

MGMT 4135  
**Project Management**

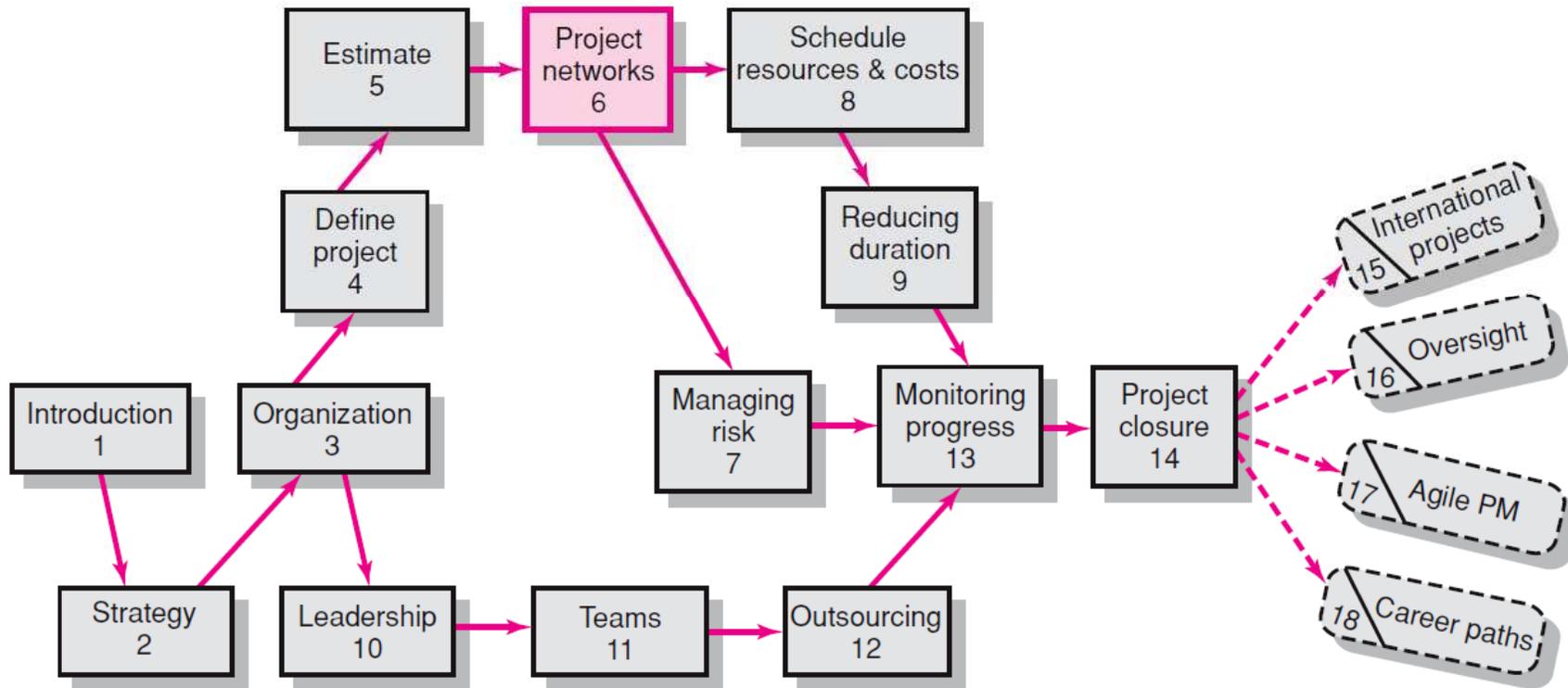
**Chapter-6** | **Developing a Project Plan**

# Chapter-6

## Developing a Project Plan



### Where we are now...





## **Developing the Project Network – Why?**

- Graphical representation of the tasks to be completed
- Logical sequence of activities
- Determines dependencies
- Provides the critical path

## **Worth Doing?**

- Takes time to develop which means it costs money
- YES. Not as valuable for trivial or short duration projects
- Basis for scheduling labor and equipment
- Estimates project duration
- Determines best start and finishing times

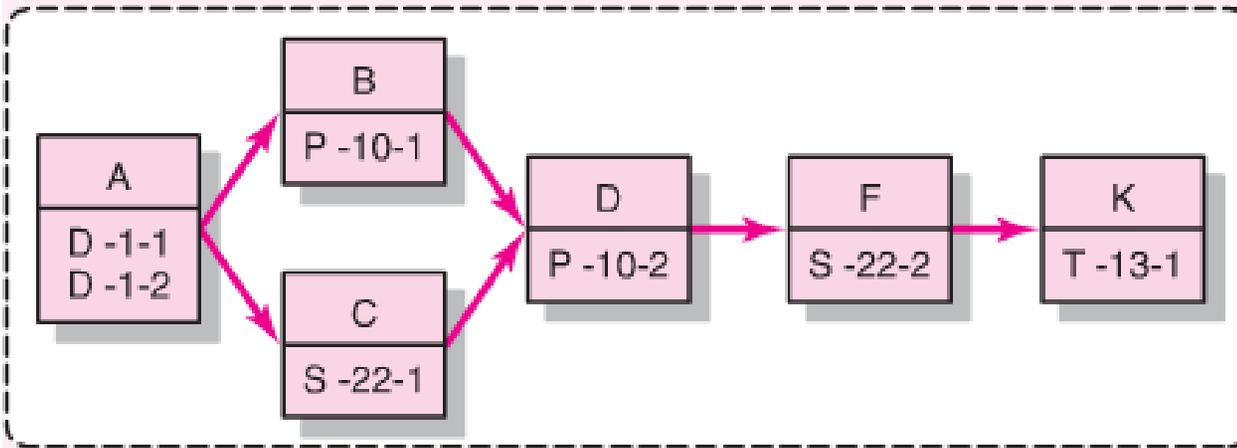


## From Work Package to Network

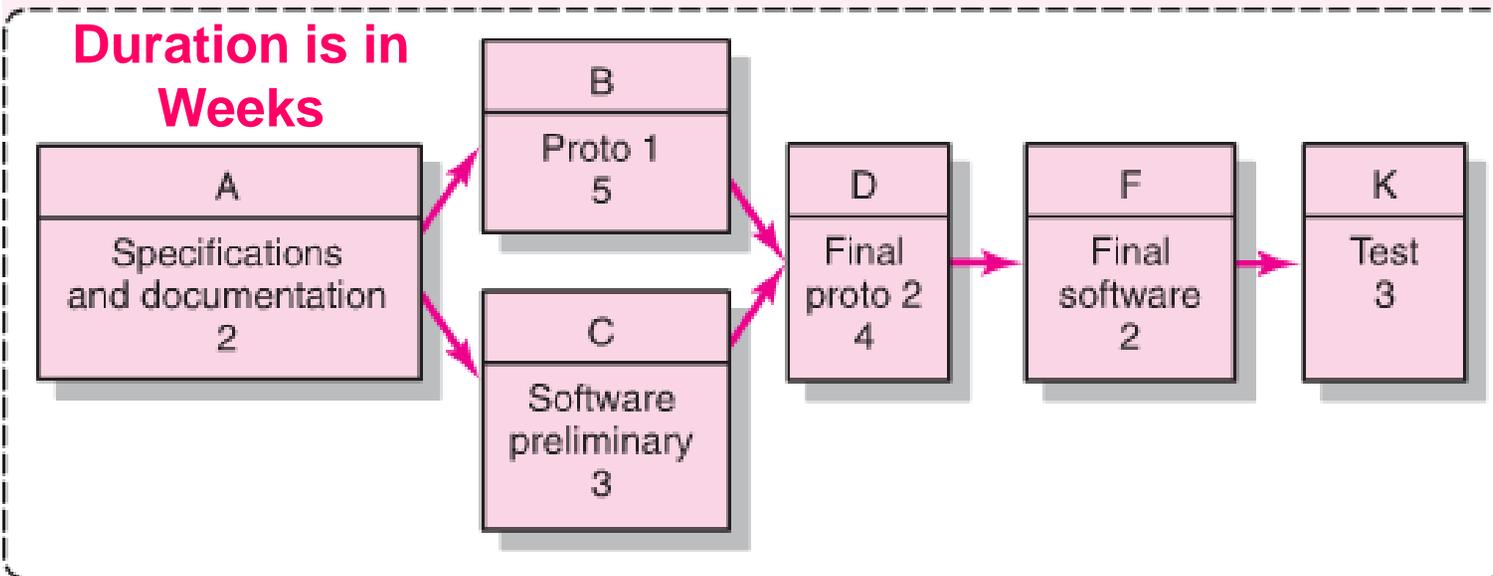
- Networks are developed from the WBS
- Work packages are the lowest level of the WBS where the activities can be found.
- An **activity** is an element in the project that consumes time.
- How does the management process often fail?
  - One group define the work packages and another defines the activities
  - WBS is poorly constructed and not deliverable oriented.
- So what is actually “networked?” All the work packages.
- Consider **Figure 6.1** pg. 159



### From Work Package to Network

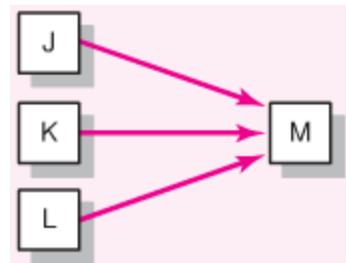
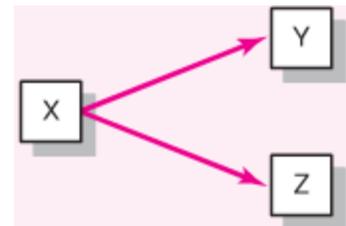
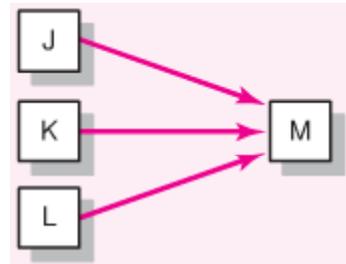


This diagram is using several work packages in its network. Some work packages can be done in parallel while others have a FS relationship, creating a linear pattern.



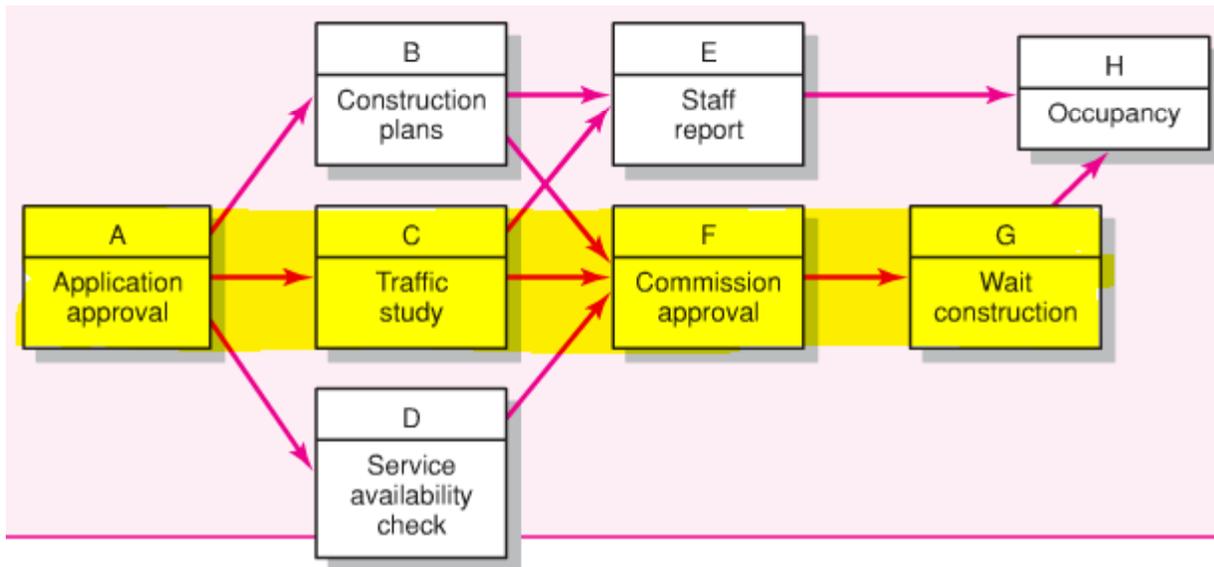
## Terminology

- An **activity** is an element in the project that consumes time. It is represented as a “node” (box).
- **Merge activities** have more than one activity immediately preceding it. “M” is a merge activity.
- **Burst activities** has more than one activity immediately following it. “X” is a burst activity.
- **Parallel activities** can take place at the same time but do not have to take place at the same time. In this case, “J-K-L” are shown as parallel activities.



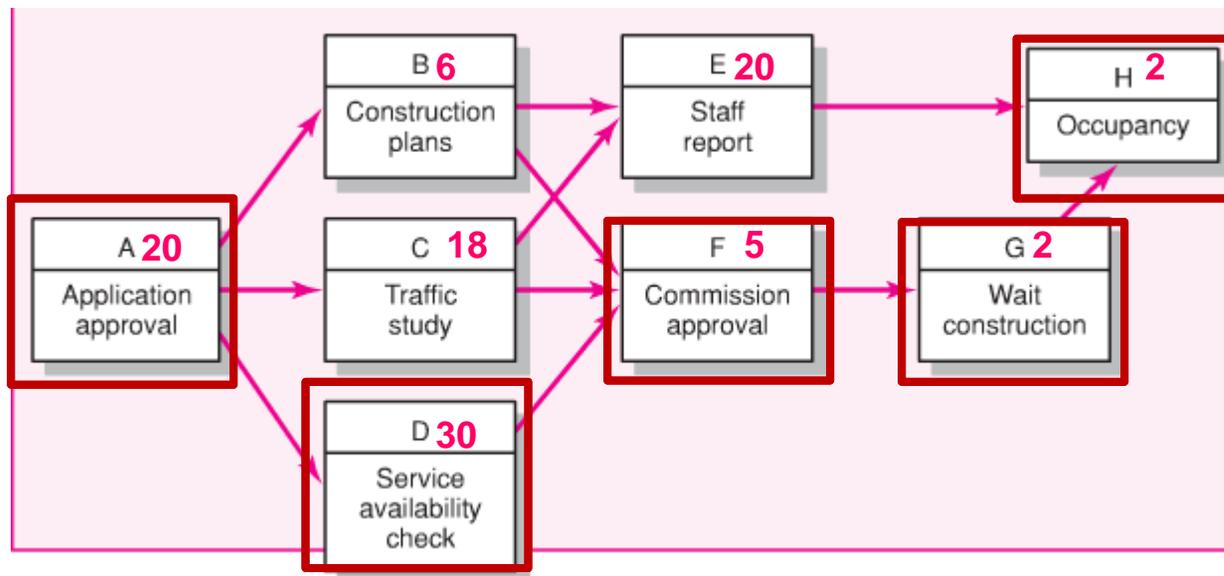
### Terminology

- **Network path** is a sequence of connected dependent activities. In this example, activities **A-C-F-G** make up one network path. How many others are there?



### Terminology

- Critical path** is the network path with the longest duration. When an activity on the critical path is delayed, the project is delayed that same amount of time. Given the example below, identify the critical path.





## **Rules for Developing Project Networks**

1. Networks flow from **left to right**
2. Activities cannot begin until all preceding connected activities have been completed
3. Arrows indicate there are predecessors and successors in the flow of activities
4. Each activity should have a unique ID *Pg. 174 “Activity Numbering”*
5. An activity identification number must be larger than that of any activities that precede it. *Consider the previous example.*
6. Looping is not allowed. *Pg. 173 “Network Logic Errors”*
7. Conditional statements are not allowed *Pg. 173 “Network Logic Errors”*
8. With multiple starts, use a common start node and similarly, use a common end node.



## Activity-on-Node (AON) Fundamentals

1. Activities that must be completed immediately before an activity are called **predecessor** activities.
2. Activities that must follow immediately after an activity are called **successor** activities.
3. Activities that can occur while an activity is take place is known as a **concurrent** or **parallel relationship**.

*See pg. 162 Figure 6.2*



## Network Computation Process

1. Activity time estimates are taken from the work package duration estimates and aggregated up as total time for that network “node.”
2. A **forward pass** detects the earliest start ES and earliest finish EF times of each activity to determine the earliest that the project can be finished.
3. A **backward pass** detects the latest start LS and the latest finish LF times of each activity in the network.



## Determining Slack (or Float)

1. When the forward and backward passes are computed, you will be able to determine which activities can be delayed without affecting the critical path and without delaying the project.
2. To determine an activity slack, simply subtract the ES from LS or EF from LF – both will yield the same answer.

In this example, activity “C” has an early start ES of 5 and an early finish EF of 10. Subtract ES from EF and you learn that activity “C” has a float of 5 weeks.

5	C	15
5	Traffic study	
10	10	20



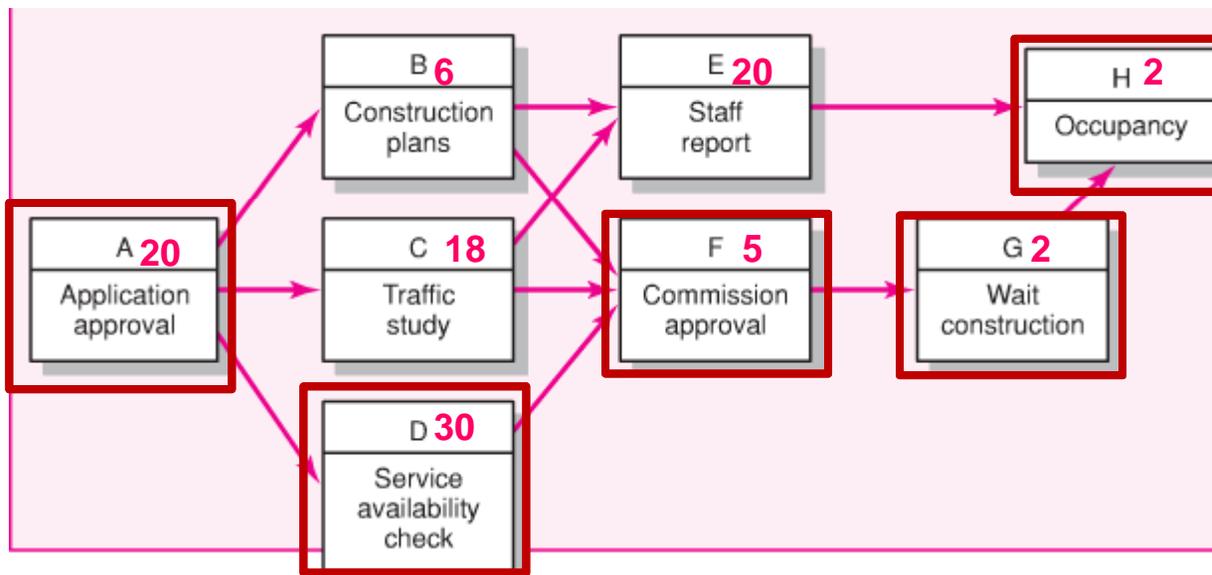
## Determining Slack (or Float)

1. If there is a likelihood that the original critical path will change, it is said to be “**sensitive**.” This is normally found when a network has more than one critical path, or near critical path(s).
  - a) **Near critical** paths are those that have the closest duration of the critical path and should be watched closely.
  - b) **Sensitivity** suggests that once the project is initiated, one of the near critical paths could end up being an additional critical path or even the new critical path should any activity slip and cause that network path duration to increase equal to or greater than the critical path duration.
2. Network schedules that have only one critical path and activities that show significant slack, not likely to affect the critical path are said to be “**insensitive**.”



### Determining Slack (or Float)

3. Going back to the previous example, consider the following paths to determine the critical path and near critical paths.



- A-B-E-H 48
- A-B-F-G-H 35
- A-C-E-H 50
- A-C-F-G-H 47
- A-D-F-G-H 59\*

It is safe to ascertain that this network is “insensitive.” Given that the second longest duration in the network is 50 weeks, there are 9 weeks slack before coming close to affecting the critical path.

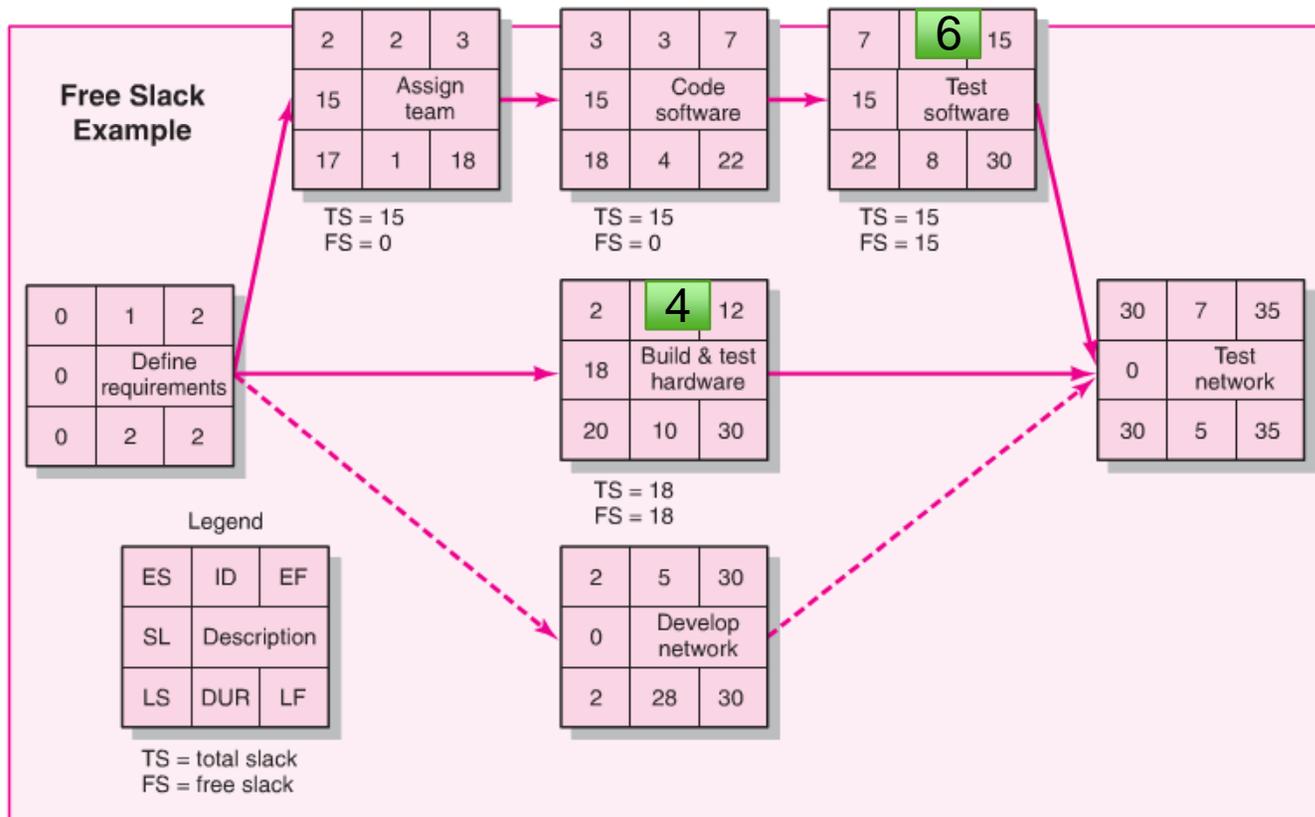
# Chapter-6

## Developing a Project Plan



### Free Slack (Float)

1. Free Slack is the amount of time the early finish of an activity can be delayed without affecting the early start of any immediate successor activity/activities. See pg. 172 Figure 6.9



- **Activity-6** as free slack of 15 time units.
- **Activity-4** has free slack of 18 time units.



## **Free Slack (Float)**

2. Free slack occurs at the end of a chain of activities before a merge occurs.
3. Notice that activities 6 and 4 are at the end of a chain before merging into activity-7. (4, 5, and 6 all merge into 7)
4. Consider also that any activity or all activities in the chain can take advantage of the free float.

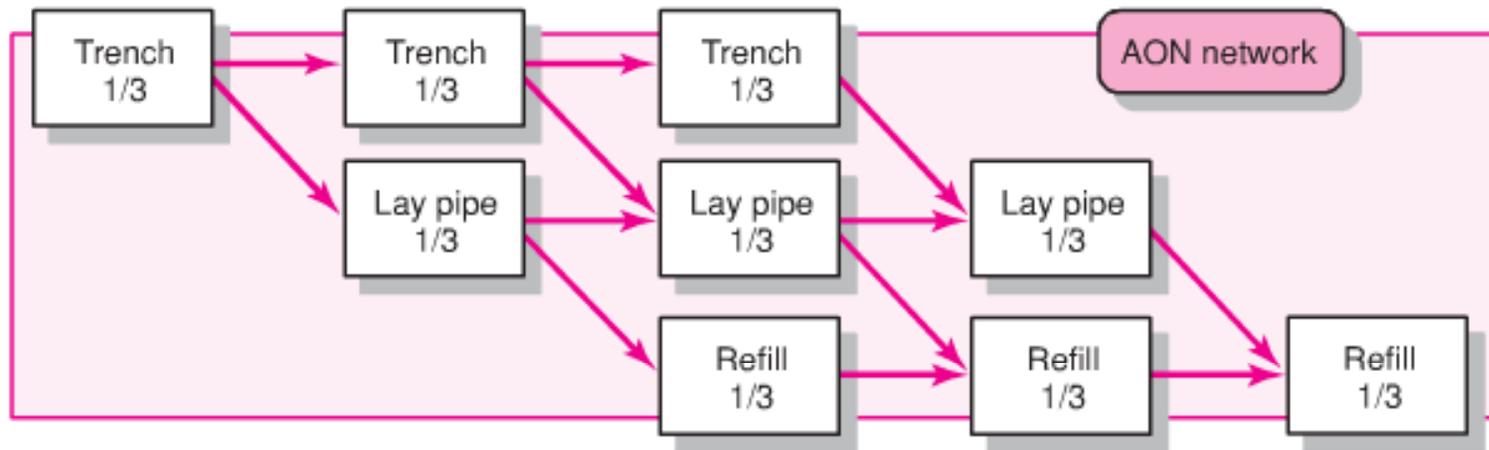
We know that Activity-6 has a free slack/float of 15 time units. That means activities 2, 3, and 6 can all share in that float. If each of those tasks float, totaling 15 time units, the successor activity (7) early start will not be affected.



## Laddering

1. Stated earlier, a finish-to-start relationship states that the activity must be 100% completed before advancing to the next activity. If that activity has a long duration and if that activity has a delay, that activity can be broken down into segments using **laddering** so that the following activity can begin sooner without delaying work.
2. Laying pipe requires that the trench must be dug, pipe laid, and the trench refilled. If the pipeline is 1-mile long, it is not necessary to dig on mile of trench before pipe laying can begin.
3. It is also true that once the pipe is in place, refilling the trench can also begin.

## Laddering



- In this **ladder** network, all activities have been broken into segments of 1/3.
- After 1/3 of the trench has been dug, the pipe laying begins.
- After 1/3 of the pipe laying is completed, the trench refilling begins.



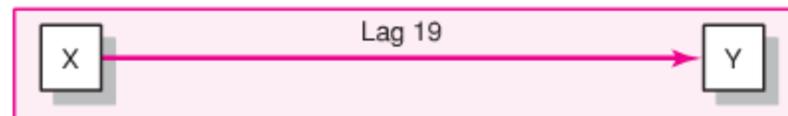
## Use of Lags

- When an long duration activity delays the start of finish of the next (or successor) activity, the network normally breaks the activity into smaller activities to avoid the long delay of the successor activity. Using **lags** can avoid such delays and reduce the network detail.
- **Lags** can be used to constrain the start and finish of an activity.
- During this lag, **no work can occur or be charged to account code.**
- The project manager normally must justify lags.

## Use of Lags: Finish-to-Start Relationship

- Ordering material (Activity-X) may take 1 day but it may take up to 19 days to receive the goods (Activity-Y).
- Activity-Y has a finish-to-start relationship with Activity-X, meaning that ordering material must be completed before receiving the goods can begin.

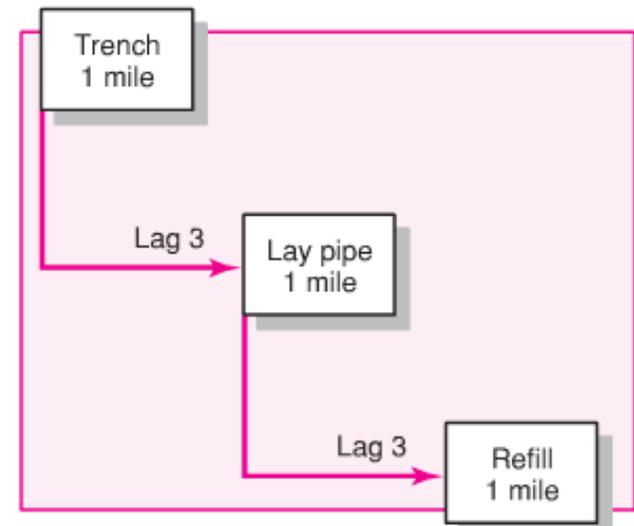
Receiving Goods: FS+19 days means that ordering material must first be completed before receiving goods can start, 19 days later.



## Use of Lags: Start-to-Start Relationship

1. Consider this depiction of using a 3-day lag for the Pipe Laying project. Essentially, each activity begins at the same time having a start-to-start relationship but there is a delay (or a lag) between the successor activities.

- Digging the 1-mile trench begins.
- Lay Pipe: SS+3 days means that laying pipe will begin the same time as the trench digging begins trench with a lag of 3 days.
- Refill trench: SS+3 days means that filling the trench will begin the same time as pipe laying begins with a lag of 3 days.



2. This is also an example of **concurrent engineering** whereby work is broken into smaller segments so that work can be performed in parallel to expedite the project.

## Use of Lags: Finish-to-Finish Relationship

1. The finish of one activity depends on the finish of another activity. While software is being developed, beta testing can be occurring in parallel. Beta testing cannot be completed until the software development is completed.
2. Given this situation, we will have a finish-to-finish lag of, say, 10 days. This means that both the software development and testing essentially have the same finish date but testing will have a finish lag of 10 days.
3. Beta Testing has a finish-to-finish relationship with Software Development plus 10 more days, or FF+10days.

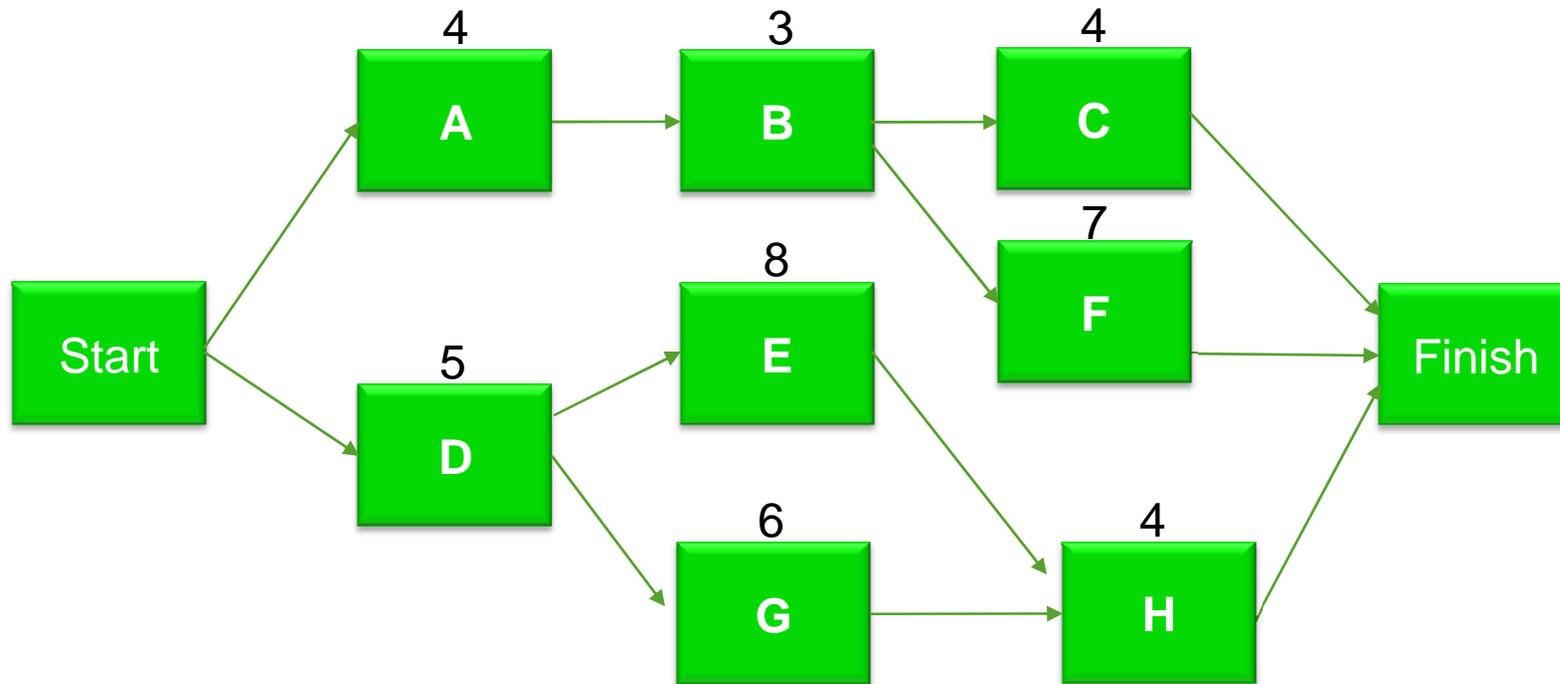




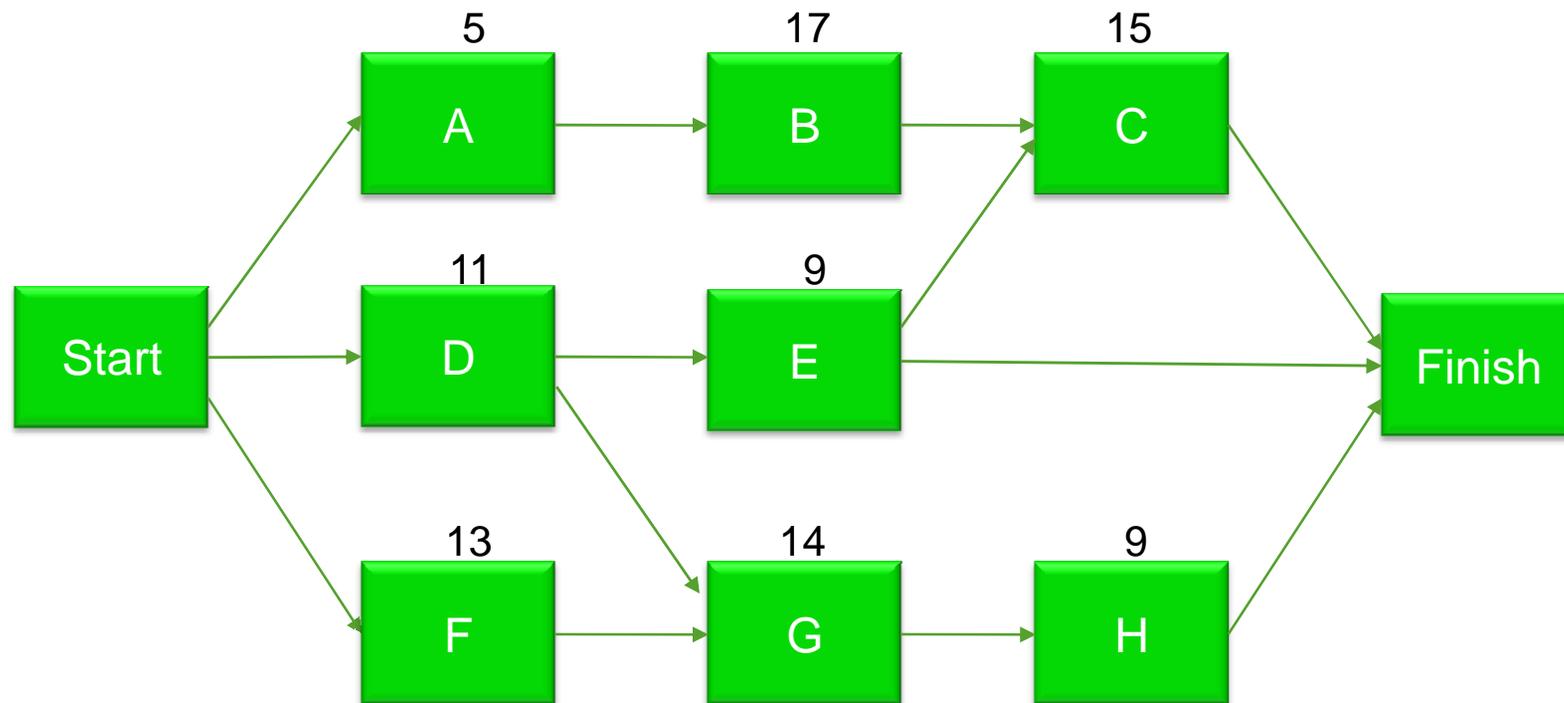
## Hammock Activities

1. **Hammock activities** derives its name because this activity spans over a segment of a project. They are used to identify fixed resources or costs over a segment of a project.
2. **Hammock activities** are very useful in assigning and controlling **indirect project costs**.
3. **Hammock activities** are determined *after* the network plan is drawn. See “Snapshot from Practice” pg. 183.

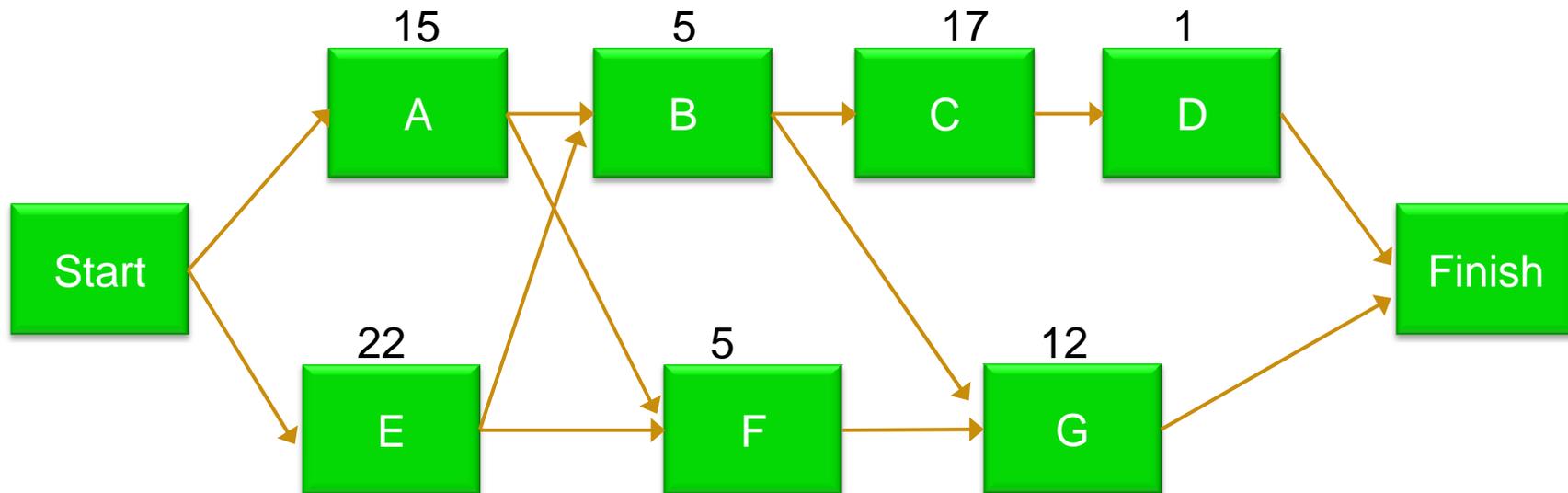
## Network Diagramming Exercise #1



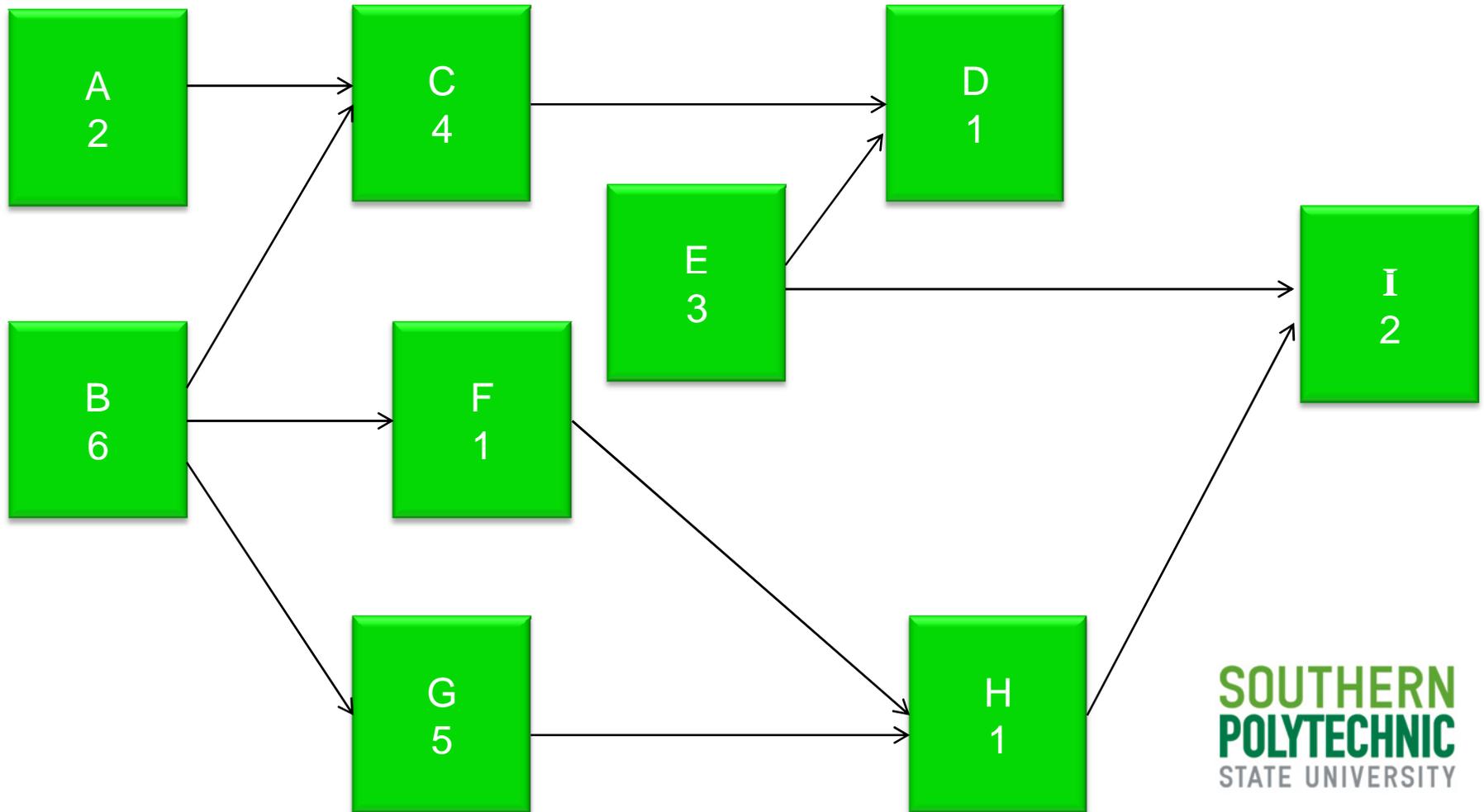
## Network Diagramming Exercise #2



## Network Diagramming Exercise #3



## Network Diagramming Exercise #4





## Chapter-5 Key terms:

---

Activity, 161	Early and late times, 159	Parallel activity, 160
Activity-on-Node, 160	Gantt chart, 174	Sensitivity, 170
Burst activity, 160	Hammock activity, 183	Total Slack/float, 169
Critical path, 160	Lag relationships, 178	Free Slack/float, 171
Concurrent Engineering, 179	Merge activity, 160	

---