

CTIS359

Principles of Software Engineering

**WBS, Precedence Diagramming
Method (PDM) Network Diagram,
ES, EF, LS,LF, Free Float and Total
Float**

***“Warning: Dates In Calendar Are
Closer Than They Appear!”***
Unknown

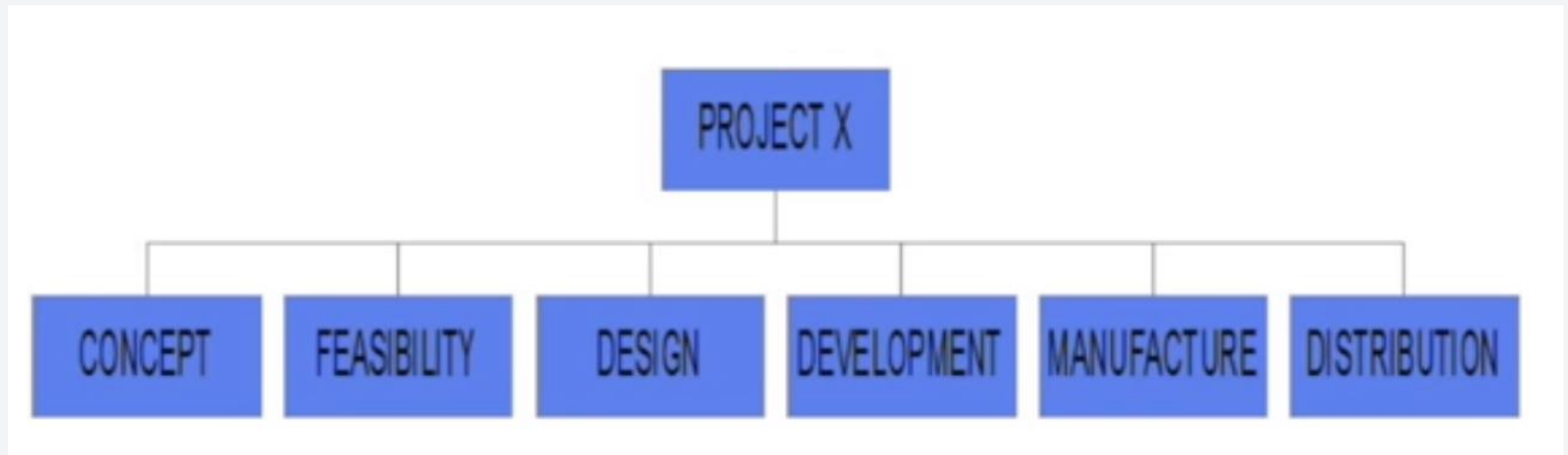
Today

- What is a WBS?
- How we can organize a WBS?
 - Based on
 - phases,
 - functional units,
 - product, and
 - mixture of these
- How we can draw Precedence Diagramming Method (**PDM**) Network Diagram for our tasks in a Project?
 - ES, EF, LS,LF, Free Float and Total Float

Work Breakdown Structures (**WBS**)

- A project planning tool.
- WBS is **a hierarchy** what needs to be achieved.
- A project can be broken down into **work packages (WP)**.
 - A WP is not to be confused with the phases of a project - although they may be the same!
- Before you start creating your WBS, are your sure that your project's objectives are clear?

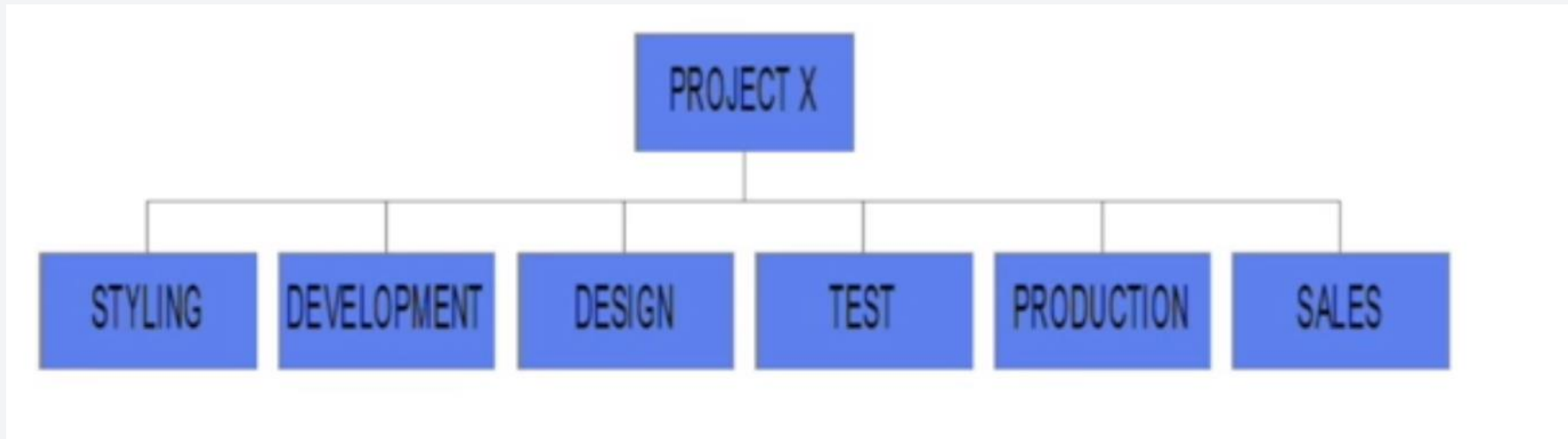
WBS based on phases



WBS based on organizational departments

- Another way of breaking down the project, is by organizational departments.
- This is particularly useful for functionally based organizations.

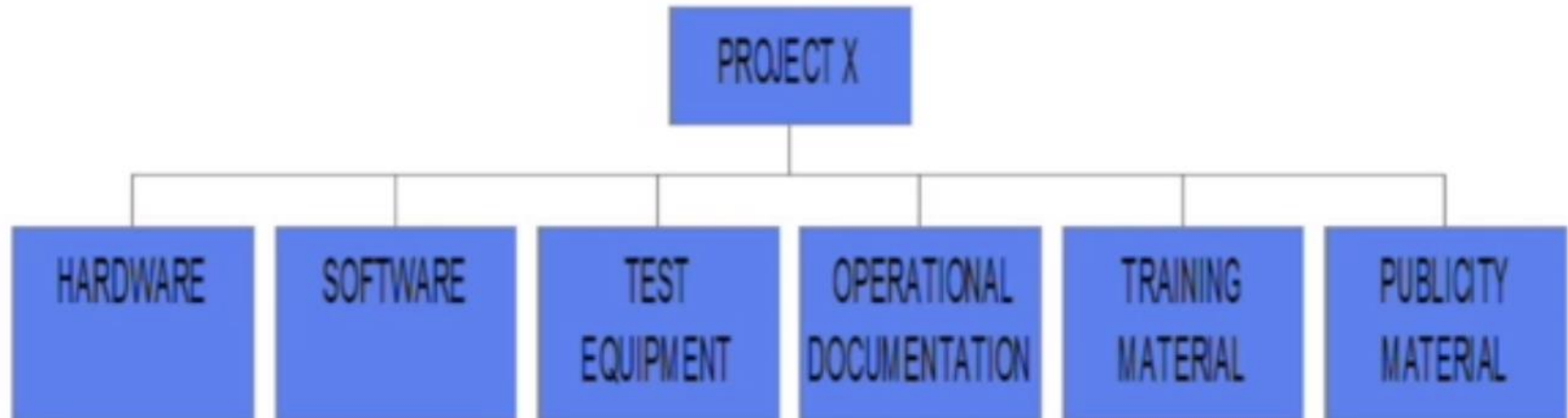
WBS based on organizational departments



WBS based on products

- A third way of representing the same project, would be to break it into its constituent products.
- This is particularly useful for complex or modular projects.

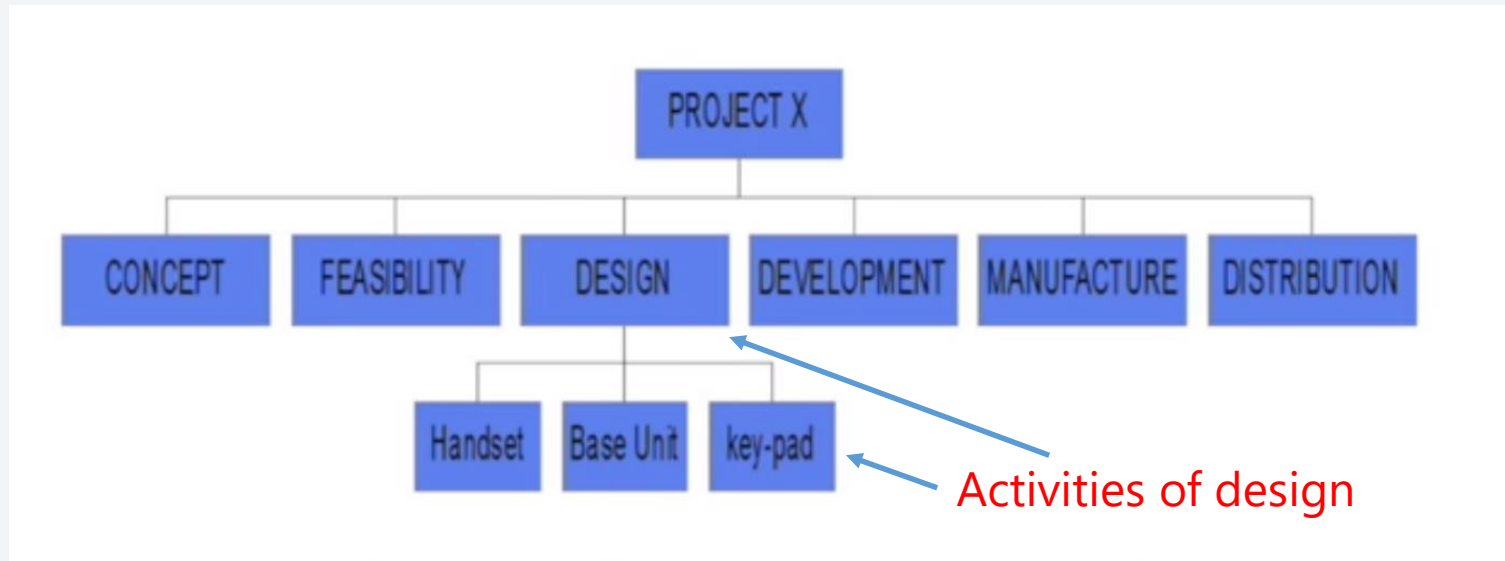
WBS based on products



Work Breakdown Structures (**WBS**)

- Any combination of these three methods can be used
 - Phase
 - Organization
 - Product
 - Any combination of the above 3
- whichever method is used, work packages can THEN be broken down into activities.

Work Breakdown Structures (WBS)

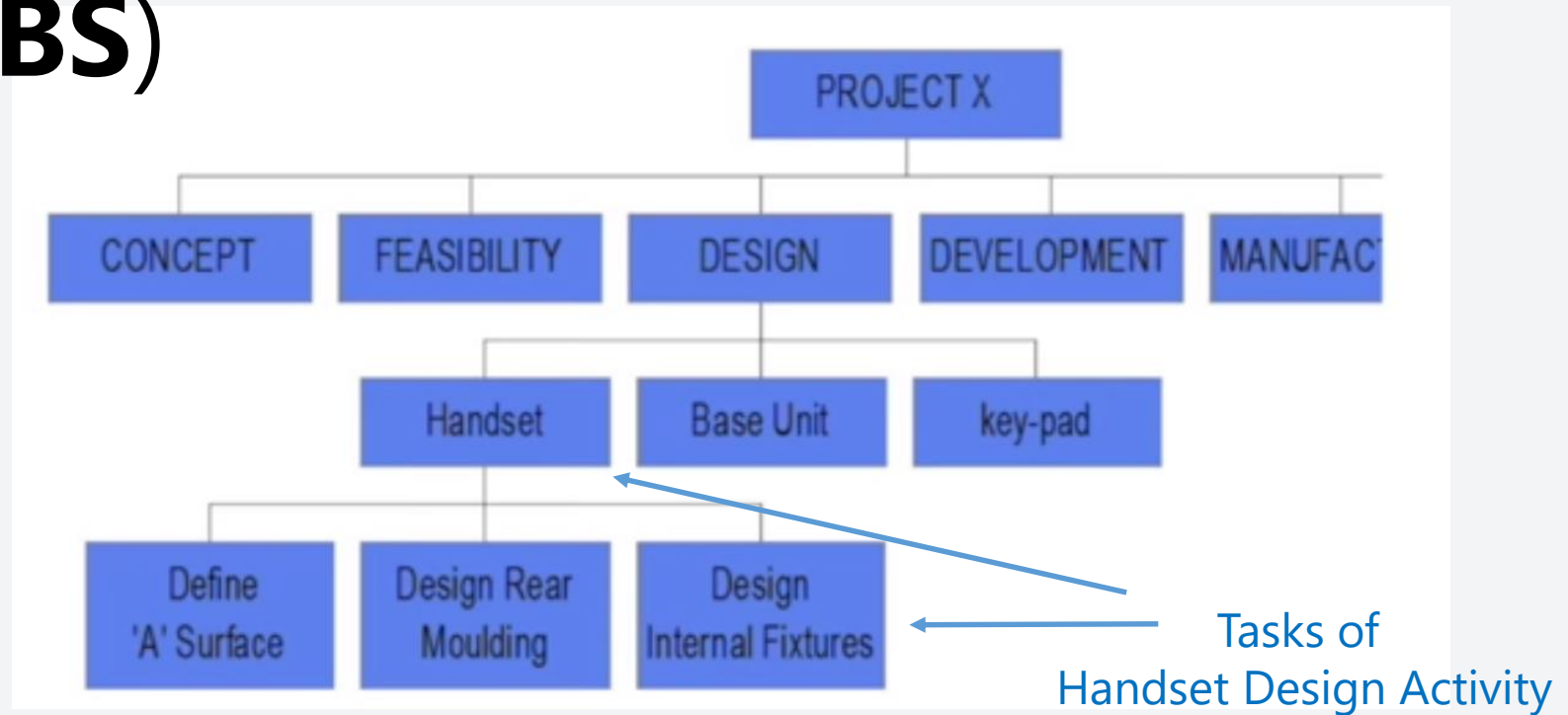


Level of detail → For operational plan

Work Breakdown Structures (**WBS**)

- Activities can then be broken **into tasks**
- If possible make these a "verb-noun" composition
 - Create detailed chassis drawing
 - Establish component costs
 - Test suspension components

Work Breakdown Structures (WBS)

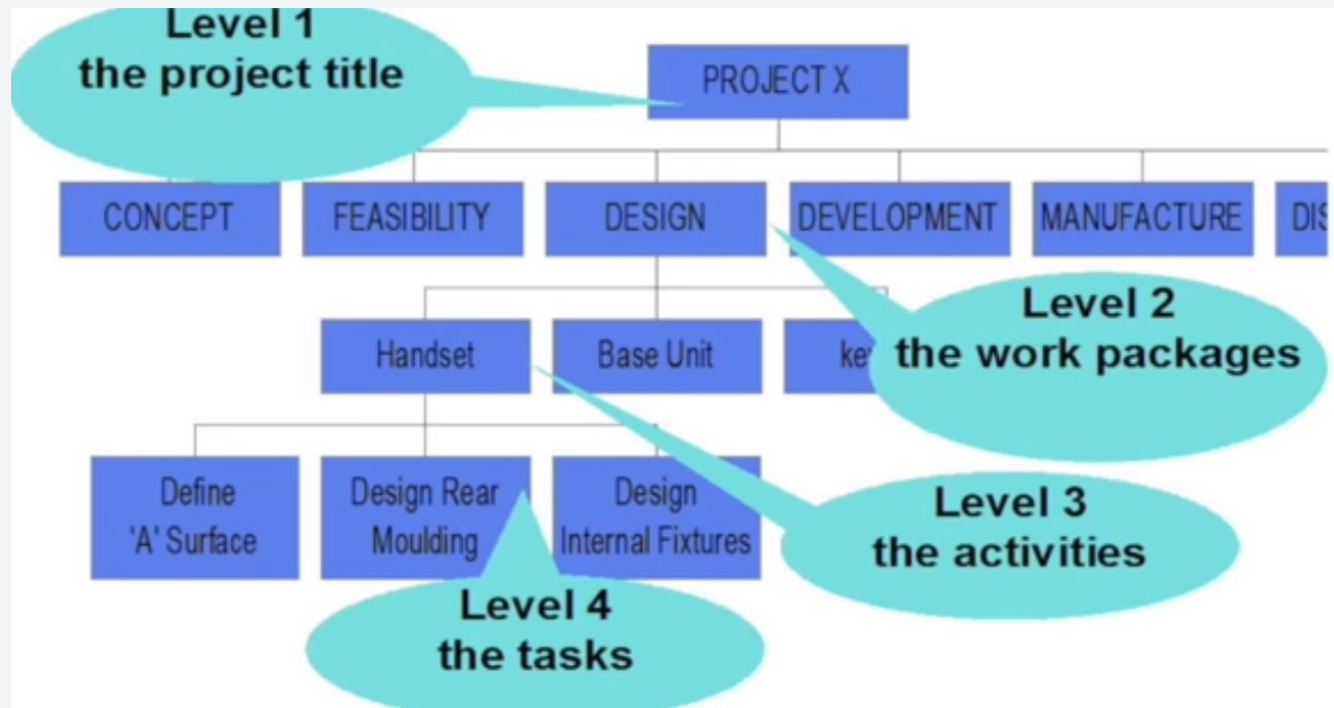


Level of detail → For day-to-day plan

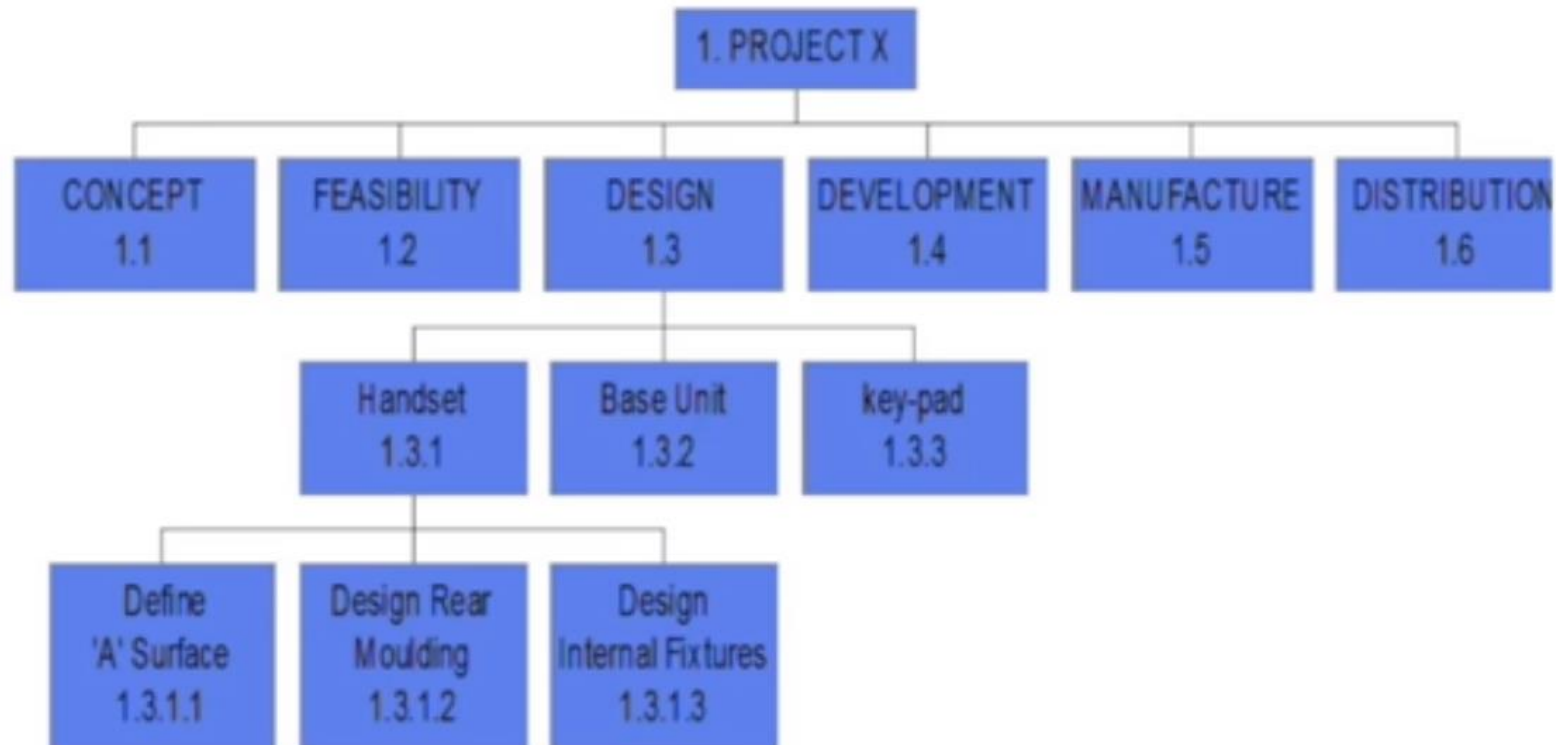
Work Breakdown Structures (**WBS**)

- At least, we have 4 levels in the WBS
 - The WBS needs to be as detailed as the project requires.
 - Large-sized projects - more details
 - 5 or 6 levels
 - 1000s of tasks

Work Breakdown Structures (WBS)



If too many levels → Tools are required →
Auto numbering



Milestones

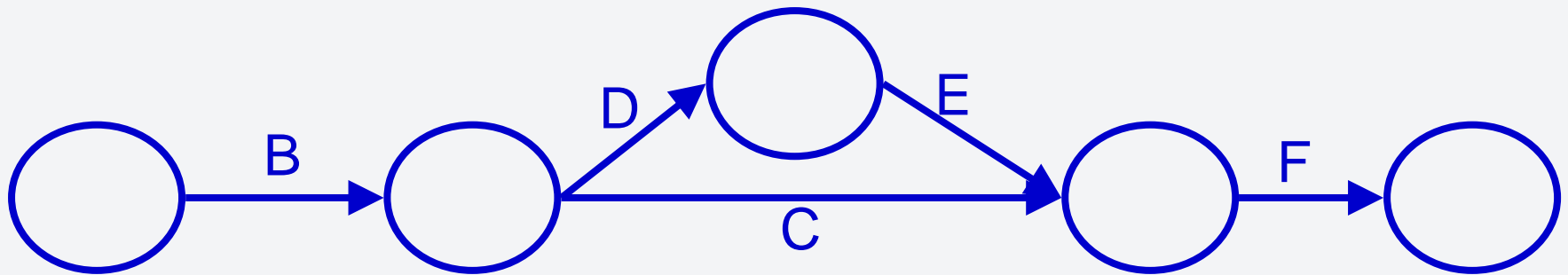
- Milestones are activities with 0 (very short) duration.
 - Approval meeting
- They are important events in the project
 - Completion of deliverables
 - One way of creating them is to use the end of the WBS work packages
 - Key decision points
- Can be used as a communication tools

How to draw a PDM network diagram?

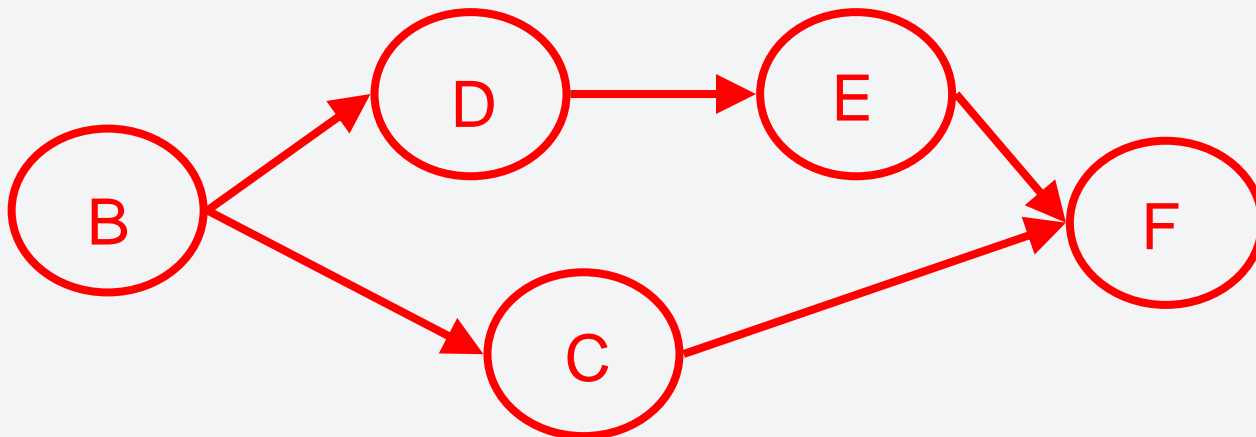
How to draw a PDM network diagram?

- Let us draw a PDM network diagram for a project.
 - Precedence Diagramming Method (**PDM**) Network Diagram
 - A kind of **flowchart** which shows which **activity/task** is leading the **other activities/tasks**.

AOA Vs. AON



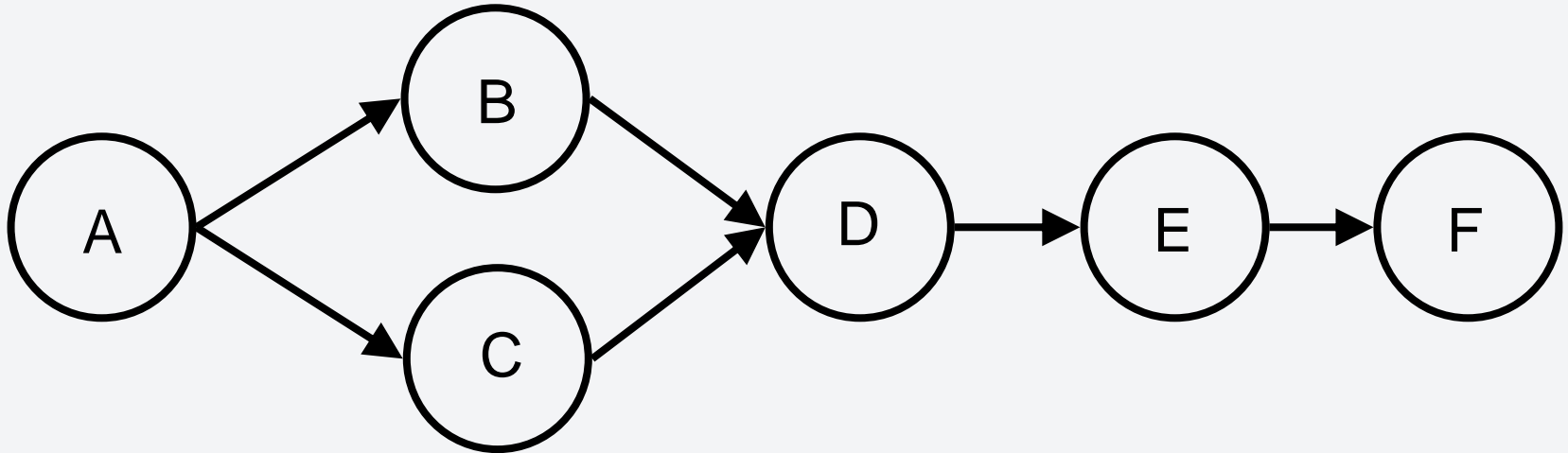
...and activities on node.



The same mini-project is shown with activities on arc...

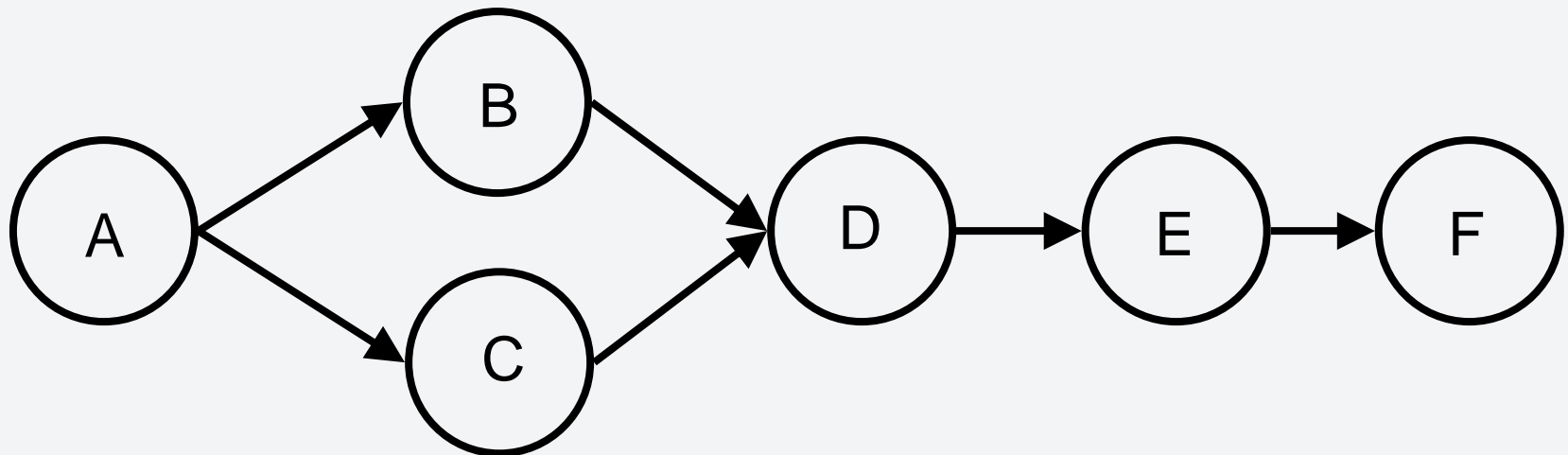
Project Scheduling Terms

- Successors
- Predecessors
- Network diagram
- Serial activities
- Concurrent activities


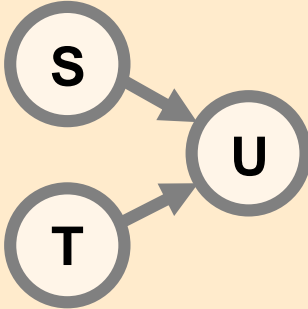


Project Scheduling Terms

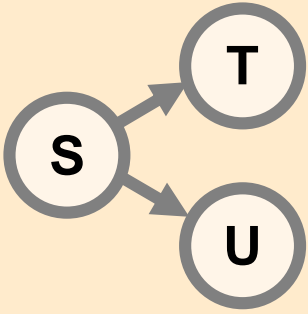
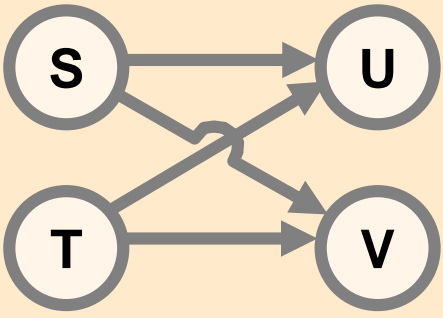
- Merge activities
- Burst activities
- Node
- Path
- Critical Path



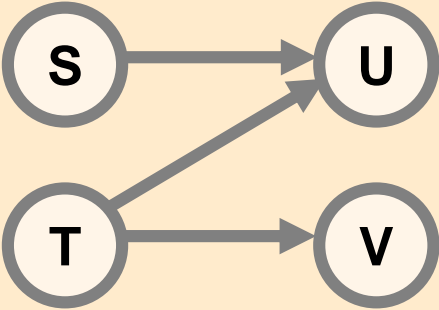
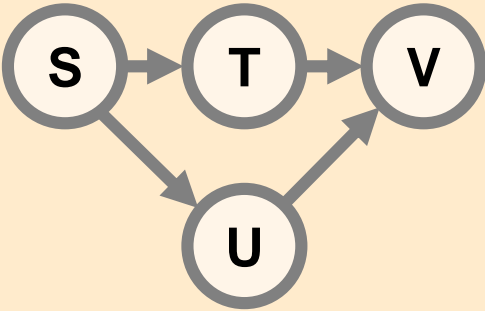
Diagramming the Network

AON	Activity Relationships
 <pre>graph LR; S((S)) --> T((T)); T --> U((U));</pre>	<p>S precedes T, which precedes U.</p>
 <pre>graph LR; S((S)) --> U((U)); T((T)) --> U;</pre>	<p>S and T must be completed before U can be started.</p>

Diagramming the Network

AON	Activity Relationships
 <pre>graph LR; S((S)) --> T((T)); S((S)) --> U((U));</pre>	<p>T and U cannot begin until S has been completed.</p>
 <pre>graph LR; S((S)) --> U((U)); S((S)) --> V((V)); T((T)) --> U((U)); T((T)) --> V((V));</pre>	<p>U and V can't begin until both S and T have been completed.</p>

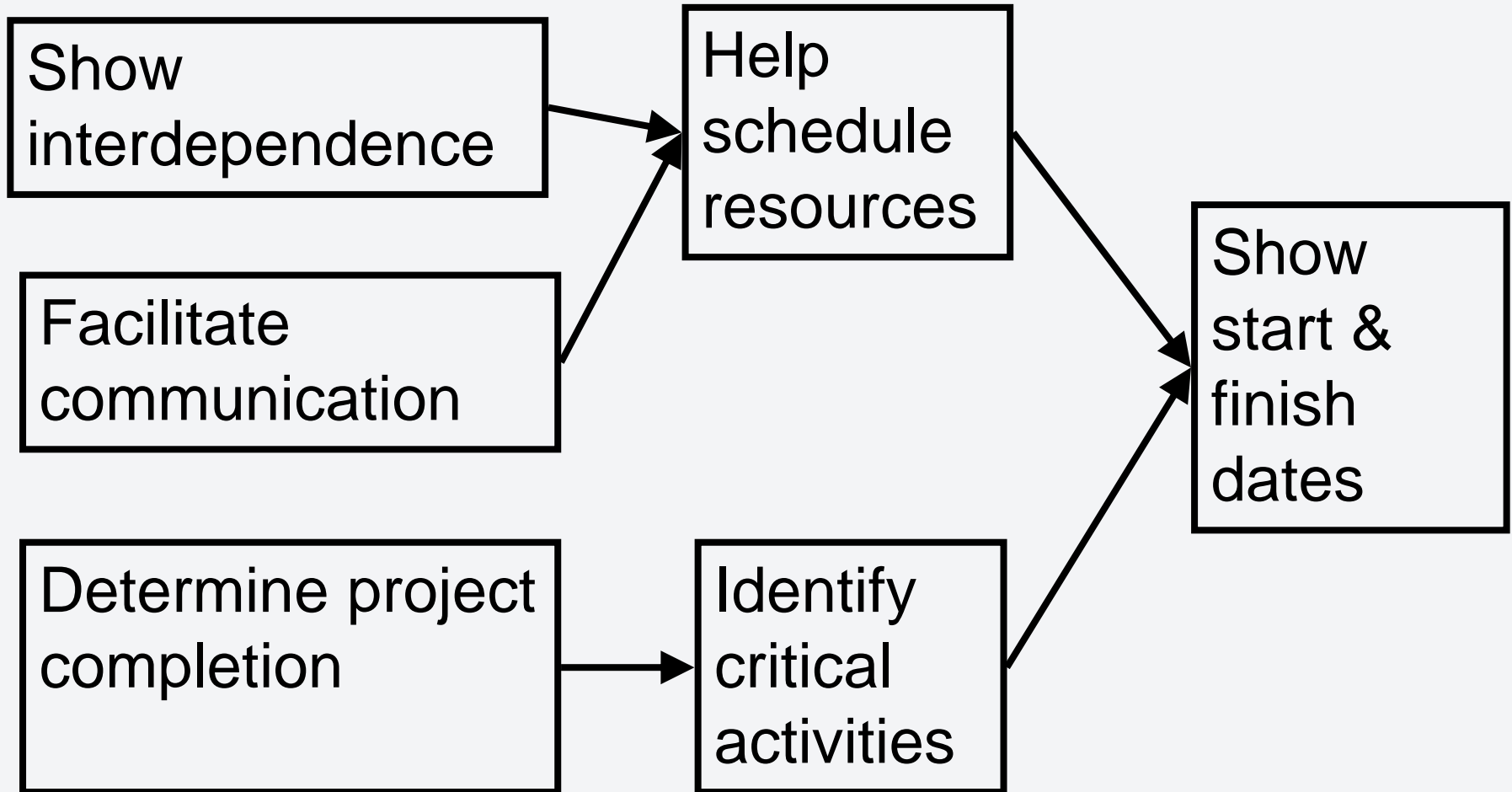
Diagramming the Network

AON	Activity Relationships
 <pre>graph LR; S((S)) --> U((U)); T((T)) --> U; T --> V((V));</pre>	<p>U cannot begin until both S and T have been completed; V cannot begin until T has been completed.</p>
 <pre>graph LR; S((S)) --> T((T)); S --> U((U)); T --> V((V)); U --> V;</pre>	<p>T and U cannot begin until S has been completed and V cannot begin until both T and U have been completed.</p>

Critical Path

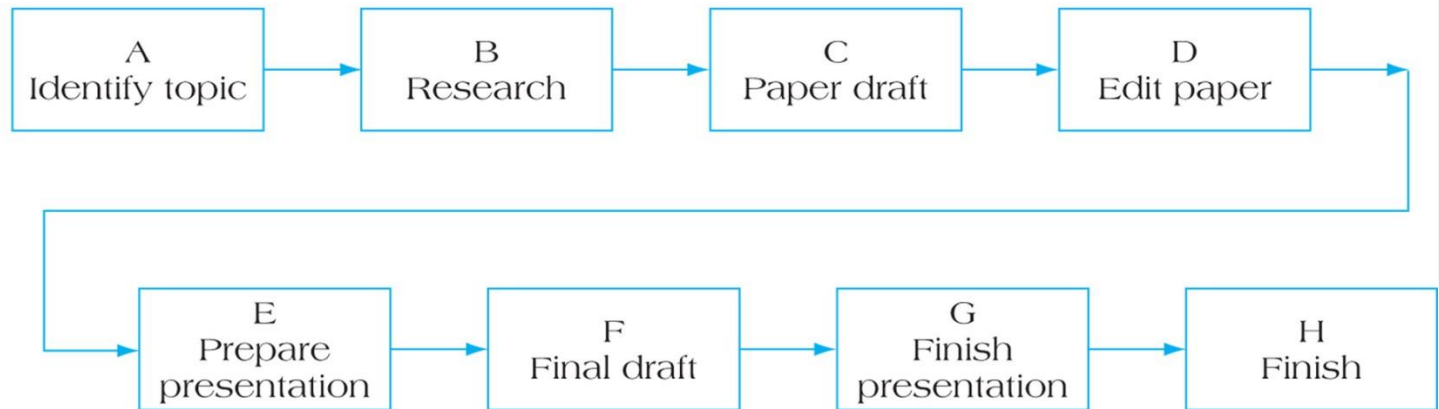
- The sequence of activities between a project's start and finish is a path
- **The critical path is the path that takes the longest time to complete**

Network Diagrams

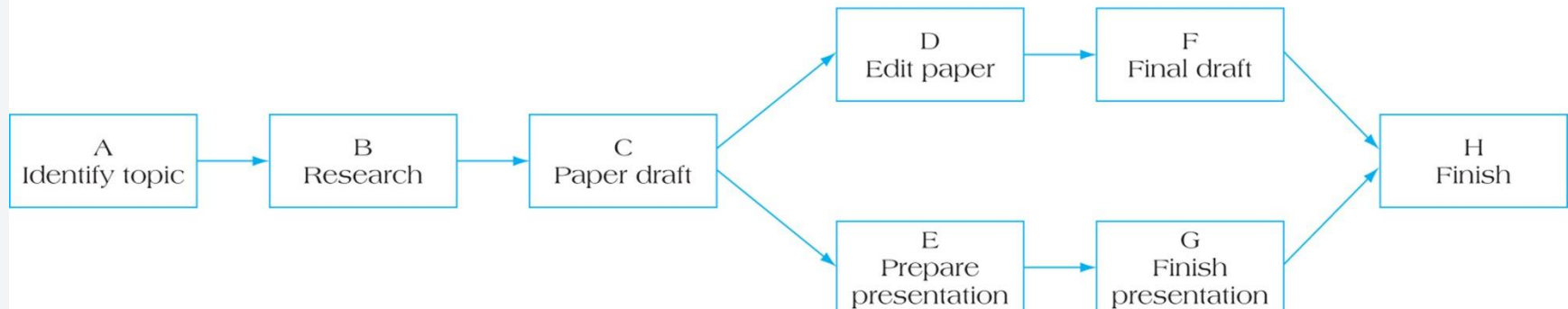


Network Diagrams

Option A: Serial Sequential Logic




Option B: Nonserial Sequential Logic



How to draw a PDM network diagram?

Task/Activity	Predecessor	Duration (Days)
A	-	5
B	A	4
C	A	5
D	B	6
E	C	3
F	D,E	4


How to draw a PDM network diagram?



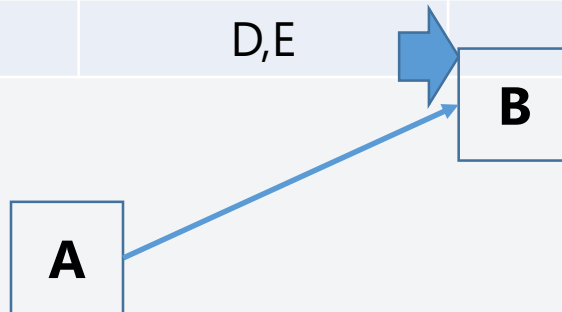
Task/Activity	Predecessor	Duration (Days)
A	-	5
B	A	4
C	A	5
D	B	6
E	C	3
F	D,E	4



How to draw a PDM network diagram?

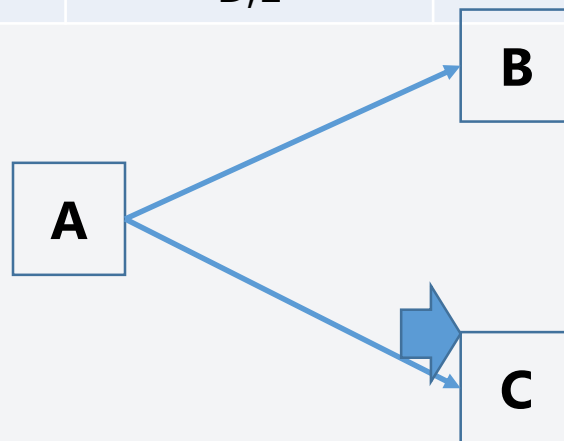


Task/Activity	Predecessor	Duration (Days)
A	-	5
B	A	4
C	A	5
D	B	6
E	C	3
F	D,E	4



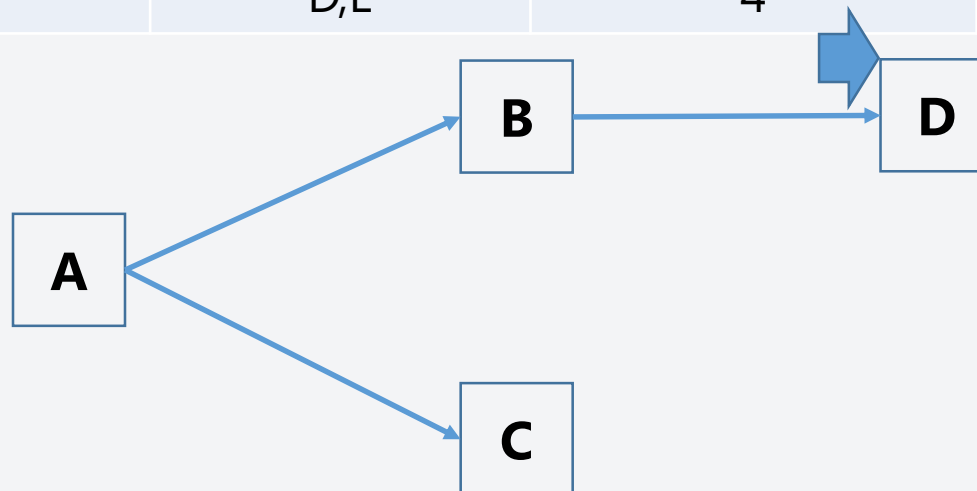
How to draw a PDM network diagram?

Task/Activity	Predecessor	Duration (Days)
A	-	5
B	A	4
C	A	5
D	B	6
E	C	3
F	D,E	4



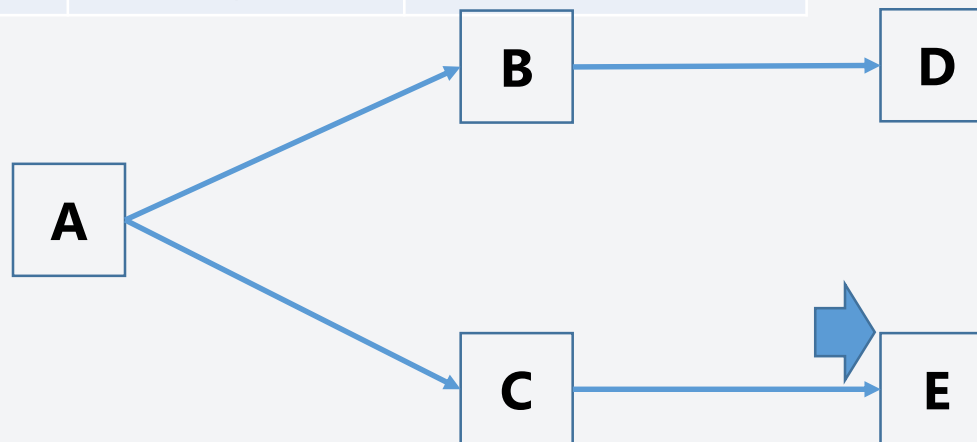
How to draw a PDM network diagram?

Task/Activity	Predecessor	Duration (Days)
A	-	5
B	A	4
C	A	5
D	B	6
E	C	3
F	D,E	4



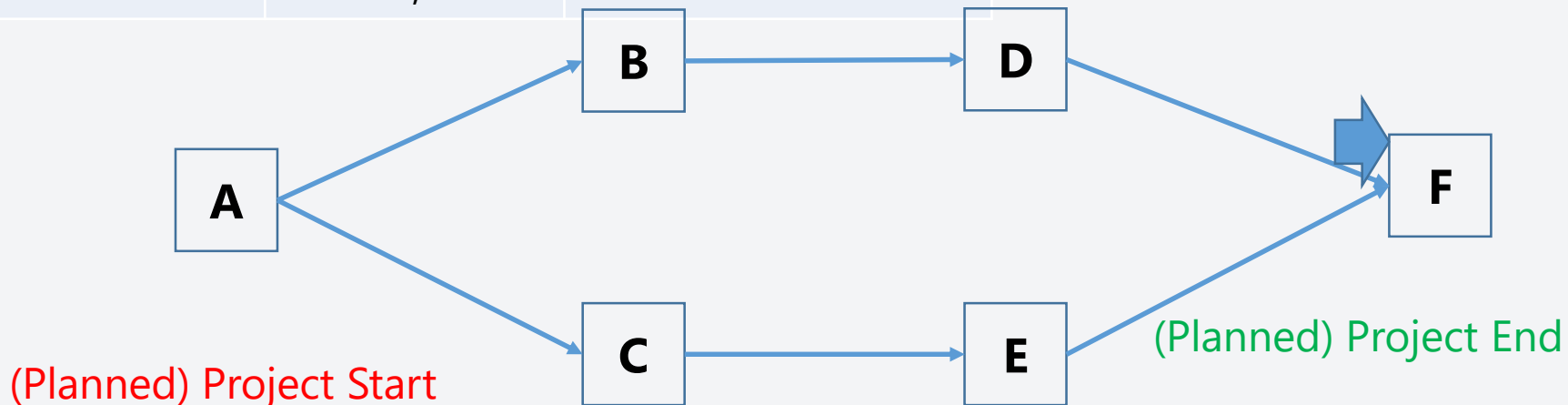
How to draw a PDM network diagram?

Task/Activity	Predecessor	Duration (Days)
A	-	5
B	A	4
C	A	5
D	B	6
E	C	3
F	D,E	4



How to draw a PDM network diagram?

Task/Activity	Predecessor	Duration (Days)
A	-	5
B	A	4
C	A	5
D	B	6
E	C	3
F	D,E	4



Late Start (LS) and Late Finish (LF) of activities in PDM network diagram

Task/Activity	Predecessor	Duration
...

ES	T/A	EF
LS	D	LF

- **T/A:** Task/Activity Name
- **D:** Duration
- **ES:** Early Start
- **EF:** Early Finish
- **LS:** Late Start
- **LF:** Late Finish → latest time any given T/A can start **without** increasing the duration!!

Constructing the Critical Path

- Forward pass – an **additive move** through the network from **start to finish**
- Backward pass – a **subtractive move** through the network from **finish to start**
- Critical path – the **longest path** from end to end which determines the **shortest project length**

Rules for Forward/Backward Pass

Forward Pass Rules (ES & EF)

- $ES + \text{Duration} = EF$
- $EF \text{ of predecessor} = ES \text{ of successor}$
- Largest preceding EF at a merge point becomes EF for successor

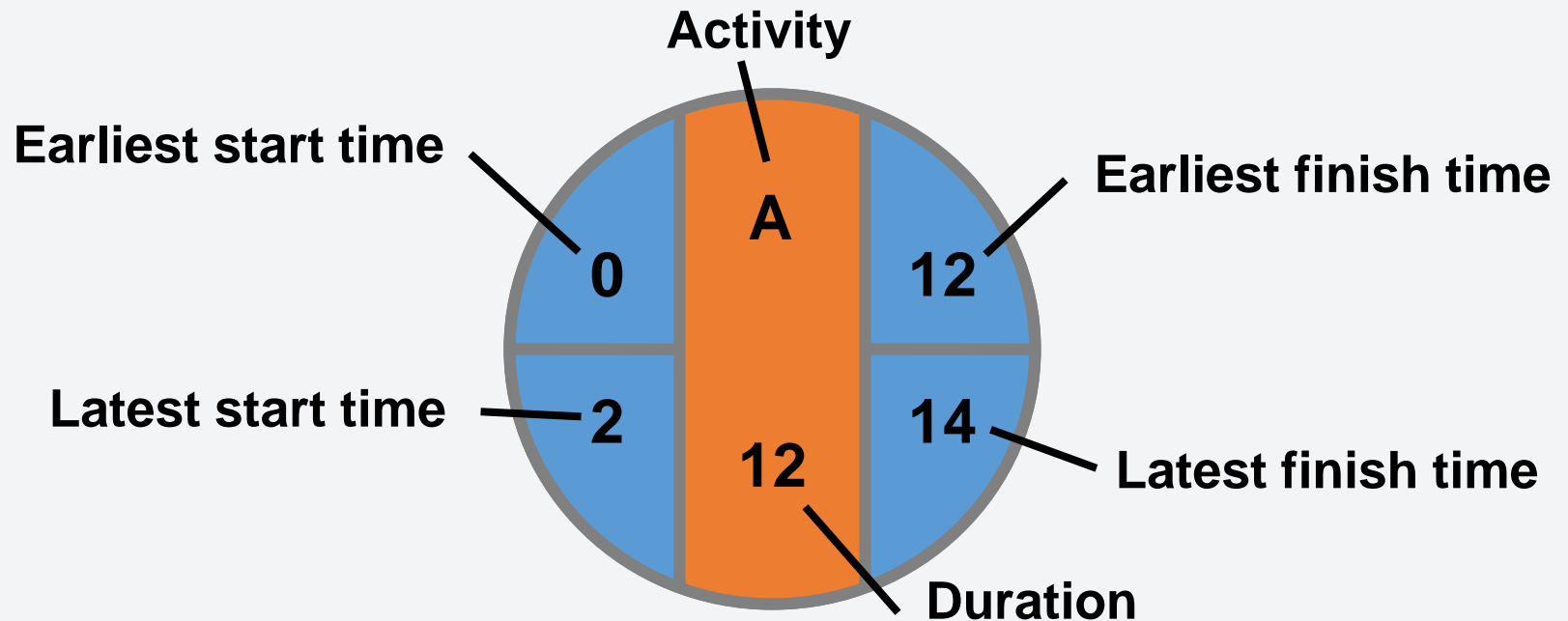
Backward Pass Rules (LS & LF)

- $LF - \text{Duration} = LS$
- $LS \text{ of successor} = LF \text{ of predecessor}$
- Smallest succeeding LS at a burst point becomes LF for predecessor

Node Labels - Alternative

Early Start	ID Number	Early Finish
Activity Float	Activity Descriptor	
Late Start	Activity Duration	Late Finish

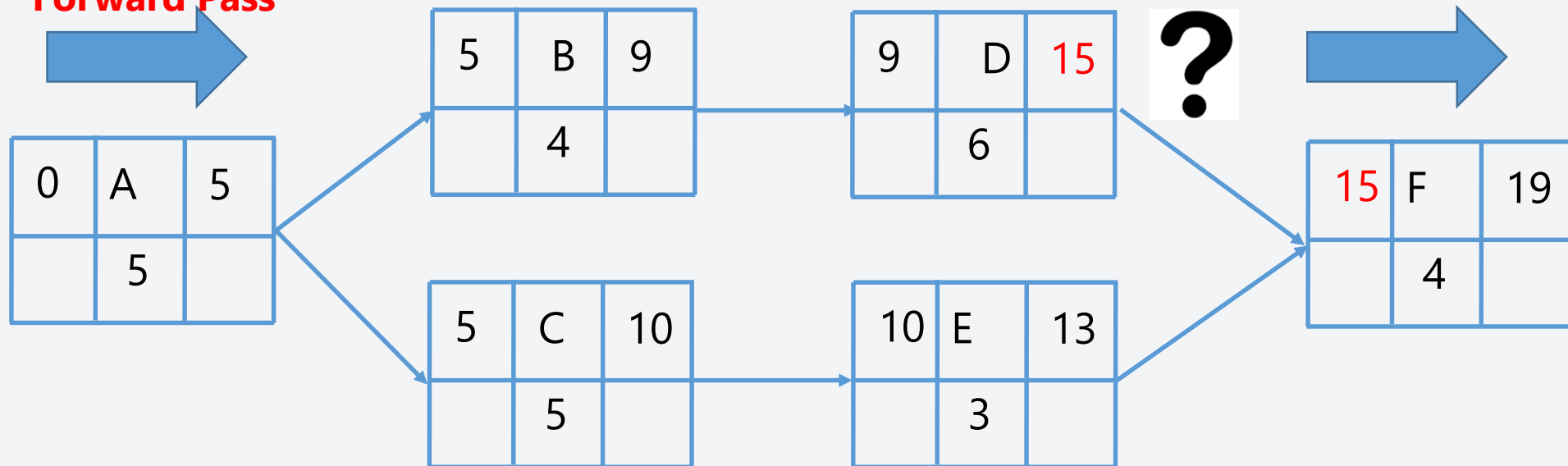
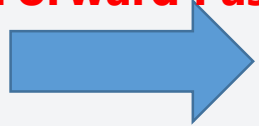
Node Labels - Alternative



Task/Activity	Predecessor	Duration (Days)
A	-	5
B	A	4
C	A	5
D	B	6
E	C	3
F	D,E	4

ES	T/A	EF
LS	D	LF

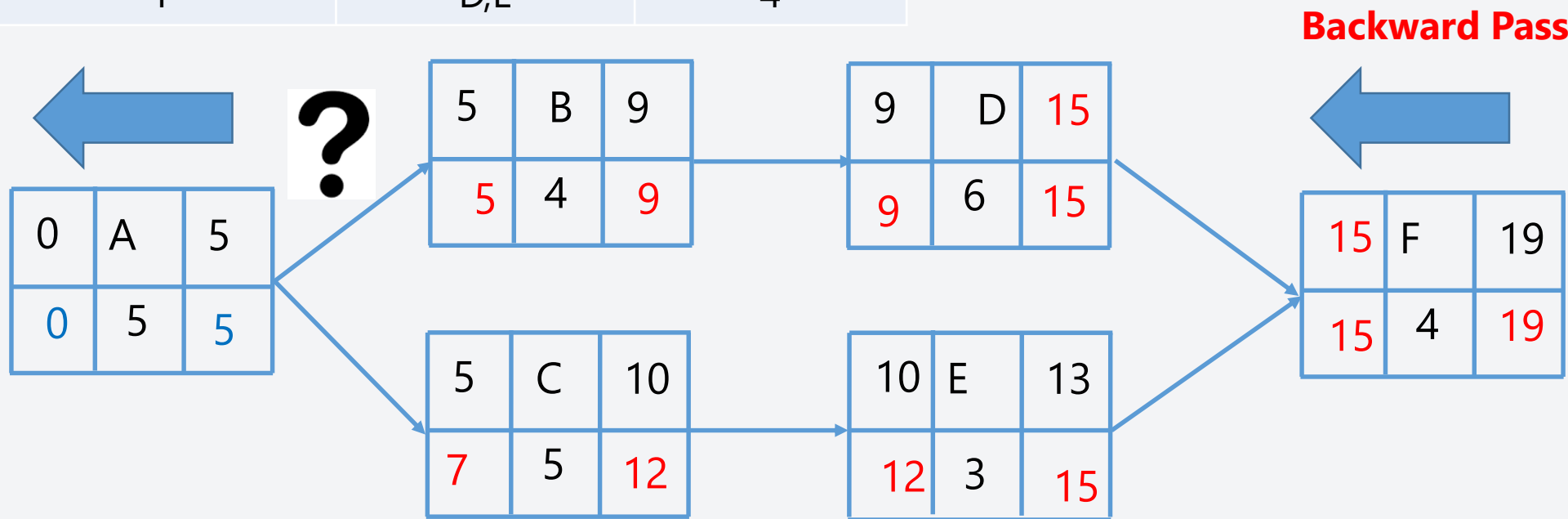
Forward Pass

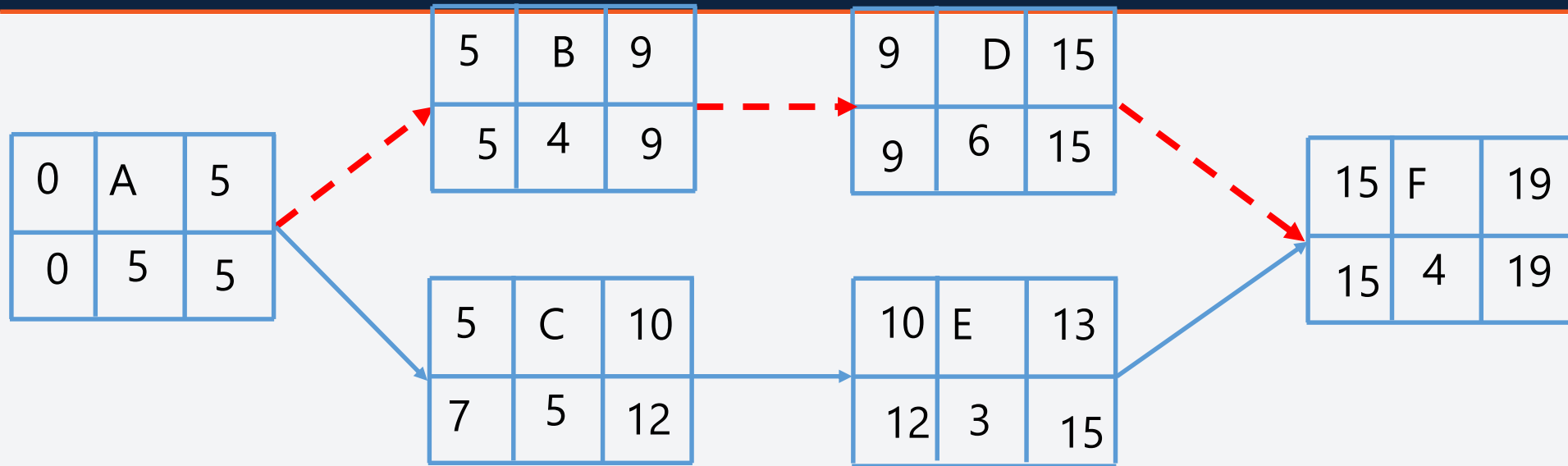


Time

Task/Activity	Predecessor	Duration (Days)
A	-	5
B	A	4
C	A	5
D	B	6
E	C	3
F	D,E	4

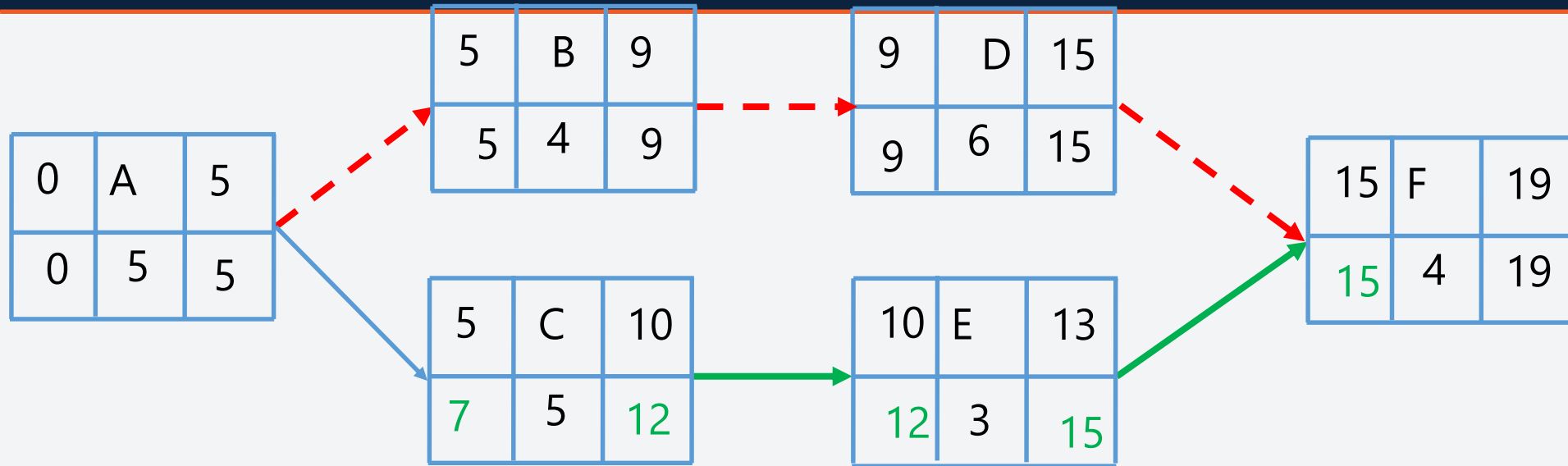
ES	T/A	EF
LS	D	LF





Critical Path

- If we were to delay (increase the duration of) any activity/task **on the critical path**, we'll have **increase** the **entire duration** of the project.



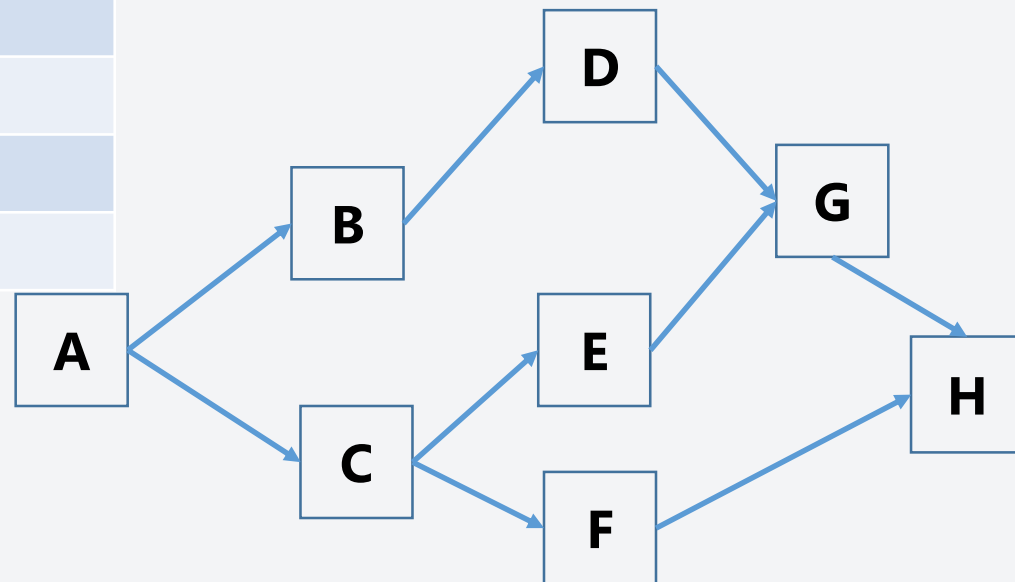
A Change in the Non-Critical Path

- **Ex:** If we delay activity/task C for 2 days, there will be no delays.
 - Start activity/task C on the 7th day not on the 5th day.

Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

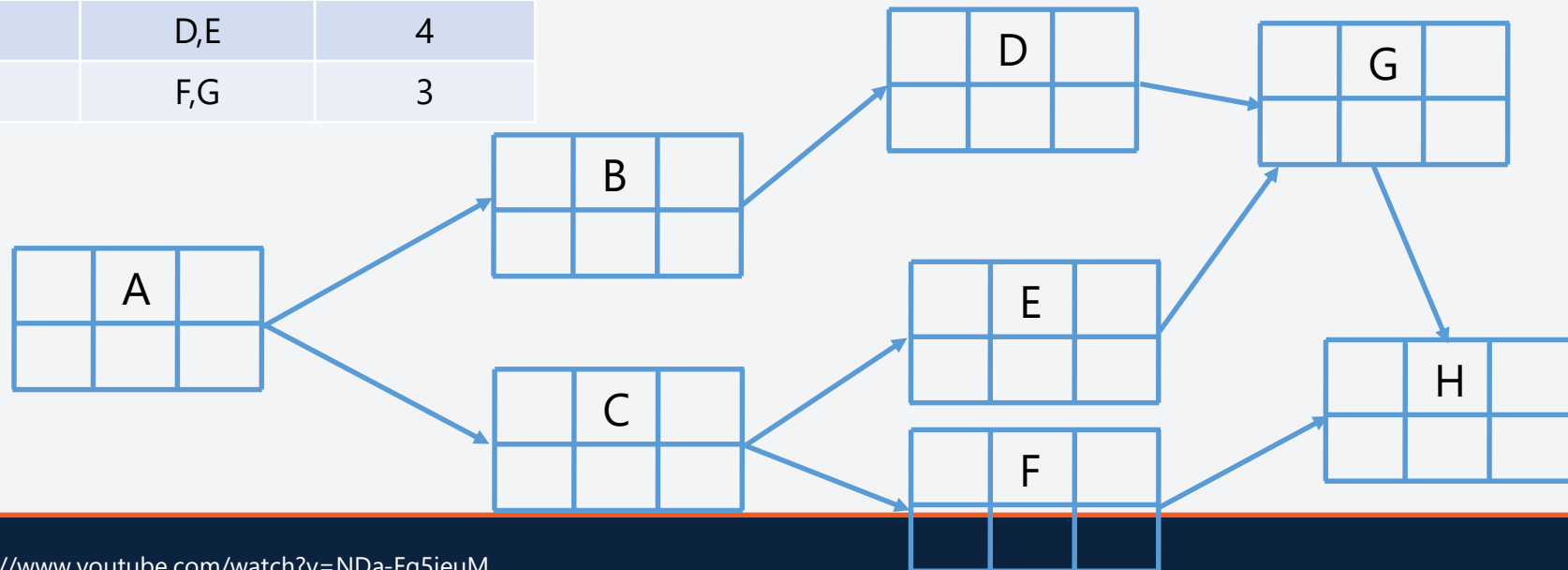
Example



Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

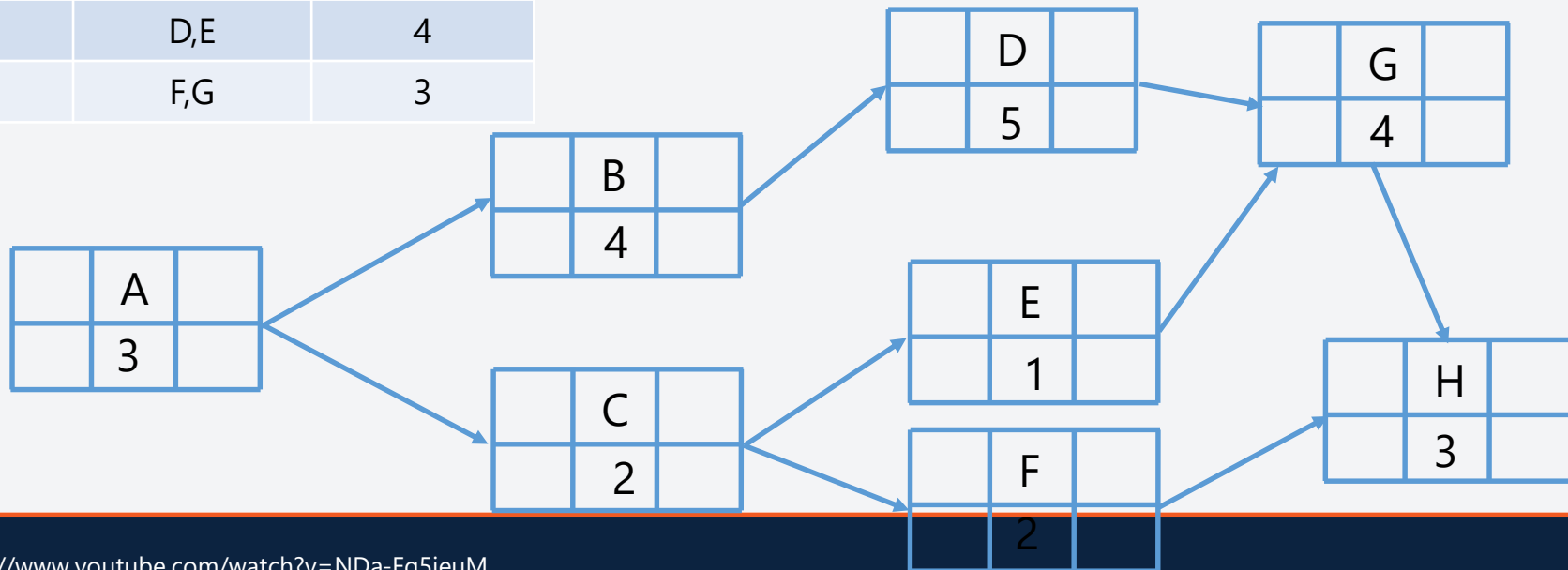
ES	T/A	EF
LS		LF



Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

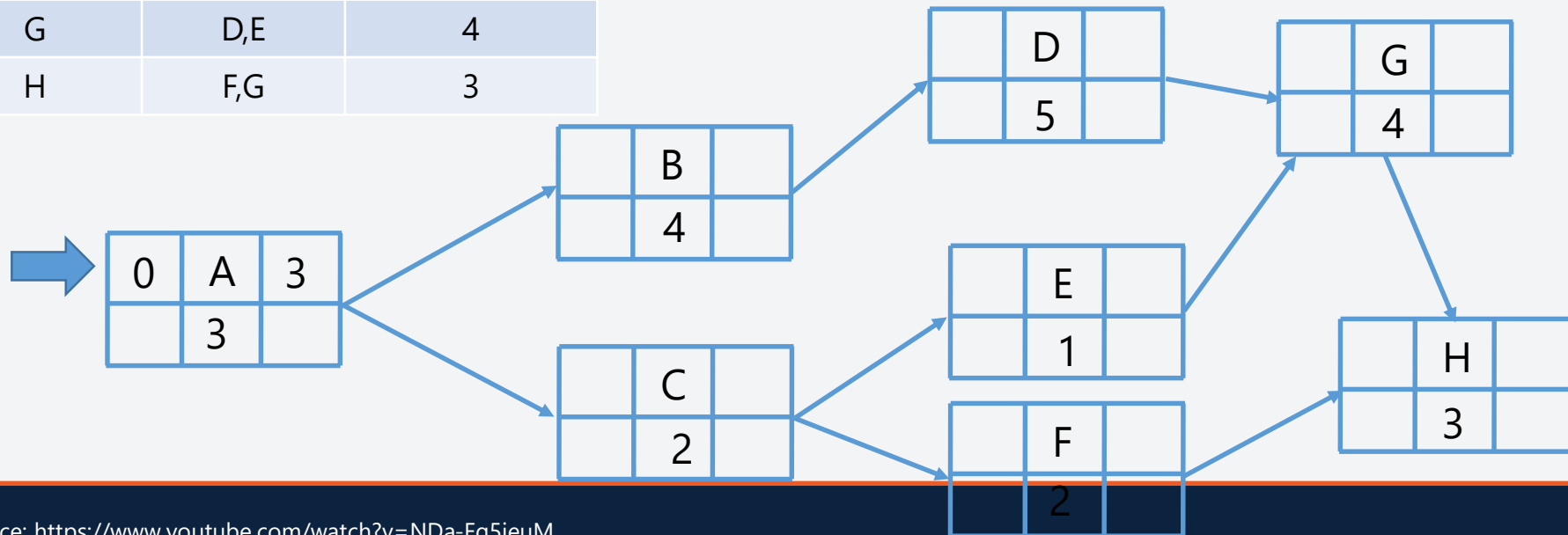
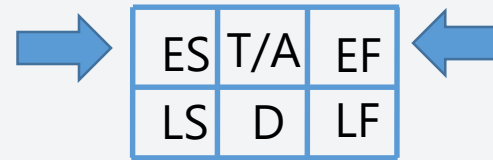
Task/Activity	Predecessor	Duration (Days)
A	-	3
B	A	4
C	A	2
D	B	5
E	C	1
F	C	2
G	D,E	4
H	F,G	3

ES	T/A	EF
LS	D	LF



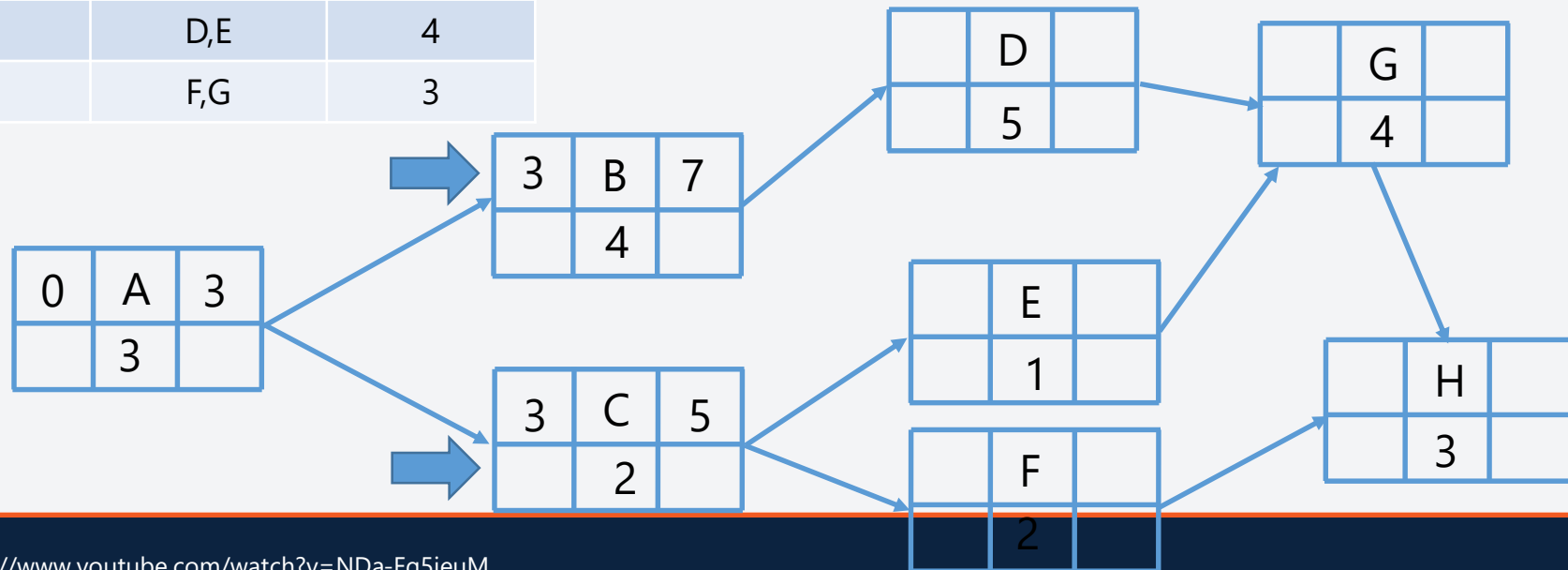
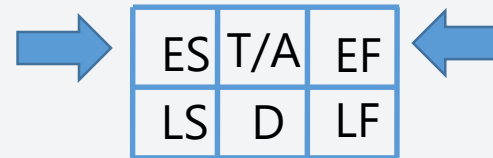
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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



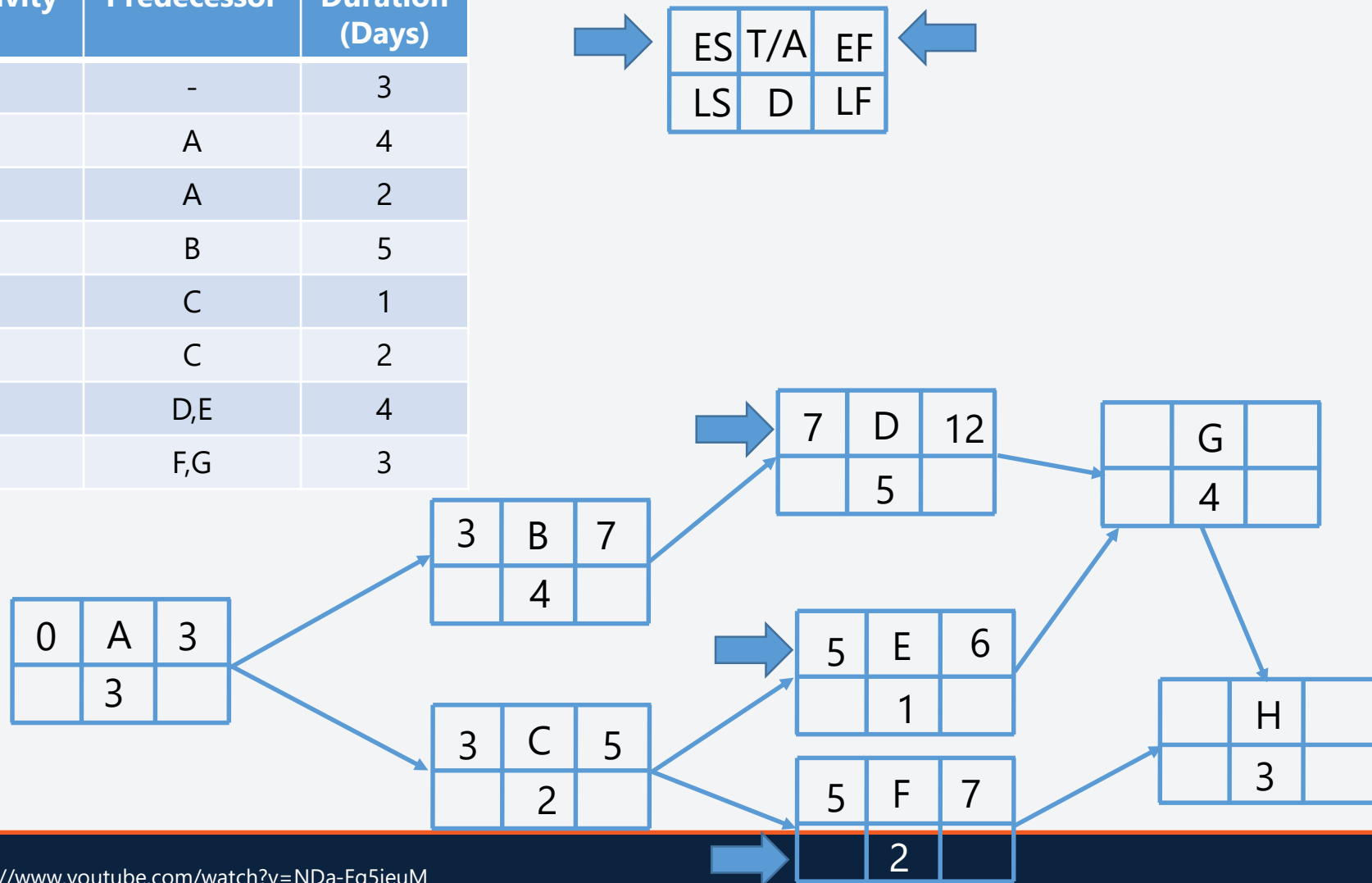
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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



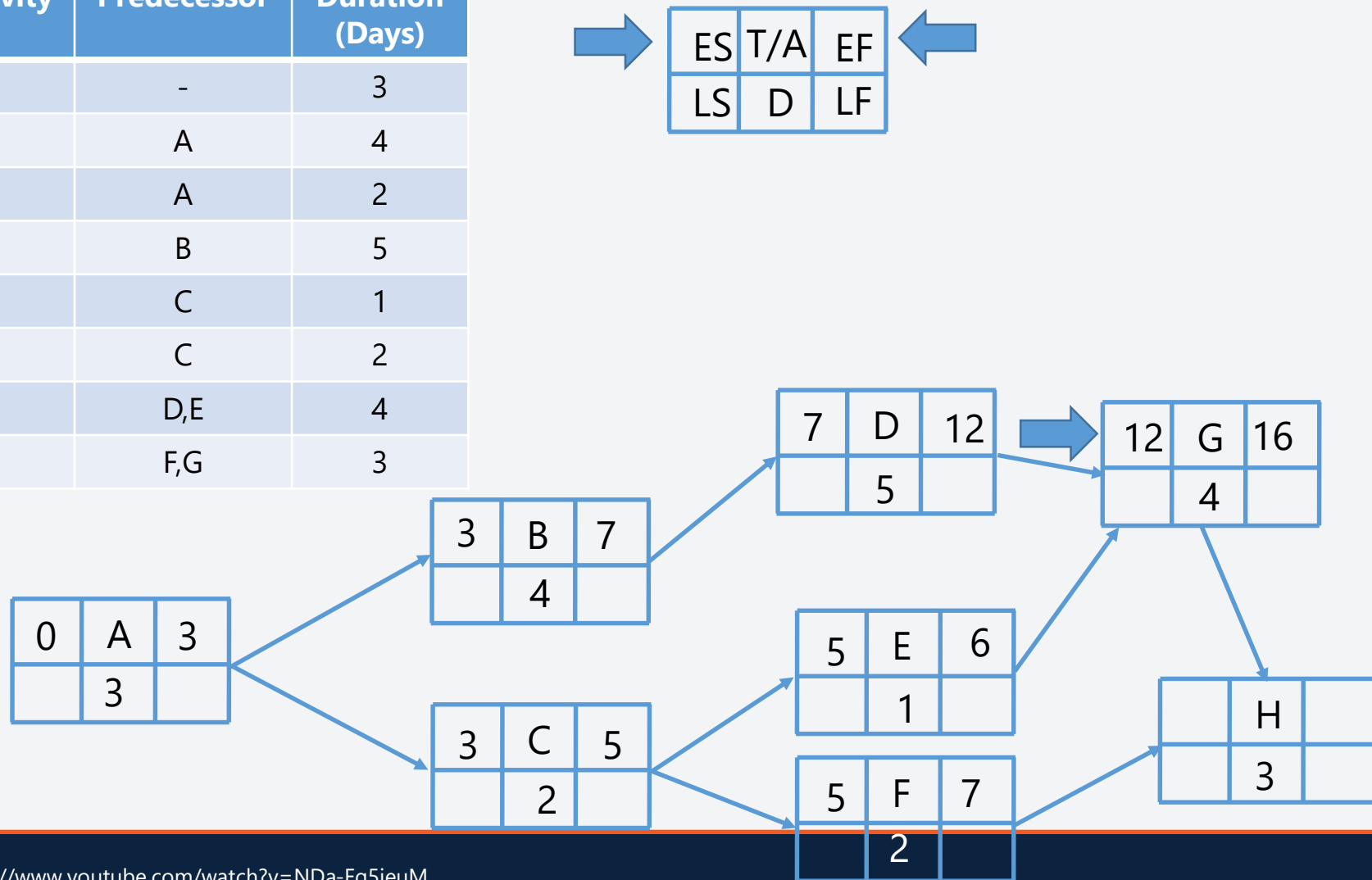
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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



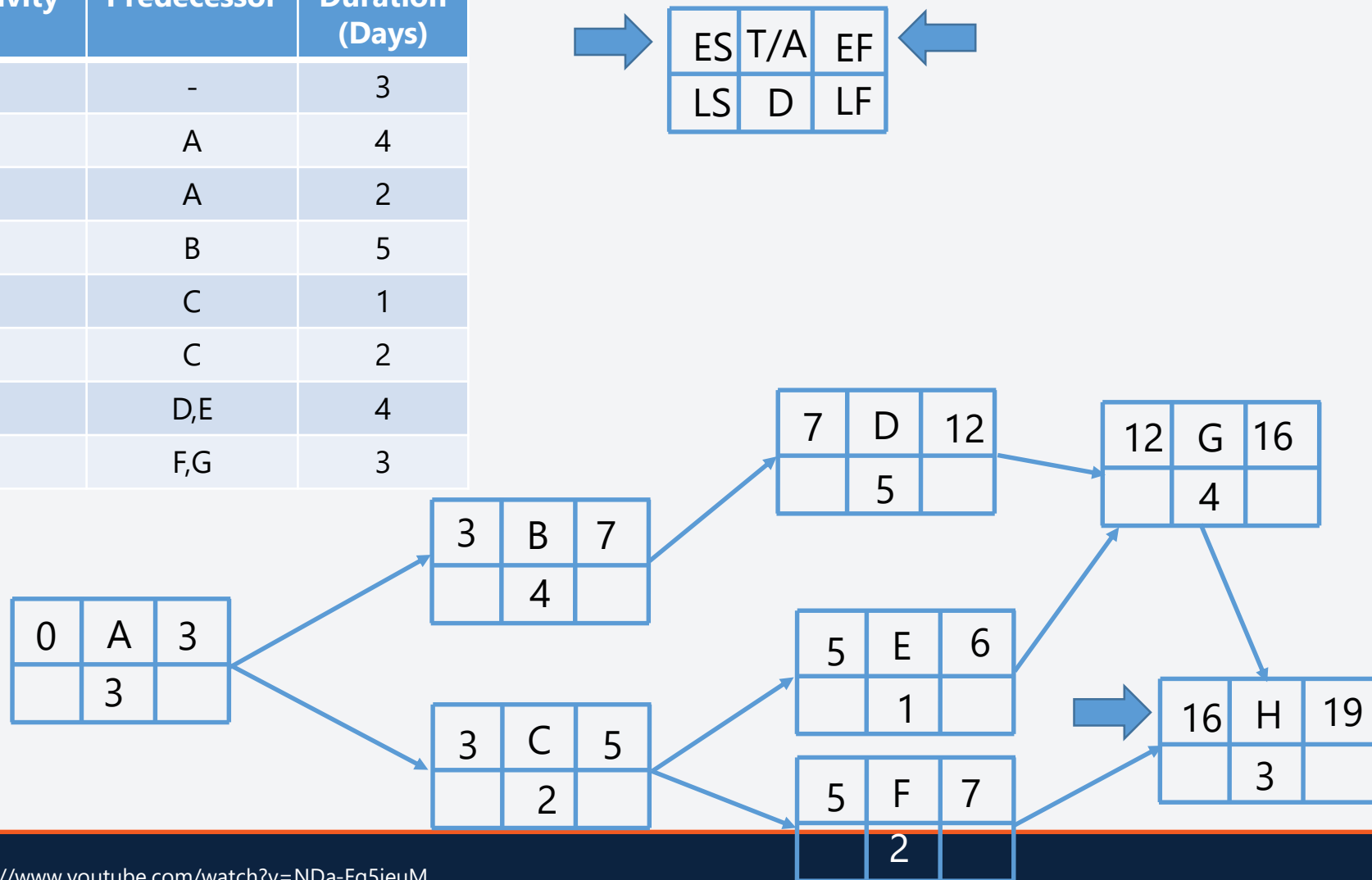
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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



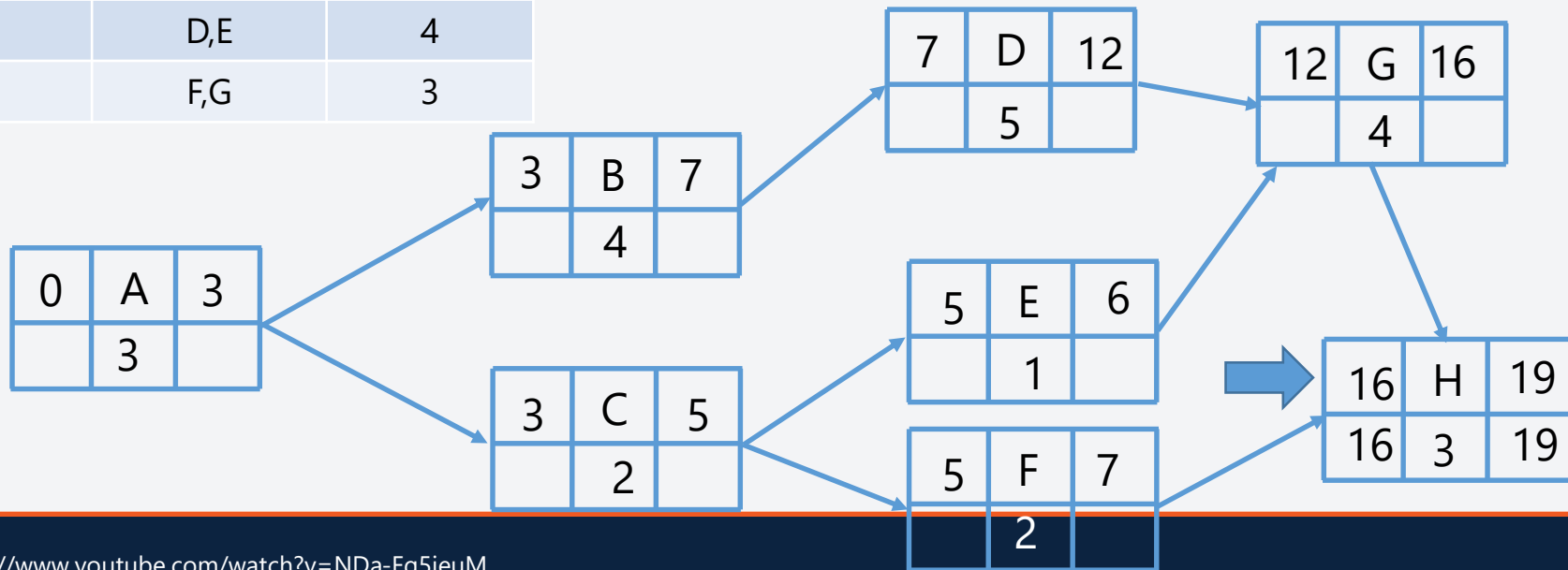
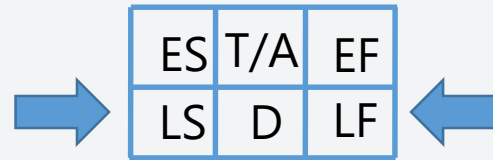
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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



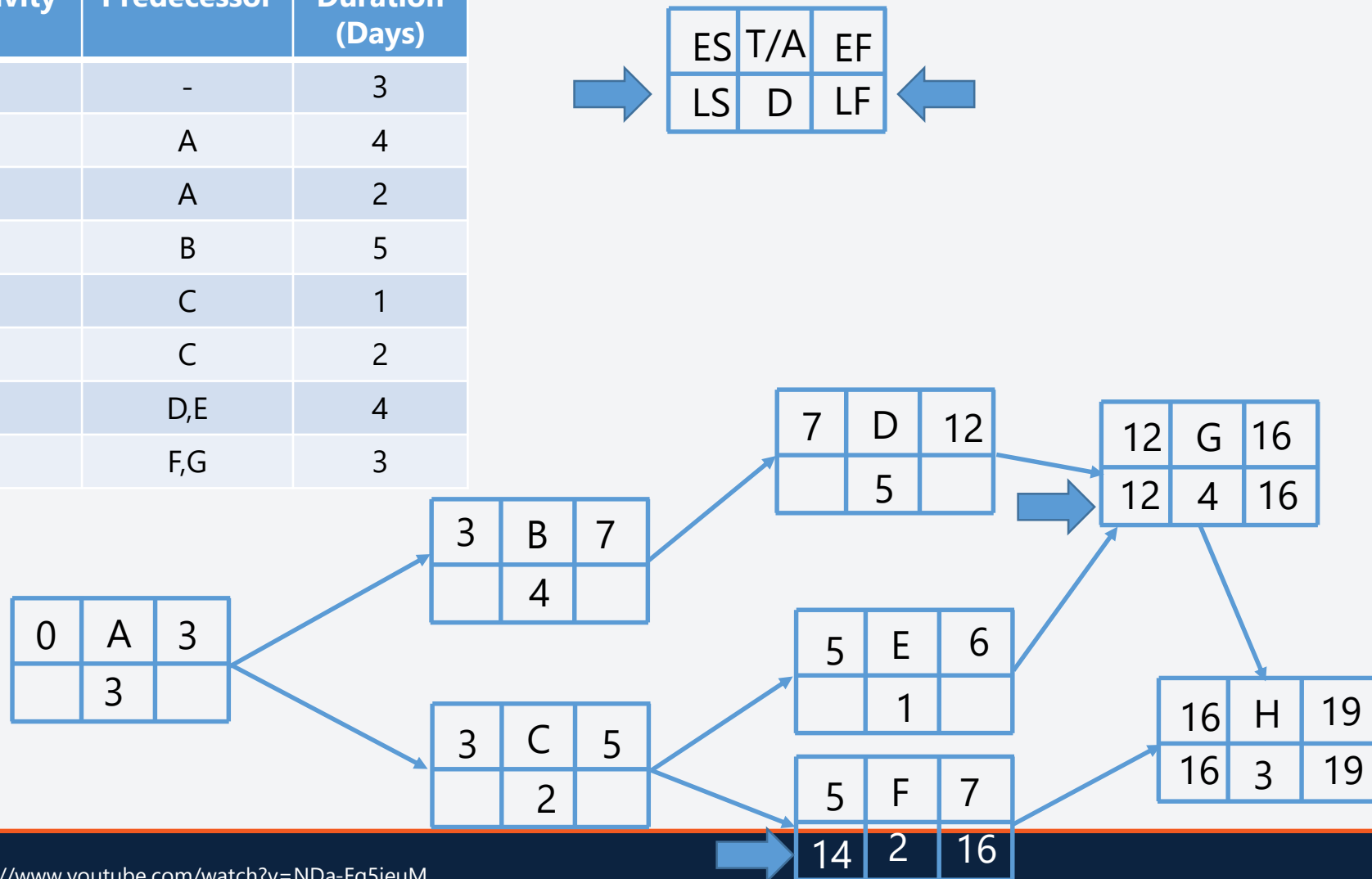
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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



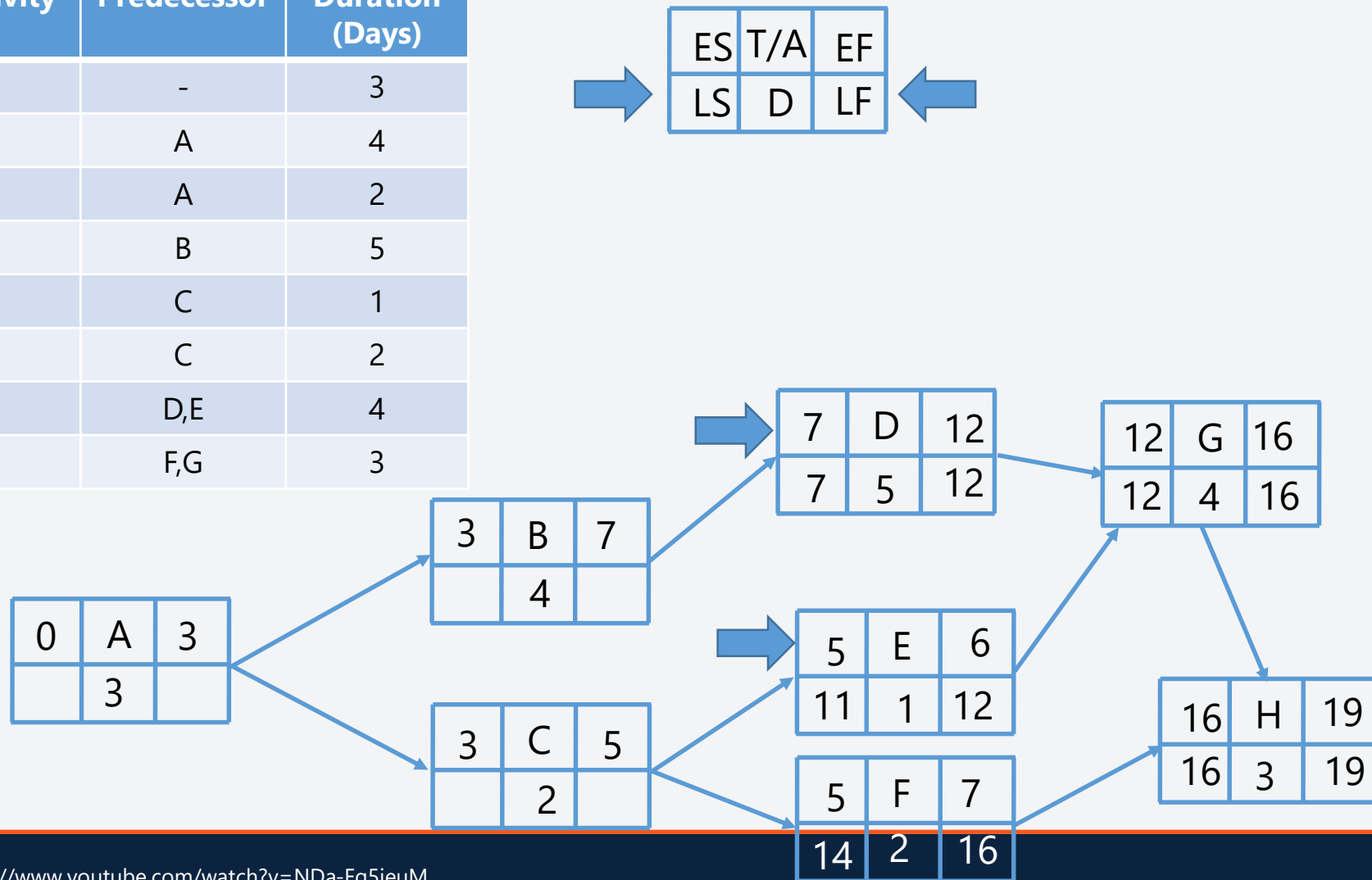
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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



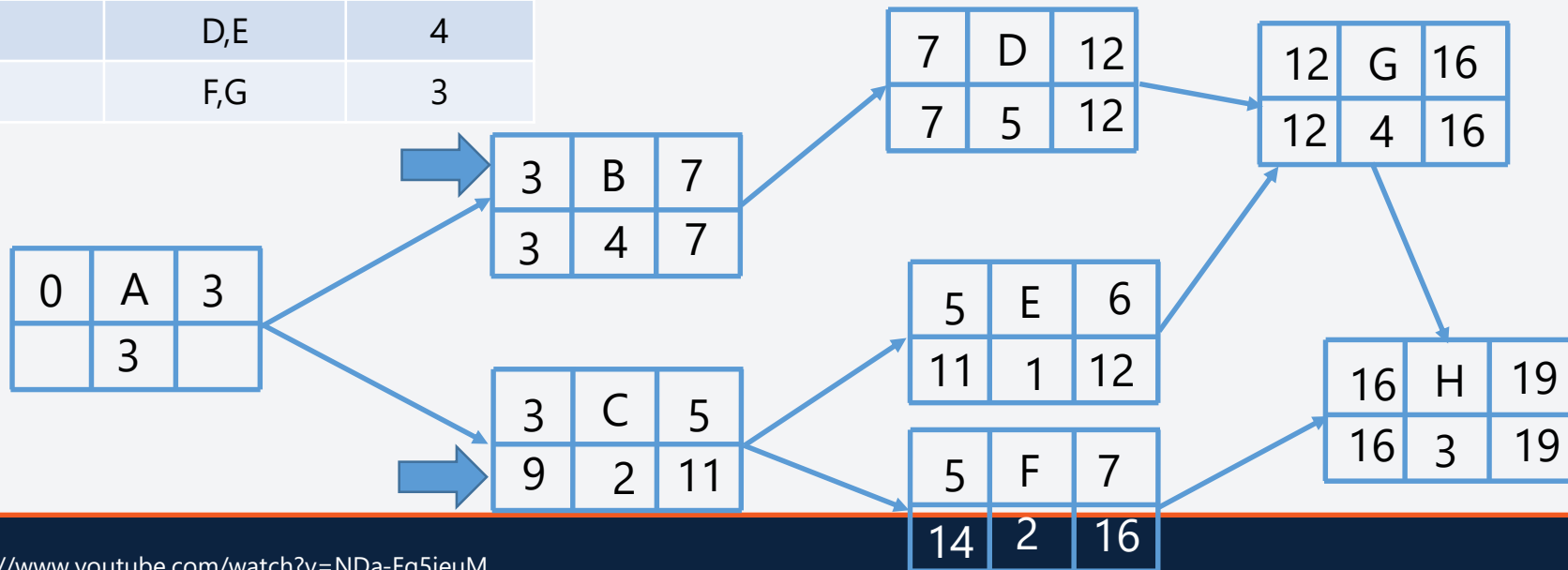
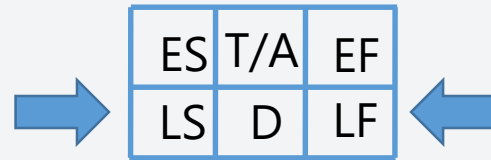
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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



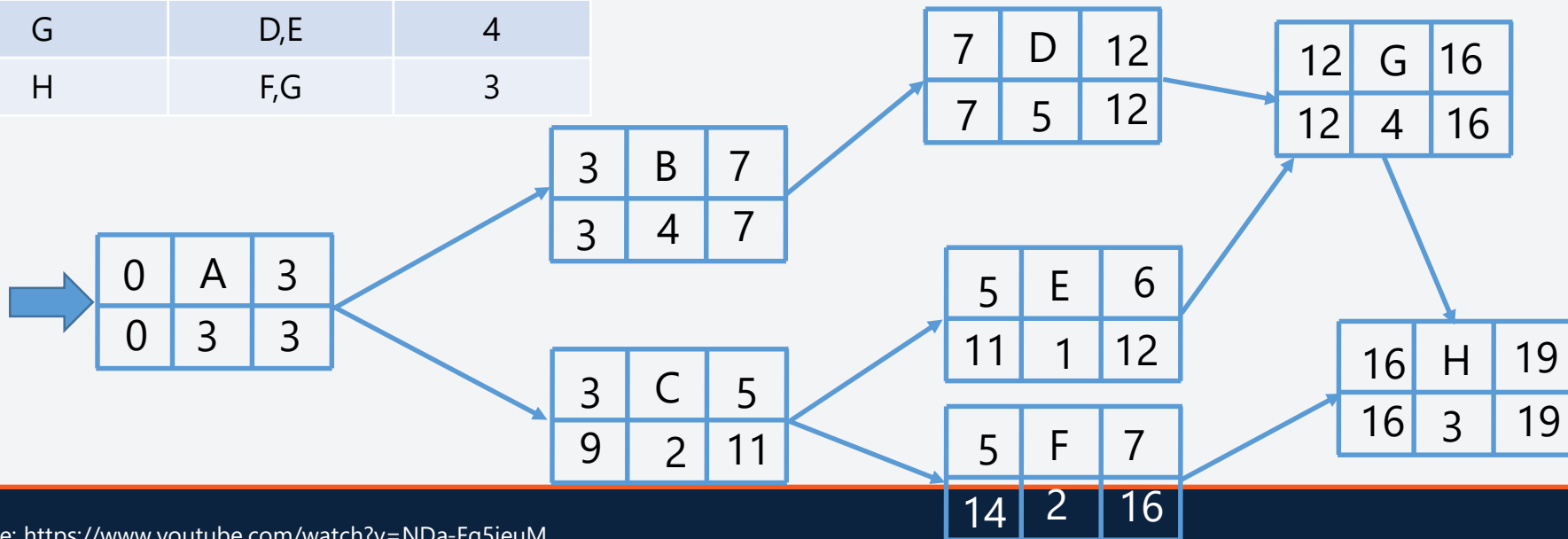
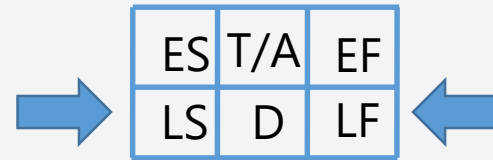
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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



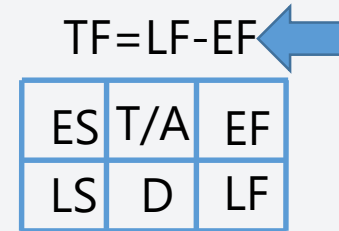
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

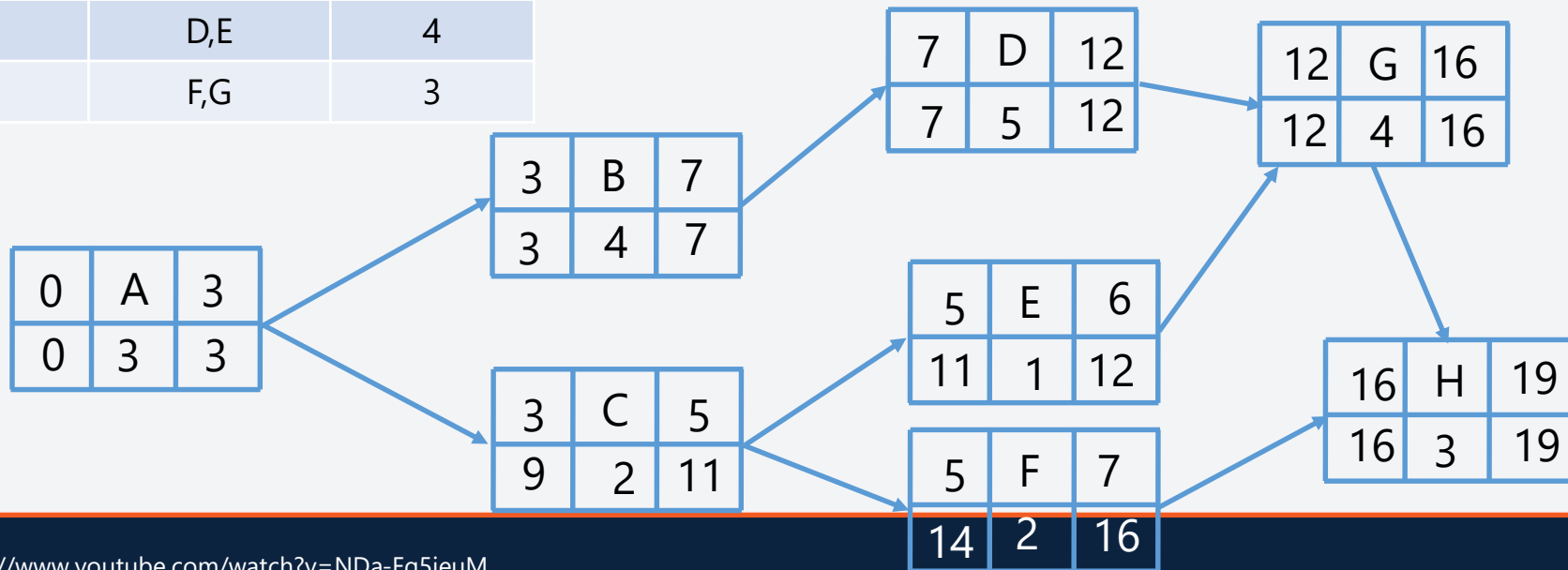


Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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



OR $TF = LS - ES$

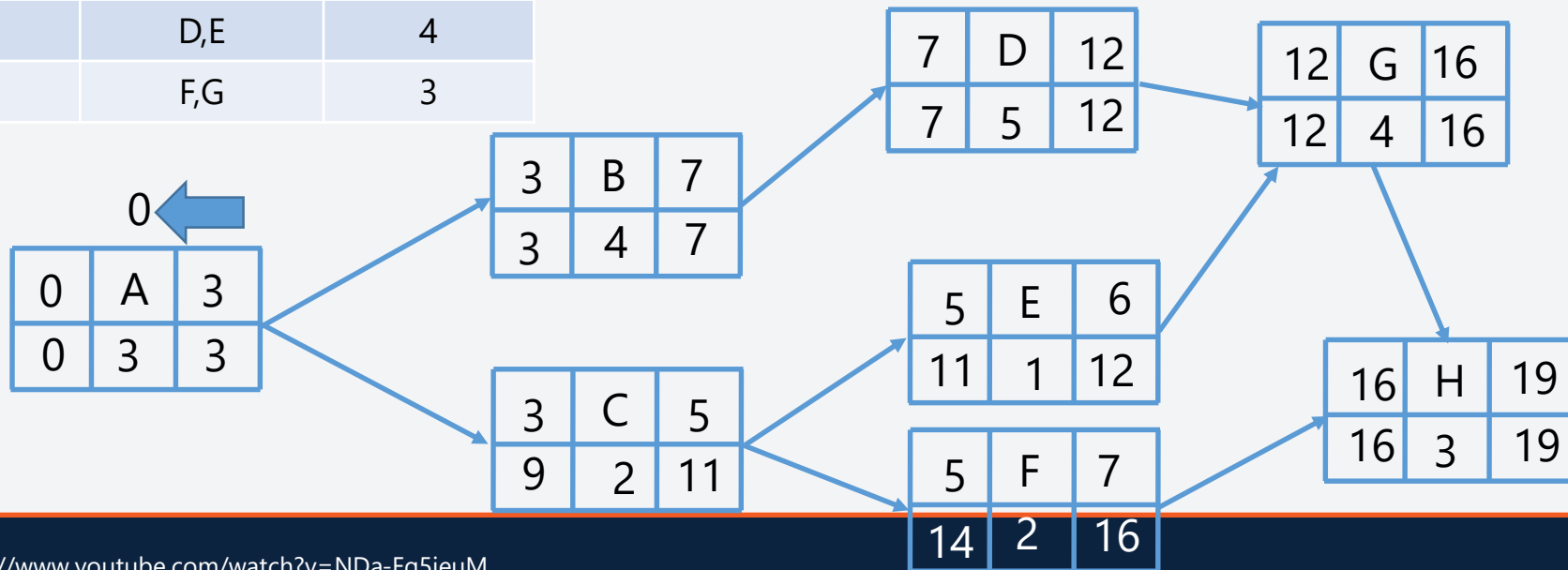


Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

$$TF = LF - EF$$

ES	T/A	EF
LS	D	LF

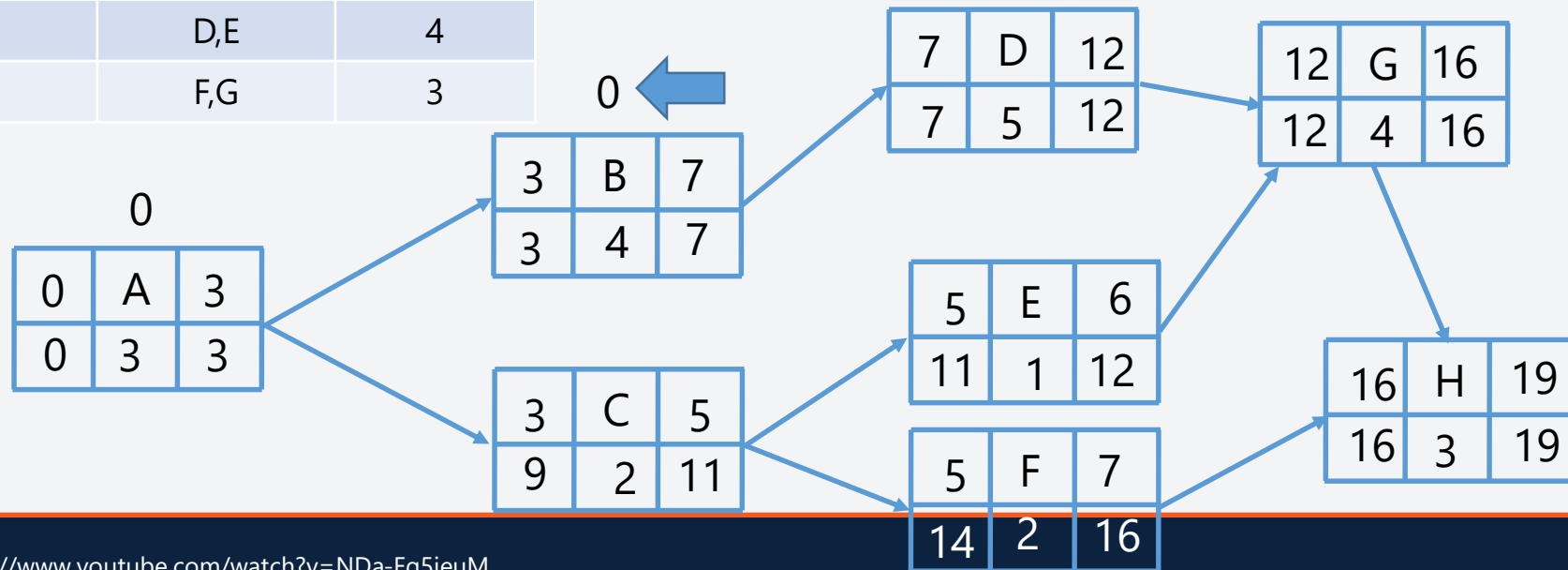


Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

$$TF = LF - EF \leftarrow$$

ES	T/A	EF
LS	D	LF

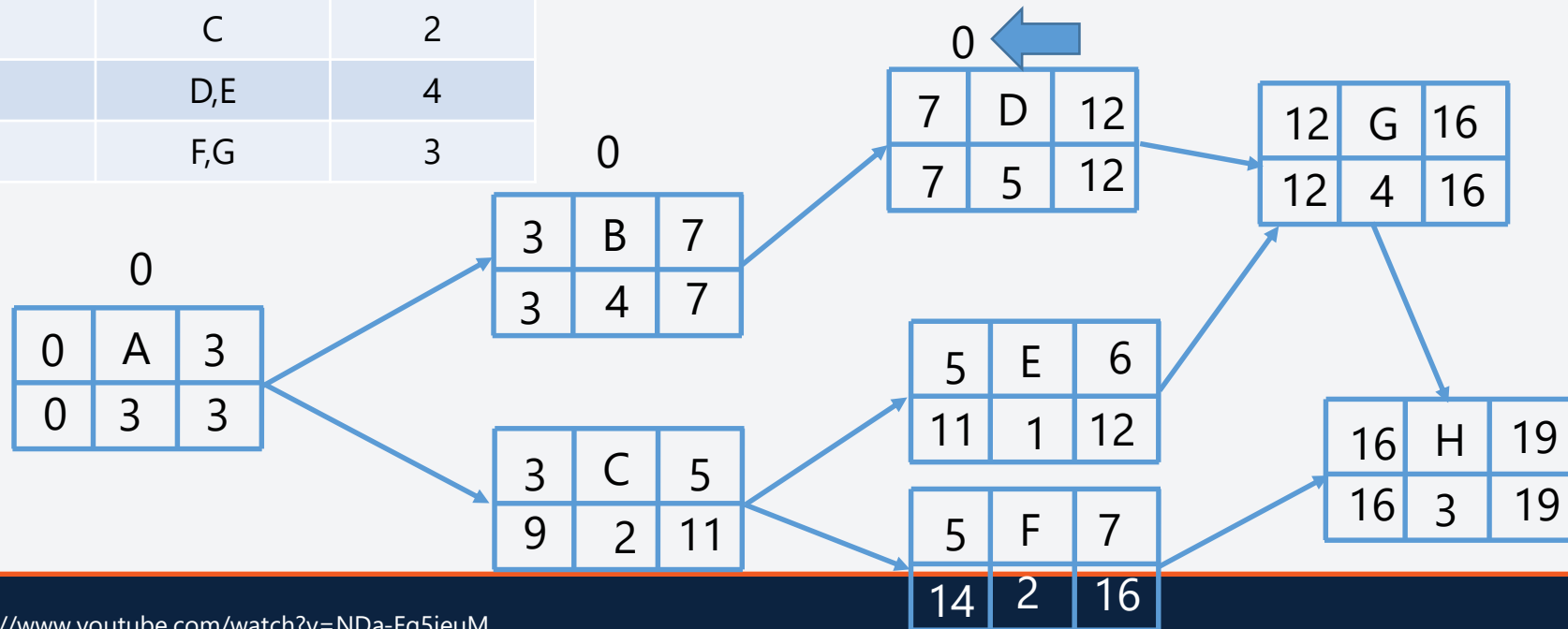


Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

$$TF = LF - EF$$

ES	T/A	EF
LS	D	LF

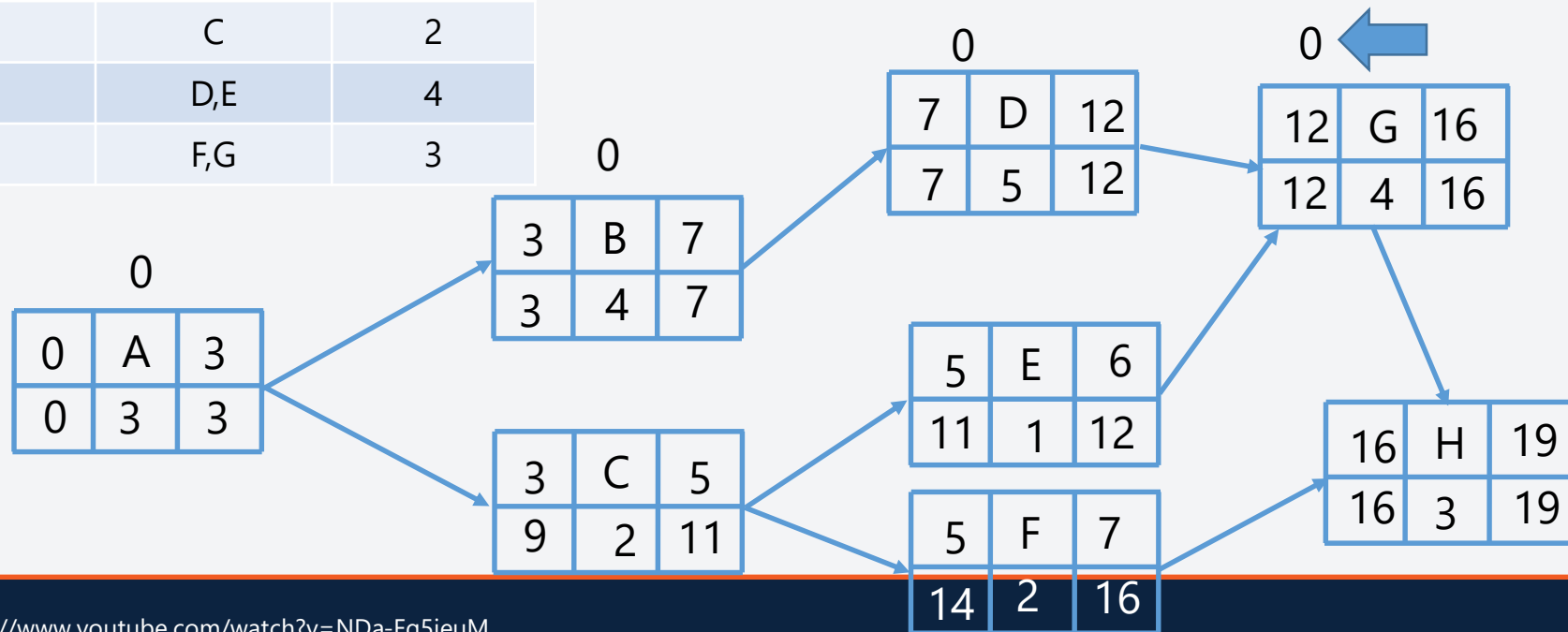


Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

$$TF = LF - EF \quad \leftarrow$$

ES	T/A	EF
LS	D	LF

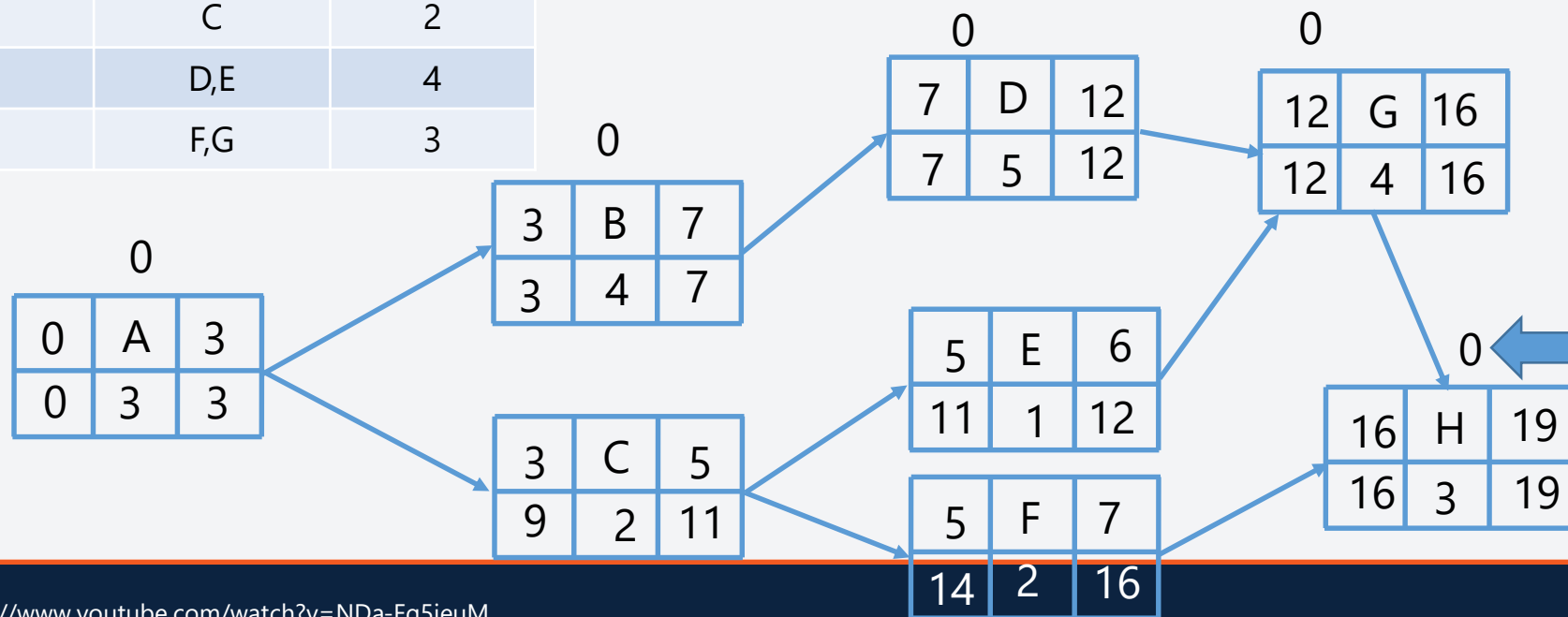


Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

$$TF = LF - EF$$

ES	T/A	EF
LS	D	LF

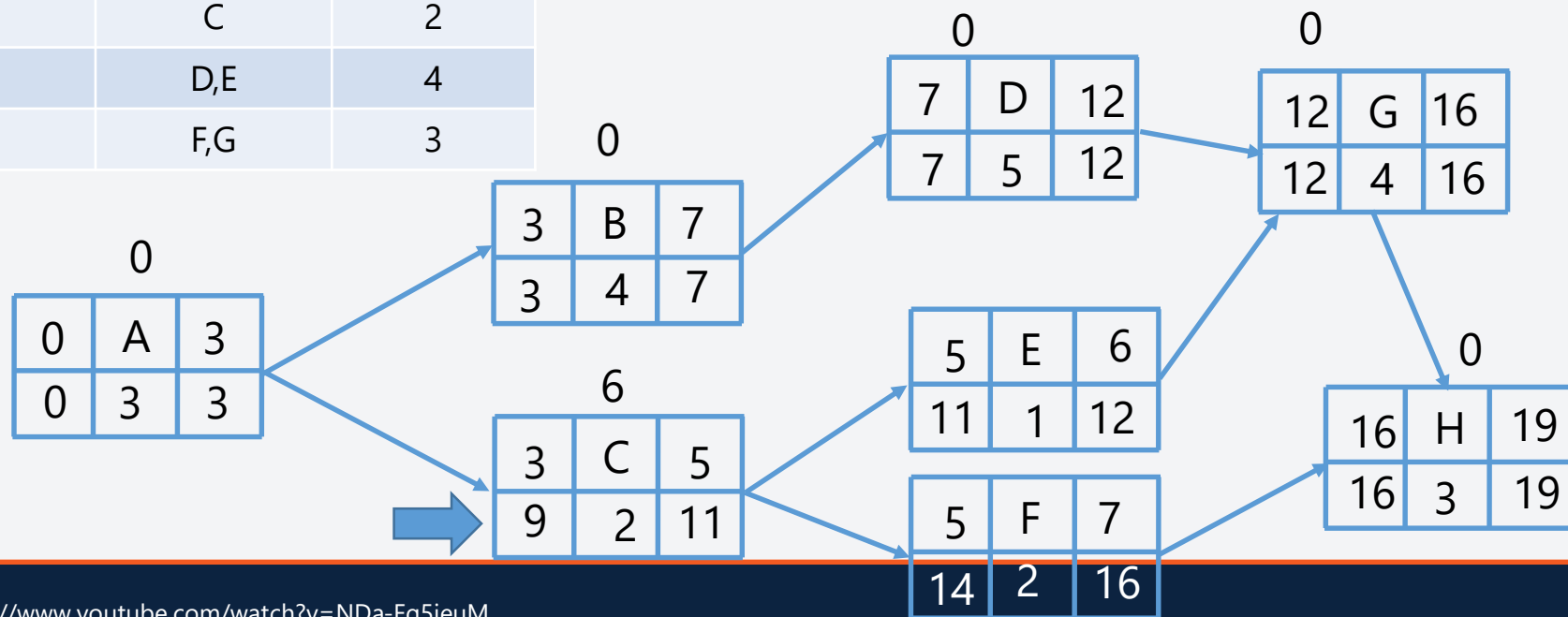


Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

$$TF = LF - EF \quad \leftarrow$$

ES	T/A	EF
LS	D	LF

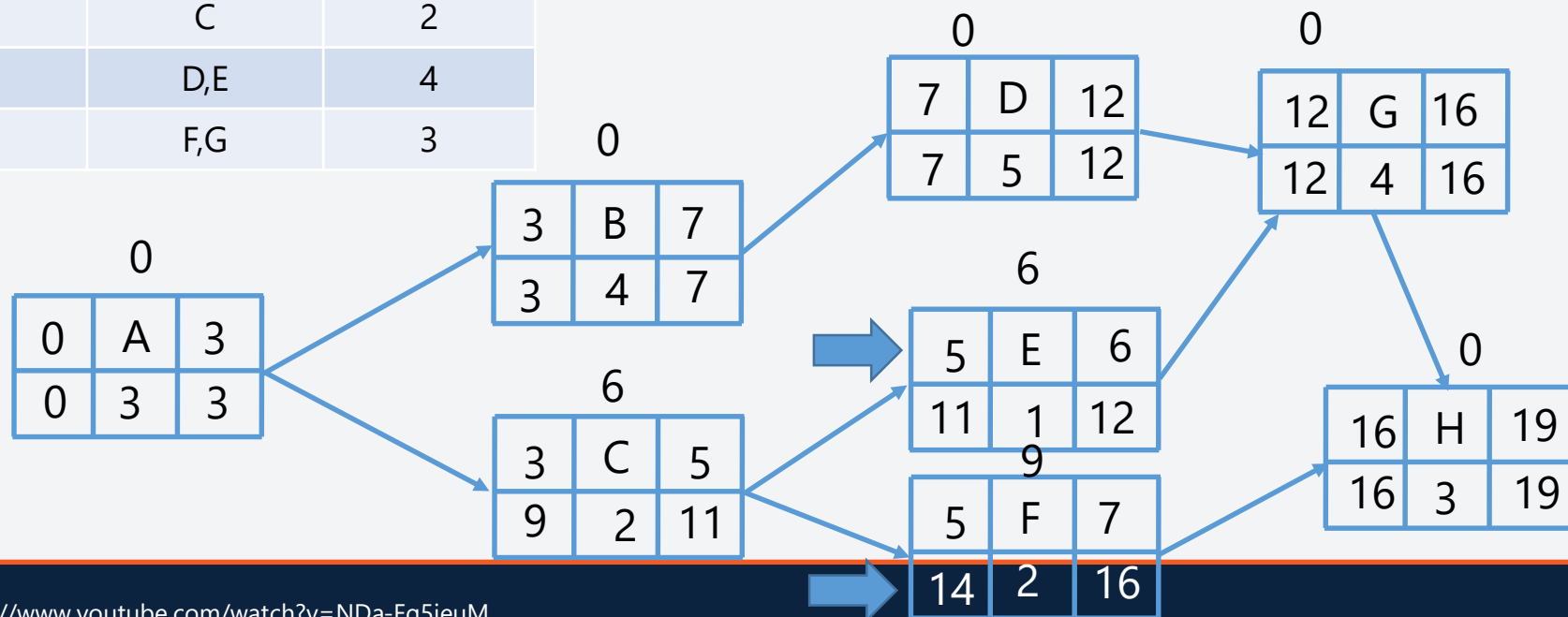


Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

$$TF = LF - EF$$

ES	T/A	EF
LS	D	LF

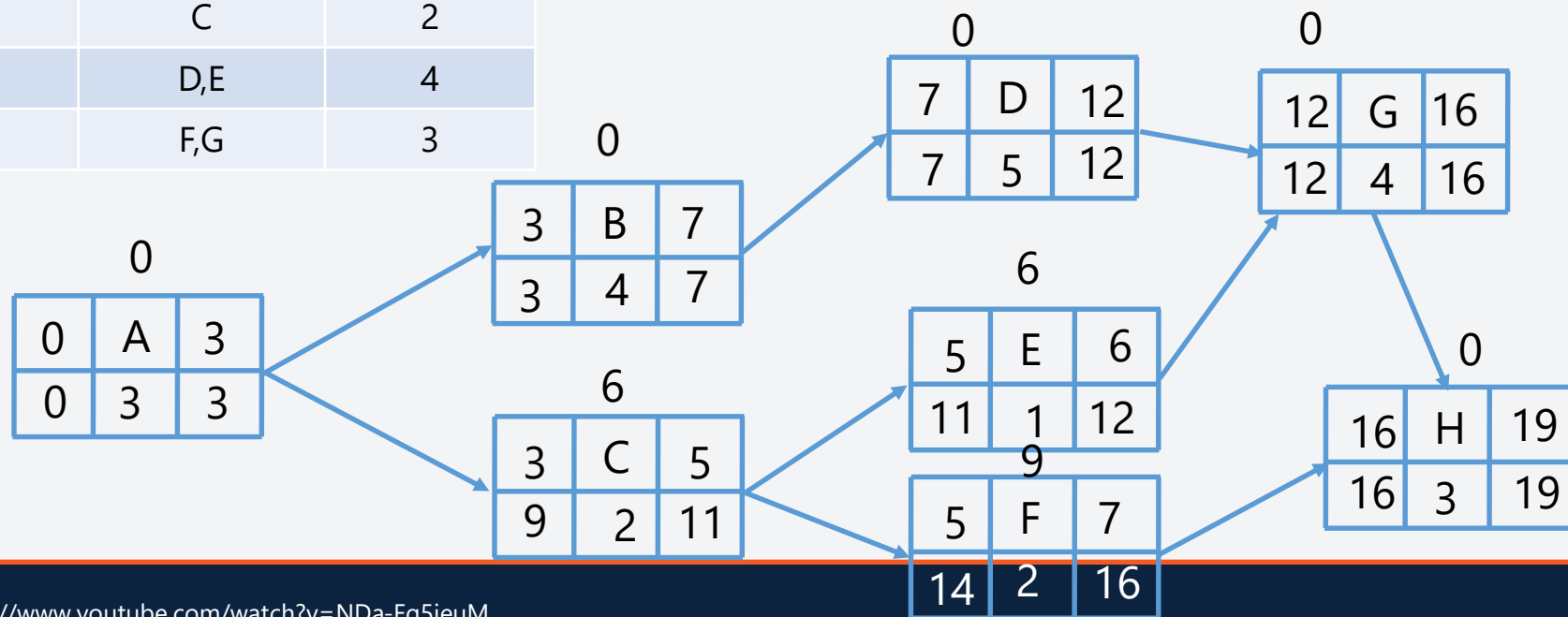


Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

$$TF = LF - EF$$

ES	T/A	EF
LS	D	LF

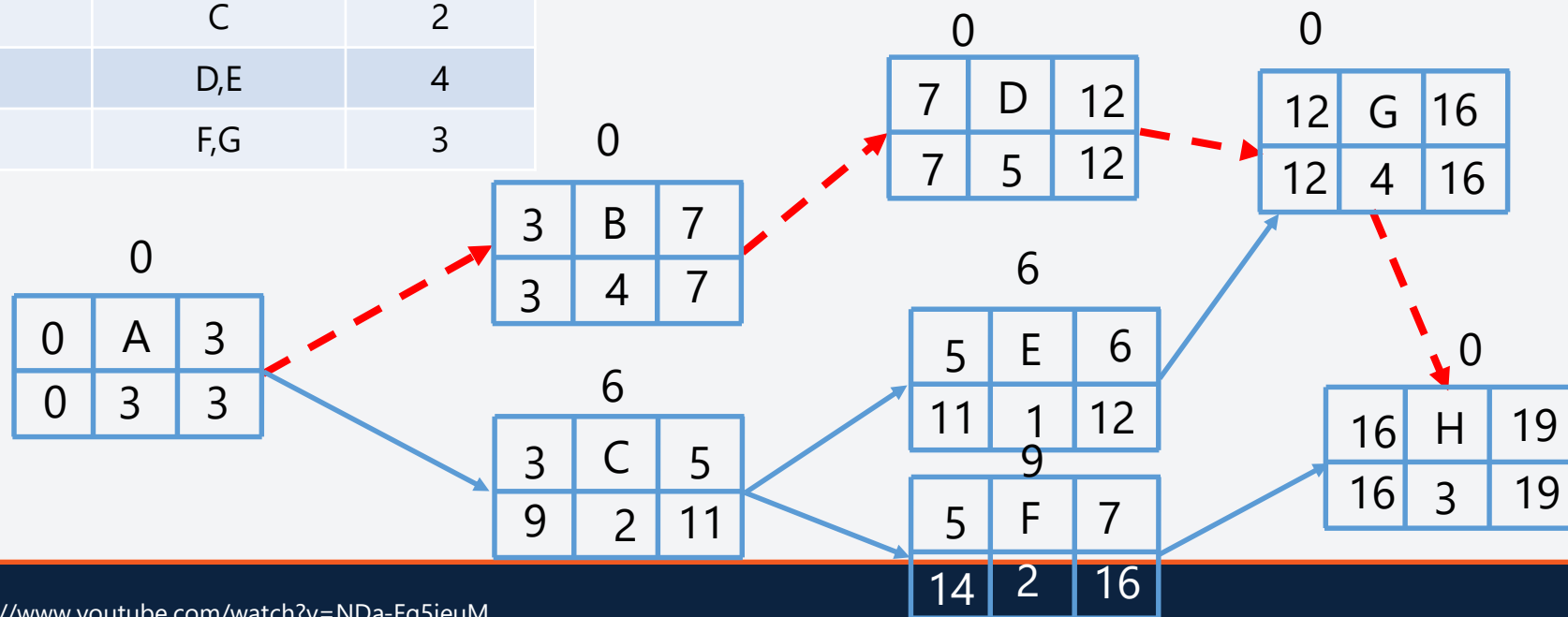


Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

Task/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

$$TF = LF - EF$$

ES	T/A	EF
LS	D	LF



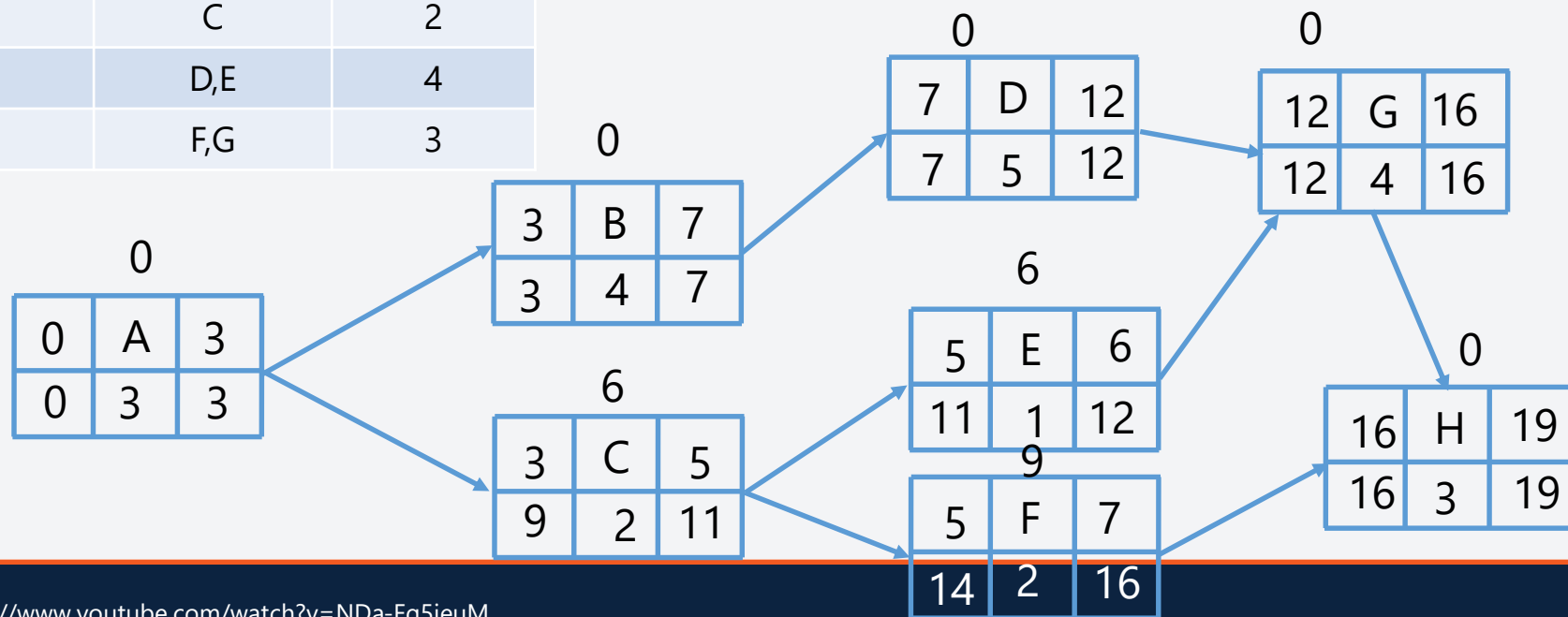
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

FF = Min(Successors' ESs) - ES - D 


ES	T/A	EF
LS	D	LF

FF = < TF

Task/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



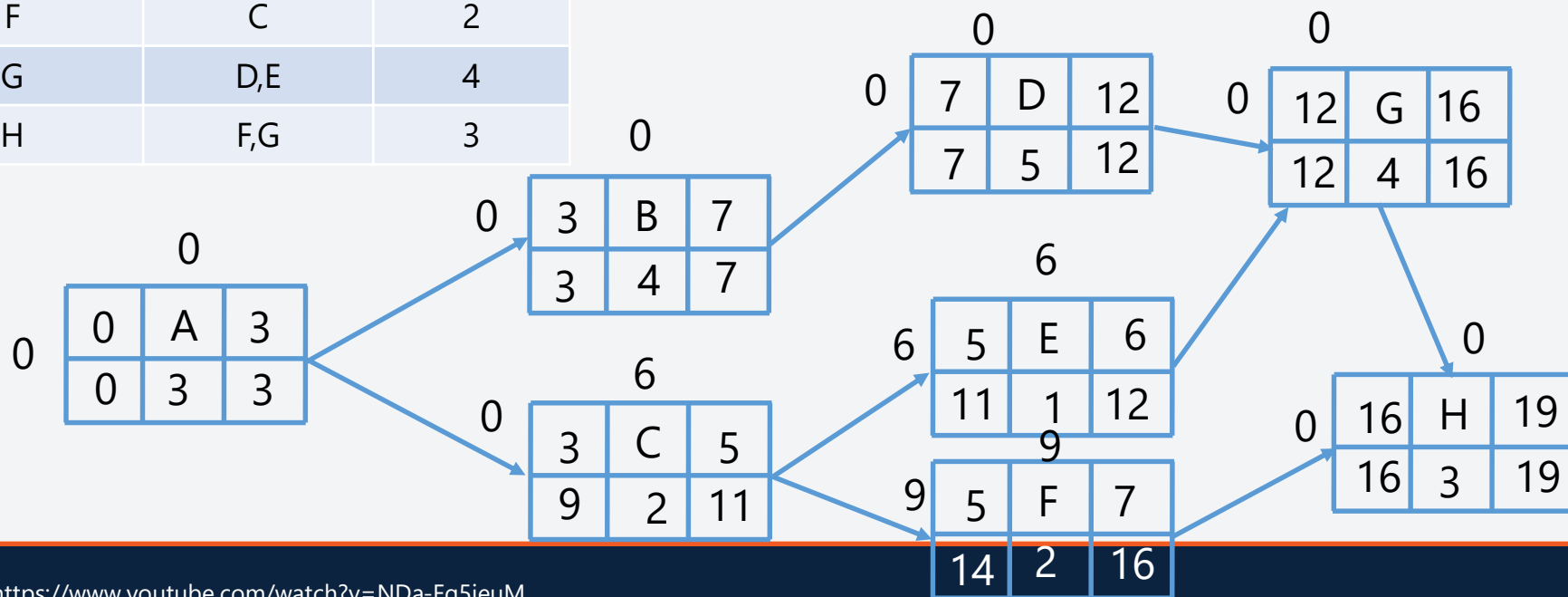
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

FF = Min(Successors' ESs) - ES - D 

Task/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

ES	T/A	EF
LS	D	LF

FF = < TF



Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

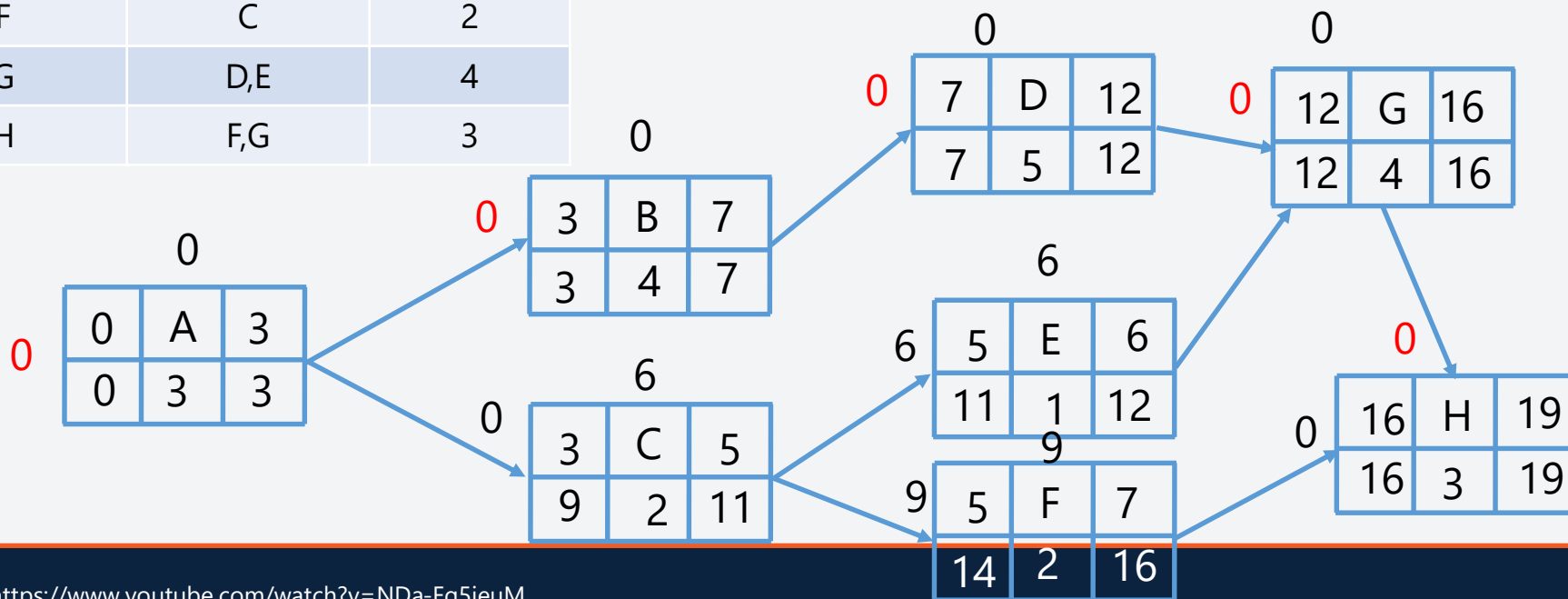
Task/Activity	Predecessor	Duration (Days)
A	-	3
B	A	4
C	A	2
D	B	5
E	C	1
F	C	2
G	D,E	4
H	F,G	3

FF=Min(Successors' ESs)-ES-D ←

ES	T/A	EF
LS	D	LF

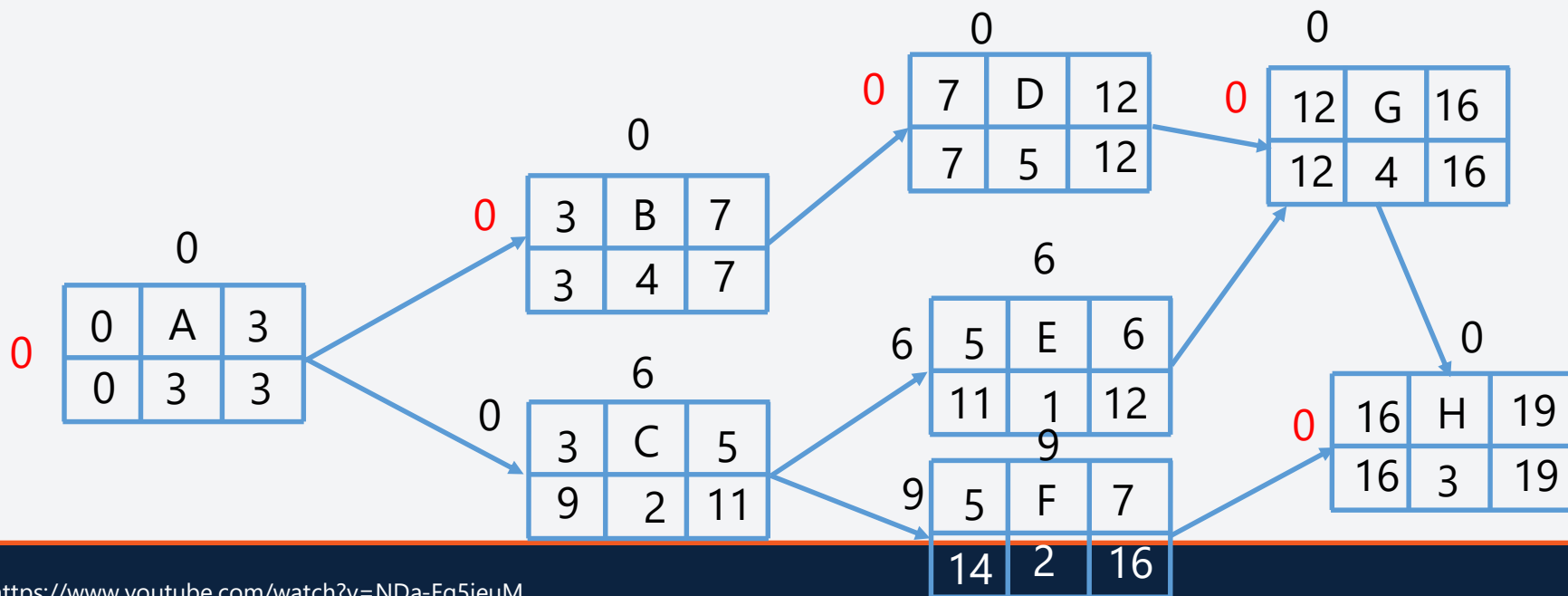
FF = < TF

On the critical path, FF is always 0.



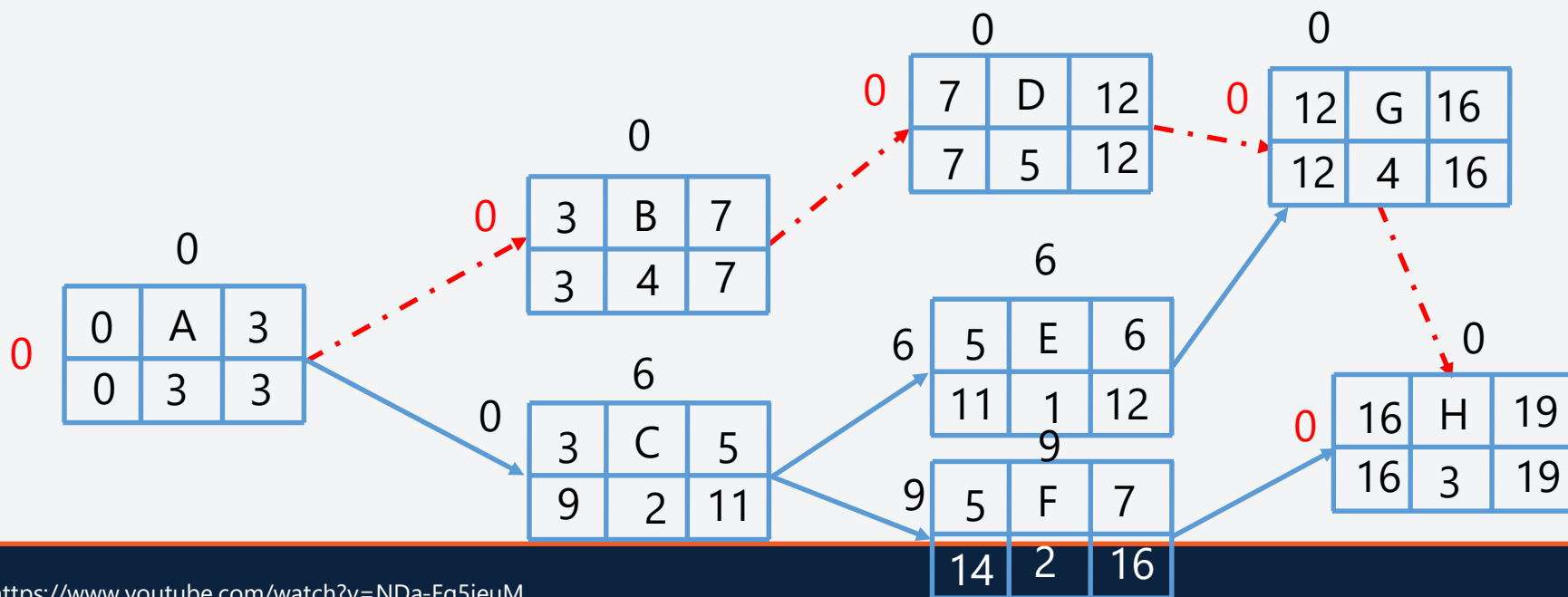
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

- **Total Float:** The amount of **time any given activity/task can be delayed without affecting the end day of the project.**
 - That is, anything on the critical path will be 0.
 - But, we can delay activity E up to 6 days.



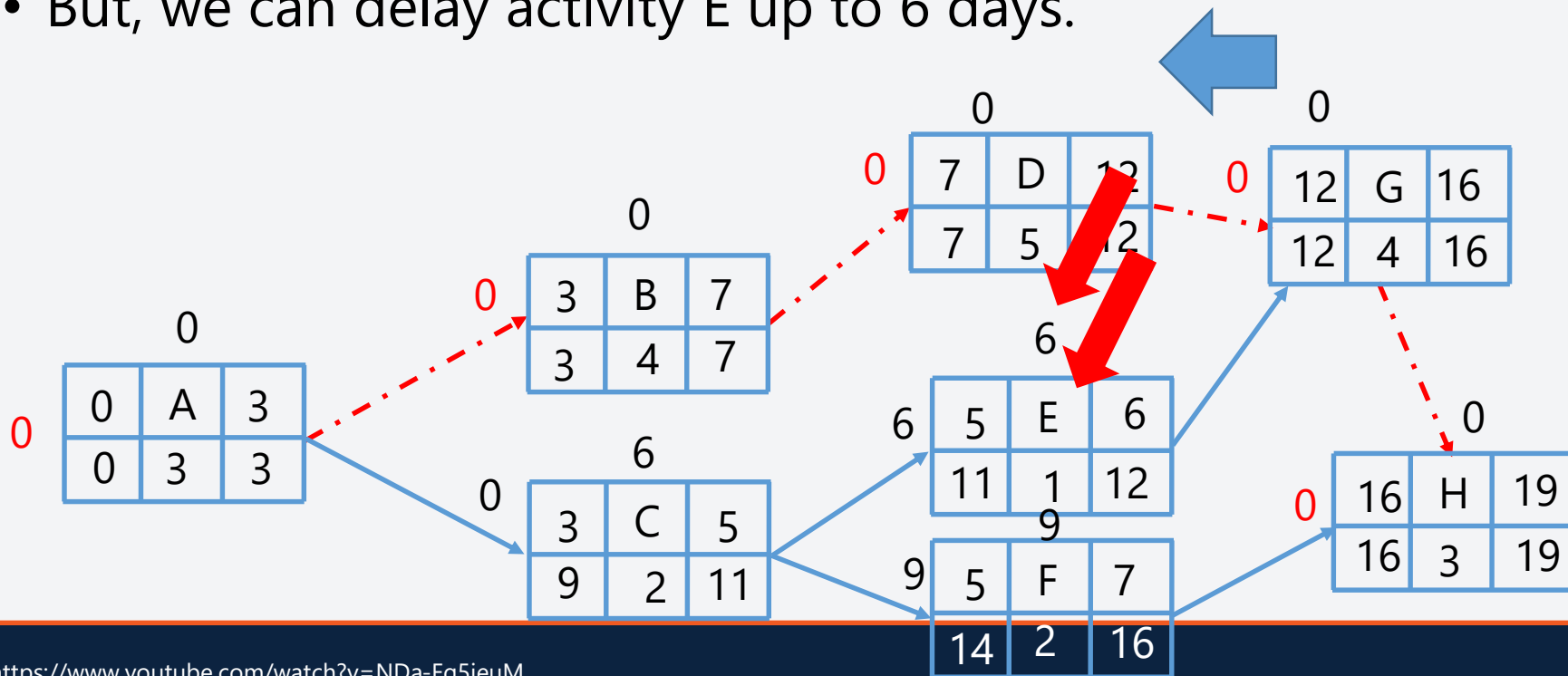
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

- **Total Float:** The amount of **time any given activity/task can be delayed without affecting the end day of the project.**
 - That is, anything on the critical path will be 0.
 - But, we can delay activity E up to 6 days.



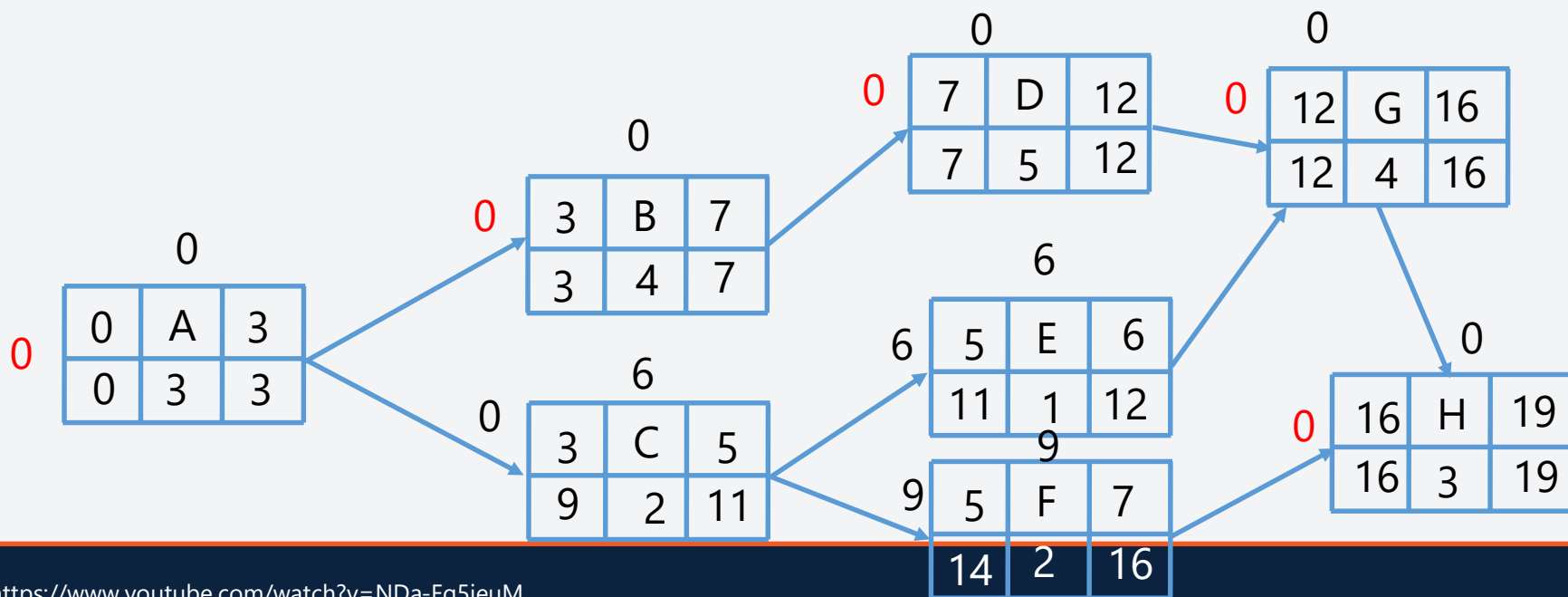
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

- **Total Float:** The amount of **time any given activity/task can be delayed without affecting the end day of the project.**
 - That is, anything on the critical path will be 0.
 - But, we can delay activity E up to 6 days.



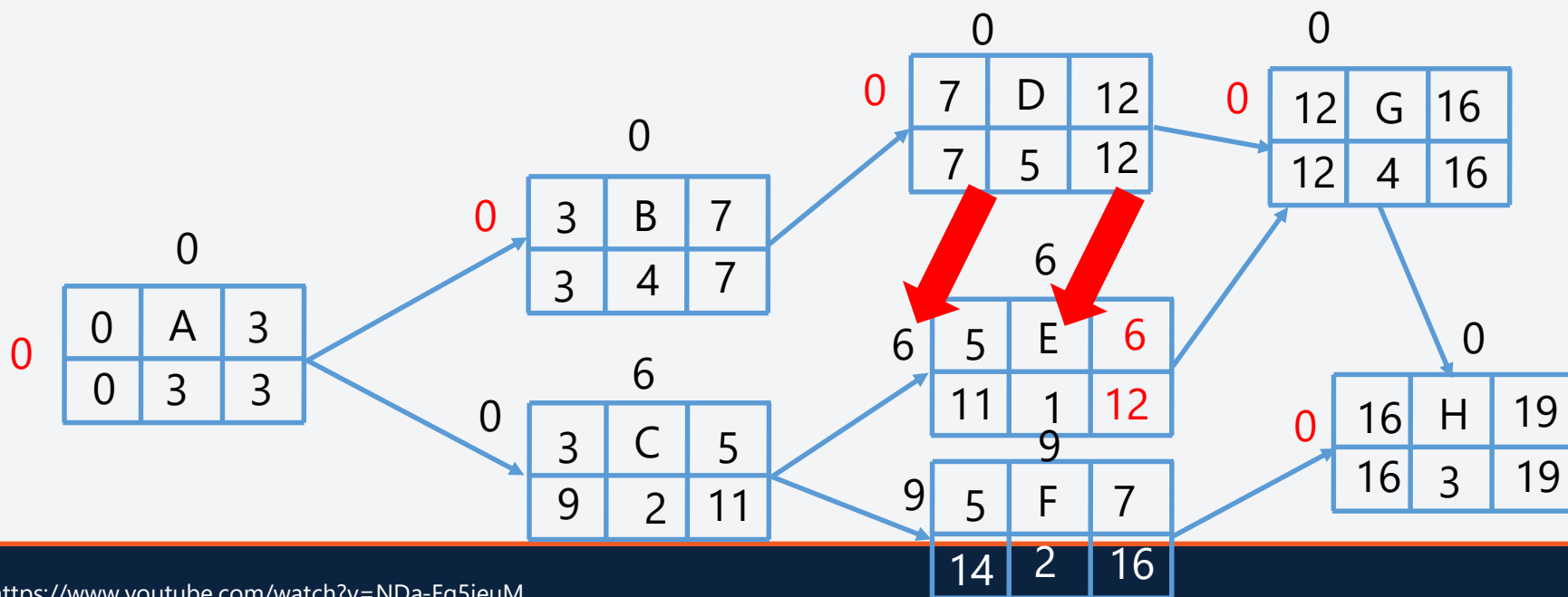
Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

- **Free Float:** The amount of **time any given activity/task can be delayed without affecting the earliest start of any of its successors.**



Total Float (a.k.a. slack) & Free Float (a.k.a. slack)

- **Free Float:** The amount of **time any given activity/task can be delayed without affecting the earliest start of any of its successors.**



St. John's Hospital Project

Activity	Immediate Predecessors	Activity Times (wks)	Responsibility
ST. JOHN'S HOSPITAL PROJECT			Kramer
START		0	
ORGANIZING and SITE PREPARATION			Stewart
A. Select administrative staff	START	12	Johnson
B. Select site and survey	START	9	Taylor
C. Select medical equipment	A	10	Adams
D. Prepare final construction plans	B	10	Taylor
E. Bring utilities to site	B	24	Burton
F. Interview applicants for nursing and support staff	A	10	Johnson
PHYSICAL FACILITIES and INFRASTRUCTURE			
G. Purchase and deliver equipment	C	35	Walker
H. Construct hospital	D	40	Sampson
I. Develop information system	A	15	Casey
J. Install medical equipment	E, G, H	4	Murphy
K. Train nurses and support staff	F, I, J	6	Pike
FINISH	K	0	Ashton

Path	Estimated Time (weeks)
------	------------------------

A-I-K	33
-------	----

A-F-K	28
-------	----

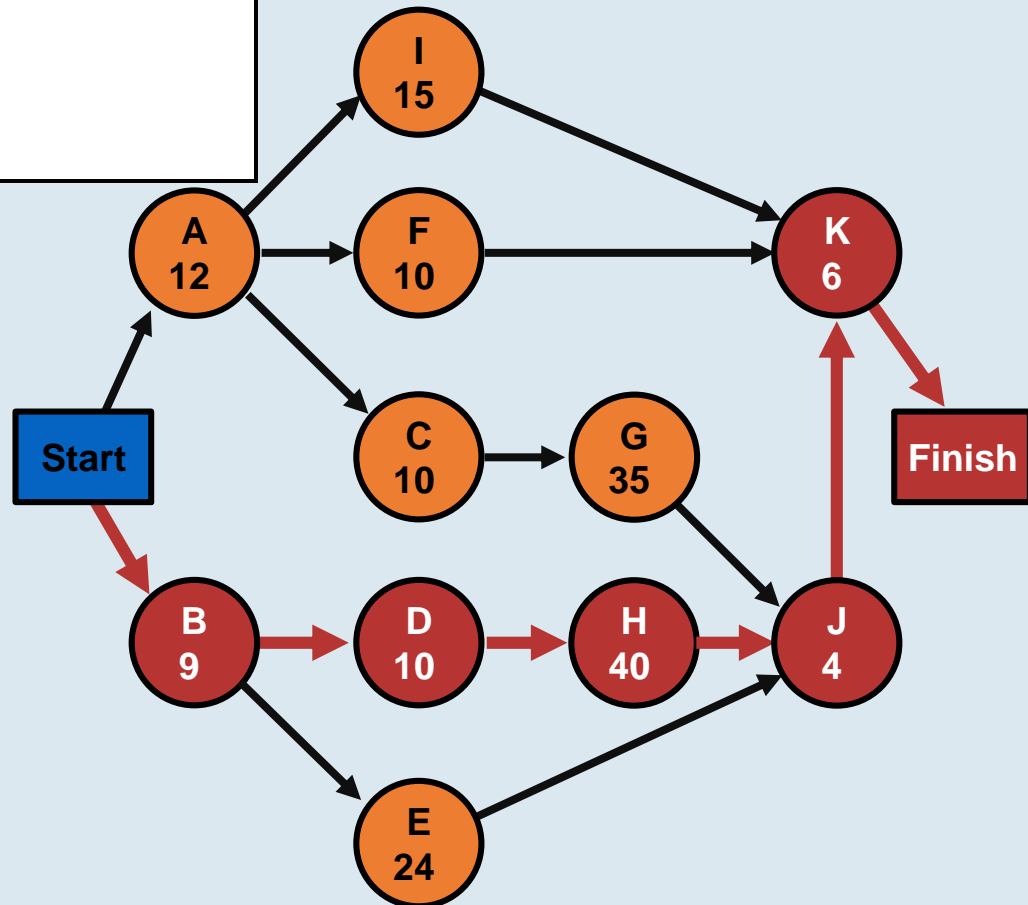
A-C-G-J-K	67
-----------	----

B-D-H-J-K	69
-----------	----

B-E-J-K	43
---------	----

al Project

Immediate Activity Times Responsibility



- A. Selected
- B. Selected
- C. Selected
- D. Preparation
- E. Bring
- F. Interview
- supp
- PHYSICAL
- G. Purchase
- H. Construction
- I. Development
- J. Installation
- K. Training
- FINISH

Exercise

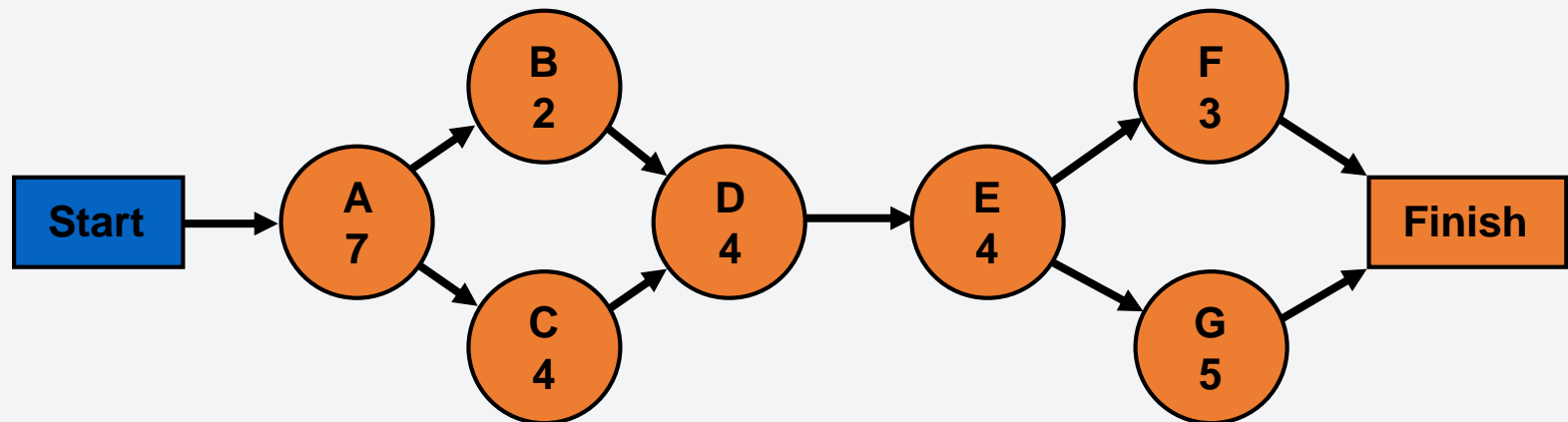
The following information is known about a project

Activity	Activity Time (days)	Immediate Predecessor(s)
A	7	—
B	2	A
C	4	A
D	4	B, C
E	4	D
F	3	E
G	5	E

Draw the network diagram for this project

Exercise

Activity	Activity Time (days)	Immediate Predecessor(s)
A	7	—
B	2	A
C	4	A
D	4	B, C
E	4	D
F	3	E
G	5	E



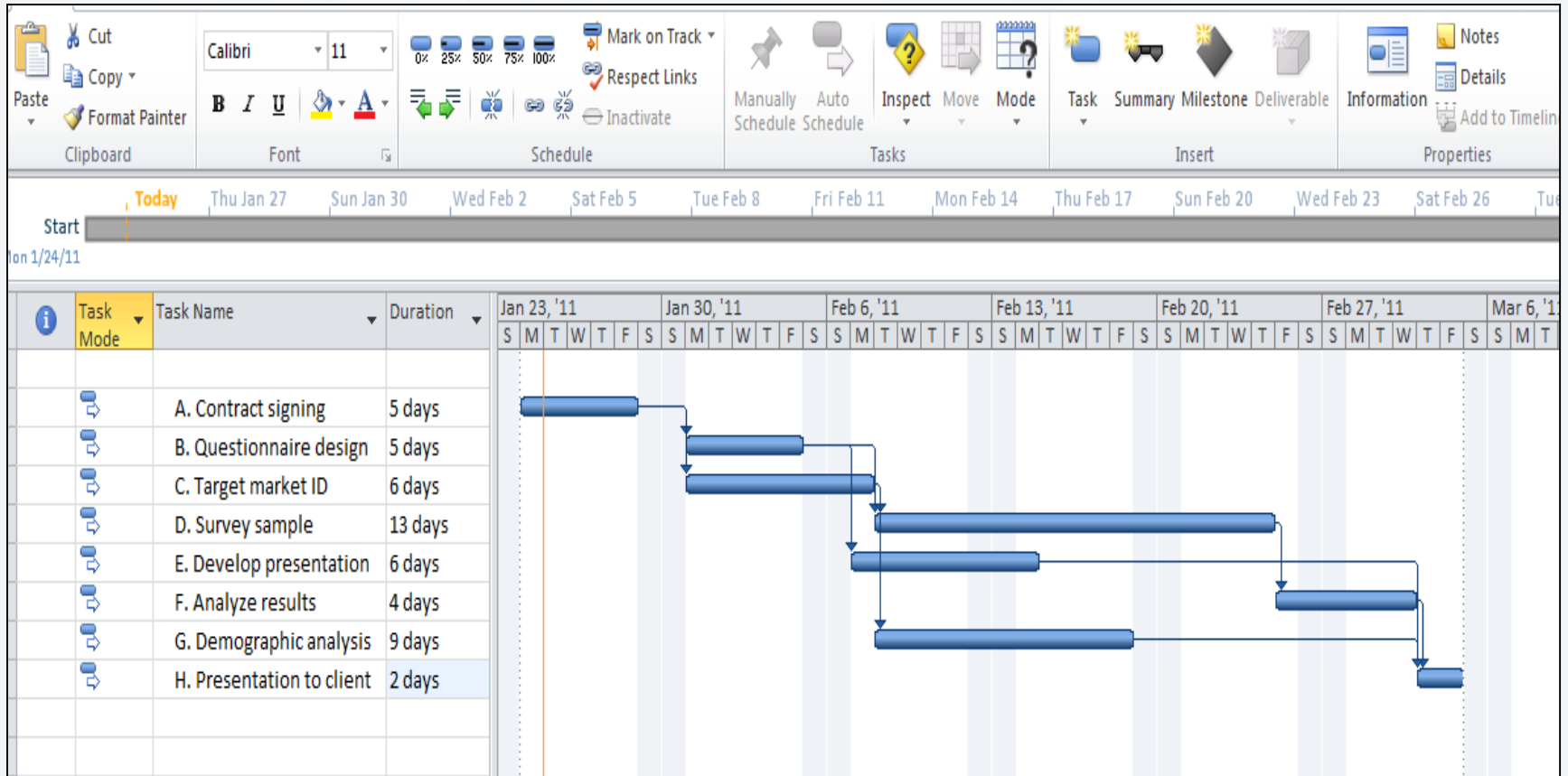
Gantt Charts

- ✓ Establish a ***time-phased network***
- ✓ Can be used as a ***tracking tool***

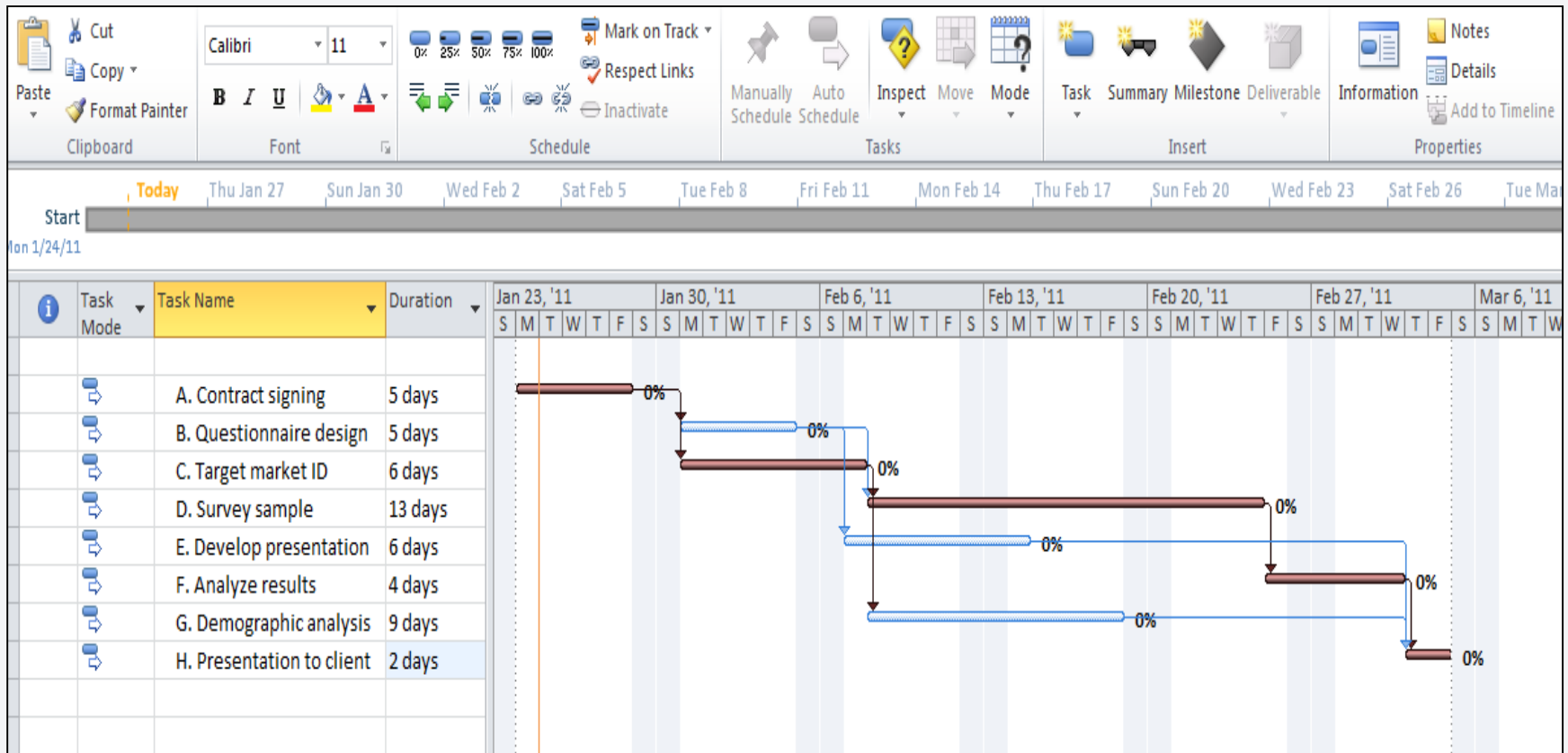
Benefits of Gantt charts

1. Easy to ***create*** and ***comprehend***
2. Identify the schedule ***baseline*** network
3. Allow for ***updating*** and ***control***
4. Identify ***resource needs***

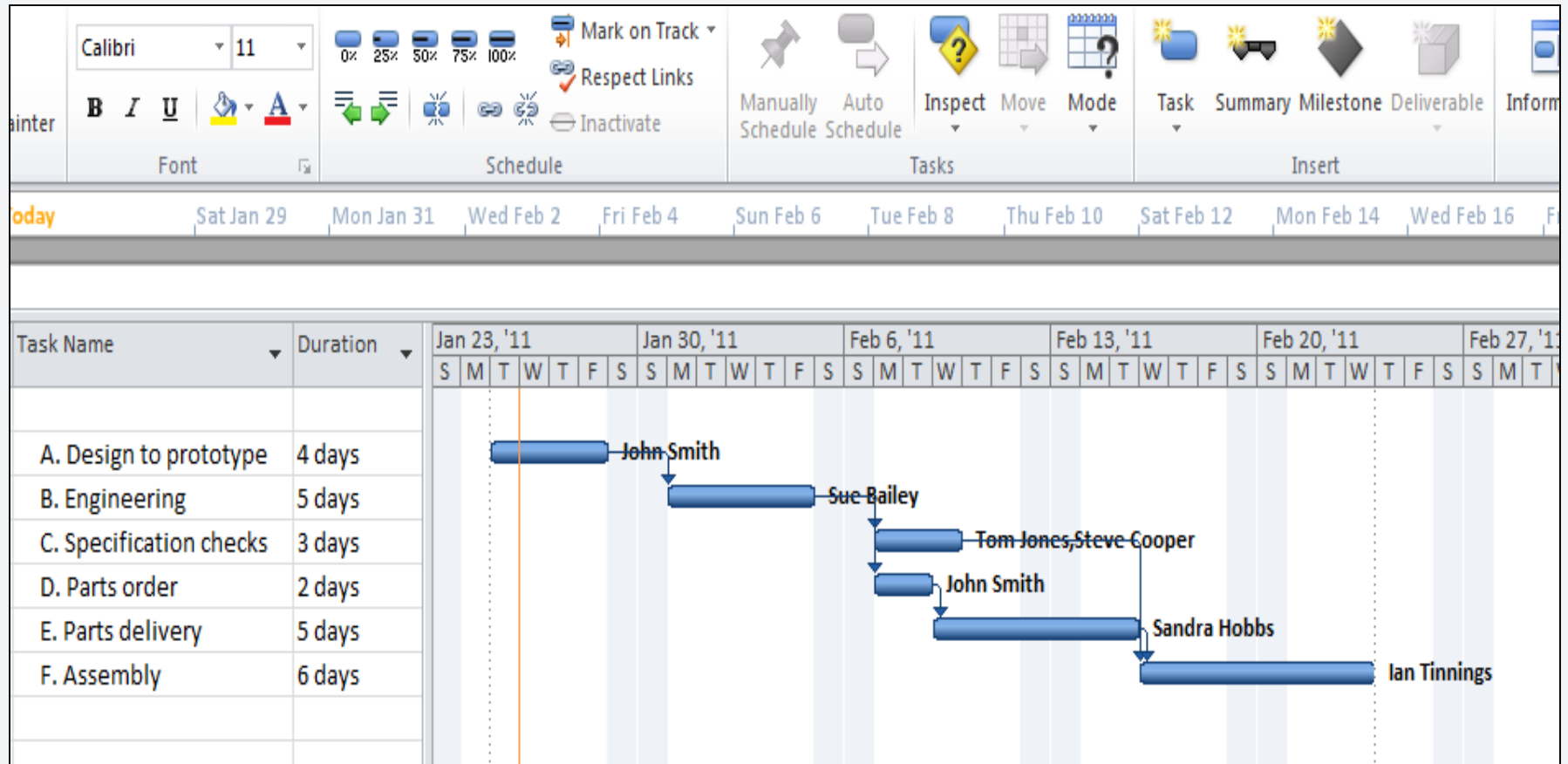
Completed Gantt Chart for Project Delta



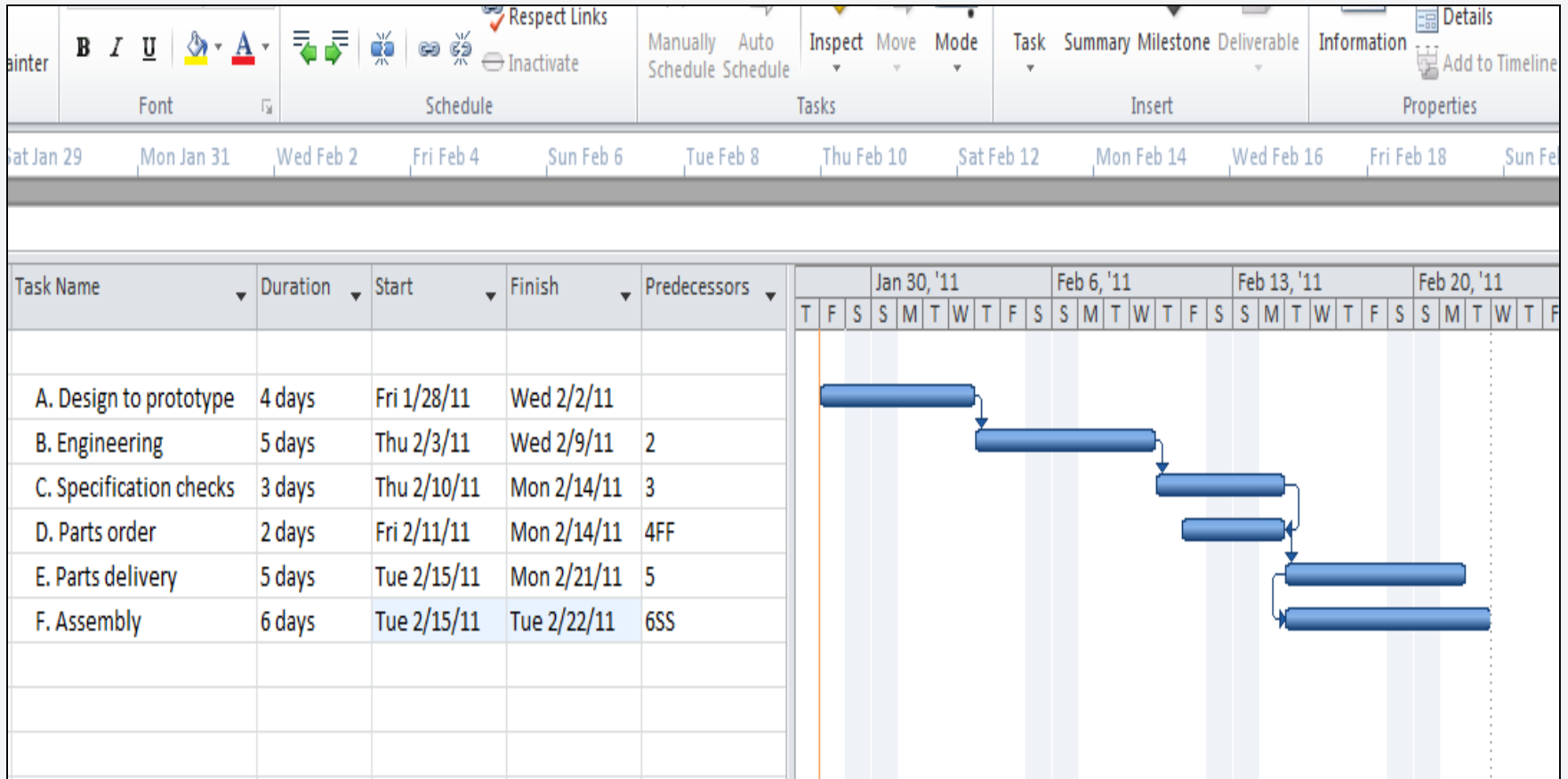
Gantt Chart for Project Delta with Critical Path Highlighted



Gantt Chart with Resources Specified



Gantt Chart with Lag Relationships



Crashing

The *process of accelerating* a project

Principal methods for crashing

- Improving existing resources' *productivity*
- Changing work *methods*
- Compromise *quality* and/or reduce *project scope*
- Institute *fast-tracking*
- Work *overtime*
- Increasing the *quantity* of resources

Managerial Considerations

- Determine activity **fixed and variable costs**
- The **crash point** is the fully expedited activity
- Optimize **time-cost tradeoffs**
- Shorten activities on the **critical path**
- Cease crashing when
 - the **target completion time** is reached
 - the **crashing cost exceeds the penalty cost**

How long does this project take and what is the cost?

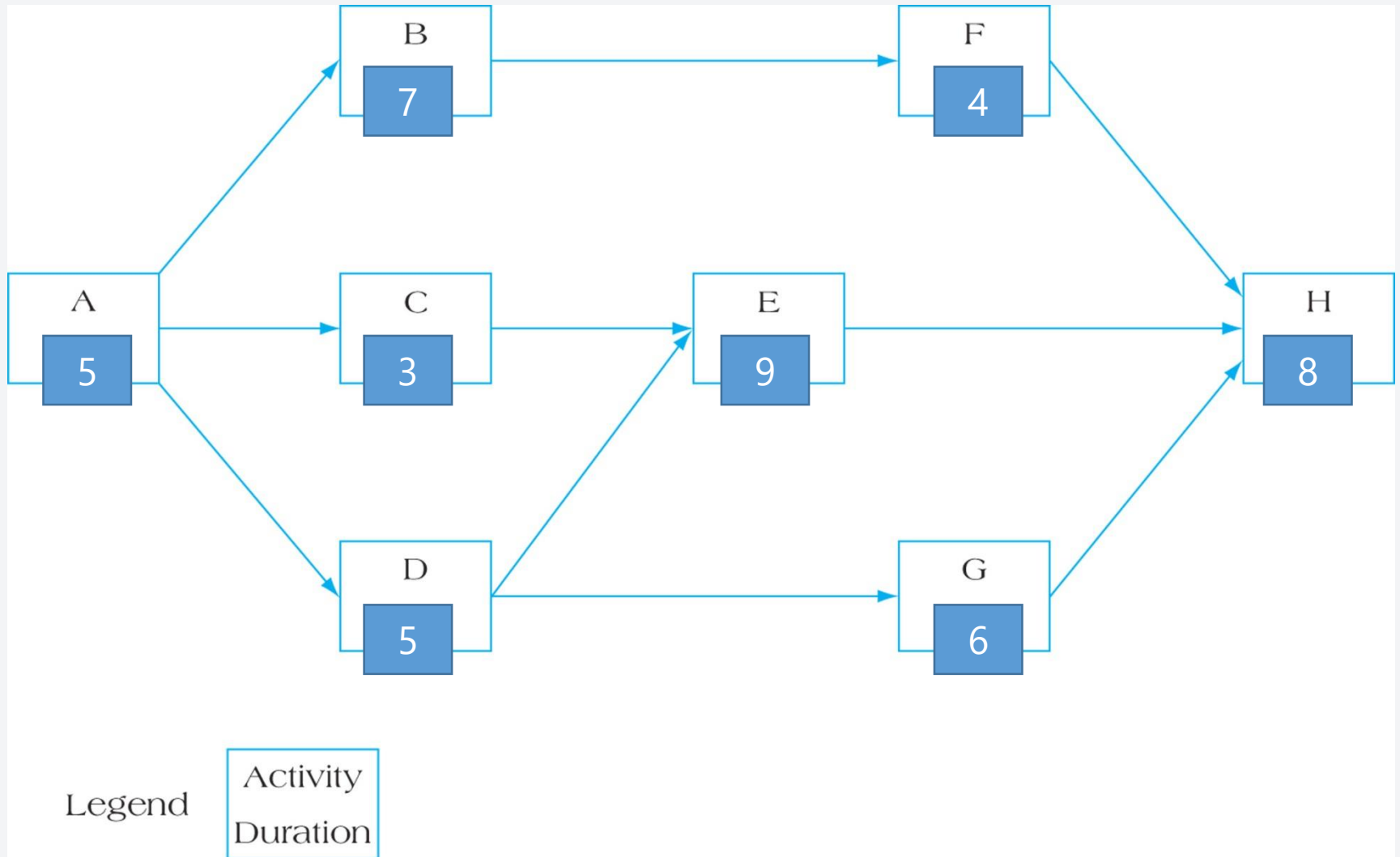
Activity	Pred	Normal Time	Min Time	Normal Cost	
A	--	14	9	500	
B	A	5	2	1000	
C	A	10	8	2000	
D	B, C	8	5	1000	
E	D	6	5	1600	
F	D	9	6	1500	
G	E, F	7	4	600	
H	G	15	11	1600	

What is the lowest cost to complete this project in 53 weeks? Times are in weeks and costs in dollars.

Activity	Pred	Normal Time	Min Time	Normal Cost	Crash Cost
A	--	14	9	500	1500
B	A	5	2	1000	1600
C	A	10	8	2000	2900
D	B, C	8	5	1000	2500
E	D	6	5	1600	1900
F	D	9	6	1500	3000
G	E, F	7	4	600	1800
H	G	15	11	1600	3600

Project Activities and Costs

Activity	Normal		Crashed	
	Duration	Cost	Duration	Cost
A	5 days	\$ 1,000	3 days	\$ 1,500
B	7 days	700	6 days	1,000
C	3 days	2,500	2 days	4,000
D	5 days	1,500	5 days	1,500
E	9 days	3,750	6 days	9,000
F	4 days	1,600	3 days	2,500
G	6 days	2,400	4 days	3,000
H	8 days	9,000	5 days	15,000
Total costs =		\$22,450		\$37,500



Project Activity Network

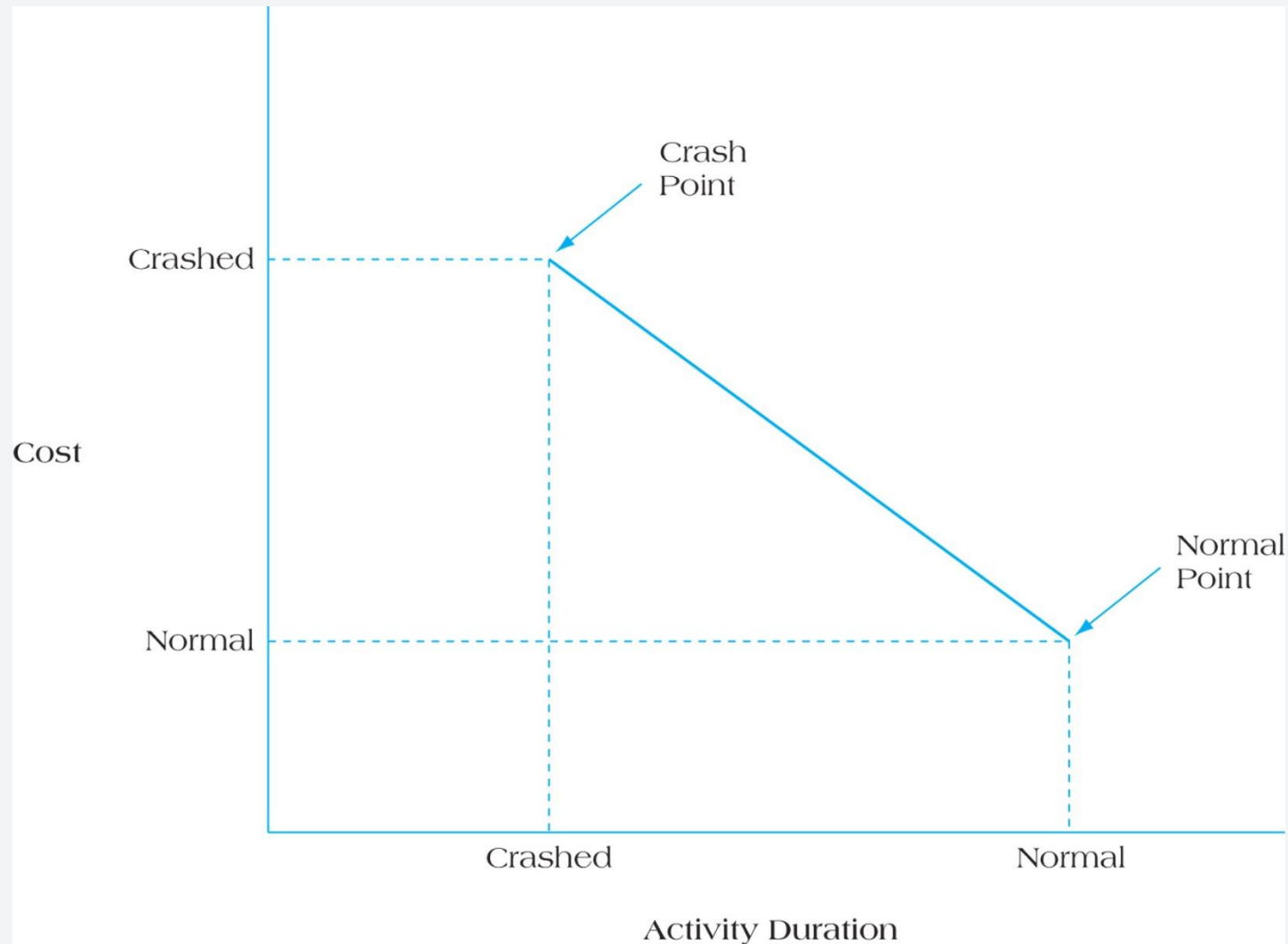
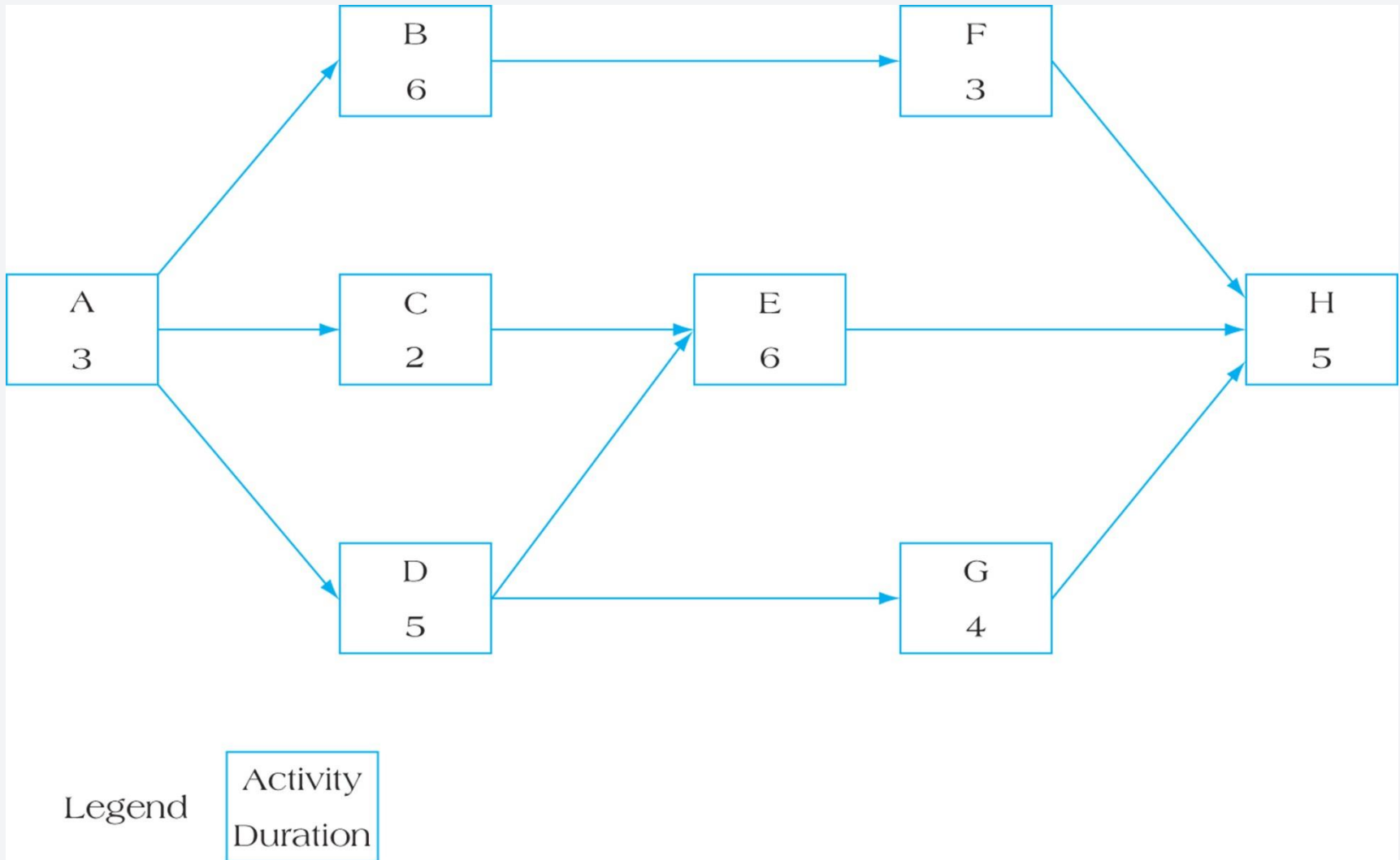
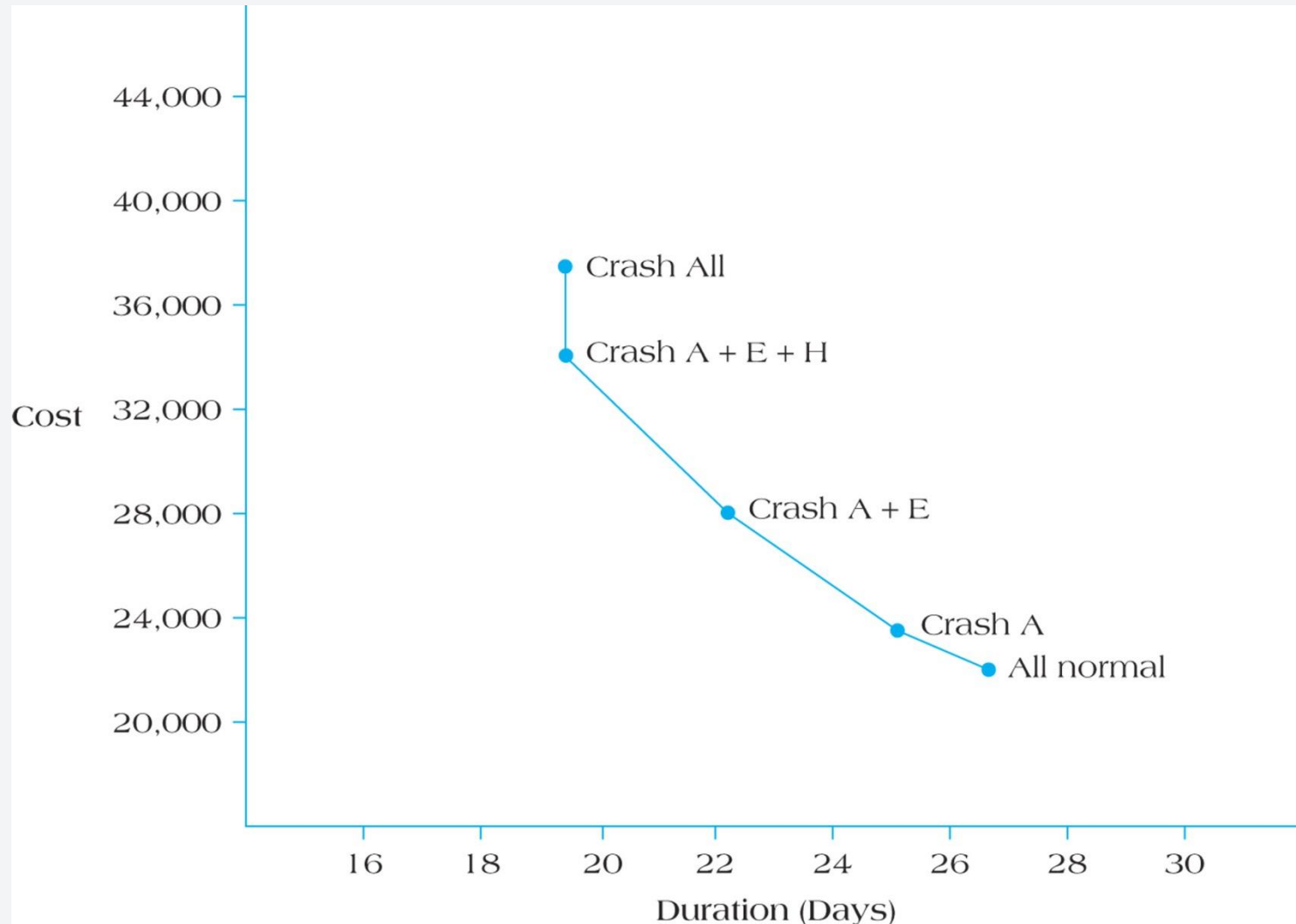


FIGURE 10.14 Time–Cost Trade-Offs for Crashing Activities



Fully Crashed Project Activity Network



Relationship Between Cost and Days Saved in a Crashed Project