SCHEDULING OF SOFTWARE

SOFTWARE PROJECT MANAGEMENT

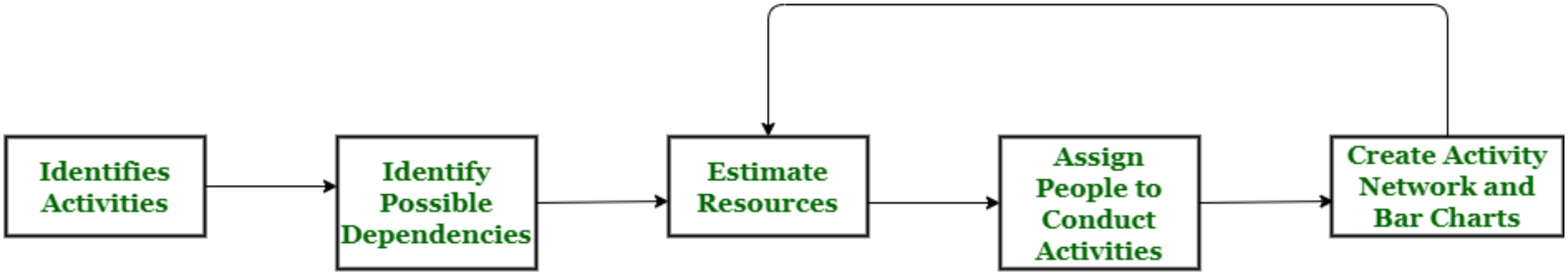
Project planning involves:

- Project scope definition (WBS)
- Software estimation (cost/effort required)
- Software scheduling (time required)
  - ✓ Bar charts/activity networks
  - ✓ Gantt charts

PROJECT SCHEDULING

- Split project into tasks
- estimate time and resources required to complete each task.
- Organize tasks concurrently to make optimal use of workforce.
- Minimize task dependencies to avoid delays between tasks.
- Dependent on project managers intuition and experience.

PROJECT SCHEDULING PROCESS



SCHEDULING PROBLEMS

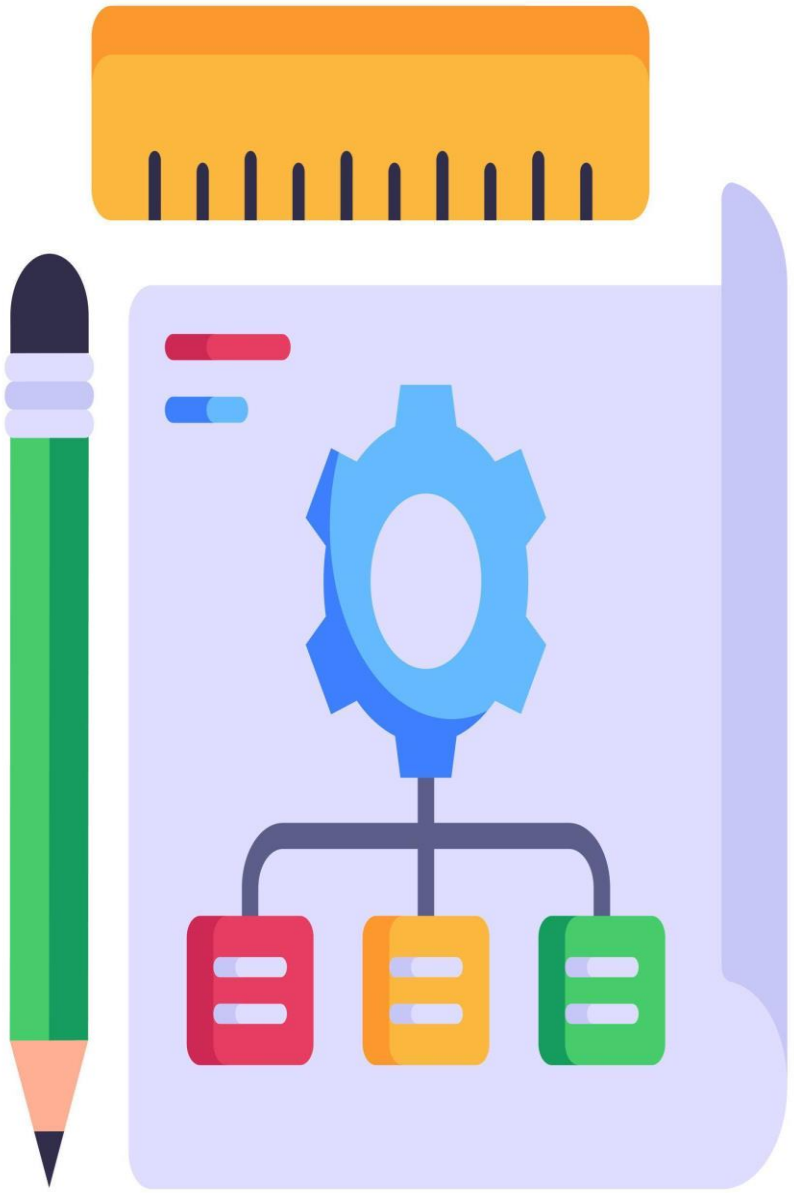
Estimating the difficulty level of problems and predicting the cost of developing a solution is challenging.

- Productivity is not proportional to the number of people working on a task.
- Adding people to a late project makes it even later because of communication overheads.
- The unexpected always happens. Always allow contingency in planning.

PROJECT SCHEDULING TECHNIQUES

BAR CHARTS & ACTIVITY NETWORKS DIAGRAMS

- Graphical notations used to illustrate the project schedule.
- Show project breakdown into tasks.
- Activity charts show project activities & task dependencies
- Bar charts show schedule against calendar time.

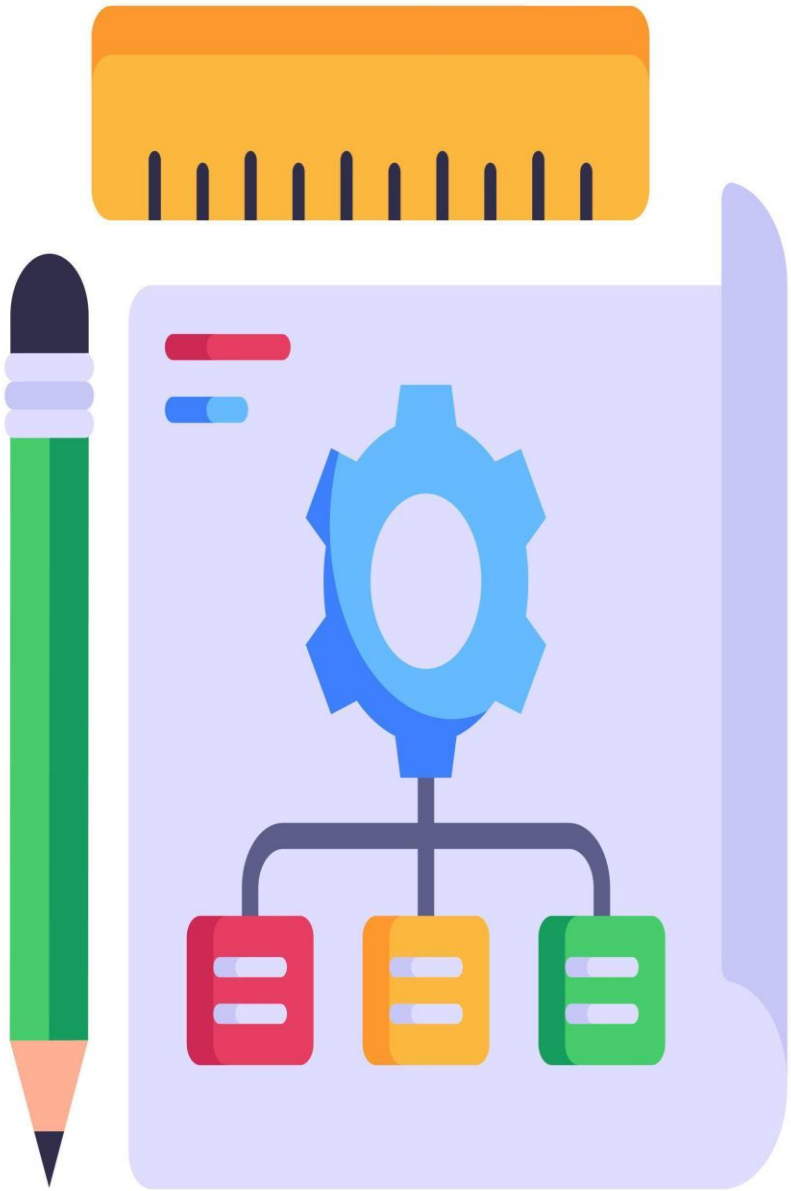


ACTIVITY NETWORK DIAGRAMS

- Model the project's activities and their relationships as a network
- 2 best known techniques:
  - ✓ CPM (Critical Path Method) or CPA (Critical Path Analysis)
  - ✓ PERT (Programme Evaluation Review Technique)
- Both approaches use Activity-on-Arrow approach
- Activities are drawn as arrows joining circles, or nodes, which depicts the possible start and/or completion of an activity or set of activities.

RULES FOR ACTIVITY NETWORK DIAGRAM CREATION

- Network flow from left to right
- Activity cannot begin until all its predecessors are done
- Arrows can overlap each other as they shows the project flow
- Every activity must have an id
- Looping is not allowed
- Every activity must have id > preceding activity.



ACTIVITY NETWORK EXAMPLE

The necessary steps are:

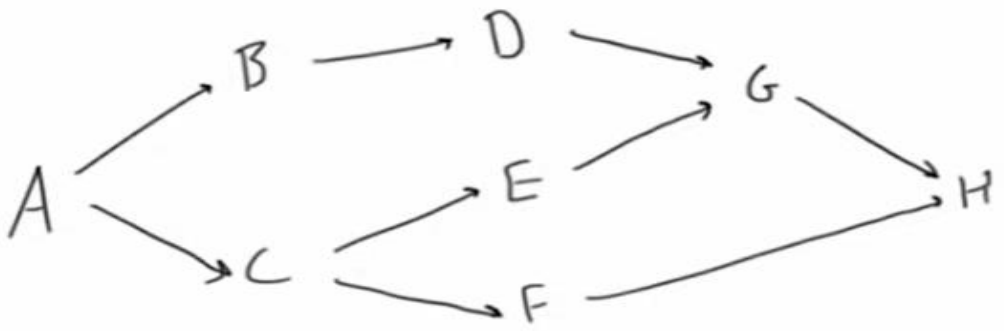
- ✓ Create a network project
- ✓ Identify critical path
- ✓ Do critical path analysis using forward and backward passes (ES, EF, LS, LF)
- ✓ Calculate float/slack values
  - ❖ Total Float-> the amount of time that an activity can be delayed from its ES date without delaying the project finish date.
  - ❖ Free float->the amount of time an activity can be freely delayed without affecting the early start of the following activity.

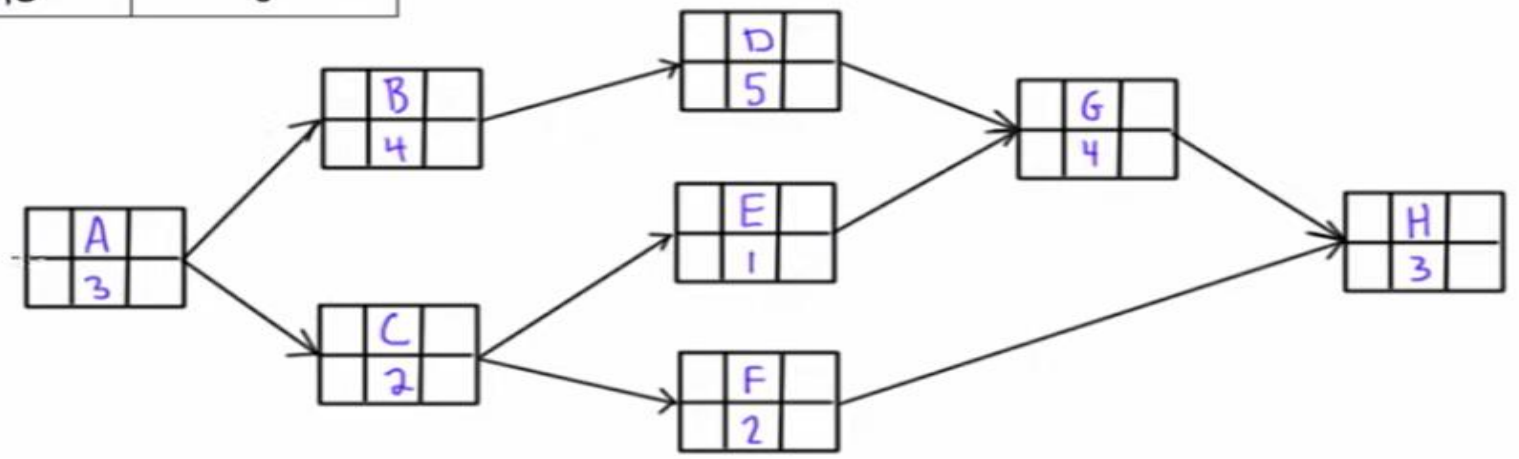
ACTIVITY NETWORK EXAMPLE

Activity	Predecessor	Duration (days)
A	-	3
B	A	4
C	A	2
D	B	5
E	C	1
F	C	2
G	D,E	4
H	F,G	3

DRAW ACTIVITY NETWORK

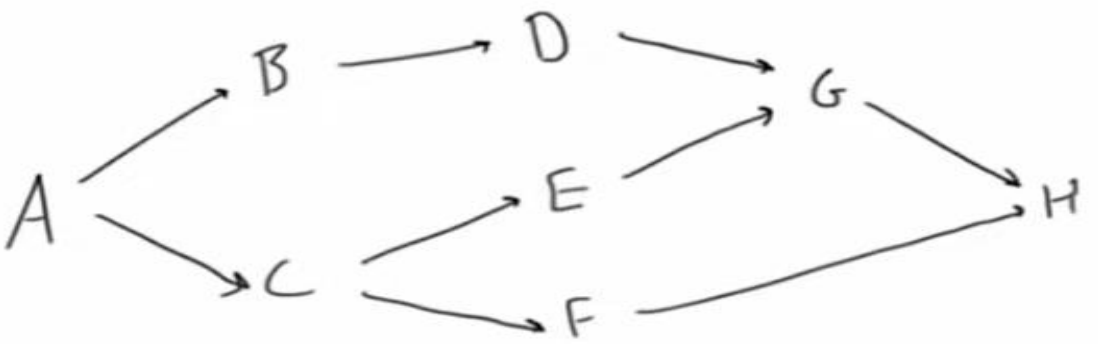
Activity	Predecessor	Duration (days)
A	-	3
B	A	4
C	A	2
D	B	5
E	C	1
F	C	2
G	D,E	4
H	F,G	3

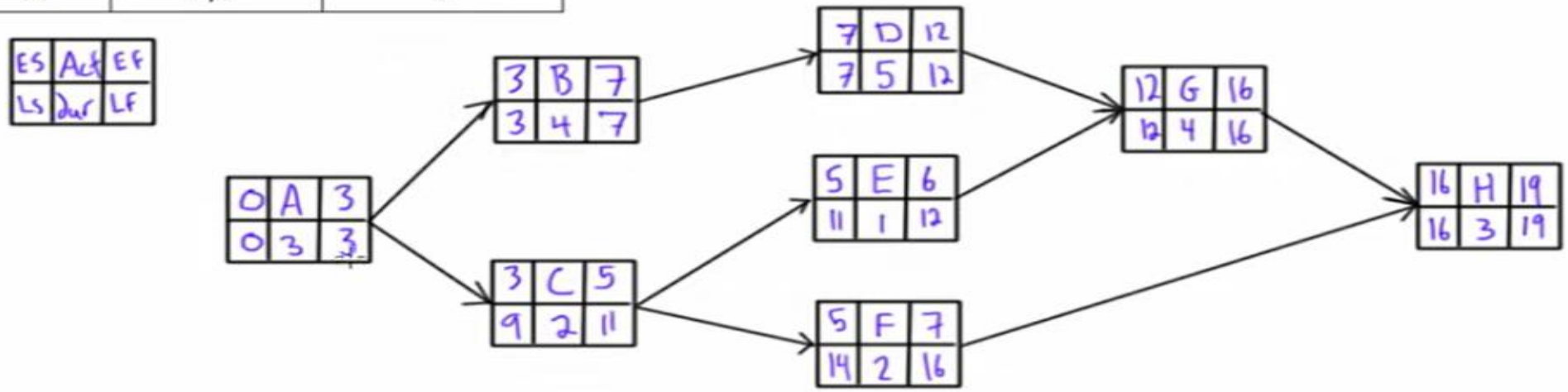




MENTION ES, EF, LS, LF

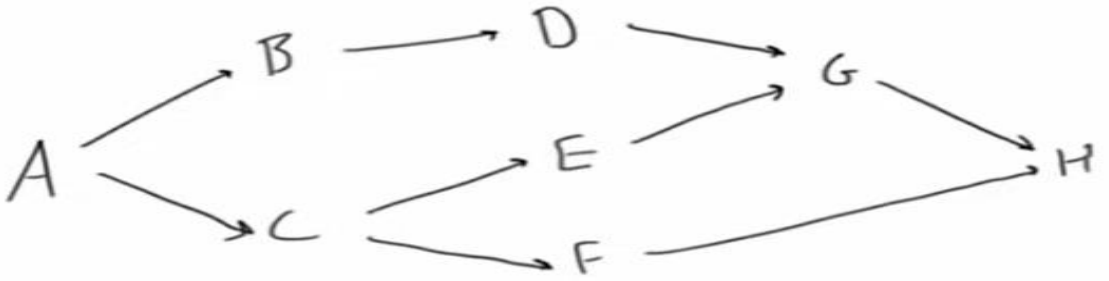
Activity	Predecessor	Duration (days)
A	-	3
B	A	4
C	A	2
D	B	5
E	C	1
F	C	2
G	D,E	4
H	F,G	3

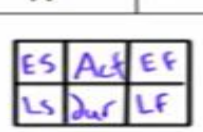


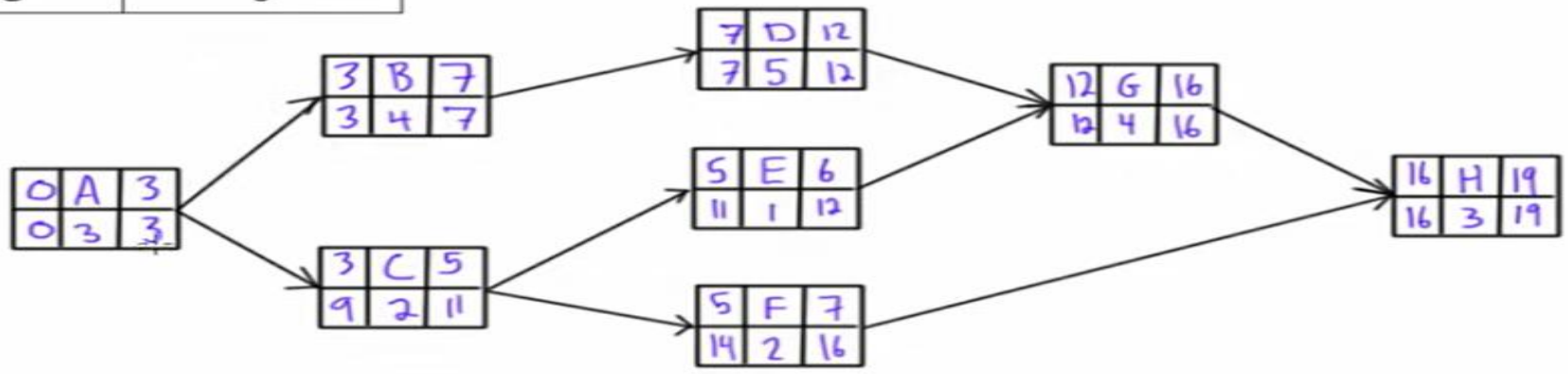


IDENTIFY PATHS AND LABEL CRITICAL PATH

Activity	Predecessor	Duration (days)
A	-	3
B	A	4
C	A	2
D	B	5
E	C	1
F	C	2
G	D,E	4
H	F,G	3







A,B,D,G,H = 19

A,C,E,G,H = 13

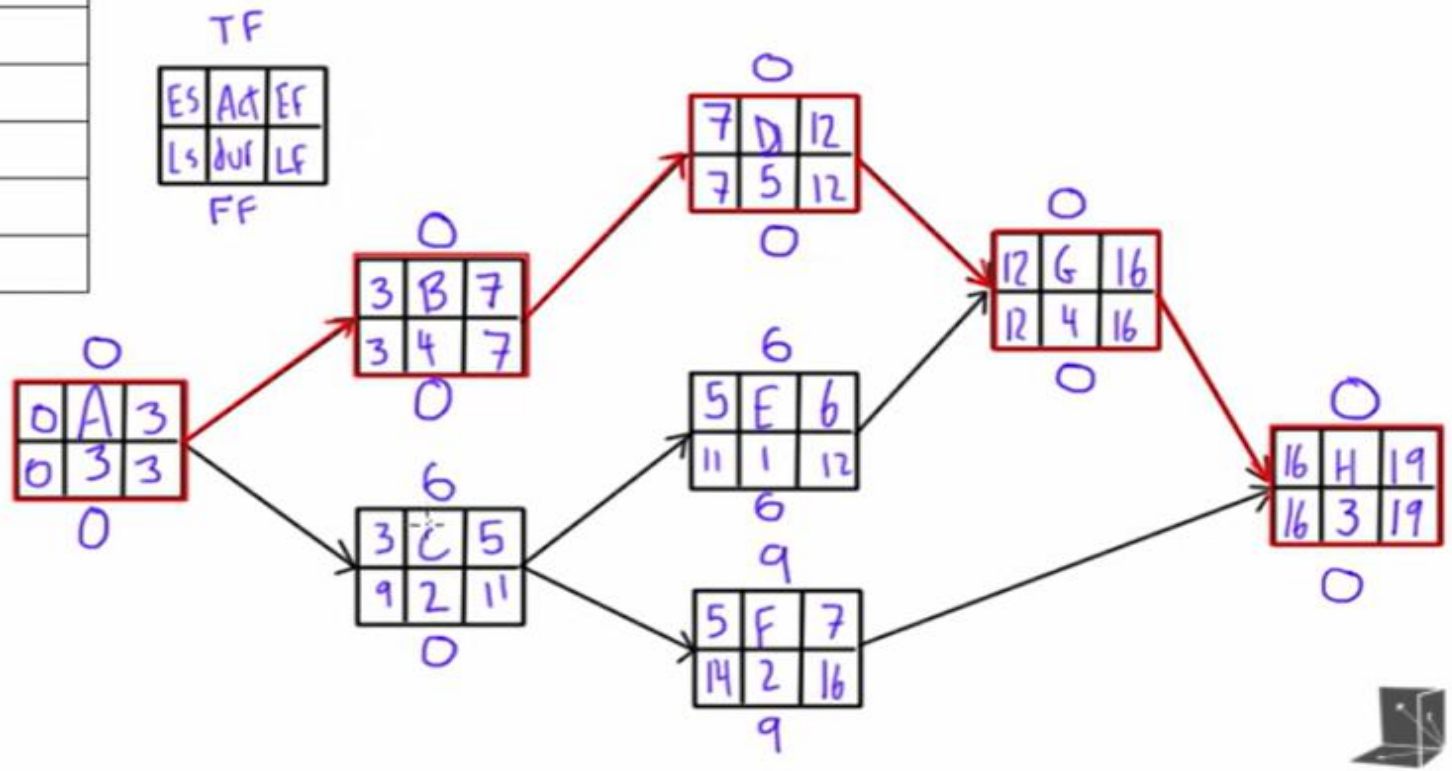
A,C,F,H = 10

IDENTIFY TOTAL SLACK/FLOAT, FREE SLACK/FLOAT

Activity	Predecessor	Duration (days)
A	-	3
B	A	4
C	A	2
D	B	5
E	C	1
F	C	2
G	D,E	4
H	F,G	3

TF = LF-EF OR LS-ES

FF = MIN ($ES_{SUCCESSOR}$) - $ES_{ACTIVITY}$ - $DURATION_{ACTIVITY}$



EXAMPLE # 2

Activity	Predecessor	Duration
A	-	5
B	A	4
C	A	5
D	B	6
E	C	3
F	D,E	4

- Draw activity diagram
- Identify early start or finish and late start or finish date
- Identify all paths and mention critical path
- Calculate slack values