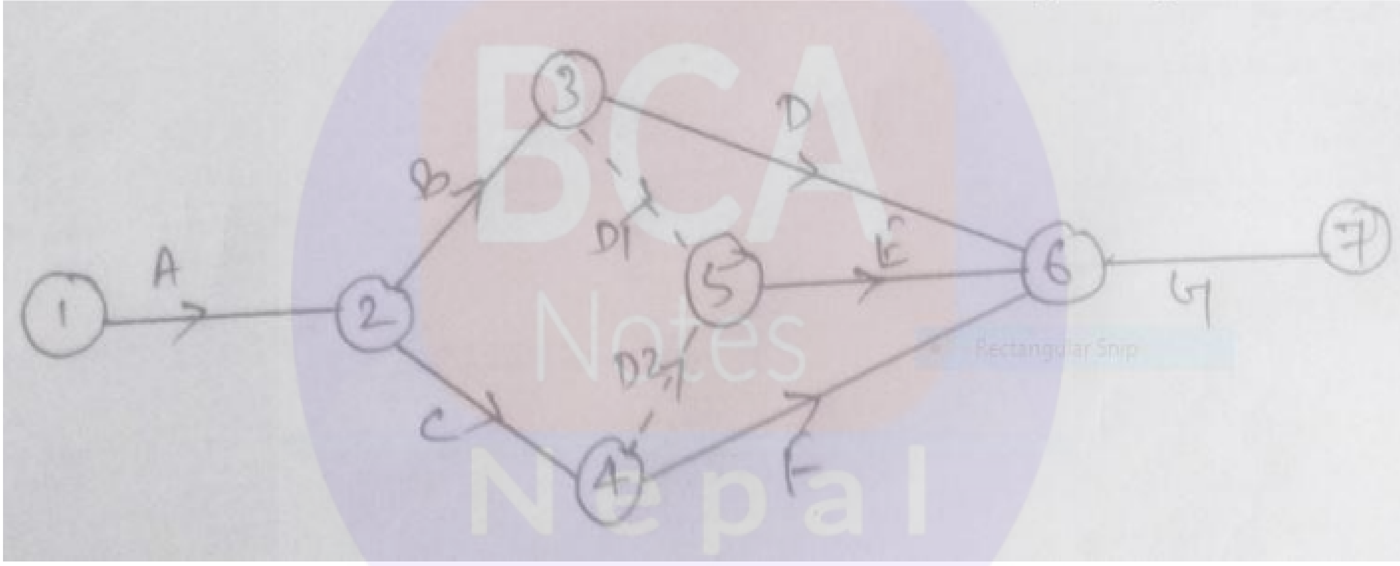
Activity on Node [AON]



Example 2

Activity	Predecessor Activity
A	-
B	A
C	A
D	B
E	B,C
F	C
G	D,E,F

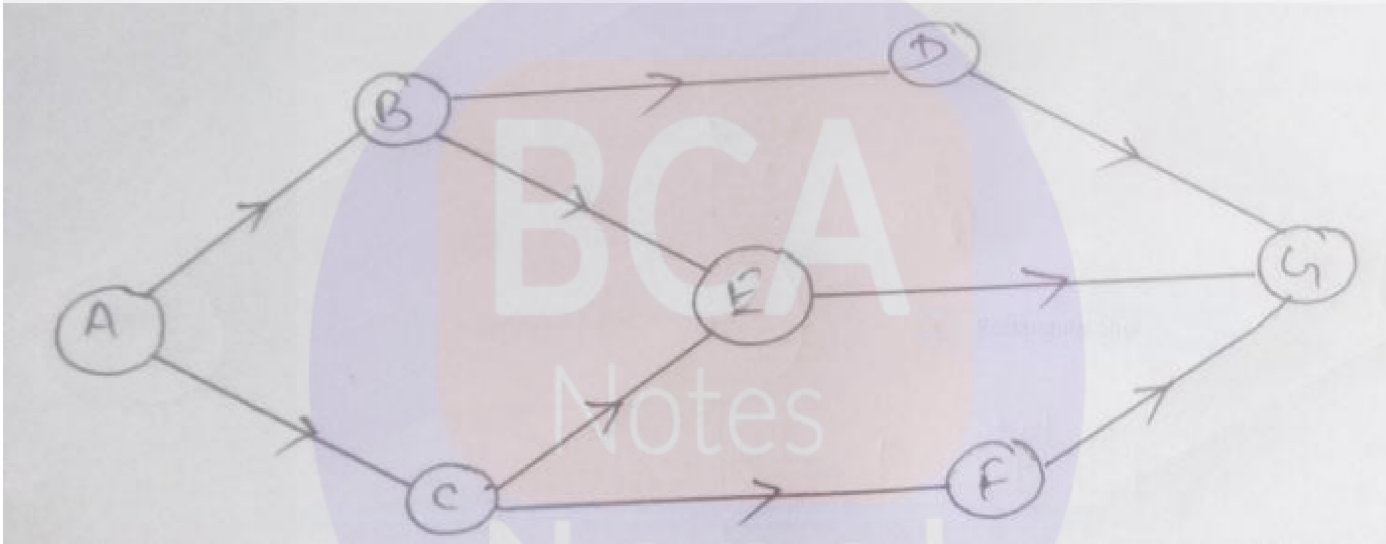
Activity on Arrow [AOA]



Example 2

Activity	Predecessor Activity
A	-
B	A
C	A
D	B
E	B,C
F	C
G	D,E,F

Activity on Node
[AON]



Similarities between PERT and CPM

- Both follow the same steps and use network diagrams
- Both are used to plan the scheduling of individual activities that make up a project
- They can be used to determine the earliest/latest start and finish times for each activity

Differences between PERT and CPM

Feature	PERT	CPM
Focus	Uncertainty in durations, probability of completion	Deterministic durations, critical path identification
Time Estimates	Optimistic, Most Likely, Pessimistic	Single, deterministic estimate
Network Representation	AOA and AON	Typically AON
Risk Management	Explicitly considers uncertainty and risk	Identifies critical tasks for monitoring
Applications	Projects with unknown durations (e.g., R&D)	Projects with well-defined tasks and durations (e.g., construction)

The Precedence Diagramming Method (PDM)

The Precedence Diagramming Method (PDM) is a technique used to model the logical relationships between various tasks or activities within a software development project. PDM helps in depicting the sequence in which tasks should be performed and the dependencies between them.

Here's how PDM is typically applied in software project management:

1. **Task Identification:** The first step involves identifying all the tasks or activities required to complete the software project. These tasks could include requirements gathering, design, coding, testing, and deployment, among others.
2. **Relationship Identification:** Once tasks are identified, the next step is to determine the dependencies or relationships between them. This involves understanding which tasks must be completed before others can start and identifying any tasks that can be performed concurrently.
3. **Construction of the Precedence Diagram:** In PDM, tasks are represented as nodes or boxes, and the relationships between them are depicted as arrows or lines. The arrows indicate the flow of work and the dependencies between tasks. There are four types of dependencies: Finish-to-Start (FS), Finish-to-Finish (FF), Start-to-Start (SS), and Start-to-Finish (SF).
4. **Critical Path Analysis:** Once the precedence diagram is constructed, project managers can analyze it to identify the critical path. The critical path is the longest sequence of dependent tasks that determines the minimum duration required to complete the project. Any delay in tasks along the critical path will directly impact the project's overall duration.
5. **Resource Allocation and Scheduling:** PDM helps project managers in allocating resources efficiently by identifying which tasks can be performed concurrently and which tasks are dependent on others. It also aids in scheduling activities to ensure that resources are utilized optimally and project deadlines are met.
6. **Risk Management:** PDM also facilitates risk management by allowing project managers to identify potential bottlenecks, dependencies, and areas where delays may occur. This enables proactive risk mitigation strategies to be implemented to address potential issues before they impact the project schedule or budget.

Shortening the project duration (SPD)

Shortening the project duration (SPD) in software project management involves strategies to accelerate project completion while still achieving desired outcomes. However, it's essential to approach this carefully, as rushing a project can introduce risks and negatively impact quality. Here are some key considerations:

Strategies/Techniques for SPD:

- **Identify Critical Path:** Analyze the project network to pinpoint the critical path, the sequence of tasks that directly affects the overall duration. Optimizing these tasks will have the most significant impact on speeding up the project.
- **Fast-Track Activities:** Explore options to overlap or expedite non-critical tasks without compromising dependencies. Techniques like parallel development or lead-time reduction can be helpful.
- **Optimize Resource Allocation:** Ensure the right resources (people, tools, skills) are assigned to critical tasks and efficiently utilized. Consider overtime, hiring additional personnel, or outsourcing specific tasks if necessary.
- **Reduce Scope:** Carefully evaluate project scope and see if any features or functionalities can be deferred to a later release or eliminated altogether. Prioritize core functionalities for the initial release to meet the shortened timeline.
- **Simplify Processes:** Streamline communication, approval processes, and decision-making to avoid delays. Implement tools and automation wherever possible to increase efficiency.
- **Utilize Risk Management:** Proactively identify and mitigate potential risks that could derail the schedule. Contingency plans and clear communication are crucial.

There are a number of ways to shorten the duration of a project, such as:

- Crashing: This involves spending more money to reduce the time required to complete an activity.
- Fast tracking: This involves starting activities that are not dependent on each other earlier than originally planned.

Identifying critical activities (ICA)

Identifying critical activities is crucial for ensuring timely project completion and preventing potential delays. These activities, also known as "critical path tasks," directly impact the overall project duration and must be completed on schedule to stay within the established timeframe.

What are critical activities?

- Critical activities are tasks within a project's network that have zero slack time. This means that any delay in their completion directly translates to a delay in the entire project deadline.
- They lie on the critical path, which is the longest sequence of dependent tasks that must be finished sequentially to achieve project completion.

Why is it important to identify them?

Knowing your critical activities allows you to:

1. Focus resources: Prioritize resources (personnel, budget, tools) towards accomplishing critical tasks first.
2. Monitor closely: Proactively track progress of critical activities to avoid potential bottlenecks and take corrective action if needed.
3. Manage risks: Identify and mitigate potential risks that could affect critical activities and detail the project schedule.

4. Make informed decisions: Understand the impact of potential delays or scope changes on the critical path and adjust accordingly.

How to identify critical activities:

Several methods can help you pinpoint critical activities in software projects:

- **Critical Path Method (CPM):** This traditional technique involves creating a network diagram of project tasks and their dependencies, calculating task durations, and then using calculations to identify the critical path.
- **PERT (Program Evaluation and Review Technique):** This method incorporates uncertainty in task durations by using probabilistic estimates, providing a more nuanced view of the critical path and potential risks.
- **Project management software:** Many project management tools offer built-in features for network analysis and critical path identification, simplifying the process.

Forward Pass and Backward Pass

Forward Pass and Backward Pass are techniques used in conjunction with a network diagram to analyze the project schedule and identify the critical path.

Forward Pass:

- Starts at the project's beginning and moves forward through the network diagram, calculating the earliest possible start and finish dates for each activity based on its duration and dependencies on other tasks.

- It helps you understand the minimum project duration and identify free float, which is the amount of time an activity can be delayed without delaying the overall project.

Backward Pass:

- Starts at the project's end date and moves backward through the network diagram, calculating the latest possible start and finish dates for each activity without delaying the overall project.
- It helps you identify the critical path, which is the sequence of activities with zero free float. Any delay in an activity on the critical path directly delays the entire project.

Key terms:

- **Early Start (ES):** The earliest possible date an activity can begin.
- **Early Finish (EF):** The earliest possible date an activity can finish.
- **Late Start (LS):** The latest possible date an activity can begin without delaying the project.
- **Late Finish (LF):** The latest possible date an activity can finish without delaying the project.
- **Free Float:** The amount of time an activity can be delayed without delaying the project.
- **Critical Path:** The sequence of activities with zero free float, determining the overall project duration.

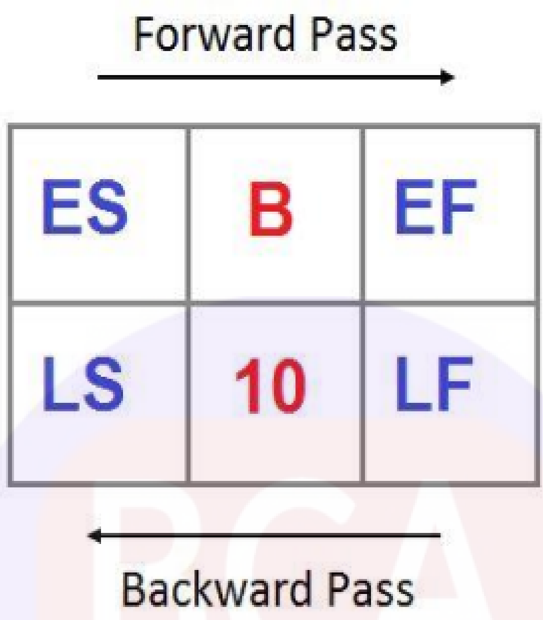
Benefits:

- Helps you understand the project schedule better.
- Identifies potential bottlenecks and areas for improvement.
- Allows you to prioritize tasks and allocate resources effectively.
- Helps you track progress and identify potential risks.

Tools:

- Many project management software applications have built-in features for performing forward and backward passes.

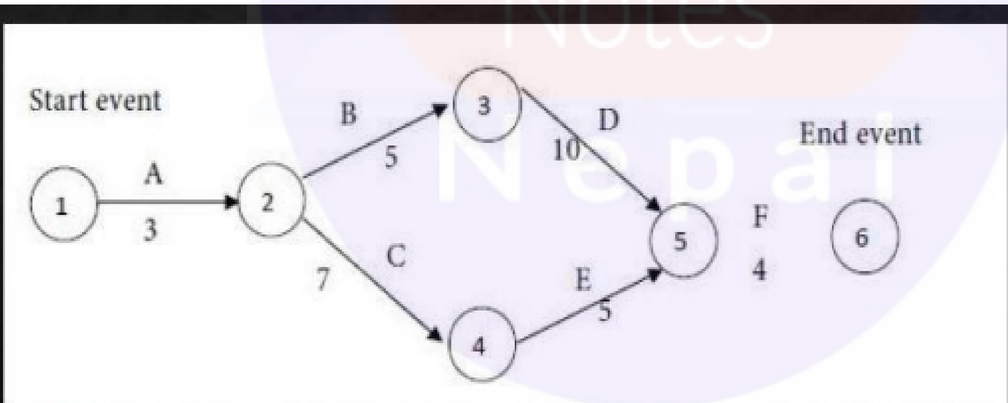
- You can also create network diagrams and perform these calculations manually using spreadsheets or other tools.

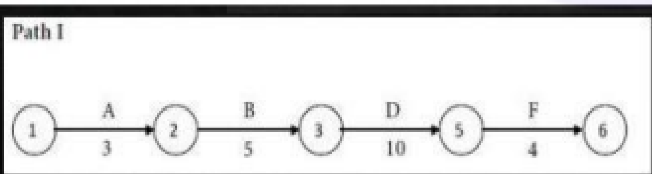


Critical Path : The critical path refers to the **longest stretch** of the activities, and a measure of them from start to finish.

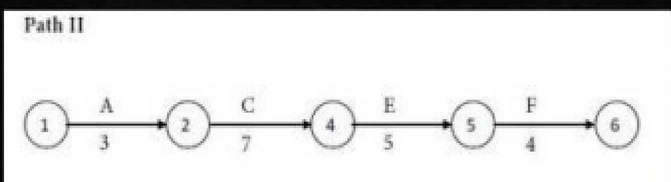
The following details are available regarding a project:

Activity	Predecessor Activity	Duration (Weeks)
A	-	3
B	A	5
C	A	7
D	B	10
E	C	5
F	D,E	4





3+5+10+4=22 weeks

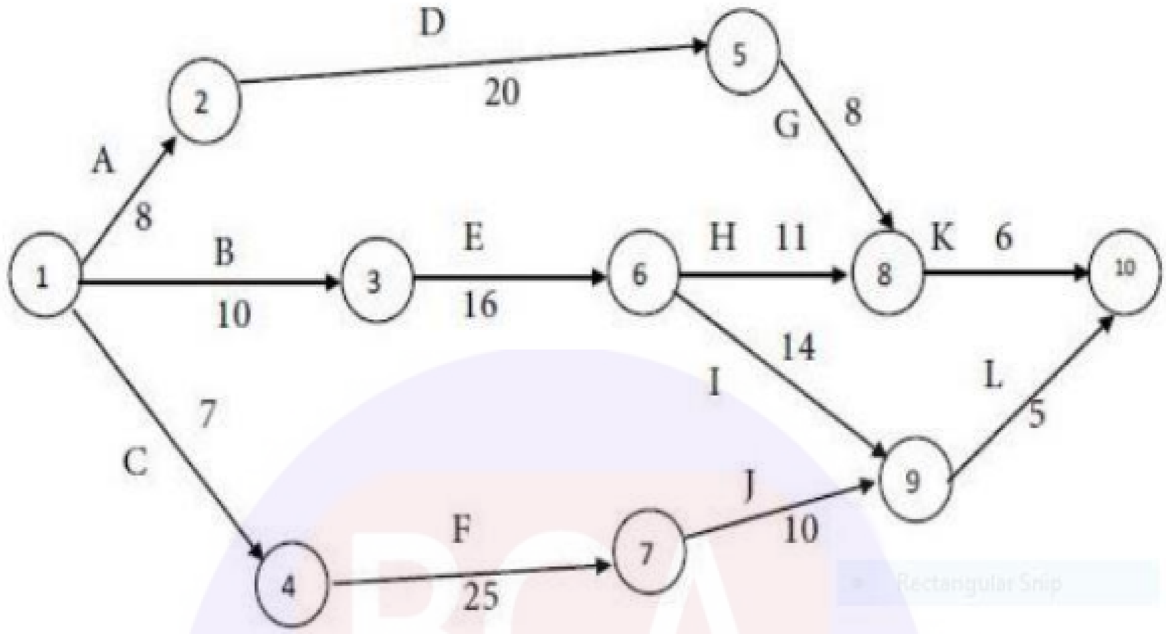


3+7+5+4=19 weeks

Hence critical Path is Path I with 22 Weeks.

Notes By Anjali Kandel
NCMT college (Nepalgunj)

Find out the completion time and the critical activities for the following project:



Solution

Path I

$$1 \xrightarrow[8]{A} 2 \xrightarrow[20]{D} 5 \xrightarrow[8]{G} 8 \xrightarrow[6]{K} 10$$

Time for the path = 8 + 20 + 8 + 6 = 42 units of time.

Path II

$$1 \xrightarrow[10]{B} 3 \xrightarrow[16]{E} 6 \xrightarrow[11]{H} 8 \xrightarrow[6]{K} 10$$

Time for the path = 10 + 16 + 11 + 6 = 43 units of time.

Path III

$$1 \xrightarrow[10]{B} 3 \xrightarrow[16]{E} 6 \xrightarrow[14]{I} 9 \xrightarrow[5]{L} 10$$

Time for the path = 10 + 16 + 14 + 5 = 45 units of time.

Path IV

$$1 \xrightarrow[7]{C} 4 \xrightarrow[25]{F} 7 \xrightarrow[10]{J} 9 \xrightarrow[5]{L} 10$$

Time for the path = 7 + 25 + 10 + 5 = 47 units of time.

Compare the times for the four paths. Maximum of {42, 43, 45, 47} = 47. We see that the following path has the maximum time and so it is the critical path:

$$1 \xrightarrow[7]{C} 4 \xrightarrow[25]{F} 7 \xrightarrow[10]{J} 9 \xrightarrow[5]{L} 10$$

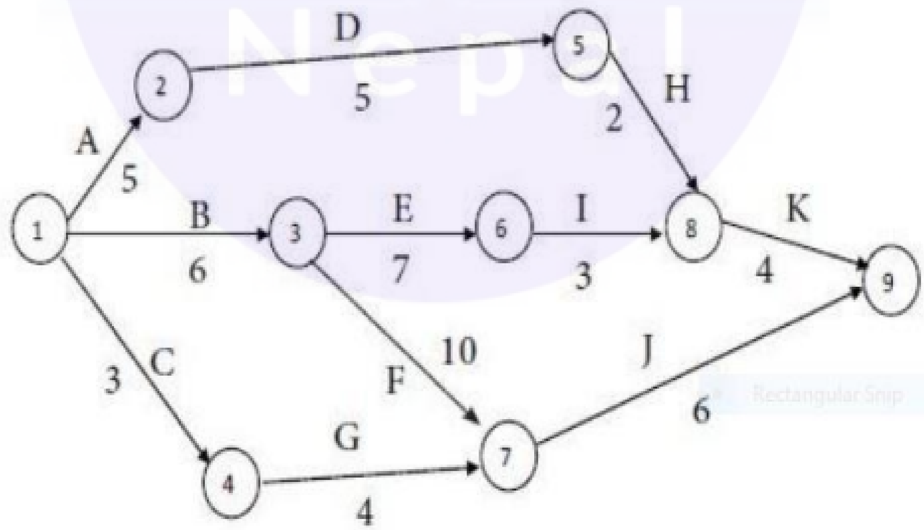
The critical activities are C, F, J and L. The non-critical activities are A, B, D, E, G, H, I and K. The project completion time is 47 units of time.

Draw the network diagram and determine the critical path for the following project:

Activity	Time estimate (Weeks)
1- 2	5
1- 3	6
1- 4	3
2 -5	5
3 -6	7
3 -7	10
4 -7	4
5 -8	2
6 -8	5
7 -9	6
8 -9	4

Question and Solution

We have the following network diagram for the project:



Path I

$$1 \xrightarrow[5]{A} 2 \xrightarrow[5]{D} 5 \xrightarrow[2]{H} 8 \xrightarrow[4]{K} 9$$

Time for the path = 5 + 5 + 2 + 4 = 16 weeks.

Path II

$$1 \xrightarrow[6]{B} 3 \xrightarrow[7]{E} 6 \xrightarrow[5]{I} 8 \xrightarrow[4]{K} 9$$

Time for the path = 6 + 7 + 5 + 4 = 22 weeks.

Path III

$$1 \xrightarrow[6]{B} 3 \xrightarrow[10]{F} 7 \xrightarrow[6]{J} 9$$

Time for the path = 6 + 10 + 6 = 16 weeks.

Path IV

$$1 \xrightarrow[3]{C} 4 \xrightarrow[4]{4} 7 \xrightarrow[6]{5} 9$$

Time for the path = 3 + 4 + 6 = 13 weeks.

Compare the times for the four paths. Maximum of (16, 22, 16, 13) = 22. We see that the following path has the maximum time and so it is the critical path:

$$1 \xrightarrow[6]{D} 3 \xrightarrow[7]{E} 6 \xrightarrow[5]{I} 8 \xrightarrow[4]{K} 9$$

The critical activities are B, E, I and K. The non-critical activities are A, C, D, F, G, H and J. The project completion time is 22 weeks.

PERT Networks (cont'd)

- Formula for Calculating Estimated PERT Times

$$T_e = \frac{T_o + 4T_m + T_p}{6}$$

T_e = Time estimate (average)

T_o = Time estimate (optimistic)

T_m = Time estimate (most likely)

T_p = Time estimate (pessimistic)

Chapter Six | 34

Standard Deviation of an Activity, $\sigma = \dfrac{P - O}{6}$

Variance of an Activity, $\sigma^2 = \left(\dfrac{P - O}{6}\right)^2$

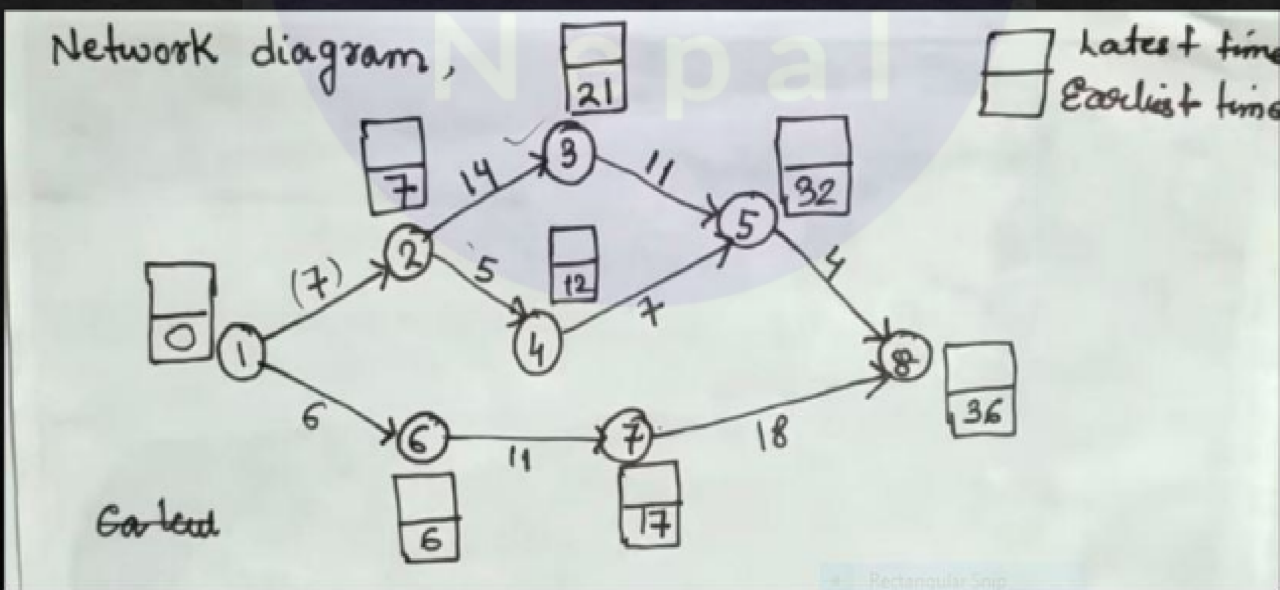
Condition in Critical Path

$$\left.\begin{aligned} E_i &= L_i \\ E_j &= L_j \\ E_j - E_i = L_j - L_i &= D_{ij} \end{aligned}\right\}$$

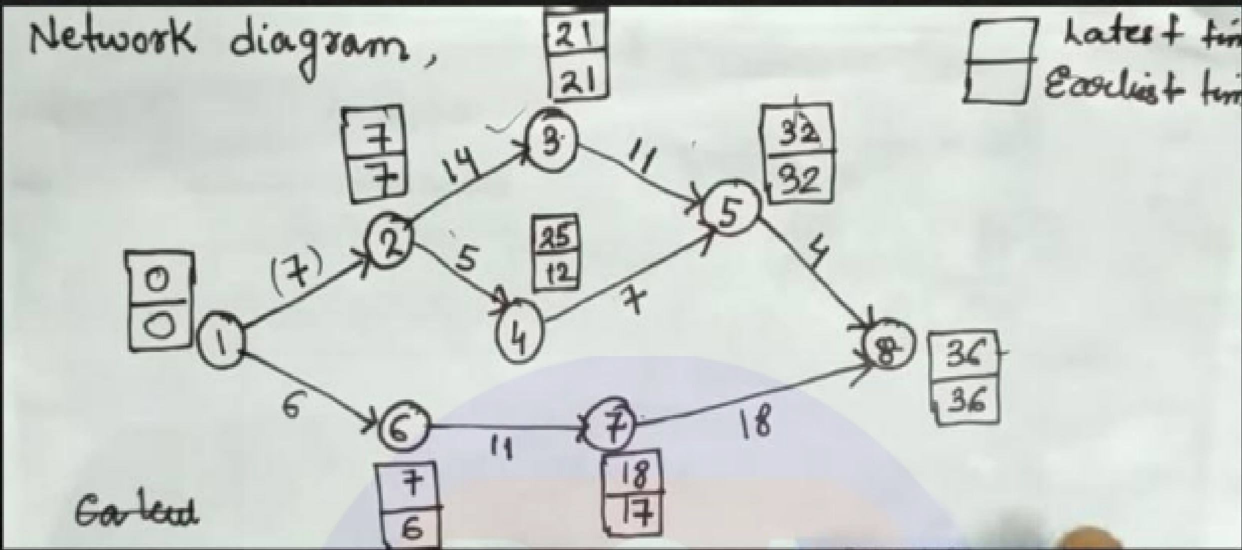
PERT (Program Evaluation and Review Technique)

Activity	Estimated Duration (Weeks)		
	Optimistic time (To)	Most Likely Time (Tm)	Pessimistic time (Tp)
1-2	1	7	13
1-6	2	5	14
2-3	2	14	26
2-4	2	5	8
3-5	7	10	19
4-5	5	5	17
6-7	5	8	29
5-8	3	3	9
7-8	8	17	32

Earliest Time



Latest Time



Critical Path=**1-2-3-5-8**

Expected project duration=7+14+11+4=36 weeks

Project length variance=4+16+4+1=25

Standard Deviation=5