INSE 6230
Total Quality Project Management

Lecture 3

Project Time Management

Project Time Management I.

- Activities
- Network diagrams
  - AOA (Activity-on-Arrow) networks
  - AON (Activity-on-Node) networks
- Developing a schedule
  - Gantt Chart
- Critical Path

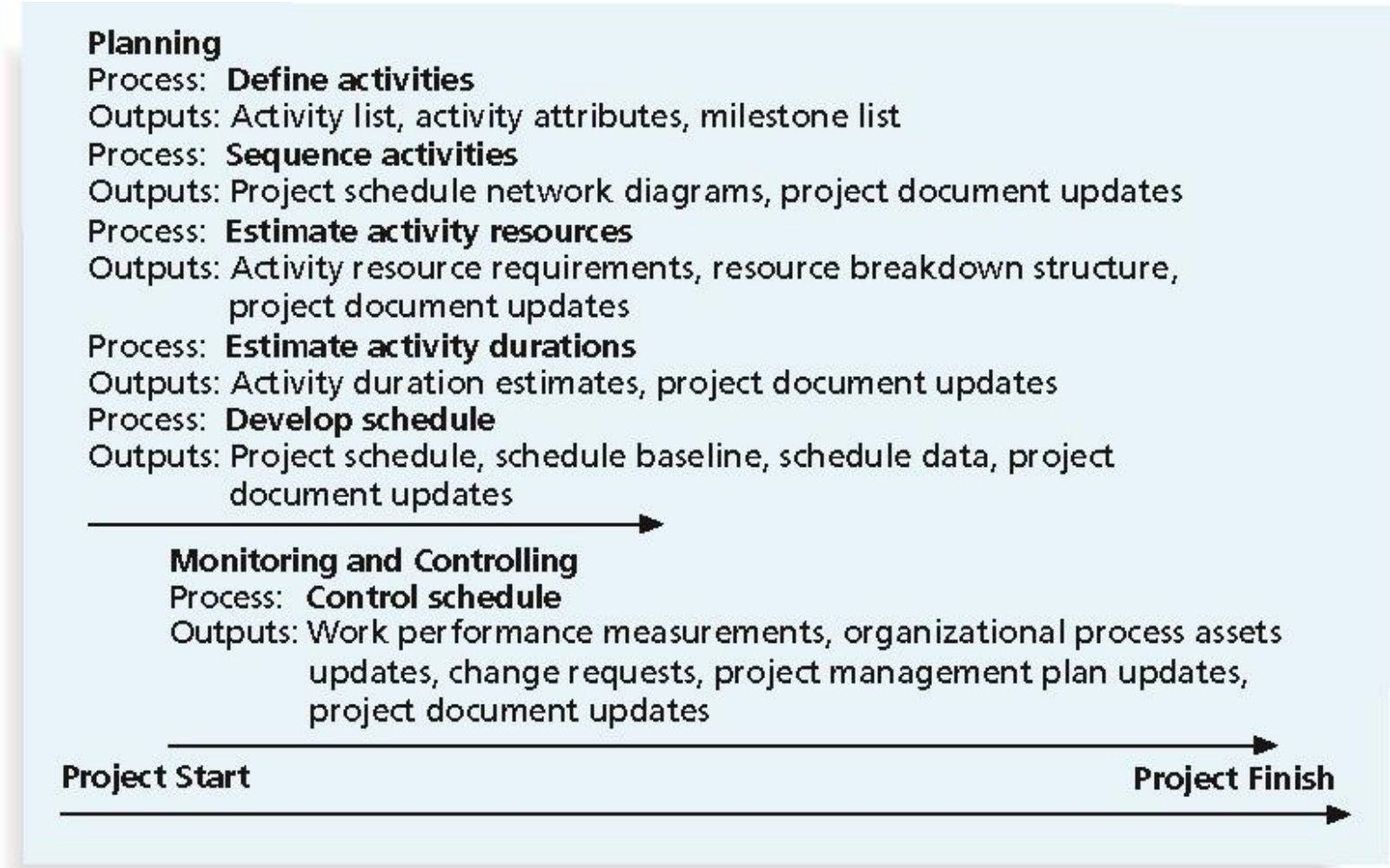
Importance of Project Schedules

- **Most IT projects exceed time estimates!**
- Time has the least amount of flexibility
- Schedule issues are **the main reason for conflicts** on projects, especially during the second half of projects
- Various attitudes towards deadlines:
  - Some people prefer to follow schedules and meet deadlines while others do not
  - Difference cultures and even entire countries have different attitudes about schedules

Project Time Management Processes

- **Defining activities**: identifying the specific activities that the project team members and stakeholders must perform to produce the project deliverables
- **Sequencing activities**: identifying and documenting the relationships between project activities
- **Estimating activity resources**: estimating how many resources a project team should use to perform project activities
- **Estimating activity durations**: estimating the number of work periods that are needed to complete individual activities
- **Developing the schedule**: analyzing activity sequences, activity resource estimates, and activity duration estimates to create the project schedule
- **Controlling the schedule**: controlling and managing changes to the project schedule

Project Time Management Summary



Defining Activities

- An **activity** or **task** is an element of work normally found on the Work Breakdown Structure (WBS)
  - It has an expected duration, a cost, and resource requirements
- Activity definition involves developing a more detailed WBS and supporting explanations to understand all the work to be done so you can develop realistic cost and duration estimates
- An **activity list** is a tabulation of activities to be included on a project schedule that includes:
  - The activity name
  - An activity identifier or number
  - A brief description of the activity
- **Activity attributes** provide more information such as predecessors, successors, logical relationships, leads and lags, resource requirements, constraints, imposed dates, and assumptions related to the activity

Sequencing Activities

- A **precedence relationship** (or **dependency**) defines the sequence constraint among activities:
  - An activity can start only after a set of other activities have been finished
- Three reasons for dependencies:
  - **Mandatory dependencies**: inherent in the nature of the work being performed on a project, sometimes referred to as *hard logic*
  - **Discretionary dependencies**: defined by the project team; sometimes referred to as *soft logic* and should be used with care since they may limit later scheduling options
  - **External dependencies**: involve relationships between project and non-project activities
- For an activity, we typically determine only its **immediate predecessor activities**

Precedence Relationships - *Example*

- Suppose that **a site preparation and concrete slab foundation construction project** consists of nine different activities:

A. Site clearing (of brush and minor debris)
B. Removal of trees
C. General excavation
D. Grading general area
E. Excavation for utility trenches
F. Placing formwork and reinforcement for concrete
G. Installing sewer lines
H. Installing other utilities
I. Pouring concrete

- Summarize the precedence relationships in the project

Predecessors
0 →A,B
A→C,D
B,C→E,F
D,E→G,H
F,G→I

Activities **A** (site clearing) and **B** (tree removal) do not have preceding activities since they depend on none of the other activities.

We assume that activities **C** (general excavation) and **D** (general grading) are preceded by activity A (site clearing).

Activities **E** (trench excavation) and **F** (concrete preparation) cannot begin until the completion of general excavation and tree removal, since they involve subsequent excavation and trench preparation.

Activities **G** (install lines) and **H** (install utilities) represent installation in the utility trenches and cannot be attempted until the trenches are prepared, and activity E (trench excavation) is thus a preceding activity. We also assume that the utilities should not be installed until grading is completed to avoid equipment conflicts, so activity D (general grading) is also preceding activities G (install sewers) and H (install utilities).

Finally, activity **I** (pour concrete) cannot begin until the sewer line is installed and formwork and reinforcement are ready, so activities F and G are preceding. Other utilities may be routed over the slab foundation, so activity H (install utilities) is not necessarily a preceding activity for activity I (pour concrete).

Precedence Relationships

- *Example*

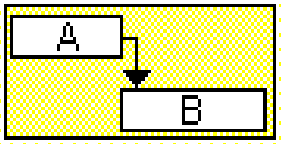
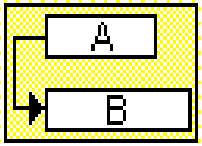
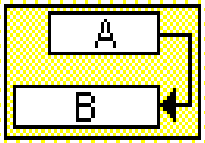
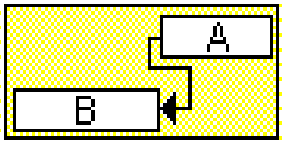
Activity	Description	Predecessors
A	Site clearing	---
B	Removal of trees	---
C	General excavation	A
D	Grading general area	A
E	Excavation for utility trenches	B,C
F	Placing formwork and reinforcement for concrete	B,C
G	Installing sewer lines	D,E
H	Installing other utilities	D,E
I	Pouring concrete	F,G

Predecessors
0 →A,B
A→CD
B,C→E,F
D,E→G,H
F,G→I

Types of Dependencies

Task dependencies

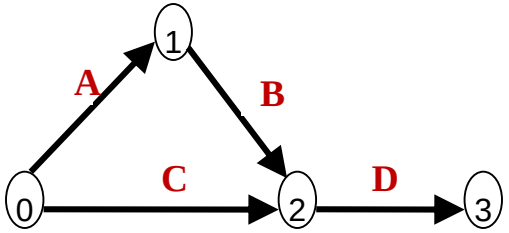
The nature of the dependencies between linked tasks. You link tasks by defining a dependency between their finish and start dates. For example, the "Contact caterers" task must finish before the start of the "Determine menus" task. There are four kinds of task dependencies in Microsoft Project:

Task dependency	Example	Description
Finish-to-start (FS)		Task (B) cannot start until task (A) finishes.
Start-to-start (SS)		Task (B) cannot start until task (A) starts.
Finish-to-finish (FF)		Task (B) cannot finish until task (A) finishes.
Start-to-finish (SF)		Task (B) cannot finish until task (A) starts.

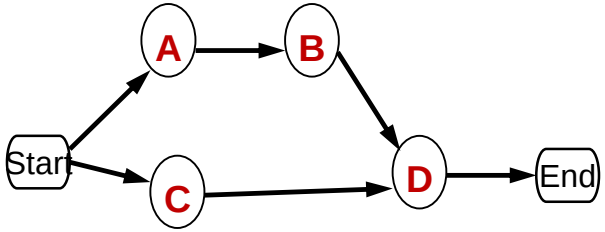
Network Diagrams

- Network diagramming is a technique for showing the activity precedence relationships in a project
- A **network diagram** is a schematic display of the logical relationships among, or sequencing of, project activities

Activity on arrow network (AOA)



Activity on node network (AON)

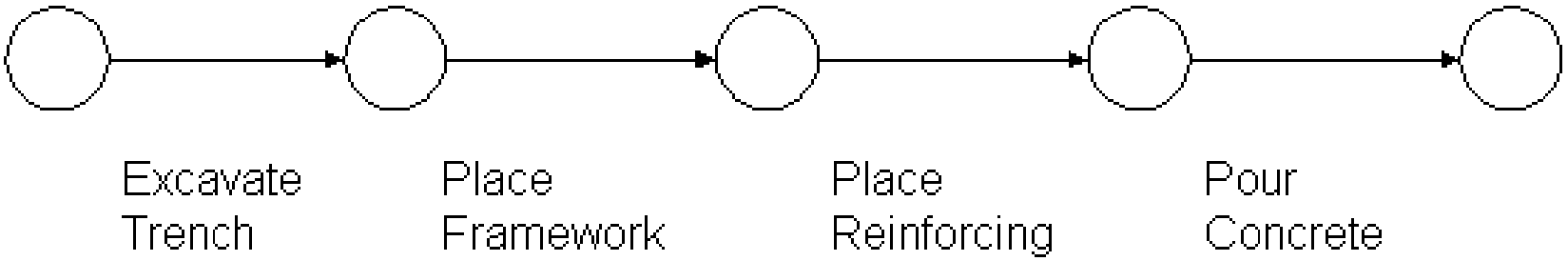


- The two networks and methods are both widely used, and both of them should be learned.

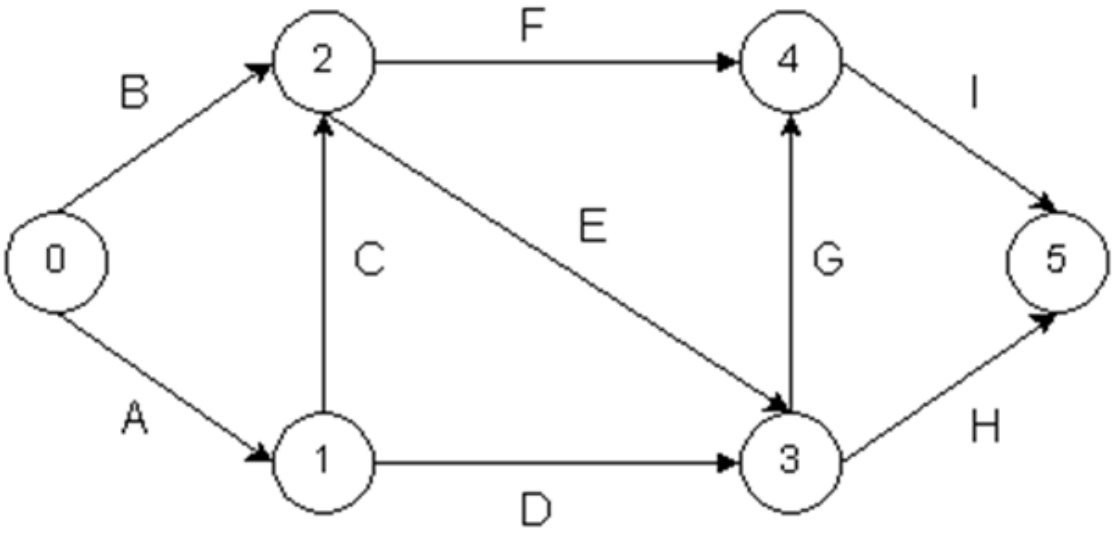
  Historically, CPM uses AON, PERT uses AOA

Activity-on-Arrow (AOA) Network

- Also called **Arrow Diagramming Method (ADM)**
- **Arrows:**
  - Arrows represent particular activities
  - Arrows show the activity sequencing
- **Nodes:**
  - Nodes are the starting and ending points of activities
    - AOA network diagram can only show finish-to-start dependencies (FS)
  - Nodes represent events, milestones, or the completion of one or a collection of activities
    - The first node signifies the start of a project, while the last node represents its end



Activity-on-Arrow - *Example*



Activity	Description	Predecessors
A	Site clearing	---
B	Removal of trees	---
C	General excavation	A
D	Grading general area	A
E	Excavation for utility trenches	B,C
F	Placing formwork and reinforcement for concrete	B,C
G	Installing sewer lines	D,E
H	Installing other utilities	D,E
I	Pouring concrete	F,G

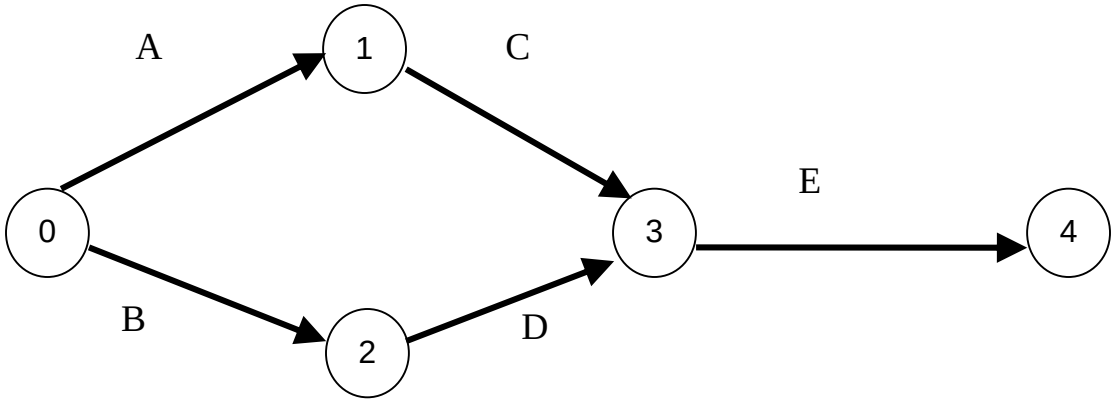
Process for Creating AOA Diagrams

1. Find all of the activities that start at node 0. Draw their finish nodes and draw arrows between node 0 and those finish nodes. Put the activity letter or name and duration estimate on the associated arrow.
2. Continue drawing the network diagram, working from left to right. Look for bursts and merges. **Bursts** occur when a single node is followed by two or more activities. A **merge** occurs when two or more nodes precede a single node.
3. Continue drawing the project network diagram until all activities that have dependencies are included on the diagram.
4. As a rule of thumb, all arrowheads should face toward the right, and no arrows should cross on an AOA network diagram.

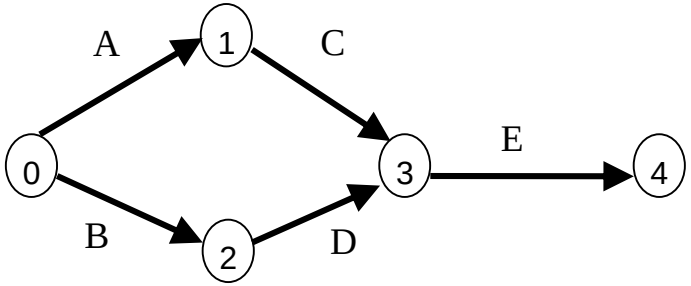
Dummy Activities in AOA Networks

Suppose a project consists of the five activities A, B, C, D and E that satisfy the following relationship:

1. Neither A nor B has any immediate predecessors
2. A is an immediate predecessor of C
3. **B is an immediate predecessor of D**
4. C and D are immediate predecessors of E



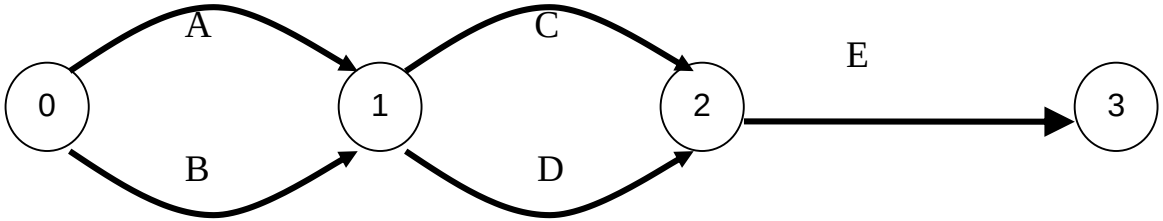
Dummy Activities in AOA Networks

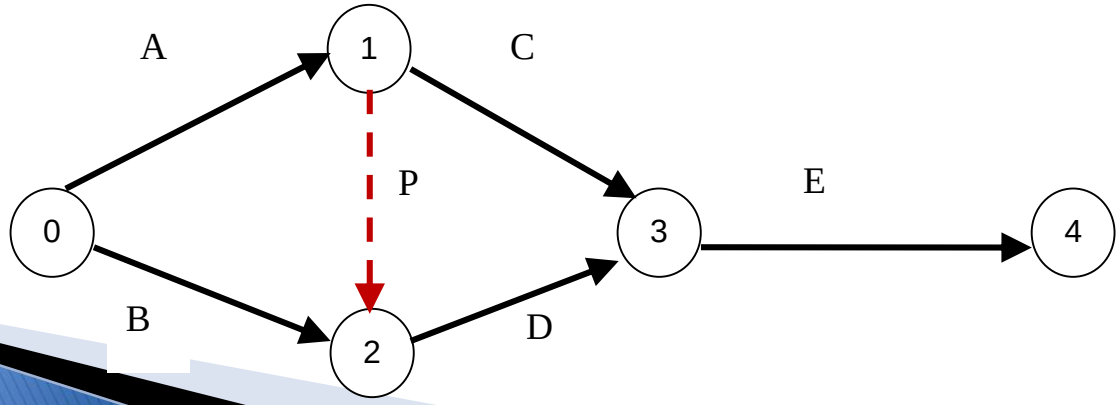


1. Neither A nor B has any immediate predecessors
2. A is an immediate predecessor of C

3'. A and B are immediate predecessors of D (revised)

4. C and D are immediate predecessors of E

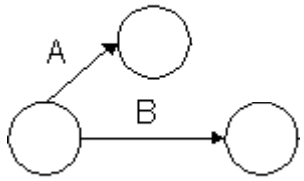


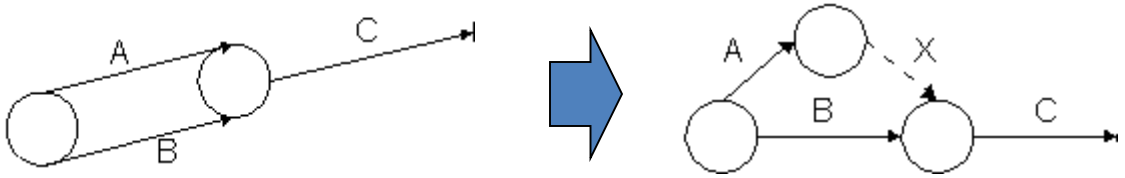


Dummy Activities in AOA Networks

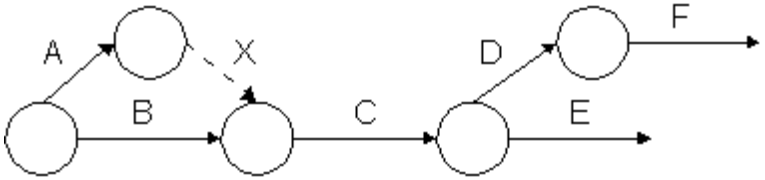
- **Dummy activity**
  - No duration and no resources
  - Graphically represented by a dashed line in a network
- **Purpose** of a dummy activity
  - Maintains the precedence relationships between activities
  - Provides unique activity designations
- Only in Activity-on-Arrow (AOA) networks!
  - AON networks do not need dummy activities

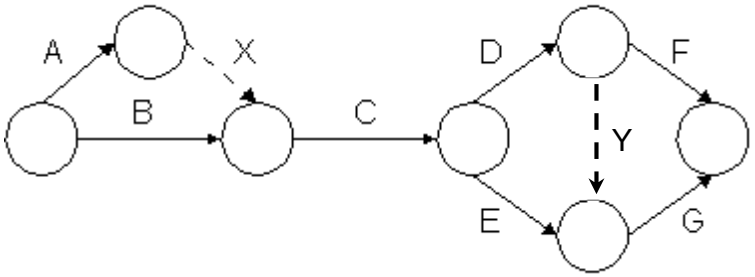
Dummy Activity – *Example 1*





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





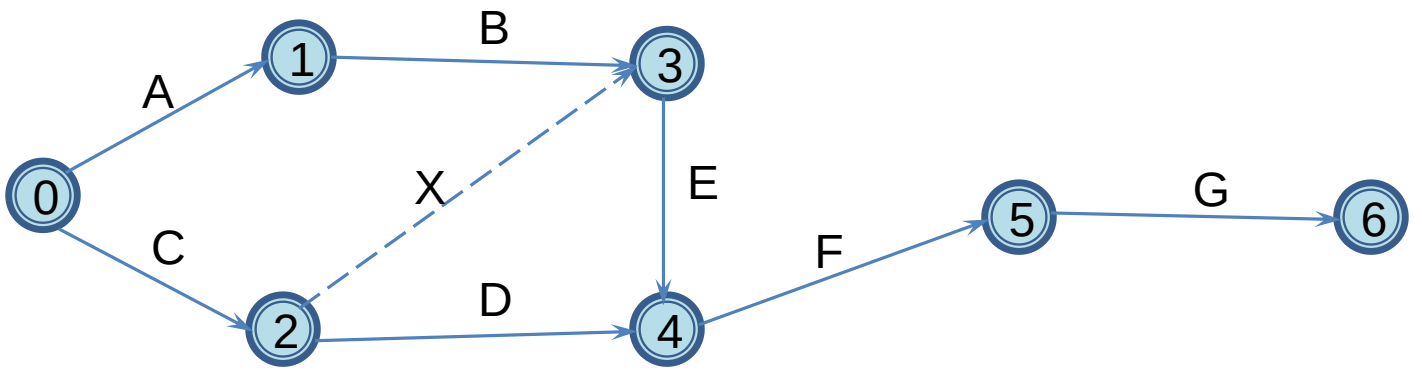
AOA Networks – *Example 2*

- Consider the activities associated with the fabrication of a steel component shown in table below. Draw the Activity-on-Arrow (AOA) network diagram for this project, determine the project duration and the critical path.

Precedence and durations of activities			
Activity	Description	Predecessors	Duration
A	Preliminary design	---	6
B	Evaluation of design	A	1
C	Contract negotiation	---	8
D	Preparation of fabrication plant	C	5
E	Final design	B, C	9
F	Fabrication of product	D, E	12
G	Shipment of product to owner	F	3

AOA Networks – *Example 2*

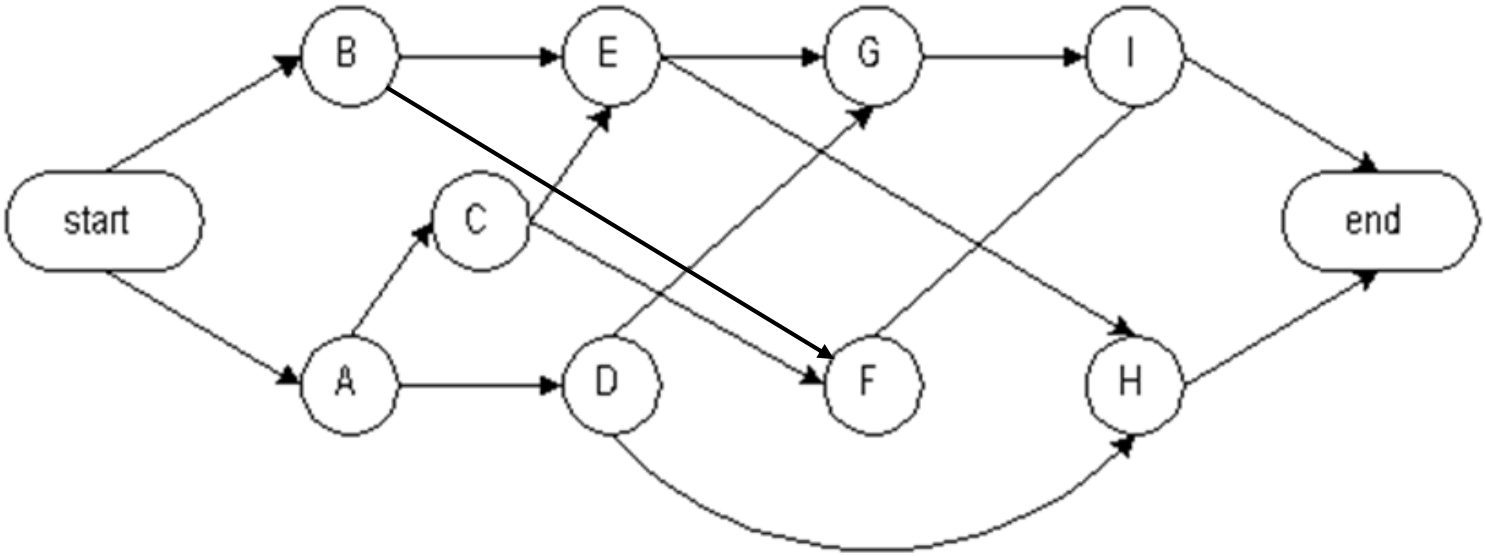
Precedence and durations of activities			
Activity	Description	Predecessors	Duration
A	Preliminary design	---	6
B	Evaluation of design	A	1
C	Contract negotiation	---	8
D	Preparation of fabrication plant	C	5
E	Final design	B, C	9
F	Fabrication of product	D, E	12
G	Shipment of product to owner	F	3



Activity-on-Node (AON) Network

- Also called **Precedence Diagramming Method (PDM)**
- **Nodes** (circles or boxes) represent activities
- **Arrows** represent precedence relationships
- More popular than ADM method and used by project management software
  - No need to use dummy activities
  - Can show different task dependency types (FS, SS, FF, SF)

Activity-on-Node Network- *Example*



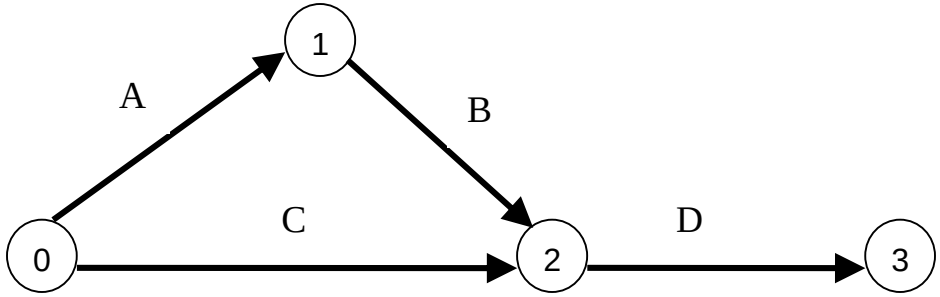
Activity	Description	Predecessors
A	Site clearing	---
B	Removal of trees	---
C	General excavation	A
D	Grading general area	A
E	Excavation for utility trenches	B,C
F	Placing formwork and reinforcement for concrete	B,C
G	Installing sewer lines	D,E
H	Installing other utilities	D,E
I	Pouring concrete	F,G

- New activity nodes representing the beginning and the end have been added

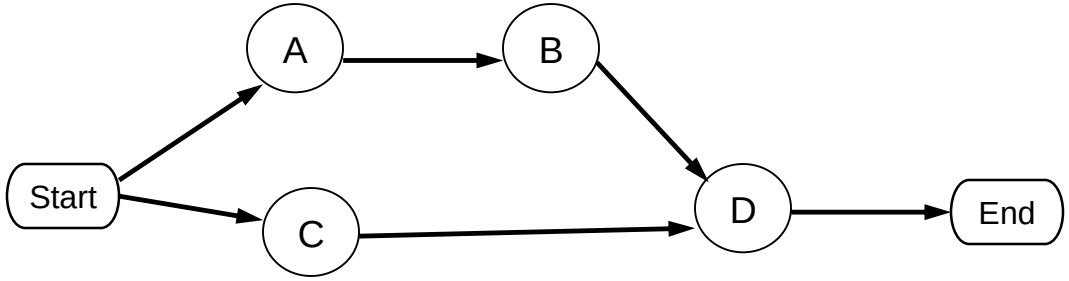
AOA and AON Networks

Activity	Predecessors
A	-
B	A
C	-
D	B, C

Activity-on-Arrow

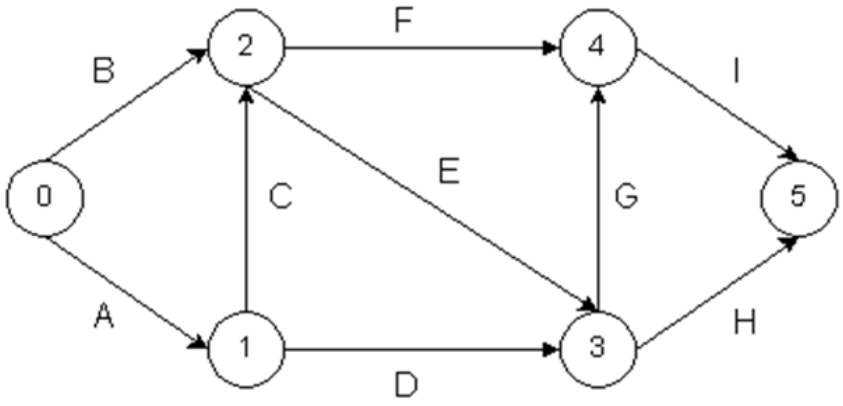


Activity-on-Node

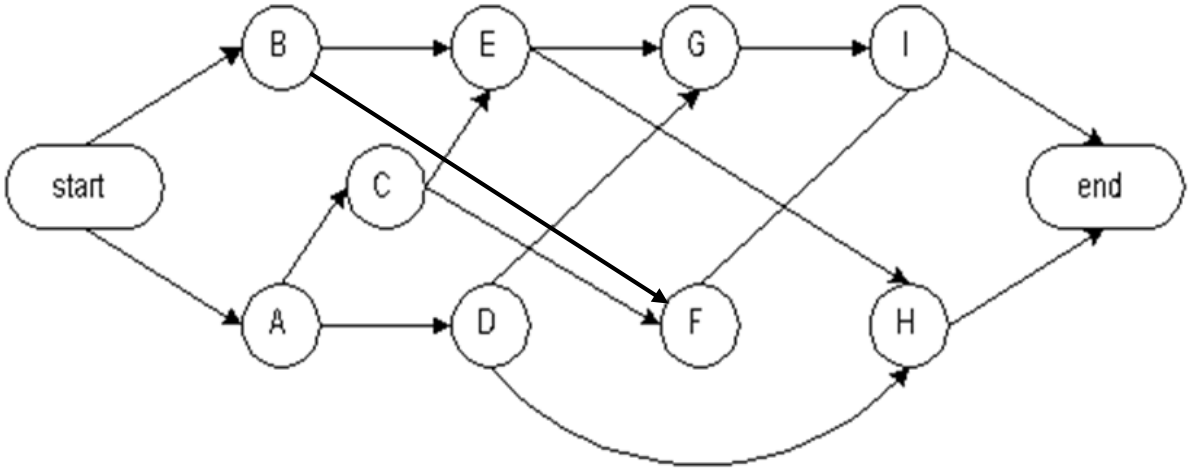


AOA and AON Networks

Activity-on-Arrow



Activity-on-Node



Estimating Activity Resources

- Determine the quantity and type of resources that will be assigned to each activity
  - Resources could be people, equipment or materials
- Consider important issues in estimating resources
  - The level of difficulty
  - The organization's history in doing similar activities
  - The availability and capability of the required resources
  - The evaluation of alternatives
  - The possibility of outsourcing
- A **resource breakdown structure** is a hierarchical structure that identifies the project's resources by category and type

Estimating Activity Duration

- **Duration** includes the actual amount of time worked on an activity *plus elapsed time*
- *Effort does not normally equal duration!*
  - Effort is the number of workdays or work hours required to complete a task
  - Duration of the task depends on the available resources
- People doing the work should help create estimates, and an expert should review them
- **Three-point estimate**
  - An estimate that includes three estimates:
    - Optimistic estimate
    - Most likely estimate
    - Pessimistic estimate
  - Three-point estimates are needed for PERT and Monte Carlo simulations

Developing a Schedule

- Determine the start and end date of the project
- Ultimate goal is to create a realistic project schedule that provides a basis for monitoring project progress for the time dimension of the project
- Important tools and techniques:
  - Gantt chart
  - Critical Path Method (CPM)
  - Critical Chain Scheduling
  - PERT analysis

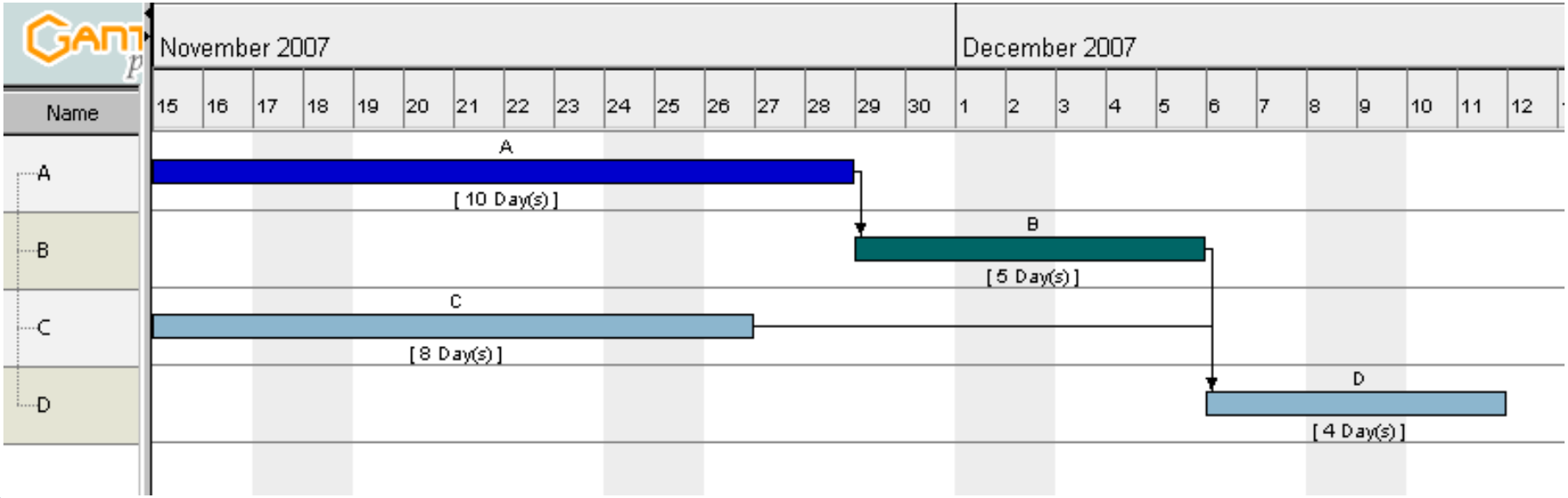
Gantt Chart

- **Gantt charts** provide a standard format for displaying project schedule information
- It is a graphical representation of the duration of tasks against the progression of time
- Gant charts monitor progress and provide immediate overview of current status
- A useful means of presenting the schedule
  - Not a very useful tool for generating schedule – dependencies are usually not shown

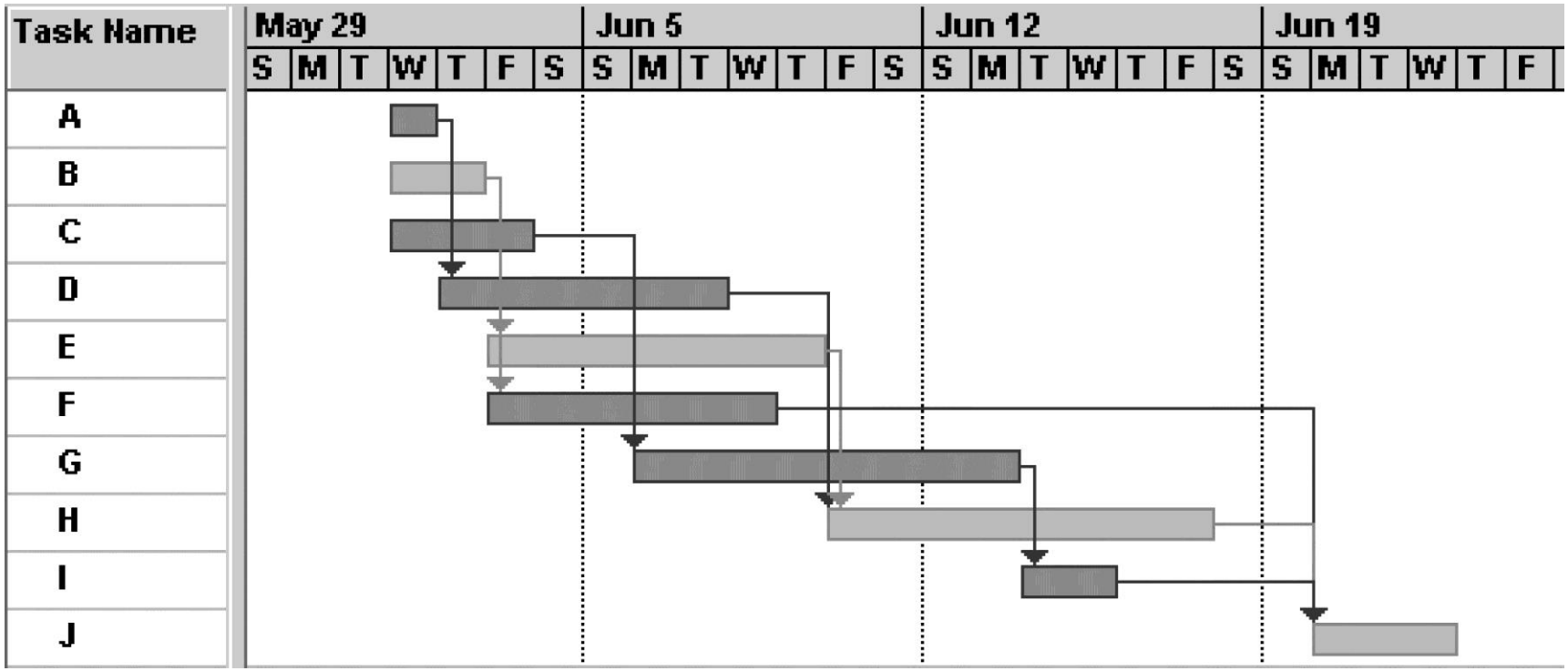
Henry L. Gantt
1861–1919

Gantt Chart

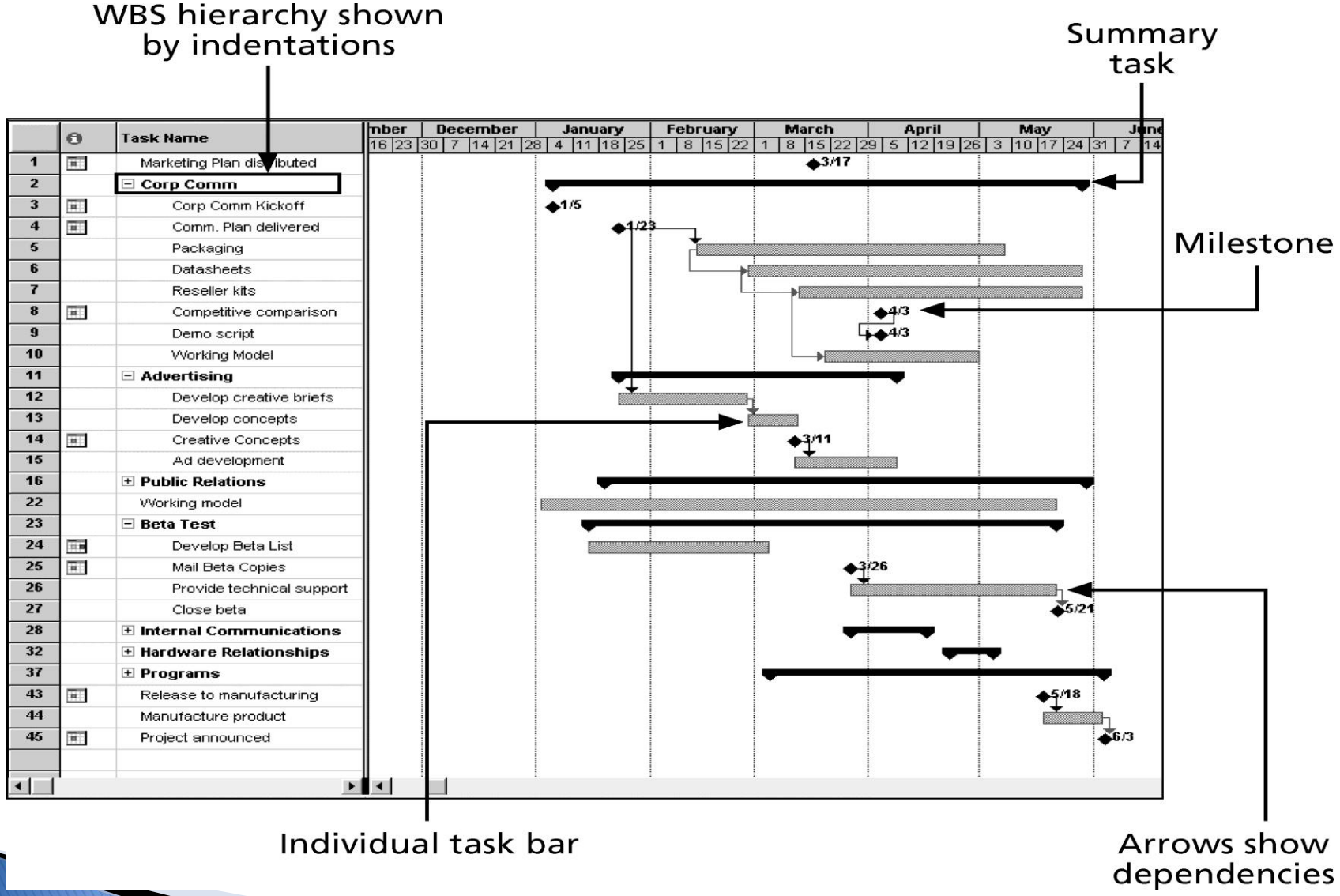
- **Inputs:** activities, durations, precedence relationships
  - Activity: represented by a bar
  - Duration of an activity: represented by the length of the bar
  - Precedence relationships: represented by arrows



Gantt Chart - *Example*



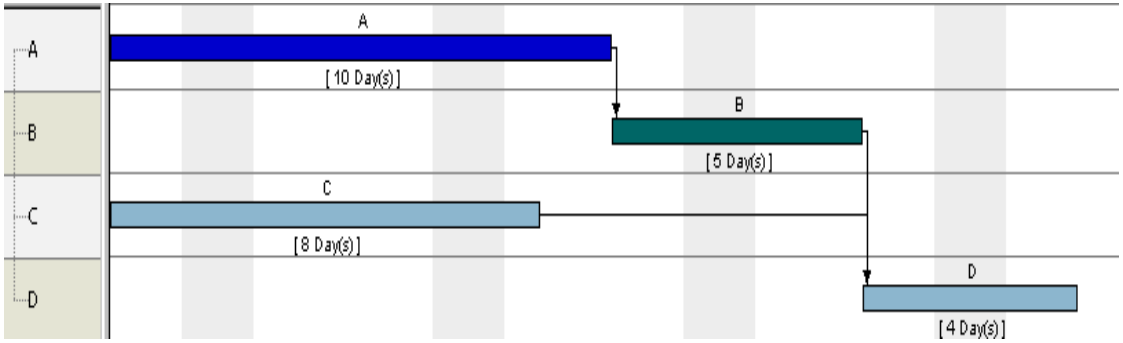
Gantt Chart - *Example*



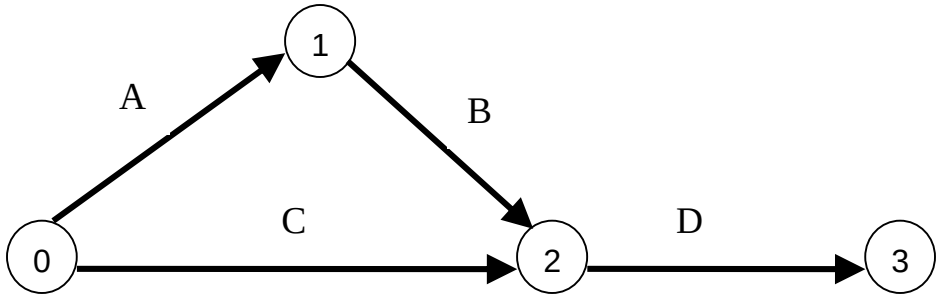
Gantt chart, AOA and AON Networks

Activity	Predecessors
A	-
B	A
C	-
D	B, C

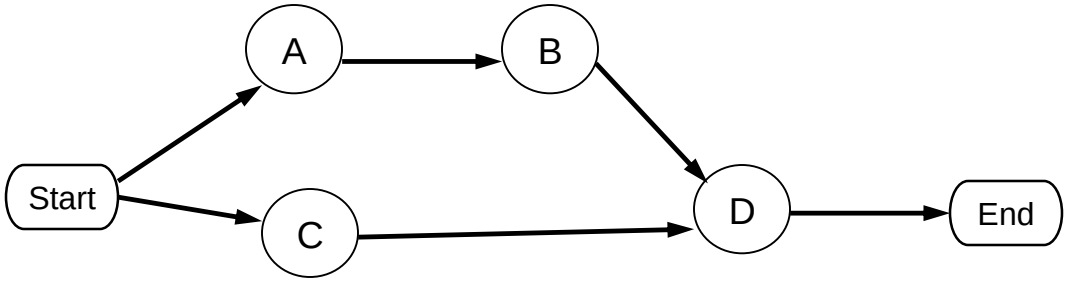
Gantt chart



Activity-on-Arrow



Activity-on-Node



Milestones

- A **milestone** is a significant event that normally has no duration
- It often takes several activities and a lot of work to complete a milestone
- They are useful tools for setting schedule goals and monitoring progress
- Examples include obtaining customer sign-off on key documents or completion of specific products
- Adding milestones to Gantt Chart
  - Many people like to focus on meeting milestones, especially for large projects
  - Milestones emphasize important events or accomplishments on projects
  - Normally create milestone by entering tasks with a zero duration, or you can mark any task as a milestone

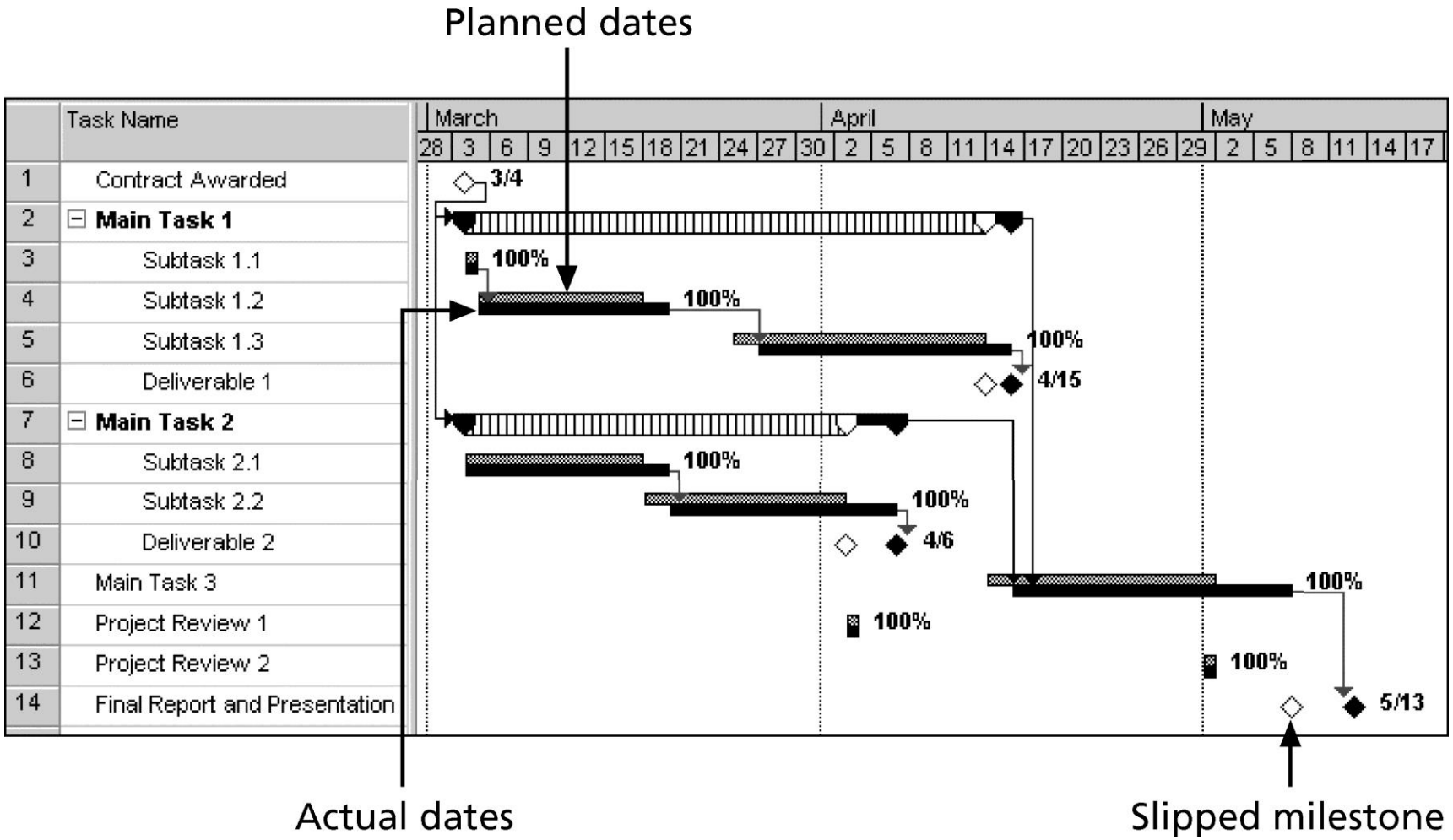
Milestones

- Milestones should be SMART:
  - **S**pecific
  - **M**easurable
  - **A**ssignable
  - **R**ealistic
  - **T**ime-framed
- Define milestones early in the project and include them in the Gantt chart to provide a visual guide.
  - Keep milestones small and frequent.
  - The set of milestones must be all-encompassing.
  - Carefully monitor the critical path.
  - Each milestone must be binary, meaning it is either complete  or incomplete 

Tracking Gantt Chart

- **Tracking Gantt Chart** is a progress evaluation tool which compares planned and actual project schedule information
- The planned schedule dates for activities are called **baseline dates** (baseline start, baseline finish)
- The entire approved planned schedule is called **schedule baseline**

Tracking Gantt Chart - *Example*



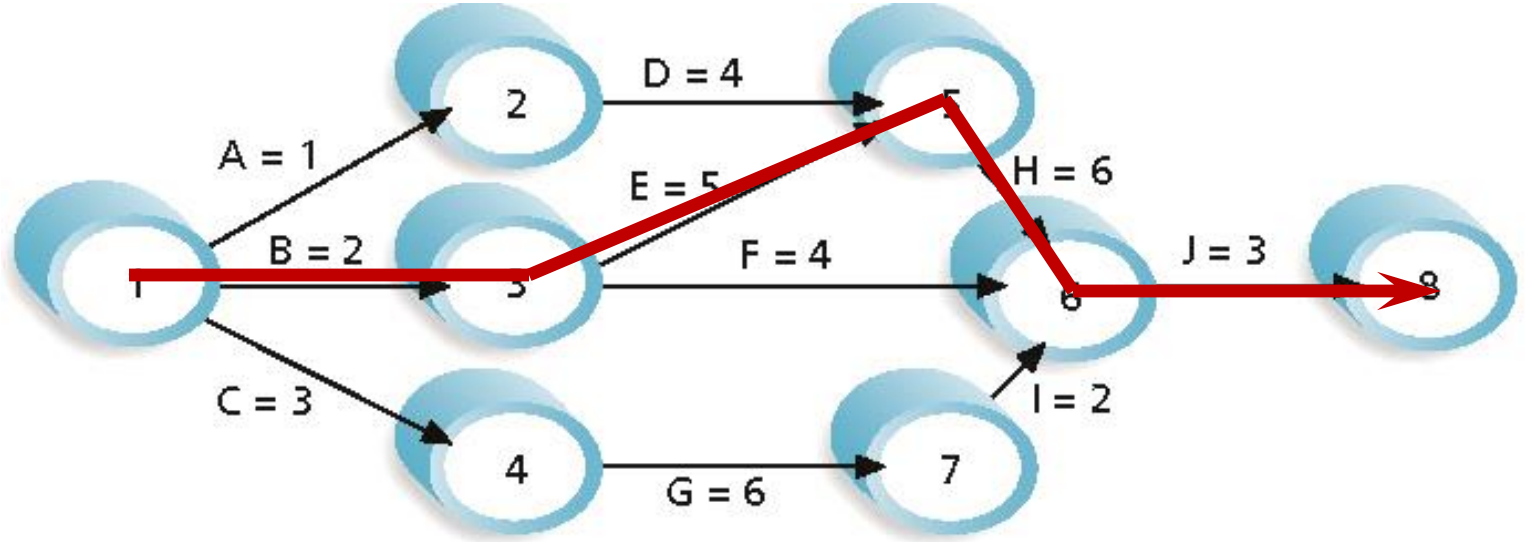
Critical Path Method (CPM)

- **CPM** is a network diagramming technique used to predict total project duration
- A **critical path** for a project is the series of activities that determines the *earliest time* by which the project can be completed
- *The critical path is the **longest path** through the network diagram and has the least amount of slack or float*
- **Slack** or **float** is the amount of time an activity may be delayed without delaying a succeeding activity or the project finish date

Calculating the Critical Path

- First develop a good network diagram
- Add the duration estimates for all activities on each path through the network diagram
- **The longest path is the critical path**
- If one (or more) of the activities on the critical path takes longer than planned, the whole project schedule will slip *unless* the project manager takes corrective action
- It is important to update project schedule information to meet time goals for a project
  - The critical path may change as the project progresses
  - If you know the project completion date will slip, negotiate with the project sponsor

Critical Path Method - *Example AOA*



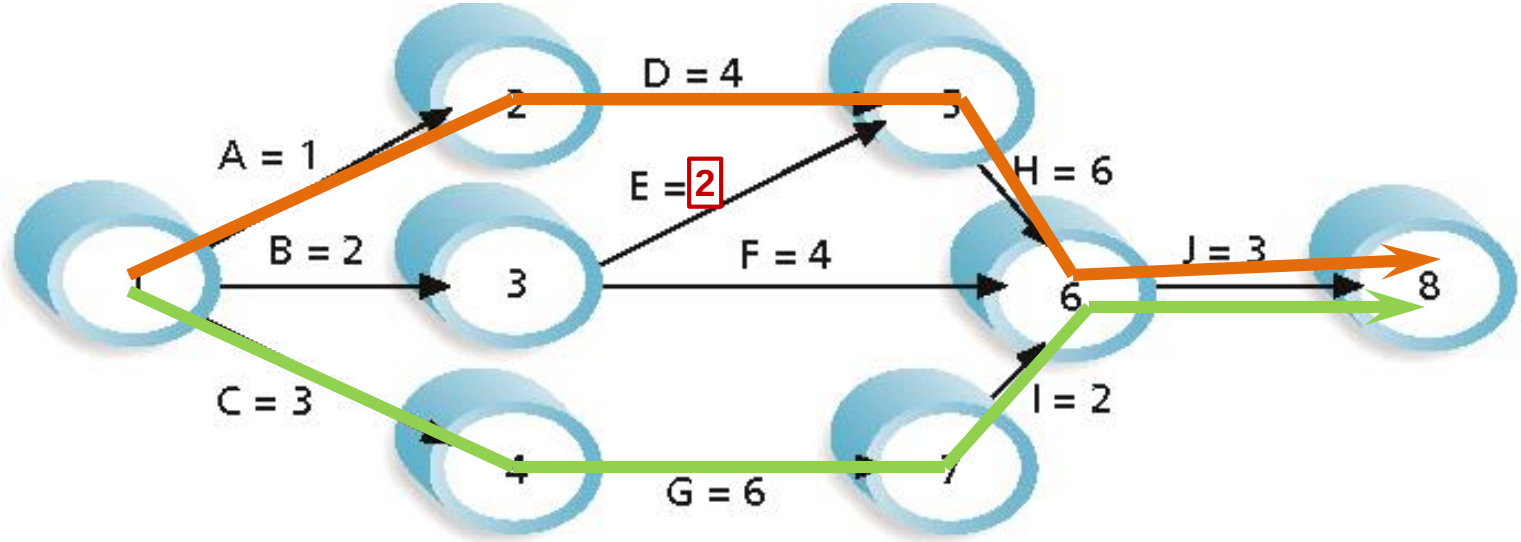
Note: Assume all durations are in days.

Path 1:	A-D-H-J	Length = 1+4+6+3 = 14 days
Path 2:	B-E-H-J	Length = 2+5+6+3 = 16 days
Path 3:	B-F-J	Length = 2+4+3 = 9 days
Path 4:	C-G-I-J	Length = 3+6+2+3 = 14 days

Since the critical path is the longest path through the network diagram, Path 2, B-E-H-J, is the critical path for Project X.

Critical Path Method - *Example AOA*

There can be more than one critical path if the lengths of two or more paths are the same!



Note: Assume all durations are in days.

Path 1:	A-D-H-J	Length = 1+4+6+3 = 14 days
Path 2:	B-E-H-J	Length = **2 + 2 + 6 + 3 = 13 days**
Path 3:	B-F-J	Length = 2+4+3 = 9 days
Path 4:	C-G-I-J	Length = 3+6+2+3 = 14 days

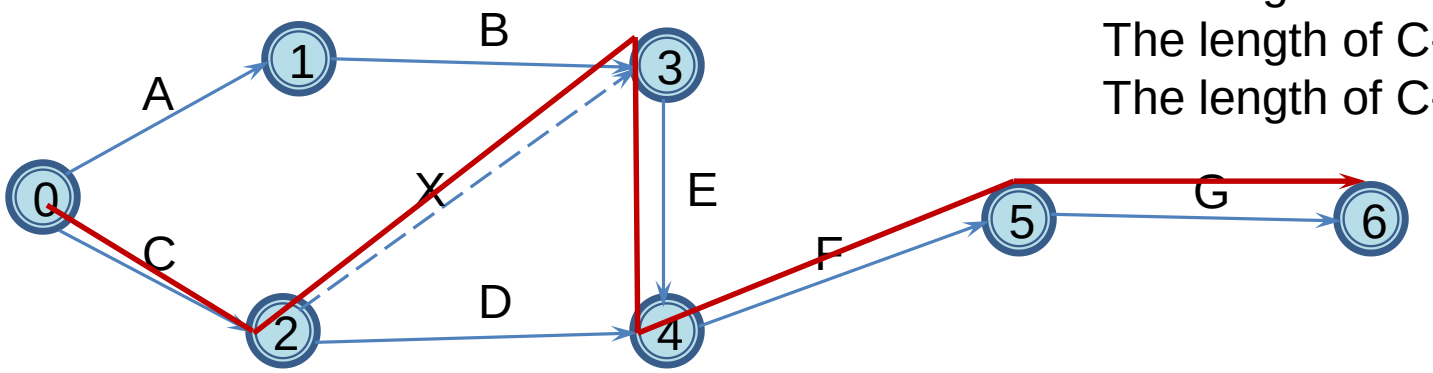
Critical Path Method

- There can be more than one critical path if the lengths of two or more paths are the same

- Common misconceptions:
  - The critical path is ***not*** the one with the most critical activities! Critical path is concerned with the time dimension of a project
  - The critical path is ***not*** the shortest path through the diagram!

AOA Networks – *Previous example*

Precedence and durations of activities			
Activity	Description	Predecessors	Duration
A	Preliminary design	---	6
B	Evaluation of design	A	1
C	Contract negotiation	---	8
D	Preparation of fabrication plant	C	5
E	Final design	B, C	9
F	Fabrication of product	D, E	12
G	Shipment of product to owner	F	3



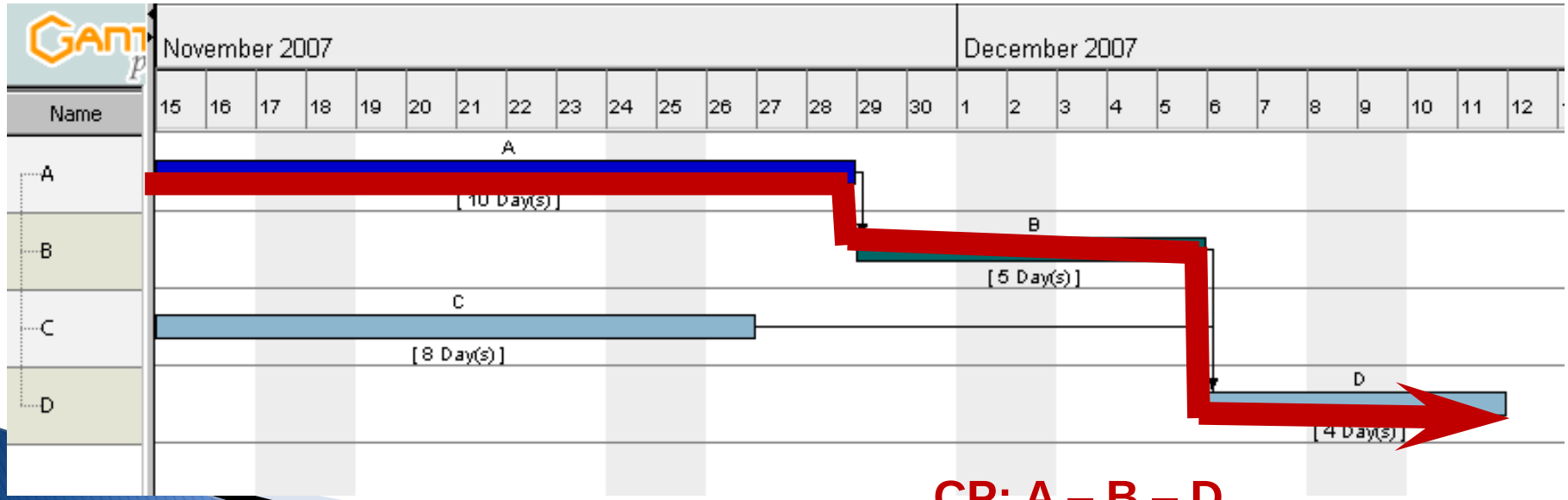
The length of A-B-E-F-G is 31
The length of C-X-E-F-G is 32
The length of C-D-F-G is 28

The critical path is **C – (X) – E – F – G**

The project duration is **32 days.**

CPM on the Gantt Chart

- How to find the critical path on the Gantt Chart?
  - Schedule all the activities at their earliest time
  - Determine the earliest time the project can finish
    - Corresponds to the longest path through the project
    - Activities on the critical path do not have slack time



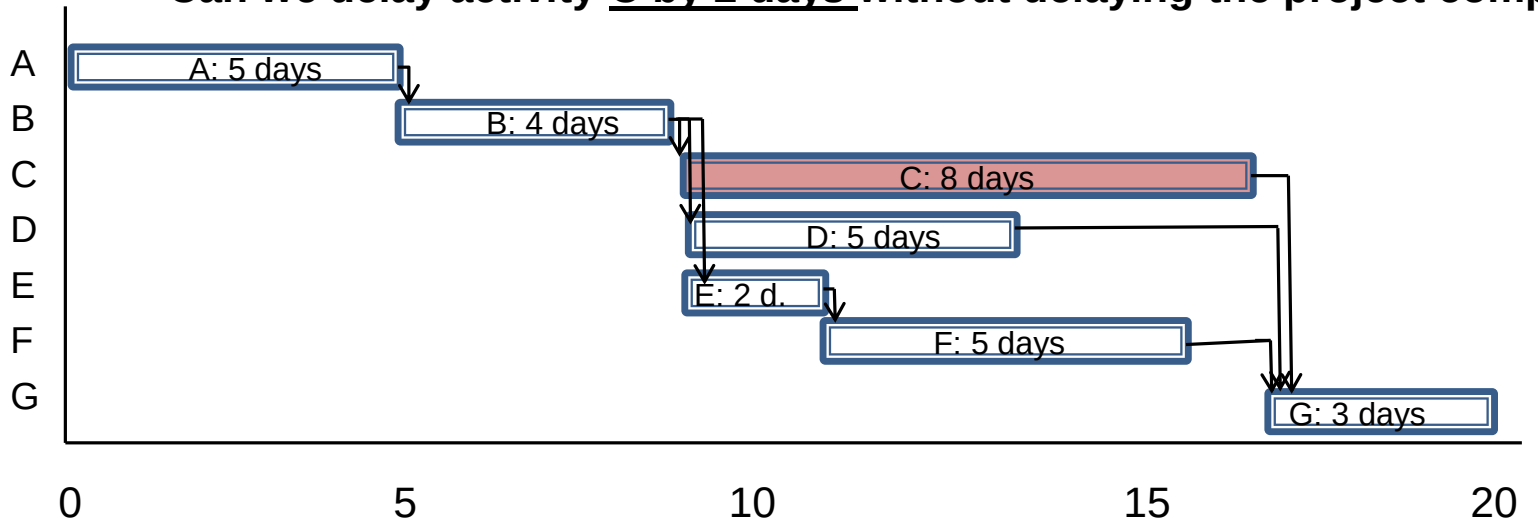
CP: A – B – D
CP = 19 days

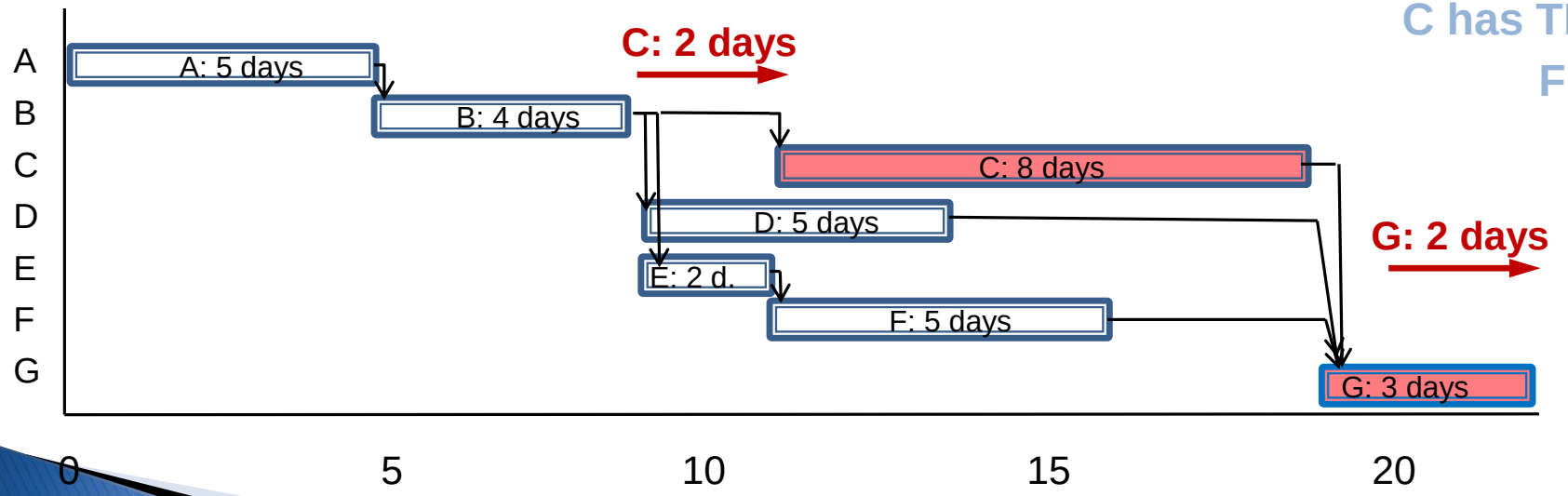
Floats

- **Slack** or **float** is the amount of time an activity may be delayed without delaying a succeeding activity or the project finish date
  - **Total Float (TF)** is the amount of time an activity can be delayed without delaying a project completion time
  - **Free Float (FF)** is the amount of time an activity can be delayed without delaying a next activity
- Activities on the critical path do not have floats!

Critical Path - *Example*

- **Can we delay activity C by 2 days without delaying the project completion?**





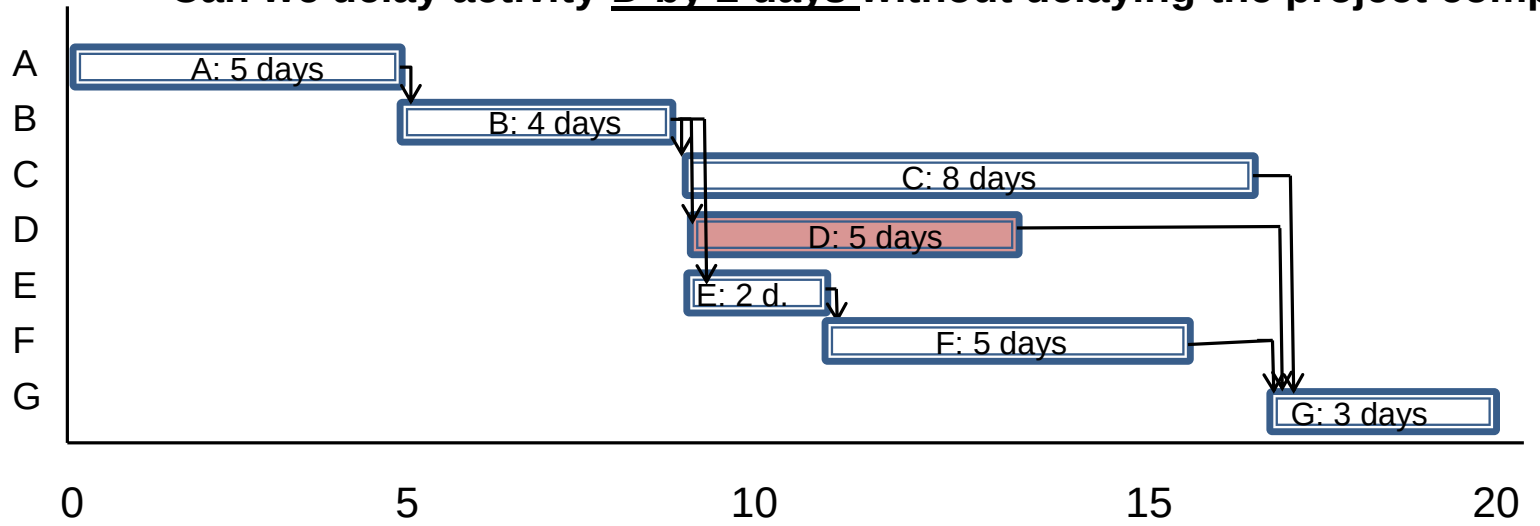
C has TF = 0

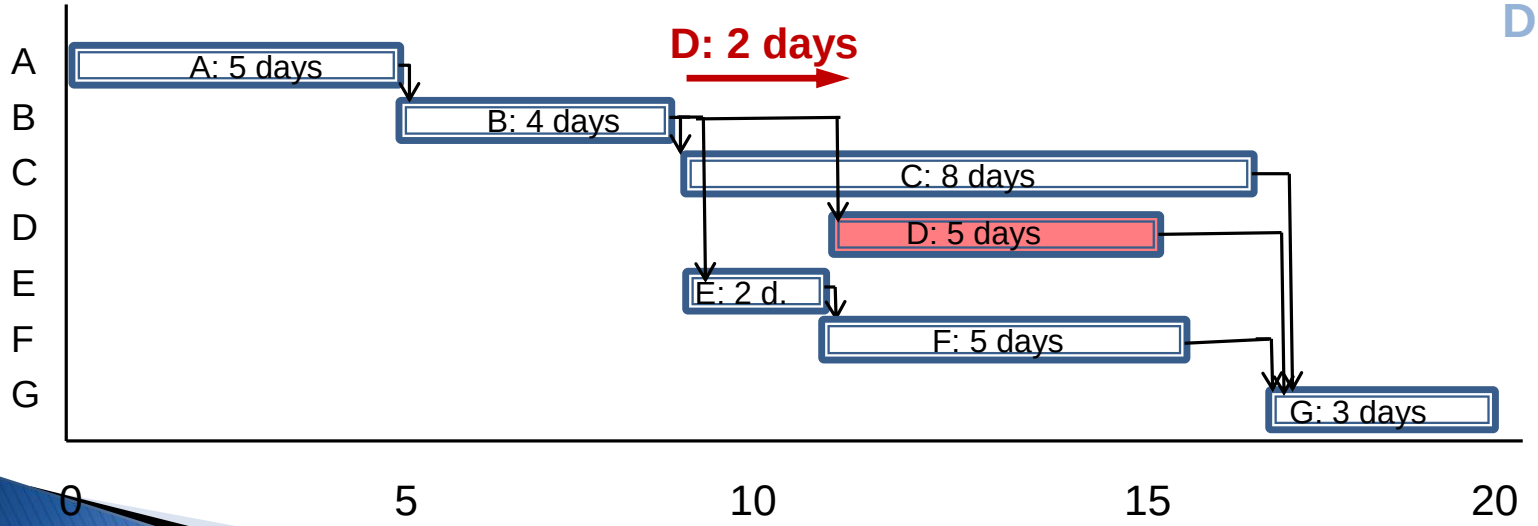
FF = 0

- **NO, C is on the CP and TF=0**
- **CP = 22 now**

Critical Path - *Example*

- **Can we delay activity D by 2 days without delaying the project completion?**





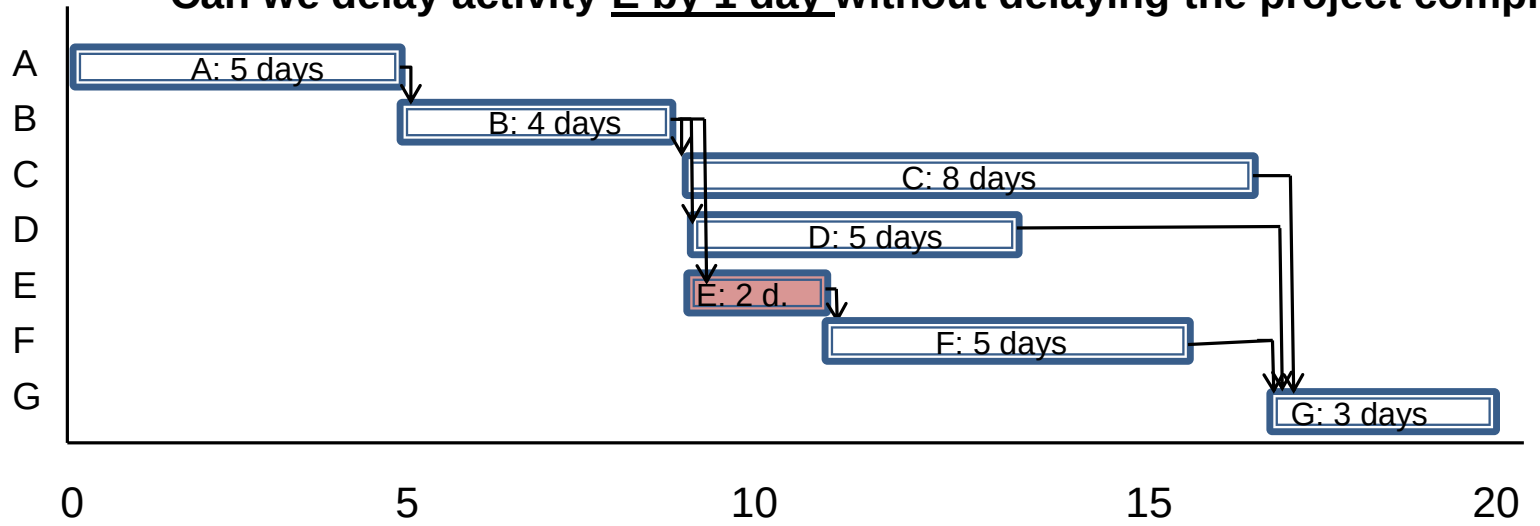
D has TF = 3

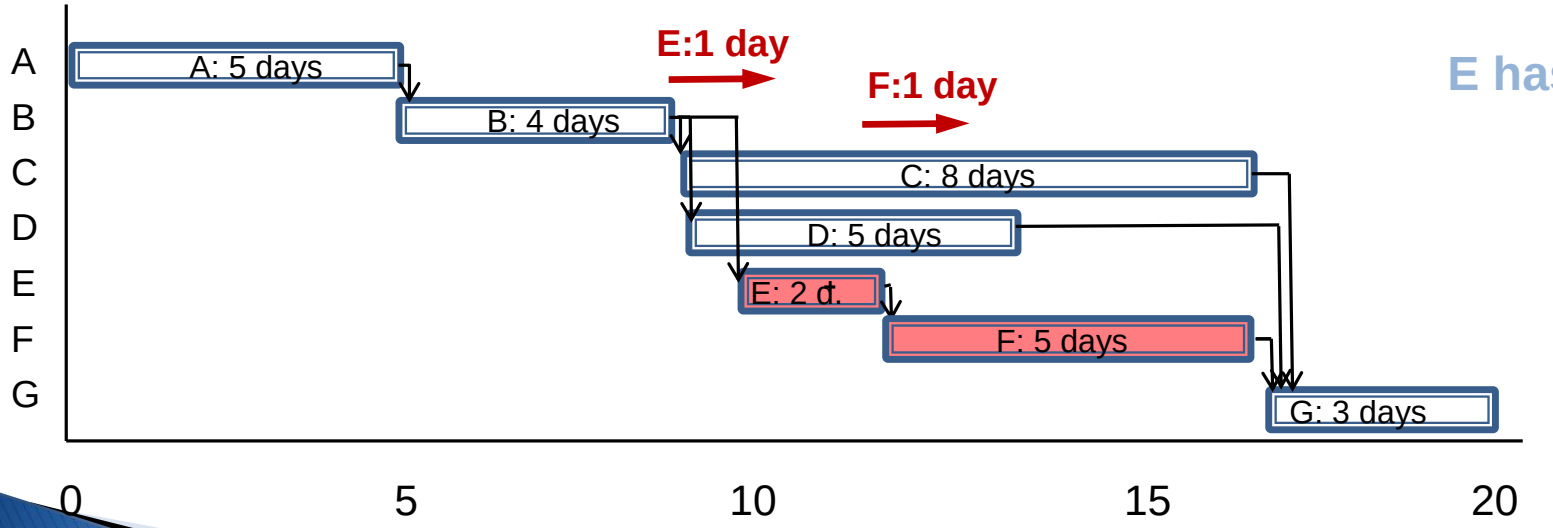
FF = 3

- **YES, D is not on the CP and TF=3**
- **CP = 20**

Critical Path - *Example*

- Can we delay activity <u>E by 1 day</u> without delaying the project completion?

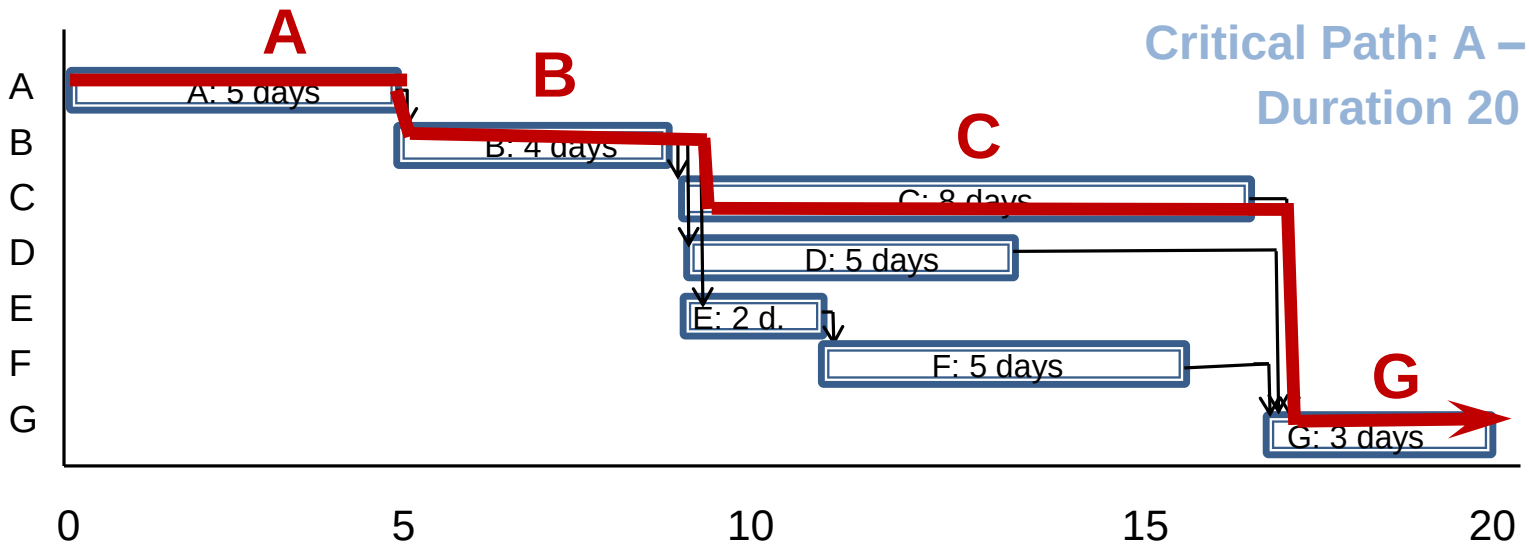




E has TF = 1
FF = 0

- **YES, E is not on the CP and TF=1**
- **CP = 20**
- **However, F was delayed, because FF=0**

Critical Path Method (CPM)



Critical Path: A – B – C – G
Duration 20 days

- Activities on a critical path do not have floats:
  - TF = 0
  - FF = 0
- **Schedule Flexibility:** The concept of float is to use part or all of this allowable range to schedule an activity without delaying the completion of the project

Critical Path Method for AON

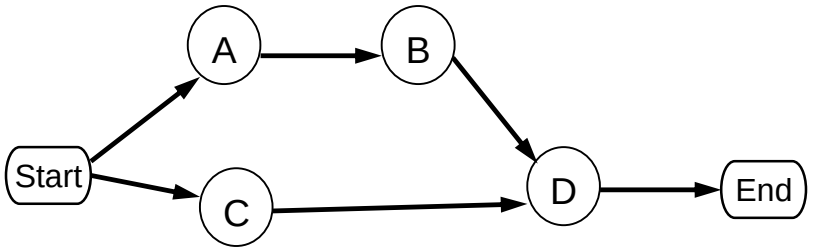
Early and Late Start and Finish

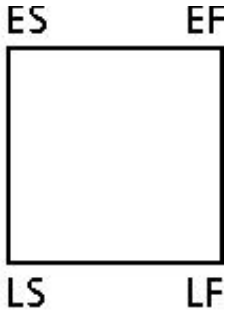
- **Duration *D(i)***
  - estimated time necessary to perform an activity
- **Early Start *ES(i)***
  - earliest time an activity can start
- **Early Finish *EF(i)***
  - earliest time an activity can finish:

$$EF(i) = ES(i) + D(i)$$

- **Late Finish *LF(i)***
  - latest time an activity can be finished without delaying the completion date of the project
- **Late Start *LS(i)***
  - latest time that an activity can be started without delaying the completion date of the project:

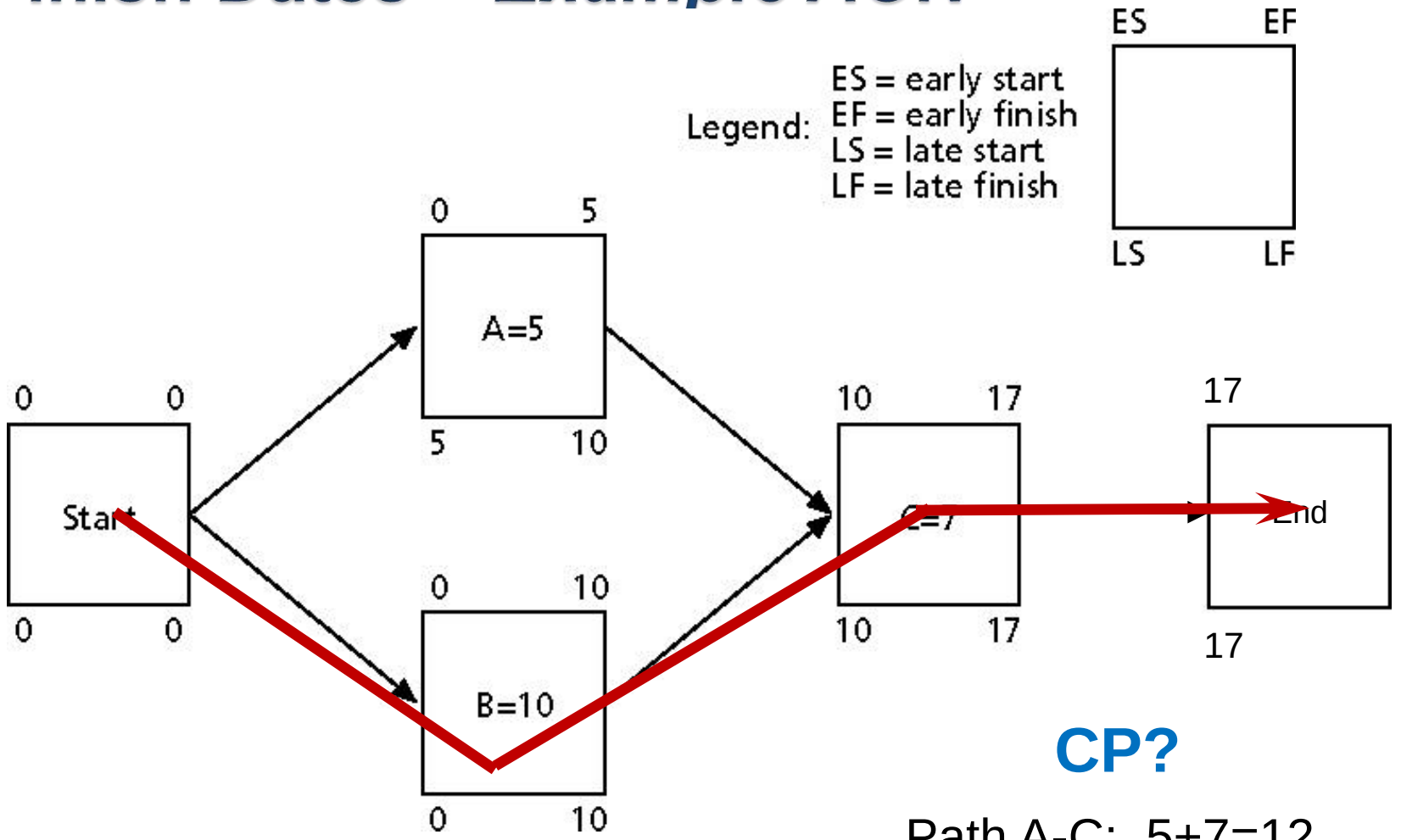
$$LS(i) = LF(i) - D(i)$$





ES		EF
Activity Number: ID		
LS	D	LF

Calculating Early and Late Start and Finish Dates – *Example AON*

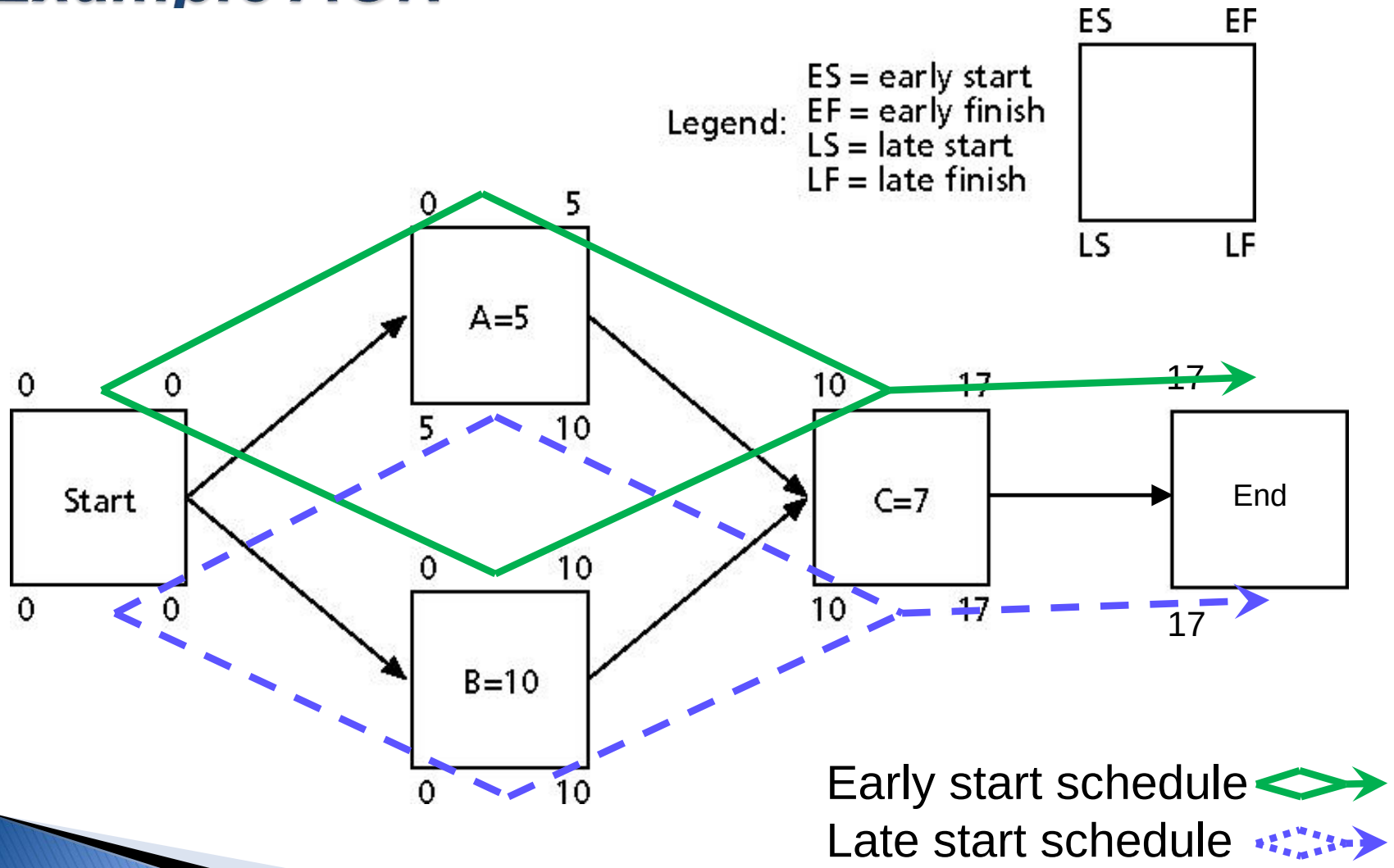


CP?

Path A-C: 5+7=12

Path B-C: 10+7=17

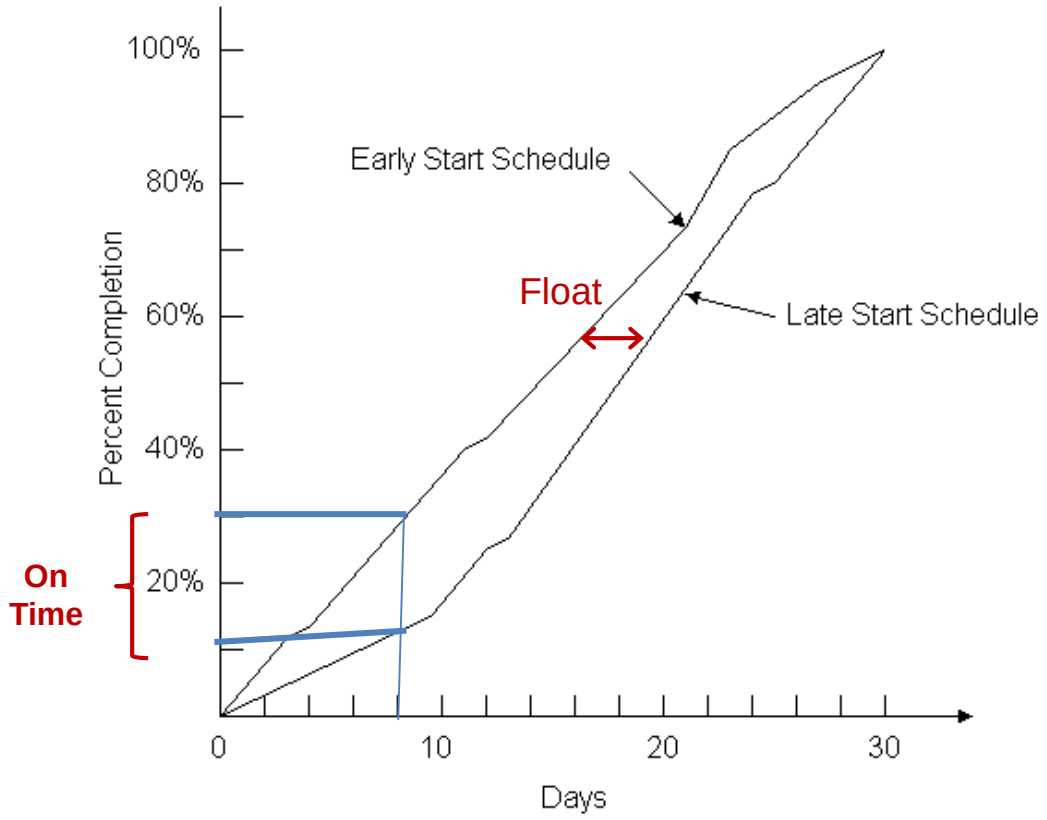
Early and Late Start Schedules – *Example AON*



Early start schedule

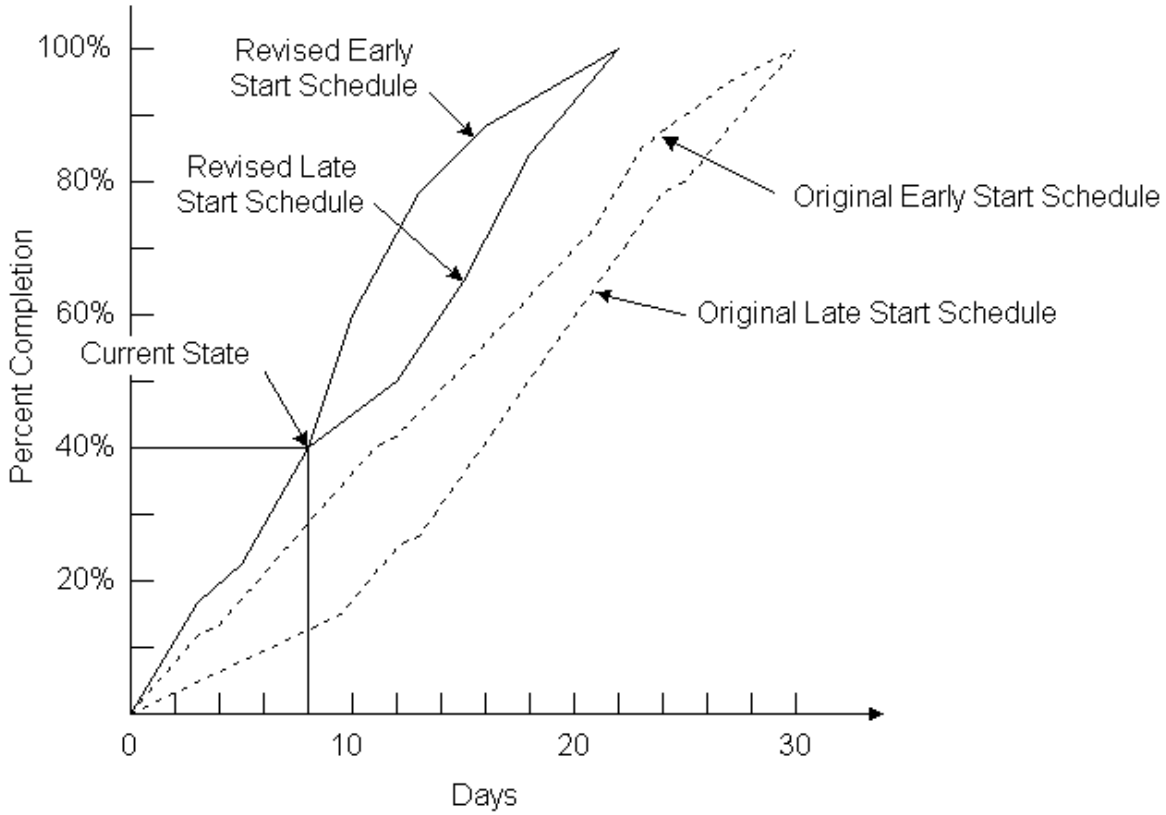
Late start schedule

Percentage Completion Based on Early and Late Start and Finish Dates - *Example*



- **Early start schedule:**
  - each activity is scheduled at its earliest start time, *ES(j)*
- **Late start schedule:**
  - each activity is scheduled at its latest start time, *LS(j)*

Percentage Completion Based on Revised Early and Late Start and Finish Dates - *Example*



CPM with AON Networks

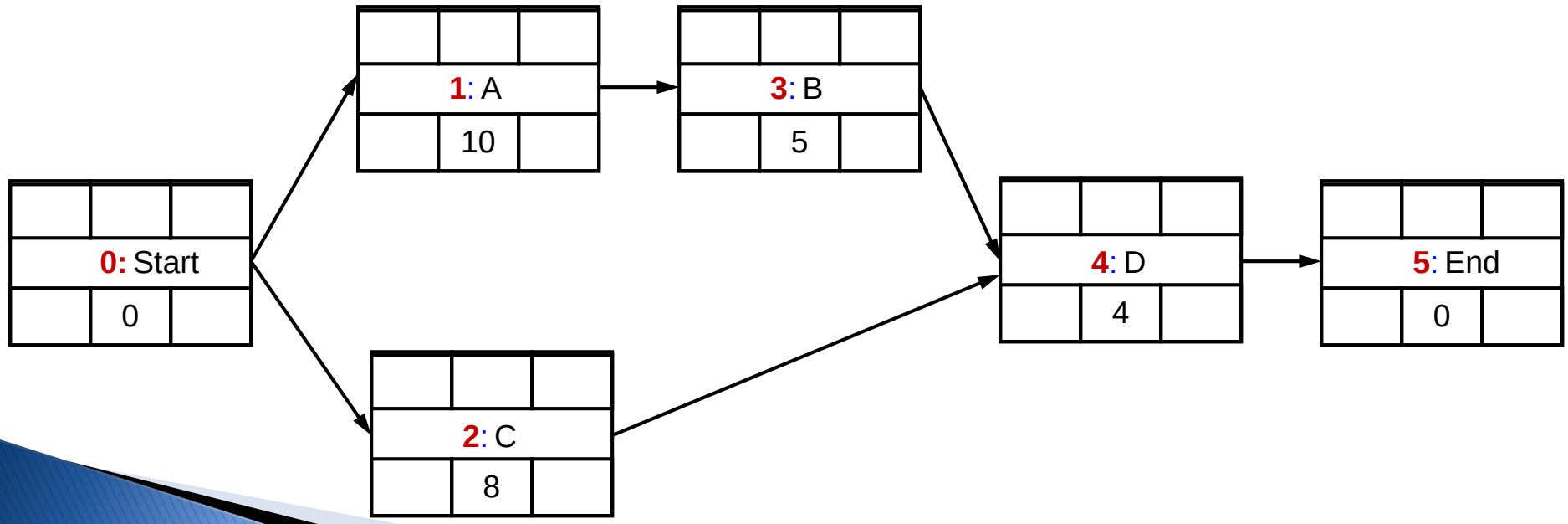
Steps for CPM Algorithm (AON)

1. Activity Numbering
2. Forward Pass *(ES & EF)*
3. Backward Pass *(LF & LS)*
4. Identifying the Critical Path
5. Float times

CPM with AON: Step 1. Activity Numbering

ES		EF
Number: ID		
LS	D	LF

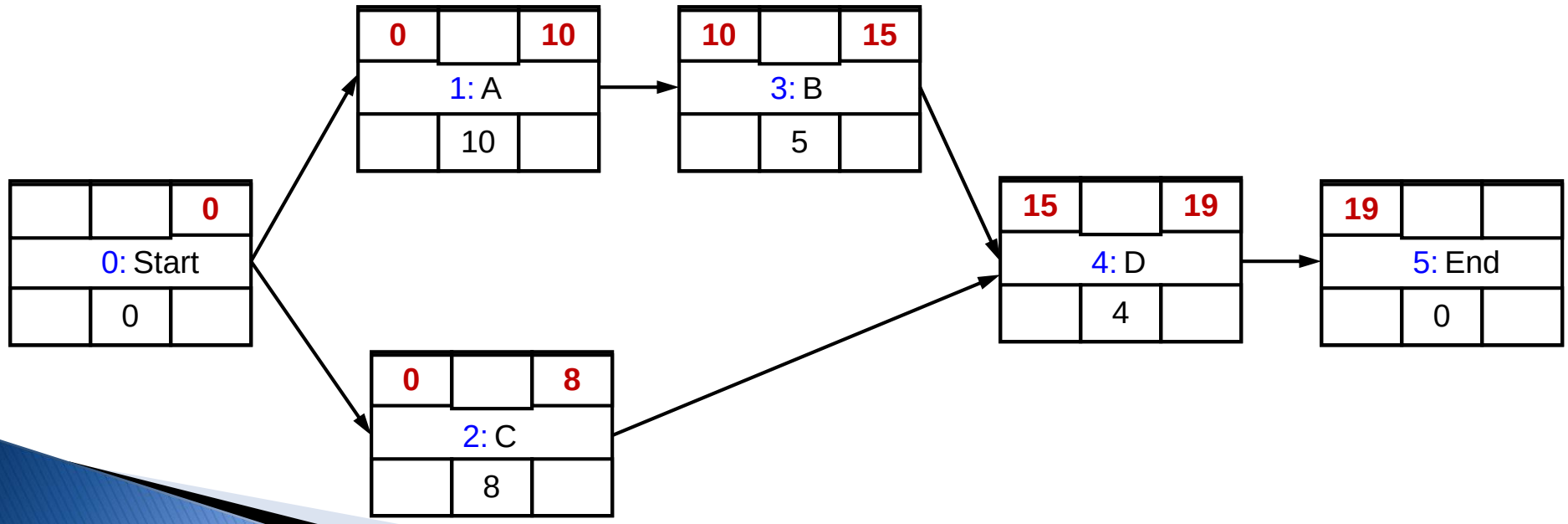
- *Step 1*: Give the starting activity number 0.
- *Step 2*: Give the next number to any unnumbered activity whose predecessor activities are each already numbered. (*e.g.*, activity 0 is a predecessor activity for 1 and 2)
  Repeat Step 2 until all activities are numbered.



CPM with AON: Step 2. Forward Pass

ES		EF
Number: ID		
LS	D	LF

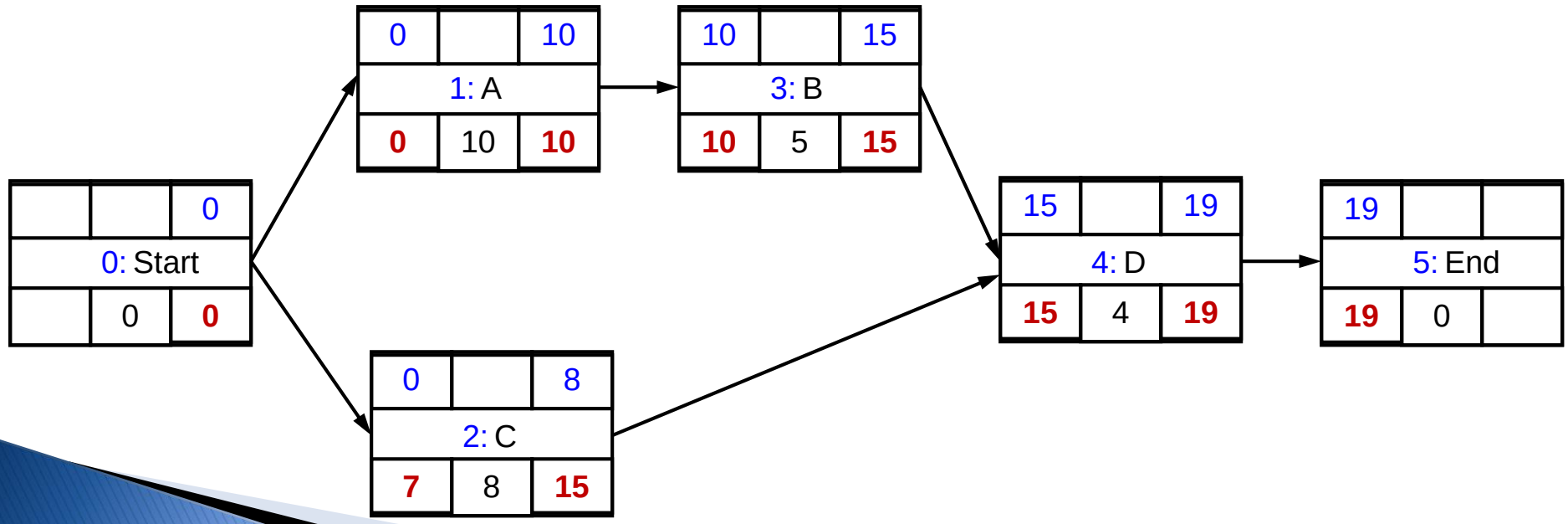
- *Step 1*: Let *ES(0) = 0*
- *Step 2*: For *j = 1,2,3,...,n* (where *n* is the End node), let
  ES(j) = maximum {EF(i)}
  where the maximum is computed over all activities (*i*) that have *j* as their successor.
- *Step 3*: $EF(j) = ES(j) + D_j$



CPM with AON: Step 3. Backward Pass

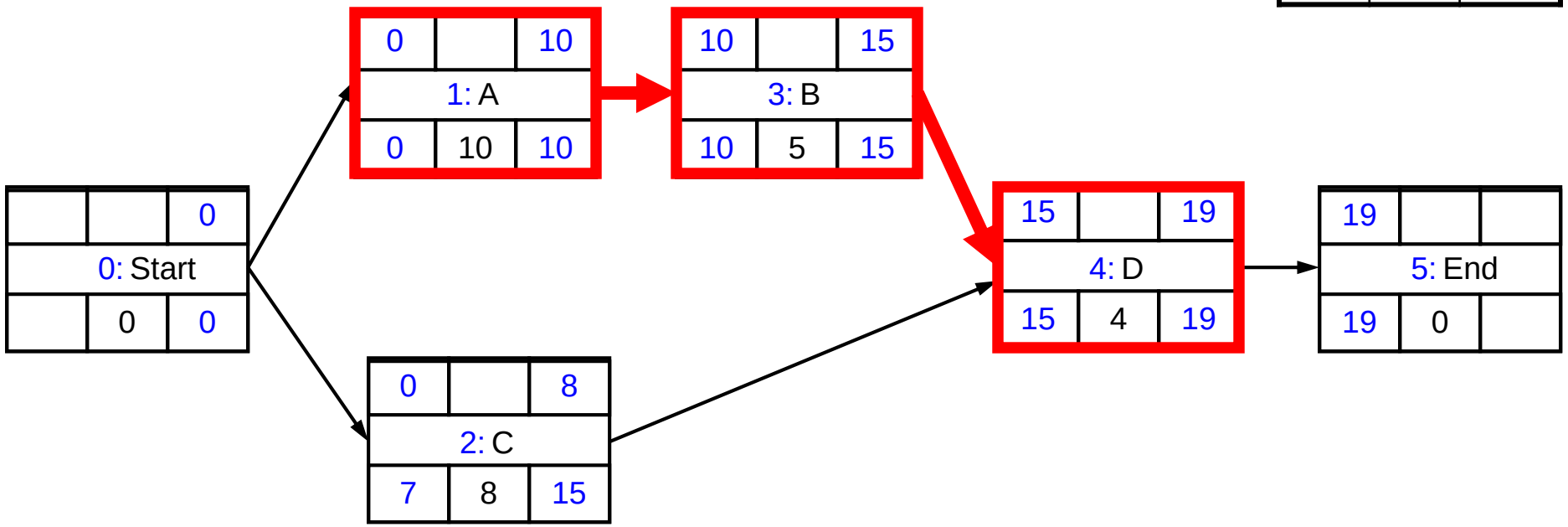
ES		EF
Number: ID		
LS	D	LF

- *Step 1*: Let *LS(n)* equal the required completion time of the project.
  (*Let LS(n) = ES(n)*)
- *Step 2*: For *i = n-1, n-2, ..., 0*, let
  LF(i) = minimum {LS(j)}
  where the minimum is computed over all activities (*j*) that have *i* as their predecessor.
- *Step 3*: *LS(i) = LF(i) - D_i*



CPM with AON: Step 4. Critical Path

ES		EF
Number: ID		
LS	D	LF



Path A-B-D: 10+5+4=19

Path C-D: 8+4=12

Condition: *ES(i) = LS(i)*

CPM with AON: Step 5. Float Times

ES		EF
Number: ID		
LS	D	LF

- **Total float (TF) = Total slack**
  the maximum amount of delay which can be assigned to any activity **without delaying the entire project**

$$TF(j) = LS(j) - ES(j)$$

- **Free float (FF) = Free slack**
  the amount of delay which can be assigned to any activity **without delaying subsequent activities**

$$FF(j) = \min_k \{ES(k)\} - EF(j)$$

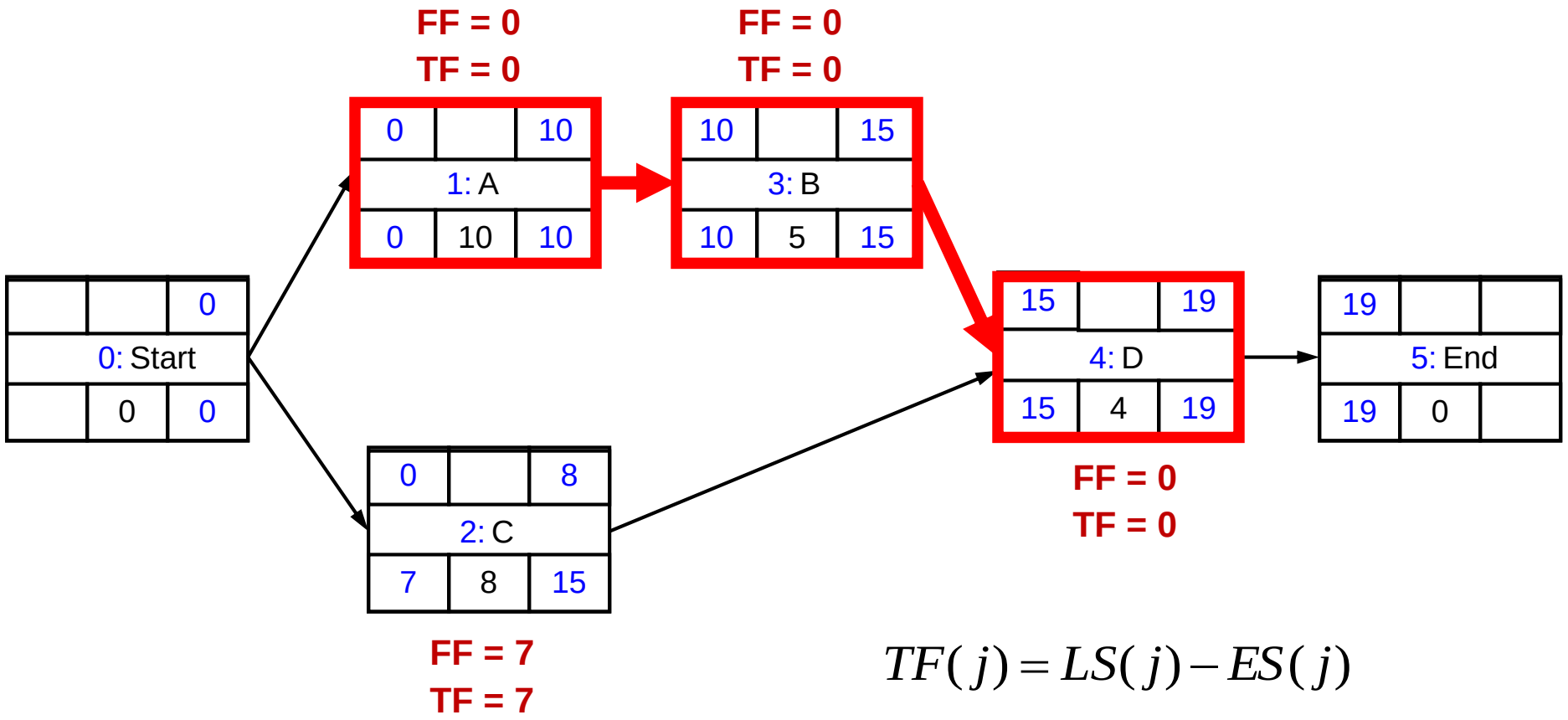
$$k : \text{the successor activities of activity } j$$

- For activities on critical path: $TF(j) = 0$

$$FF(j) = 0$$

CPM with AON: Step 5 Float Times

ES		EF
Number: ID		
LS	D	LF



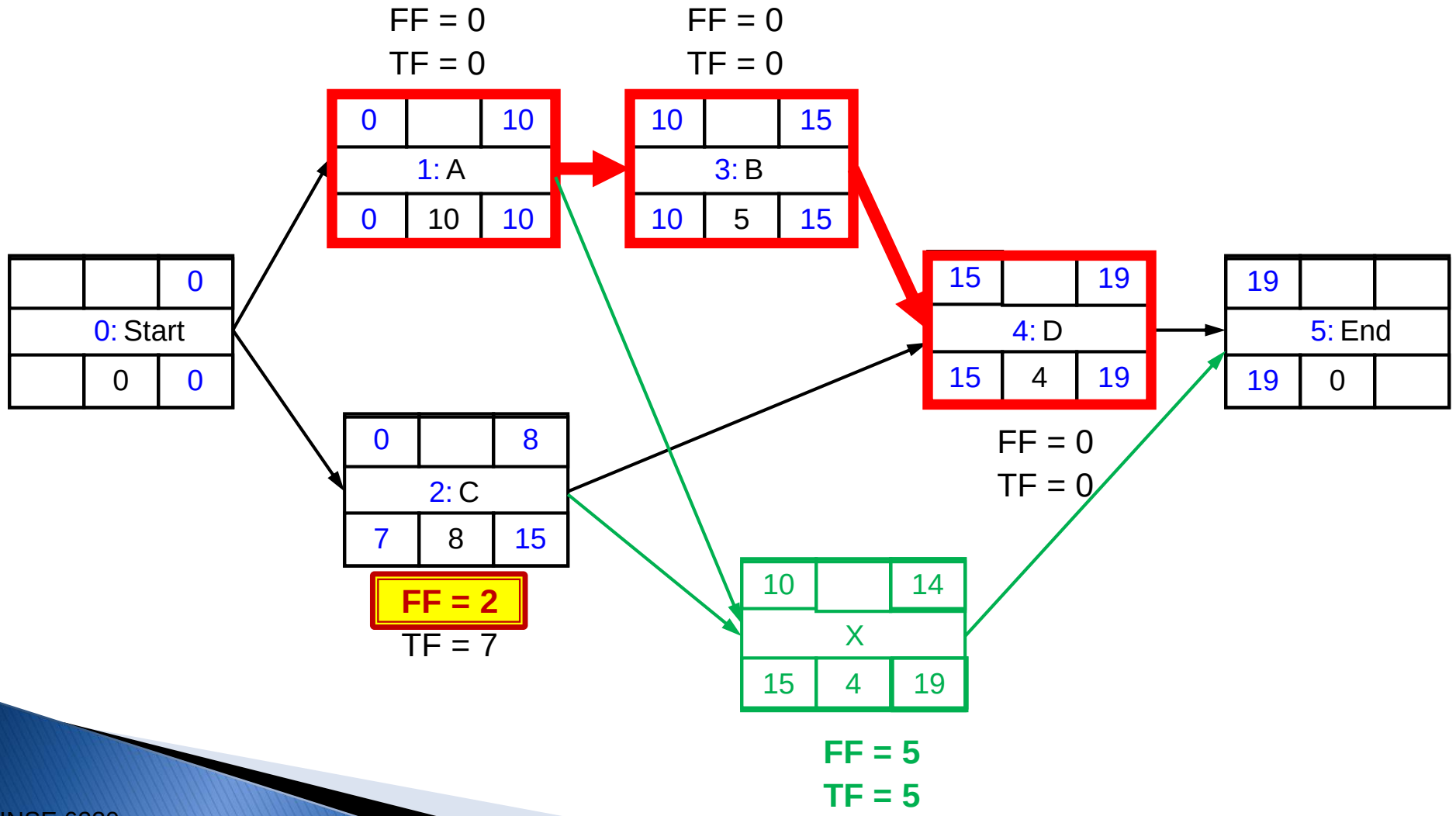
$$TF(j) = LS(j) - ES(j)$$

$$FF(j) = \min_{k}\{ES(k)\} - EF(j)$$

k : the successor activities of activity j

CPM with AON: Step 5. Float Times

ES		EF
Number: ID		
LS	D	LF



Scheduling on Critical Path (CP)

- Activities that have different earliest and latest start times (*i.e., ES(i) < LS(i)*) can be scheduled to start anytime between *ES(i)* and *LS(i)*
  - **Schedule Flexibility:** The concept of ***float*** is to use part or all of this allowable range to schedule an activity without delaying the completion of the project

- Activities on a critical path do not have floats

$$TF(j) = 0$$

$$FF(j) = 0$$

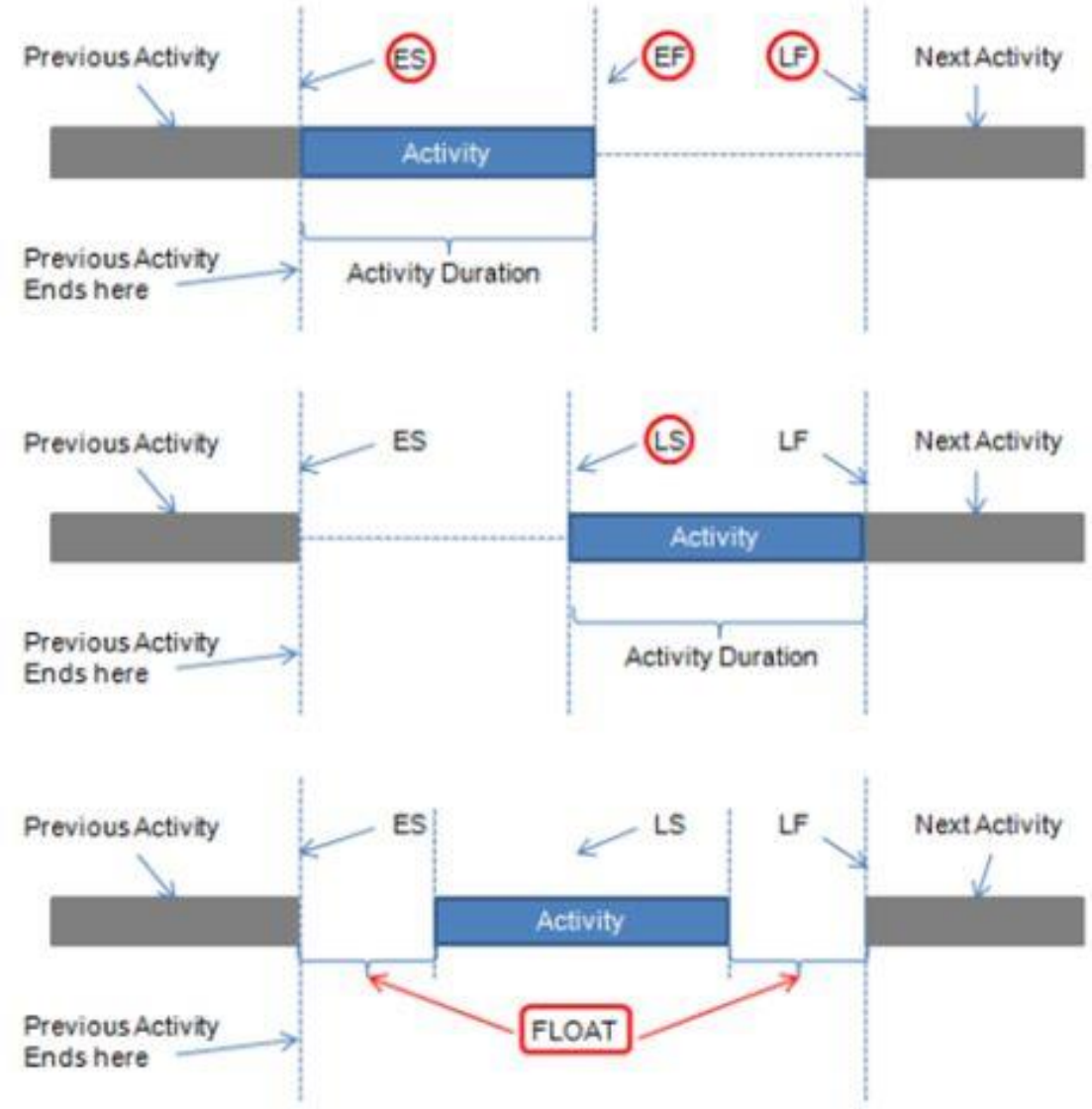
  - They have to be scheduled at their *ES(i)*

    → **no flexibility on CP!!!**

$$\mathrm{ES(i)} = \mathrm{LS(i)}$$

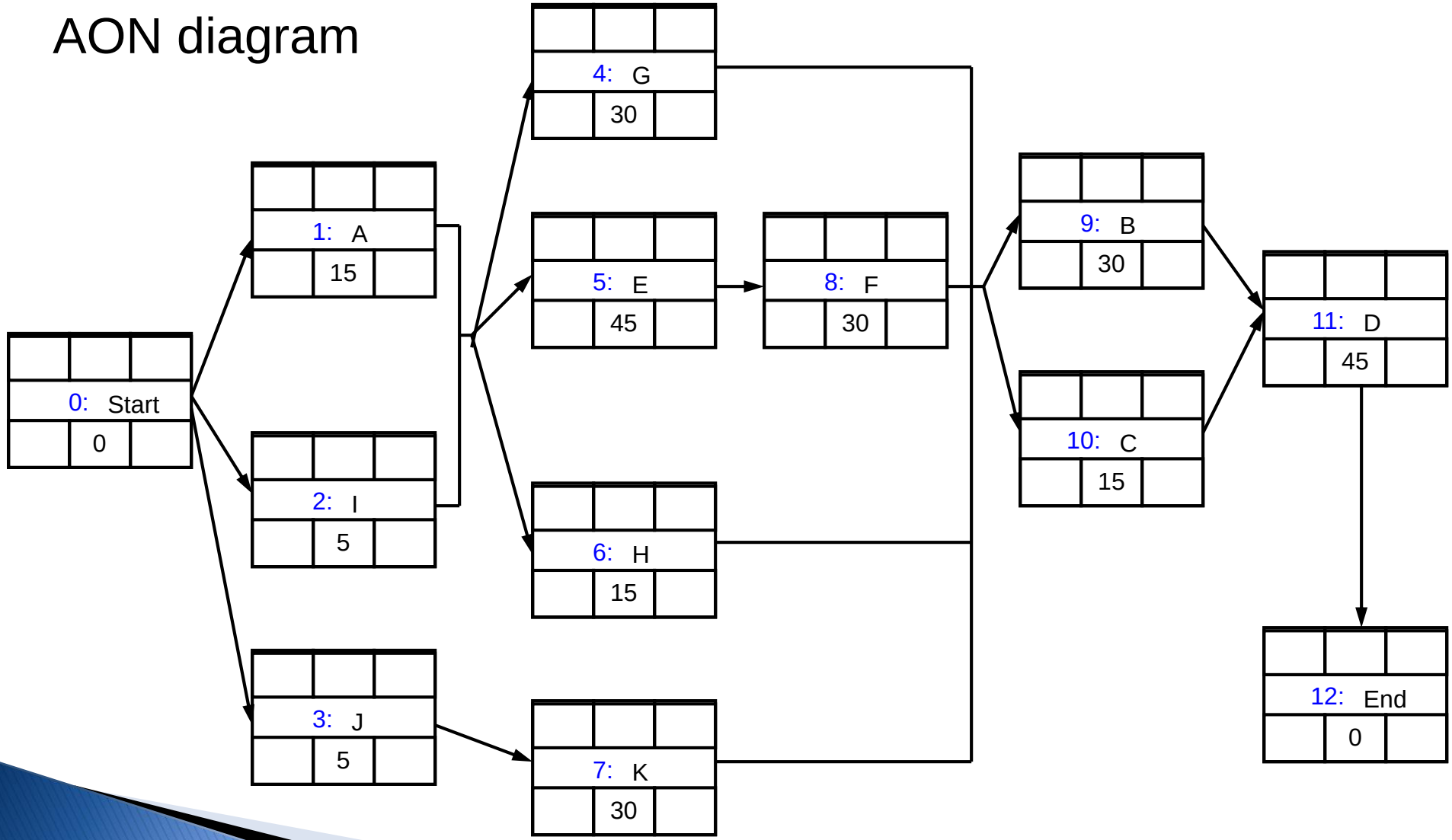
$$\mathrm{EF(i)} = \mathrm{LF(i)}$$

Floats – scheduling flexibility



CPM with AON: *Example*

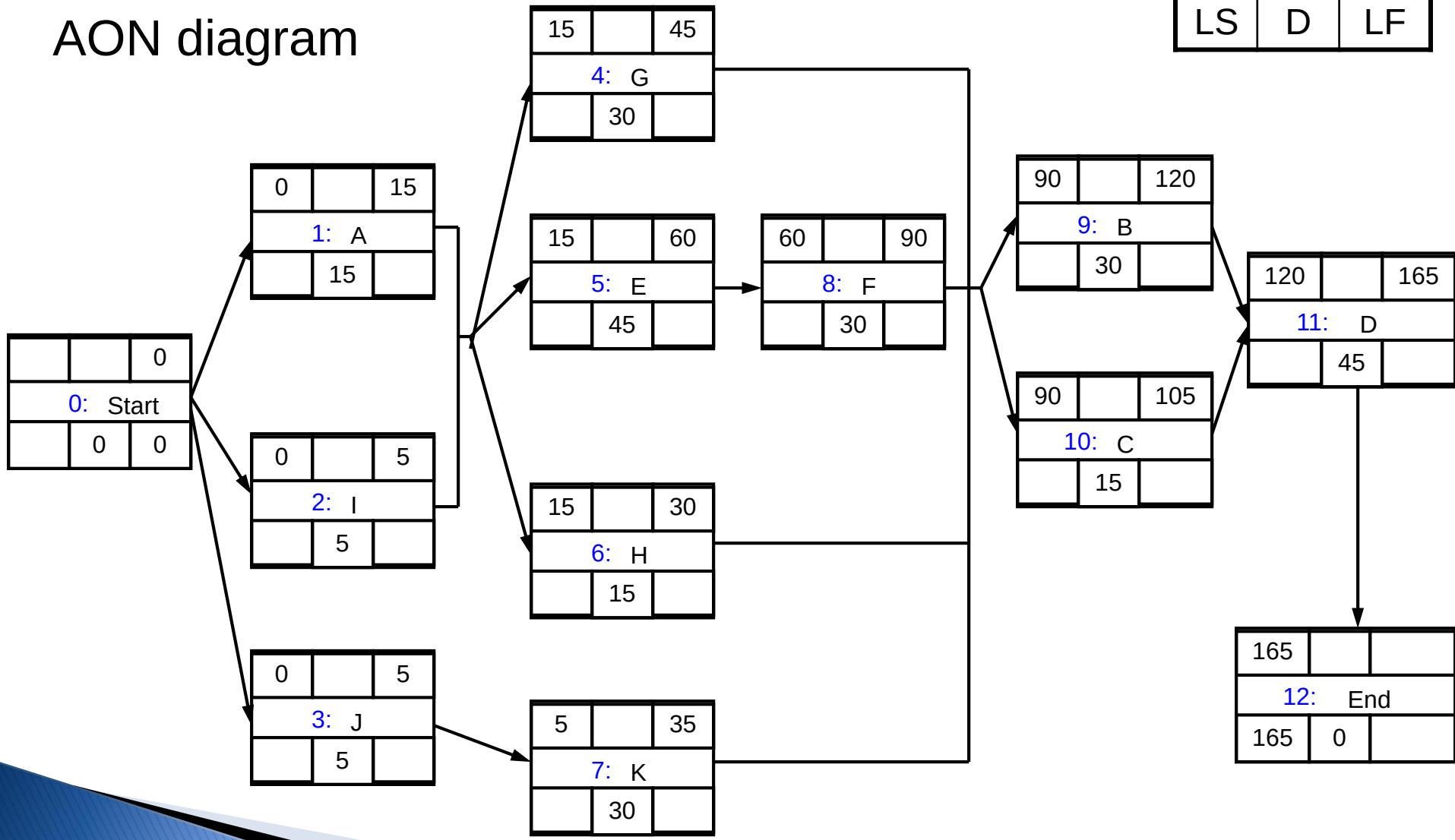
AON diagram



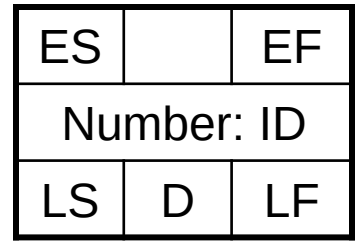
CPM with AON: *Example*

ES		EF
Number: ID		
LS	D	LF

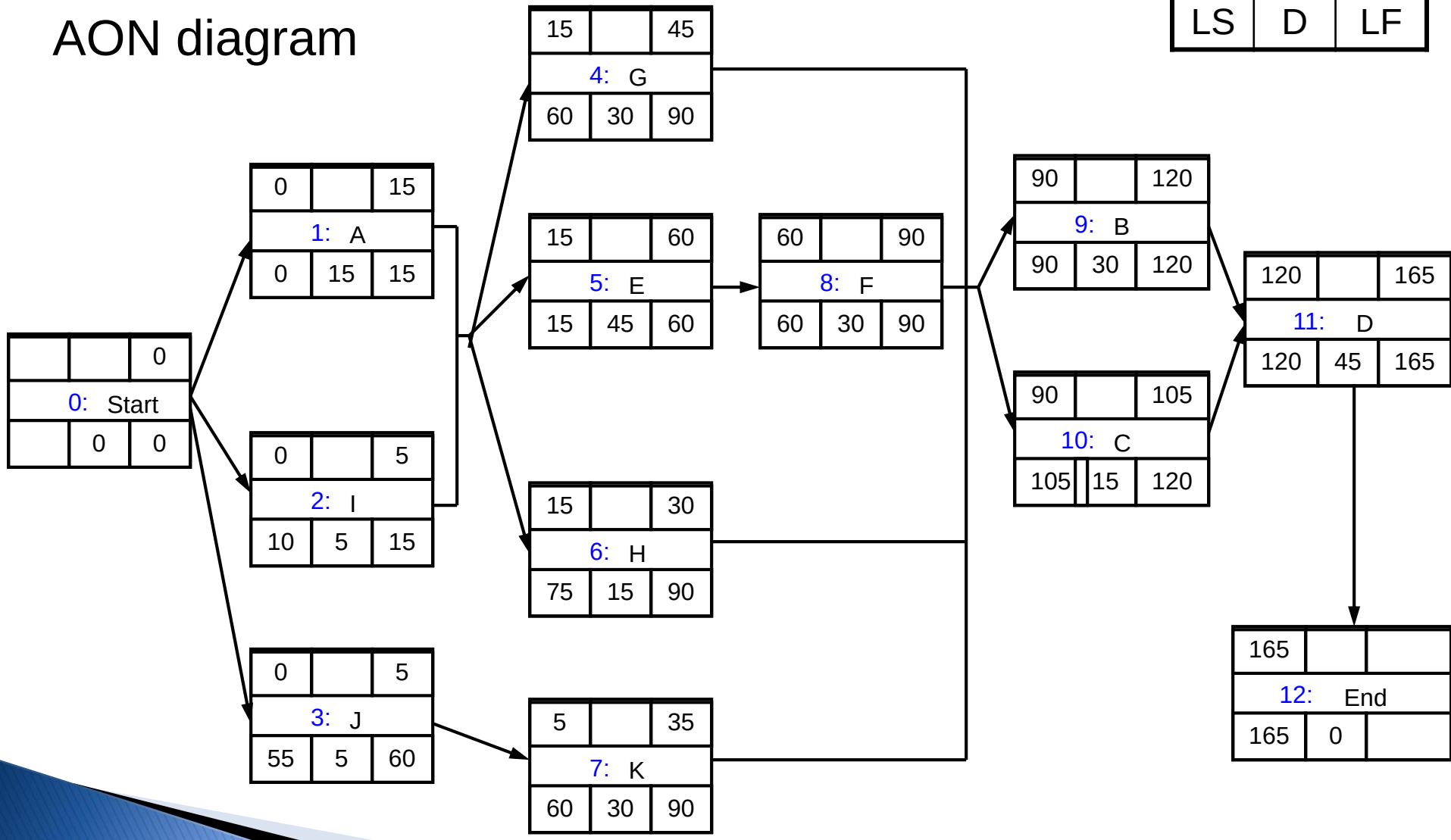
AON diagram



CPM with AON: *Example*



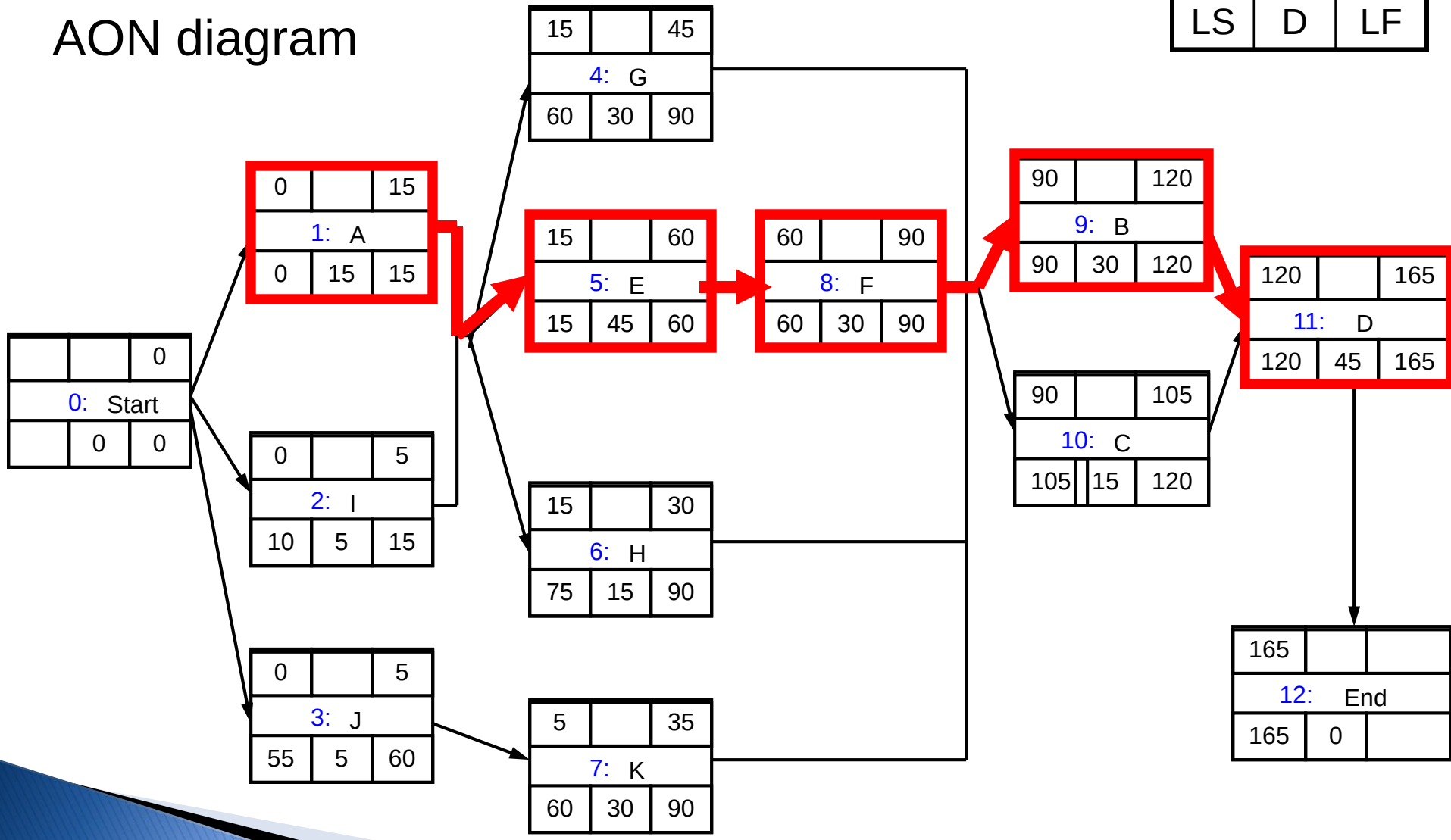
AON diagram



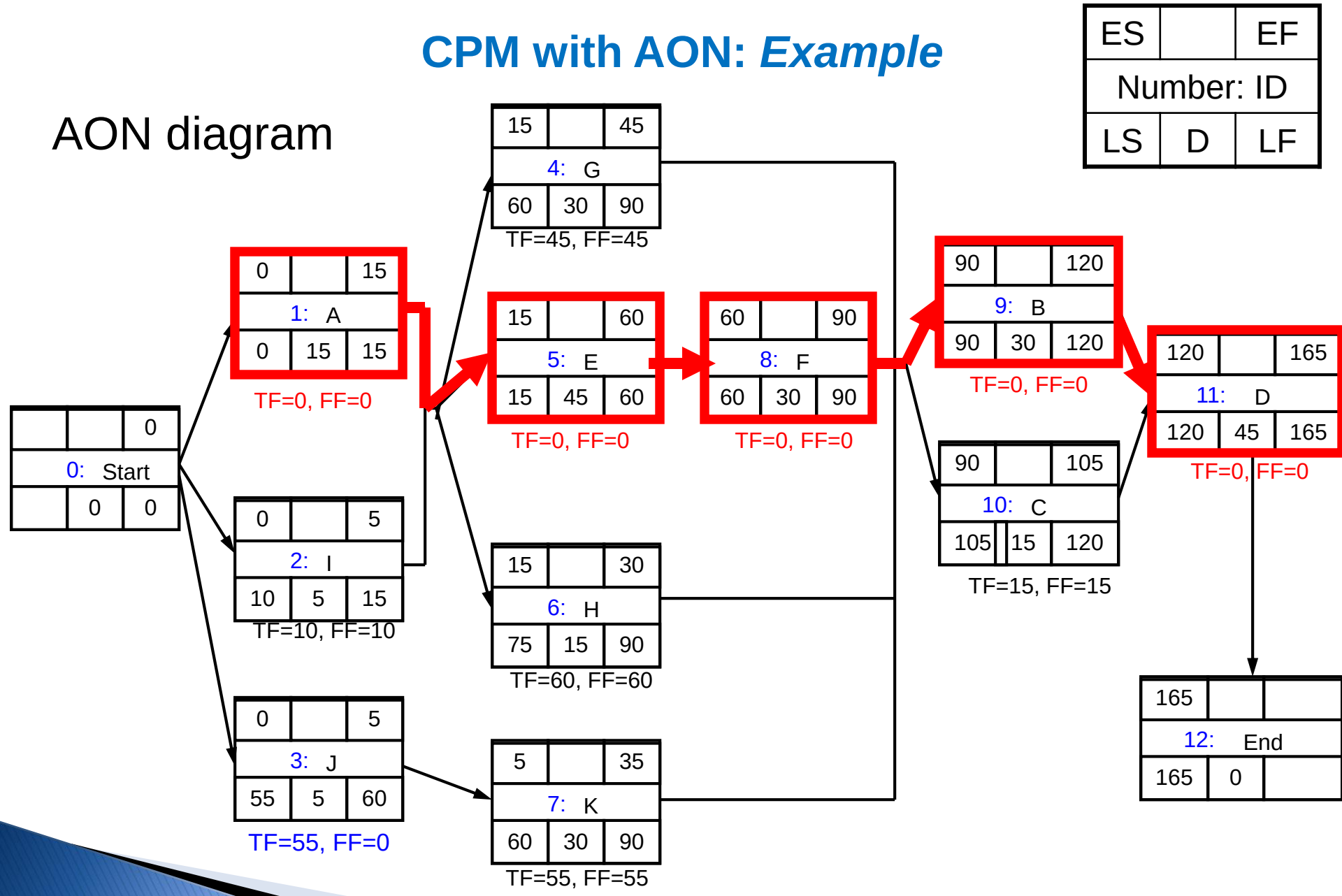
CPM with AON: *Example*

ES		EF
Number: ID		
LS	D	LF

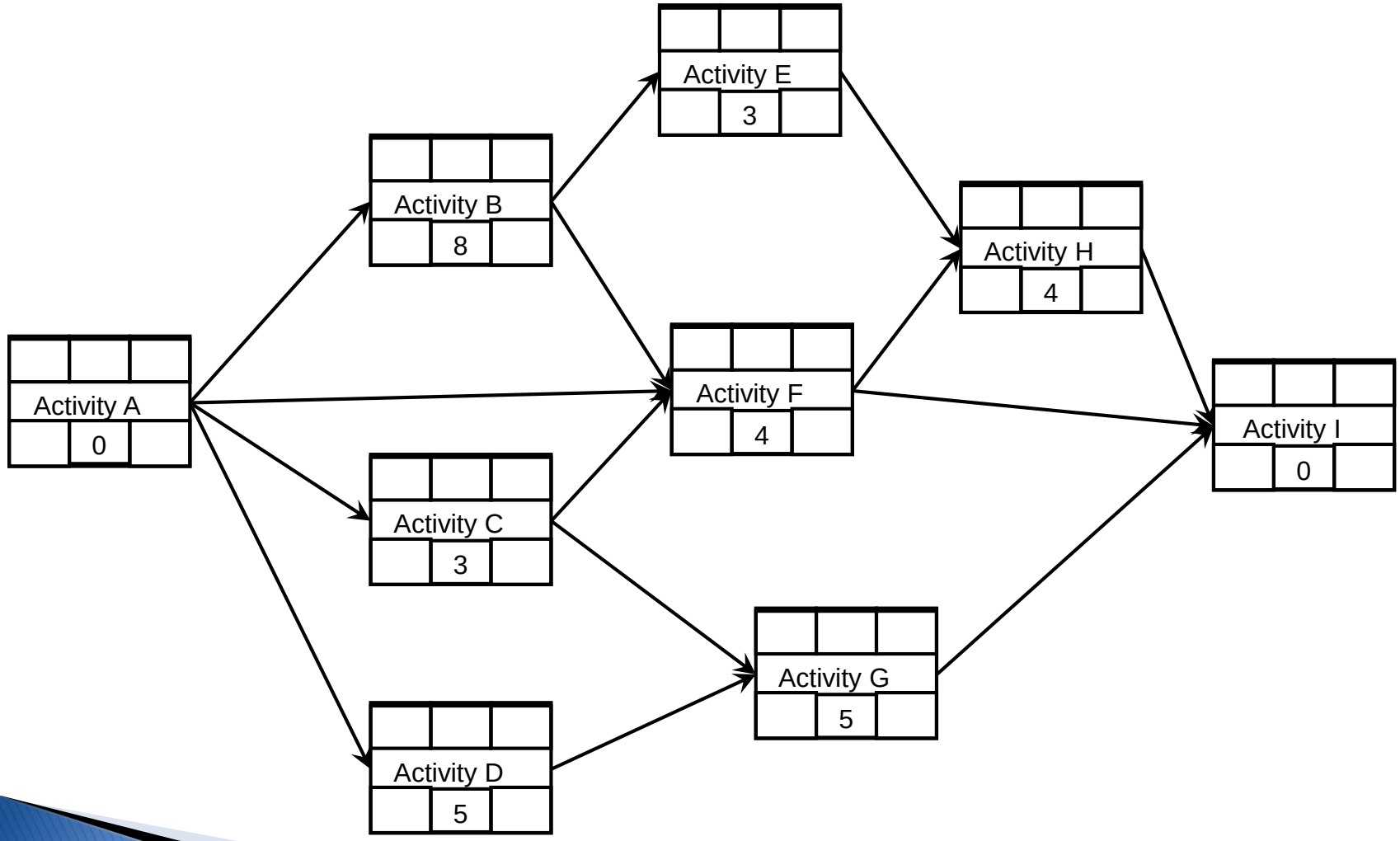
AON diagram

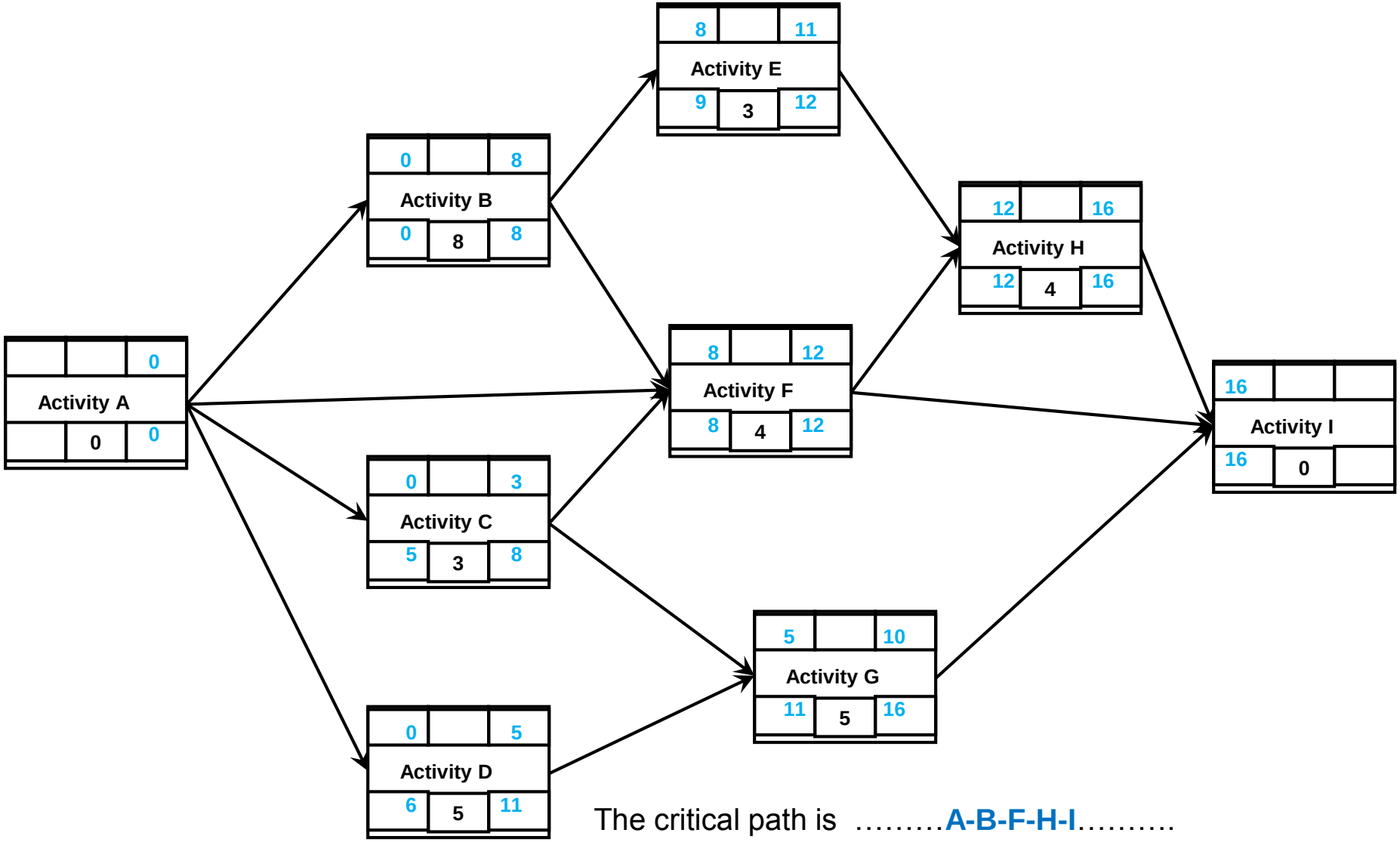


CPM with AON: *Example*



CPM with AON: *Example - midterm*





The critical path is ……**A-B-F-H-I**…….
The total duration of the project is ……**16**……..
For activity **E**, determine ES:……**8**…….. and LF ……**12**…..
For activity **G**, determine EF:……**10**…….. and LS ……**11**…
For activity **C**, determine TF:……**5**…… and FF ……**2**…..
For activity **D**, determine TF:……**6**……
For activity **F**, determine FF ……**0**……

Next Lecture

- Project Time Management II.