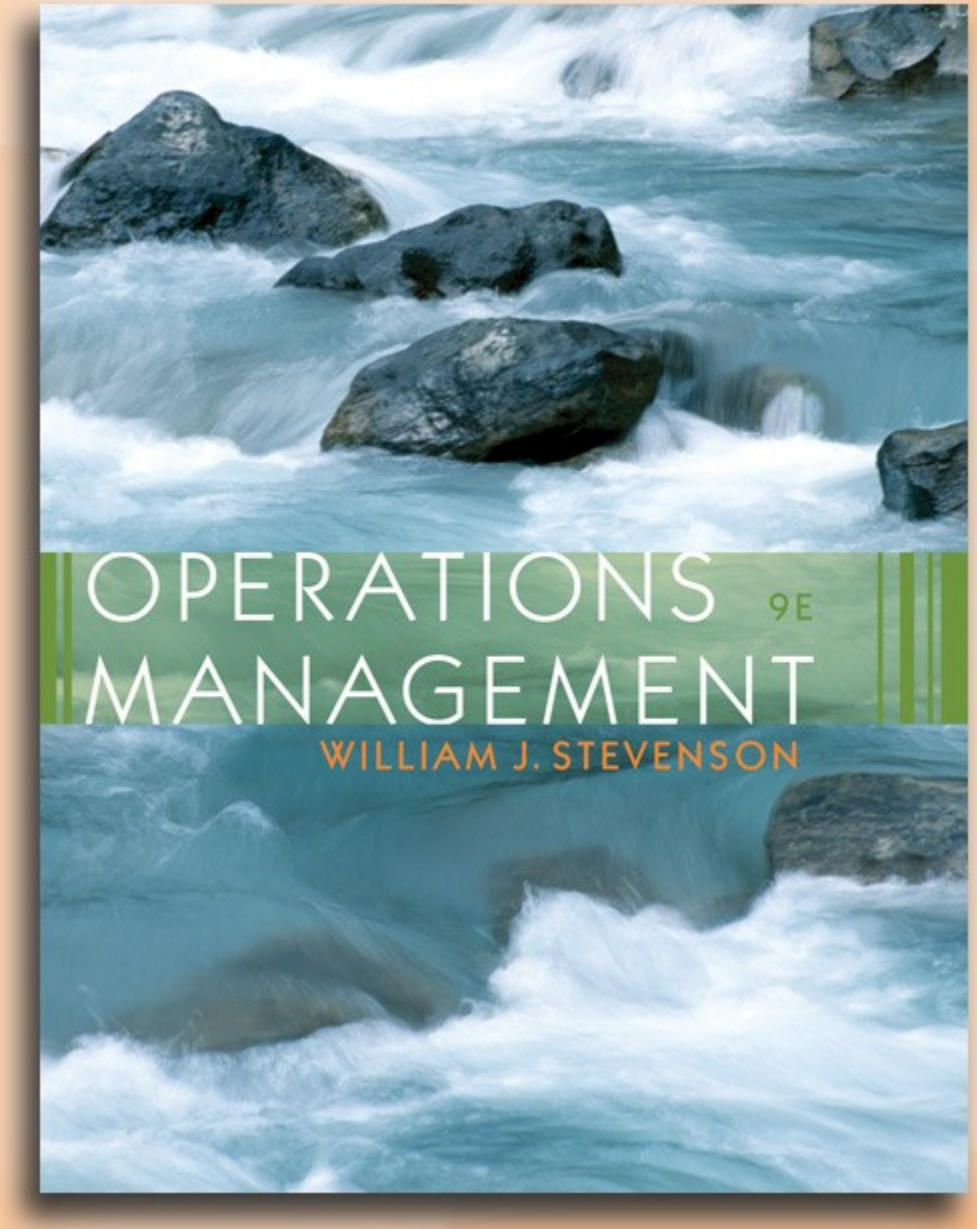
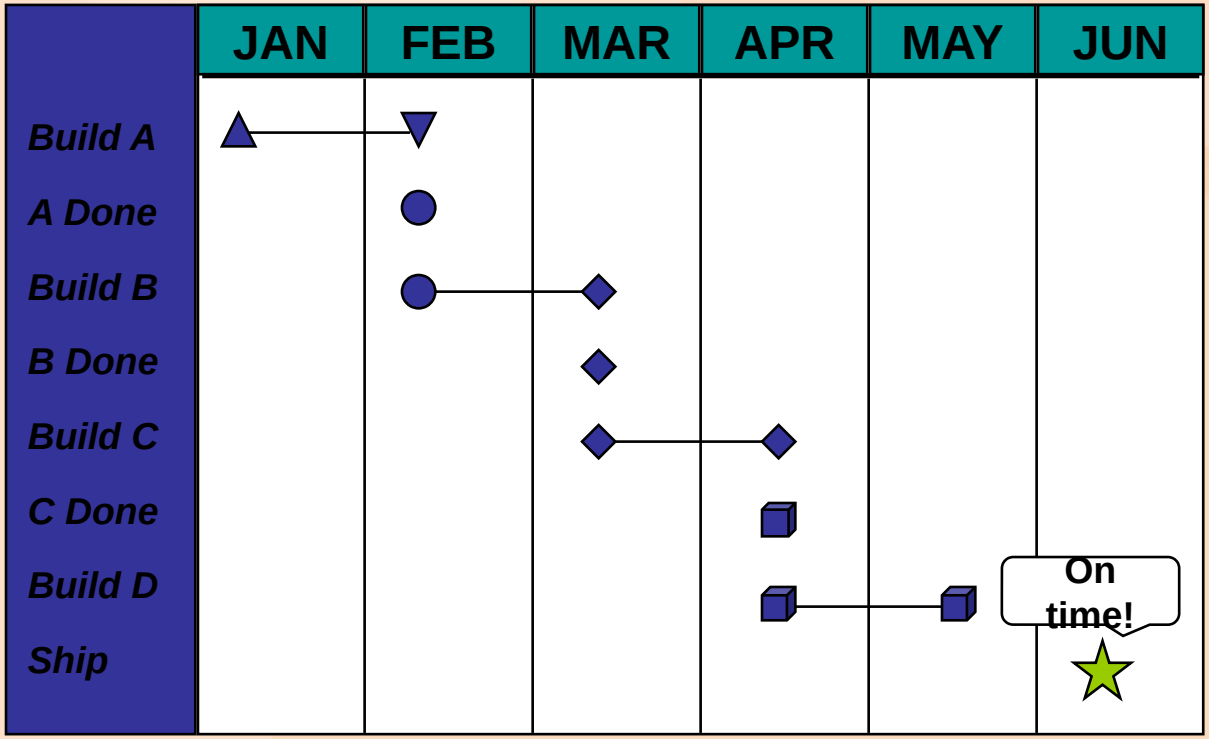



# Chapter 17

# Project Management

# Projects




Unique, one-time operations designed to accomplish a specific set of objectives in a limited time frame.

# Project Management

- How is it different?
  - Limited time frame
  - Narrow focus, specific objectives
  - Less bureaucratic
- Why is it used?
  - Special needs
  - Pressures for new or improves products or services

# Project Management

- What are the Key Metrics
  - Time
  - Cost
  - Performance objectives
- What are the Key Success Factors?
  - Top-down commitment
  - Having a capable project manager
  - Having time to plan
  - Careful tracking and control
  - Good communications

# Project Management

- What are the Major Administrative Issues?
  - Executive responsibilities
    - Project selection
    - Project manager selection
    - Organizational structure
  - Organizational alternatives
    - Manage within functional unit
    - Assign a coordinator
    - Use a matrix organization with a project leader

# Project Management

- What are the tools?
  - Work breakdown structure
  - Network diagram
  - Gantt charts
  - Risk management

# Project Life Cycle

1. Initiating
2. Planning
3. Executing
4. Monitoring and Controlling
5. Closing

# Key Decisions

- Deciding which projects to implement
- Selecting a project manager
- Selecting a project team
- Planning and designing the project
- Managing and controlling project resources
- Deciding if and when a project should be terminated

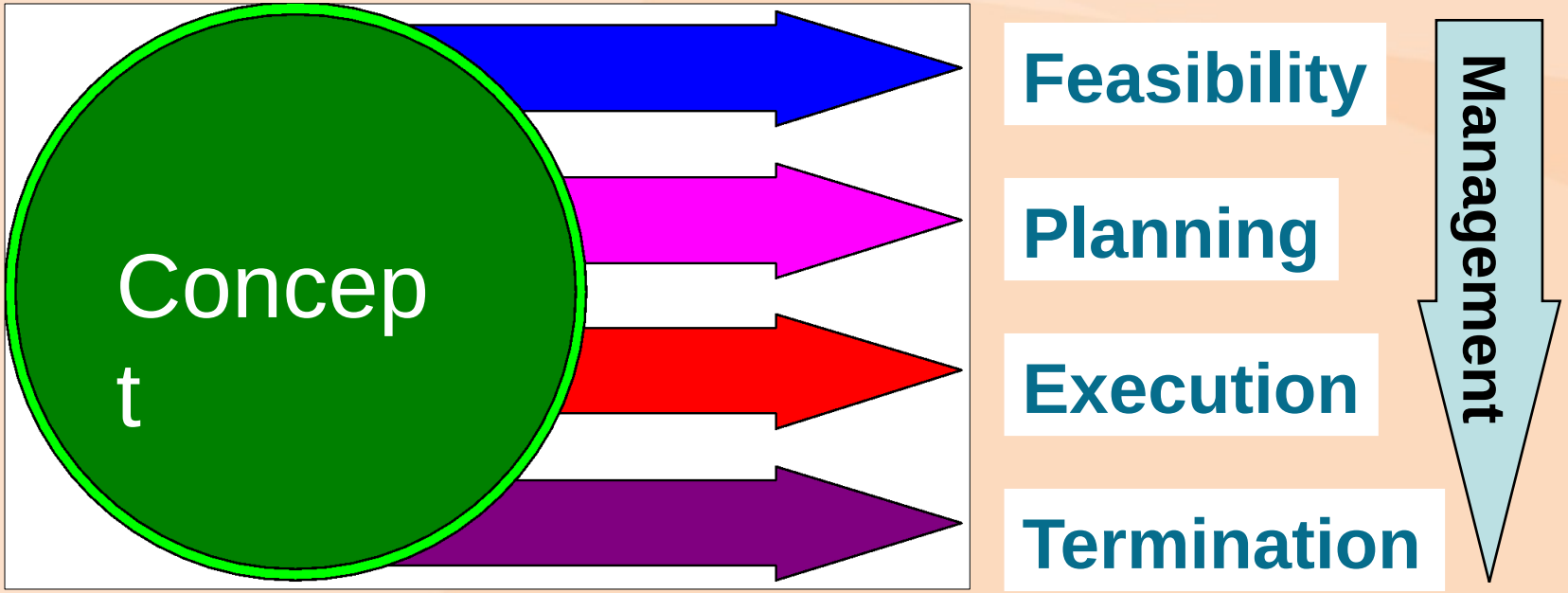
# Project Manager

Responsible for:

# Ethical Issues

- Temptation to understate costs
- Withhold information
- Misleading status reports
- Falsifying records
- Comprising workers' safety
- Approving substandard work

# Project Life Cycle

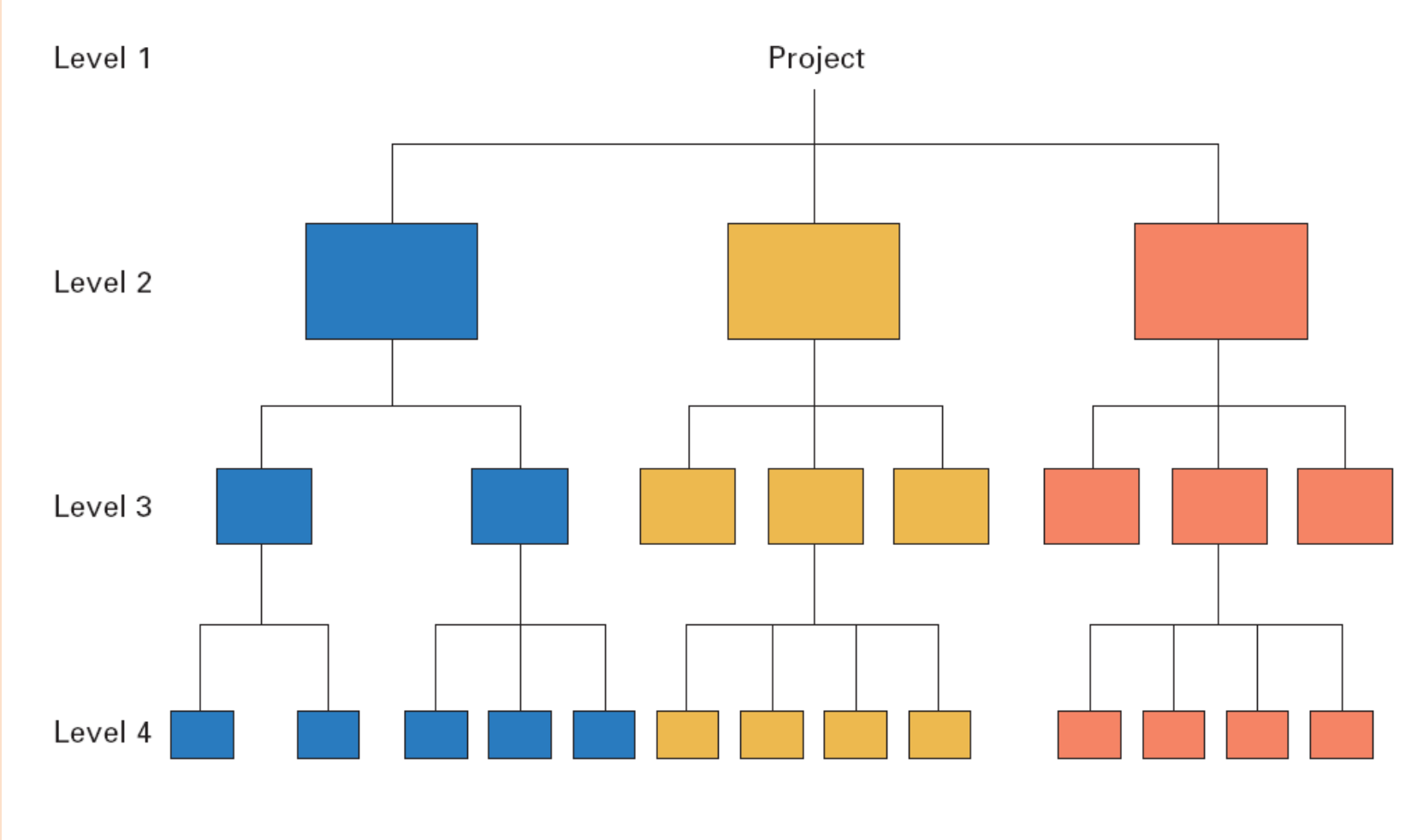


# Work Breakdown Structure (WBS)

- **WBS**
  - A hierarchical listing of what must be done during a project
    - Establishes a logical framework for identifying the required activities for the project
      1. Identify the major elements of the project
      2. Identify the major supporting activities for each of the major elements
      3. Break down each major supporting activity into a list of the activities that will be needed to accomplish it

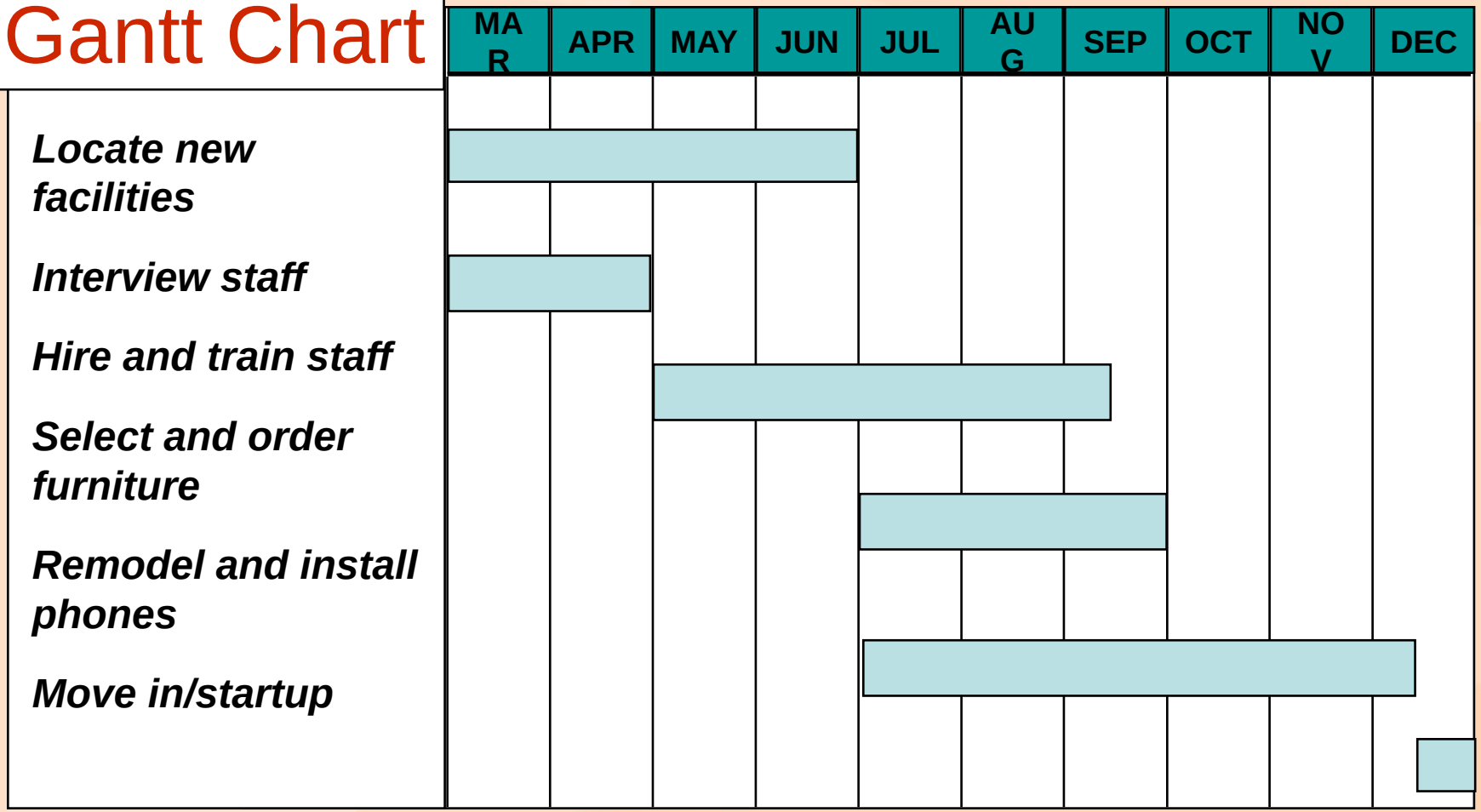
# Work Breakdown Structure

Figure 17.2



# Planning and Scheduling

## Gantt Chart



# PERT and CPM

***PERT***: Program Evaluation and Review Technique

***CPM***: Critical Path Method

By using PERT or CPM Managers can obtain:

1. A graphical display of project activities
2. An estimate of how long the project will take
3. An indication of which activities are most critical to timely project completion
4. An indication of how long any activity can be delayed without delaying the project

# The Network Diagram

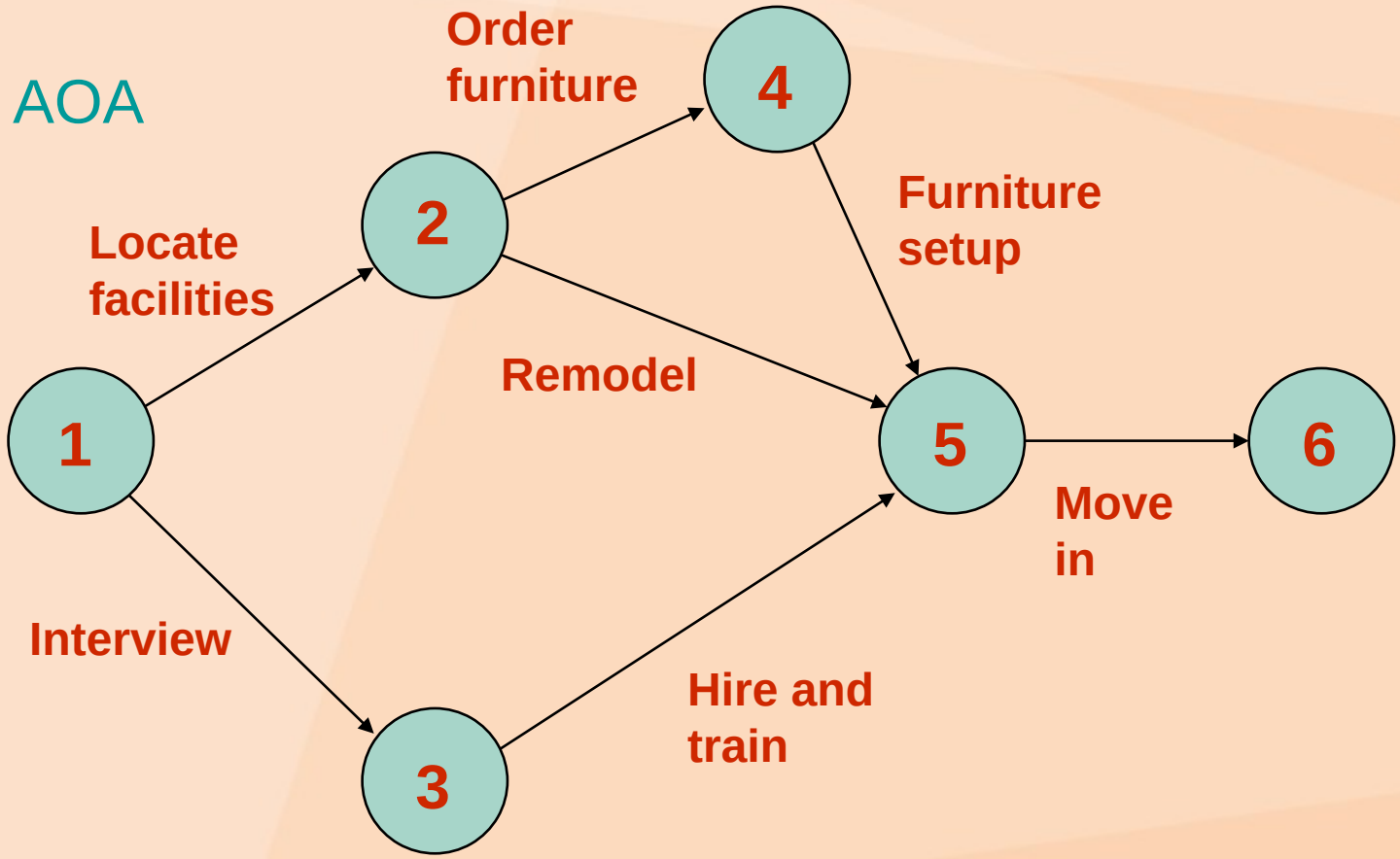
- *Network (precedence) diagram* – diagram of project activities that shows sequential relationships by the use of arrows and nodes.
- *Activity-on-arrow* (AOA) – a network diagram convention in which arrows designate activities.
- *Activity-on-node* (AON) – a network diagram convention in which nodes designate activities.
- *Activities* – steps in the project that consume resources and/or time.
- *Events* – the starting and finishing of activities, designated by nodes in the AOA convention.

# The Network Diagram (cont'd)

- Path
  - Sequence of activities that leads from the starting node to the finishing node
- Critical path
  - The longest path; determines expected project duration
- Critical activities
  - Activities on the critical path
- Slack
  - Allowable slippage for path; the difference the length of path and the length of critical path

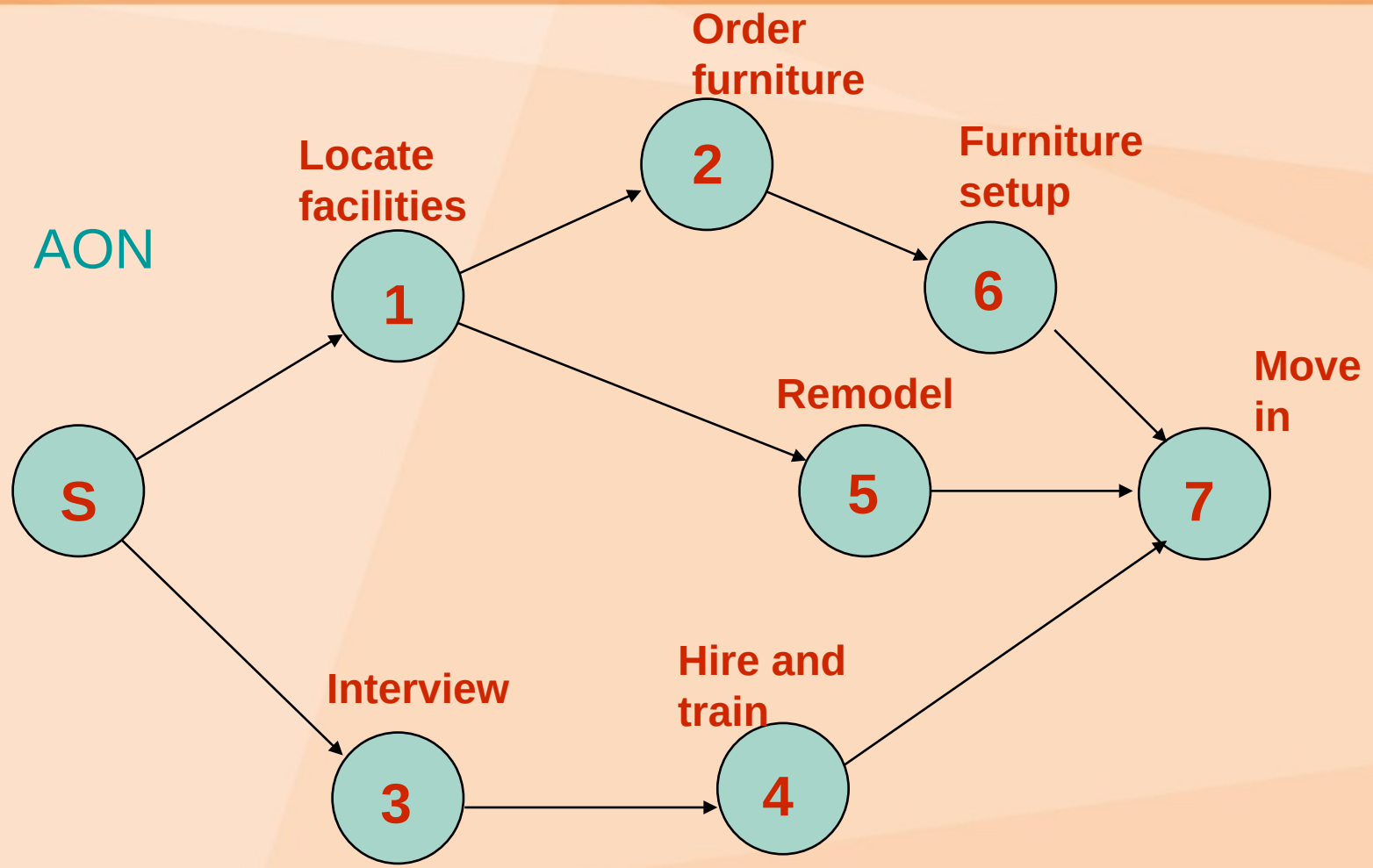
# Project Network – Activity on Arrow

Figure 17.4

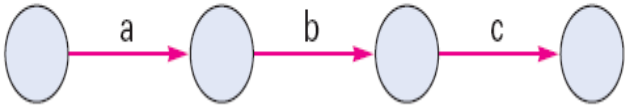
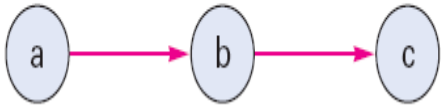
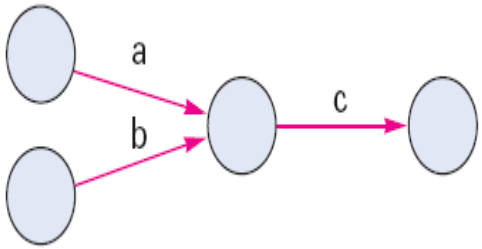
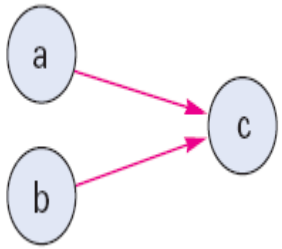
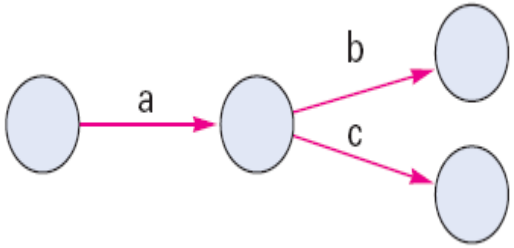
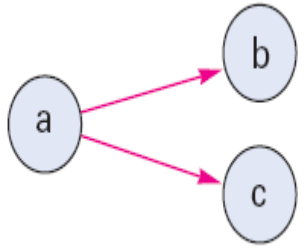


# Project Network – Activity on Node

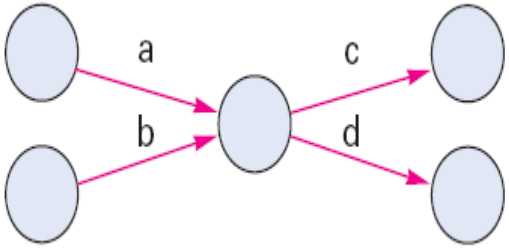
Figure 17.4



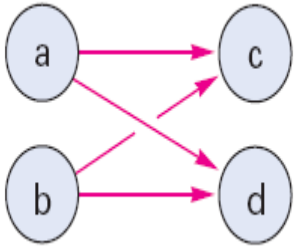
# Network Conventions

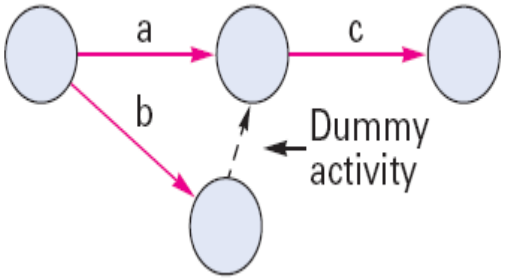
| AOA | Interpretation | AON |
| --- | --- | --- |
|  | Activities must be completed in sequence: first *a*, then *b*, and then *c*. |  |
|  | Both *a* and *b* must be completed before *c* can start. |  |
|  | Activity *a* must be completed before *b* or *c* can start. |  |

# Network Conventions




Both *a* and *b* must be completed before *c* or *d* can start.

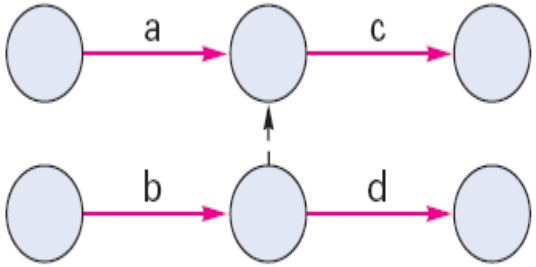



Use a dummy activity to clarify relationships:

1. To separate two activities that have the same starting and ending nodes.

(No dummy needed)




2. When activities share some, but not all, preceding activities. Here, activity *c* is preceded by activities *a* and *b*, while activity *d* is only preceded by activity *b*.

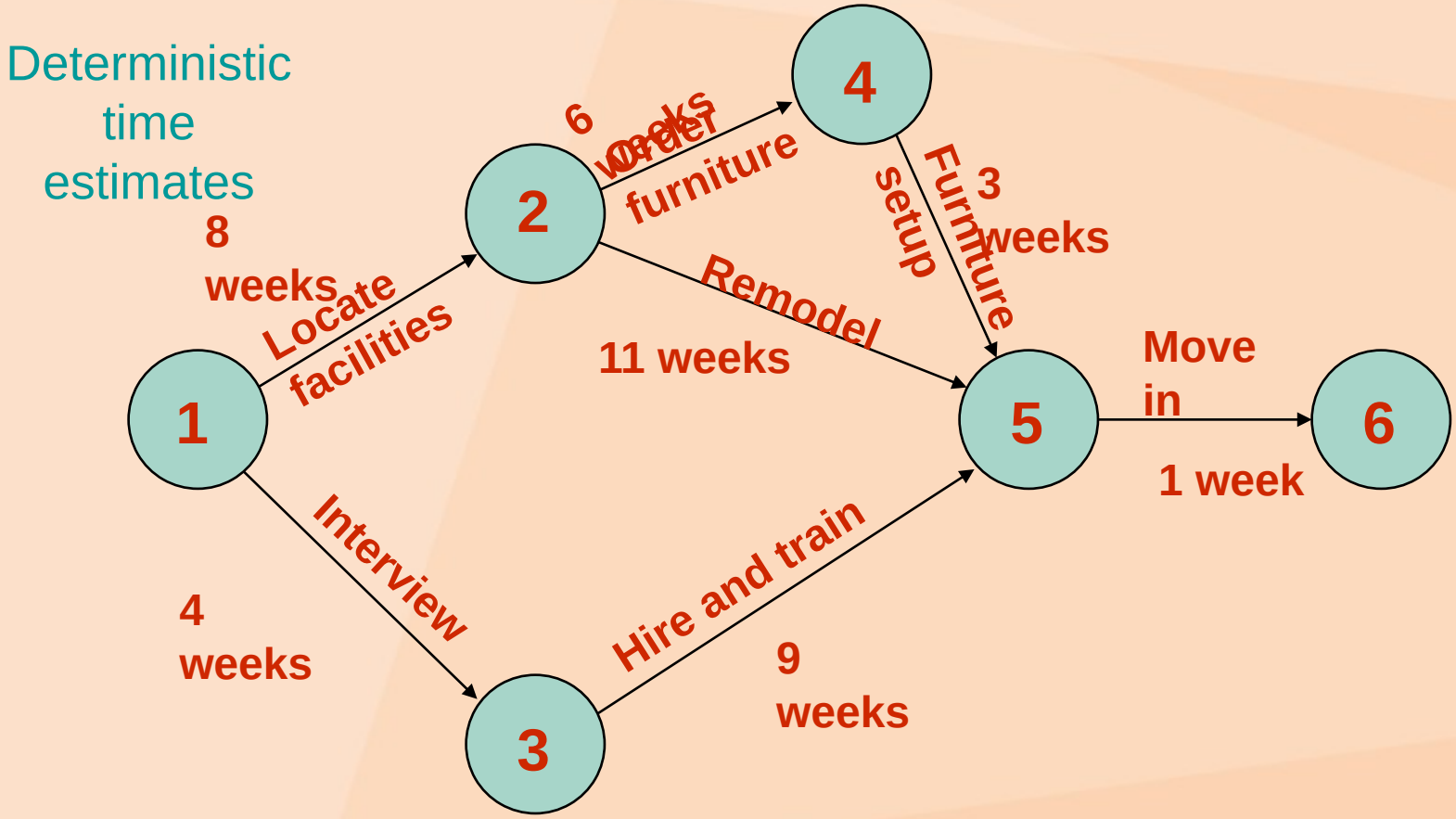
(No dummy needed)

# Time Estimates

- Deterministic
  - Time estimates that are fairly certain
- Probabilistic
  - Estimates of times that allow for variation

# Example 1

Figure 17.5



# Example 1 Solution

**Critical Path**

| Path | Length (weeks) | Slack |
|---|---|---|
| 1 - 2 - 3 - 4 - 5 - 6 | 18 | 2 |
| 1 - 2 - 5 - 6 | 20 | 0 |
| 1 - 3 - 5 - 6 | 14 | 6 |

# Computing Algorithm

- Planners use an algorithm to develop:
  - ES: early start
  - EF: early finish
  - LS: late start
  - LF: late finish
- Used to determine:
  - Expected project duration
  - Slack time
  - Critical path

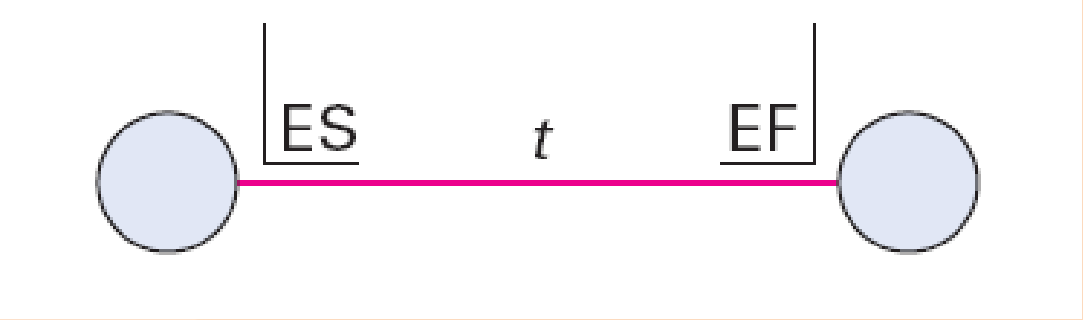
# Early Start, Early Finish

- Finding ES and EF involves a forward pass through the network diagram
  - **Early start (ES)**
    - The earliest time an activity can start
    - Assumes all preceding activities start as early as possible
      - For nodes with one entering arrow
        - **ES = EF of the entering arrow**
      - For activities leaving nodes with multiple entering arrows
        - **ES = the largest of the largest entering EF**
  - **Early finish (EF)**
    - The earliest time an activity can finish
      - **EF = ES + *t***

# Late Start, Late Finish

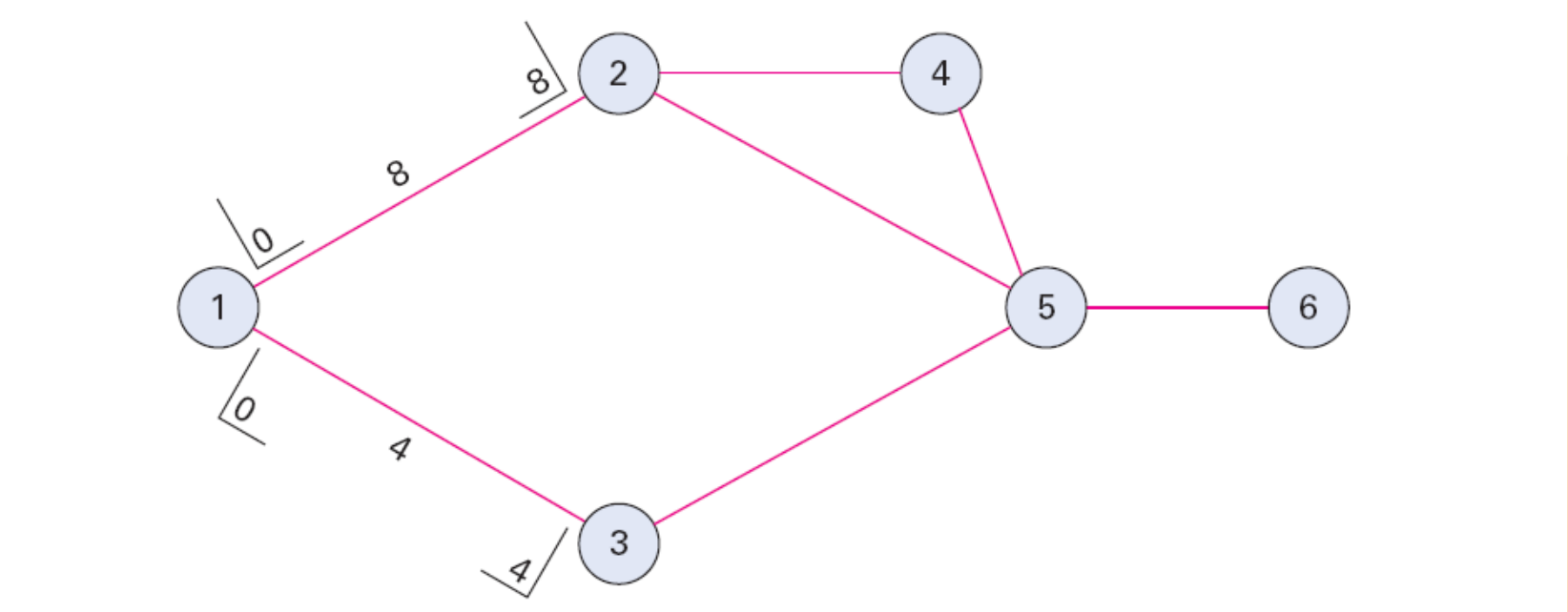
- Finding LS and LF involves a backward pass through the network diagram
  - **Late Start (LS)**
    - The latest time the activity can start and not delay the project
      - The latest starting time for each activity is equal to its latest finishing time minus its expected duration:
        - **LS = LF - *t***
  - **Late Finish (LF)**
    - The latest time the activity can finish and not delay the project
      - For nodes with one leaving arrow, LF for nodes entering that node equals the LS of the leaving arrow
      - For nodes with multiple leaving arrows, LF for arrows entering node equals the smallest of the leaving arrows

# ES and EF



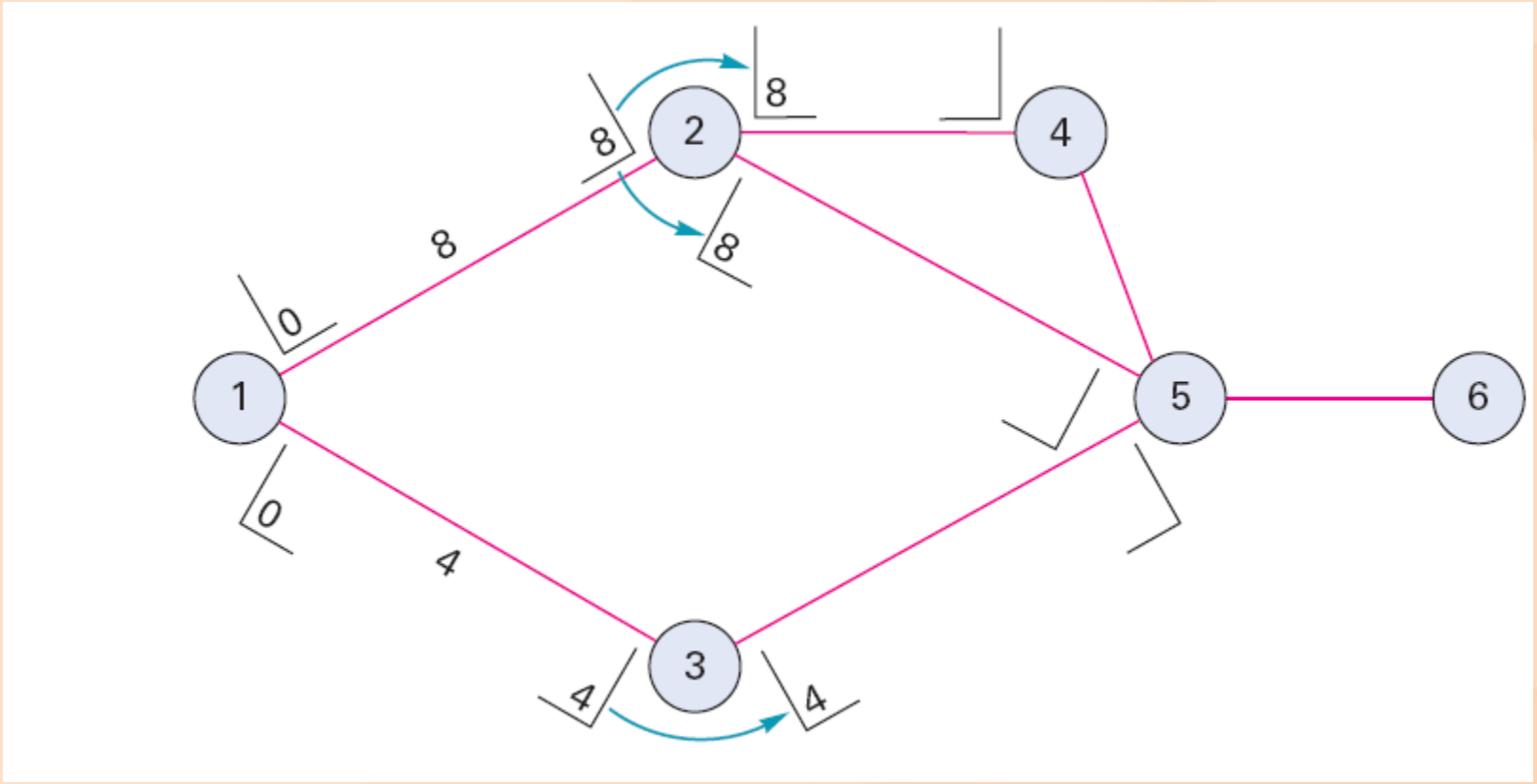

- Once ES has been determined for each activity, EF can be found by adding the activity time, *t,* to ES: ES + *t* = EF.

# ES and EF

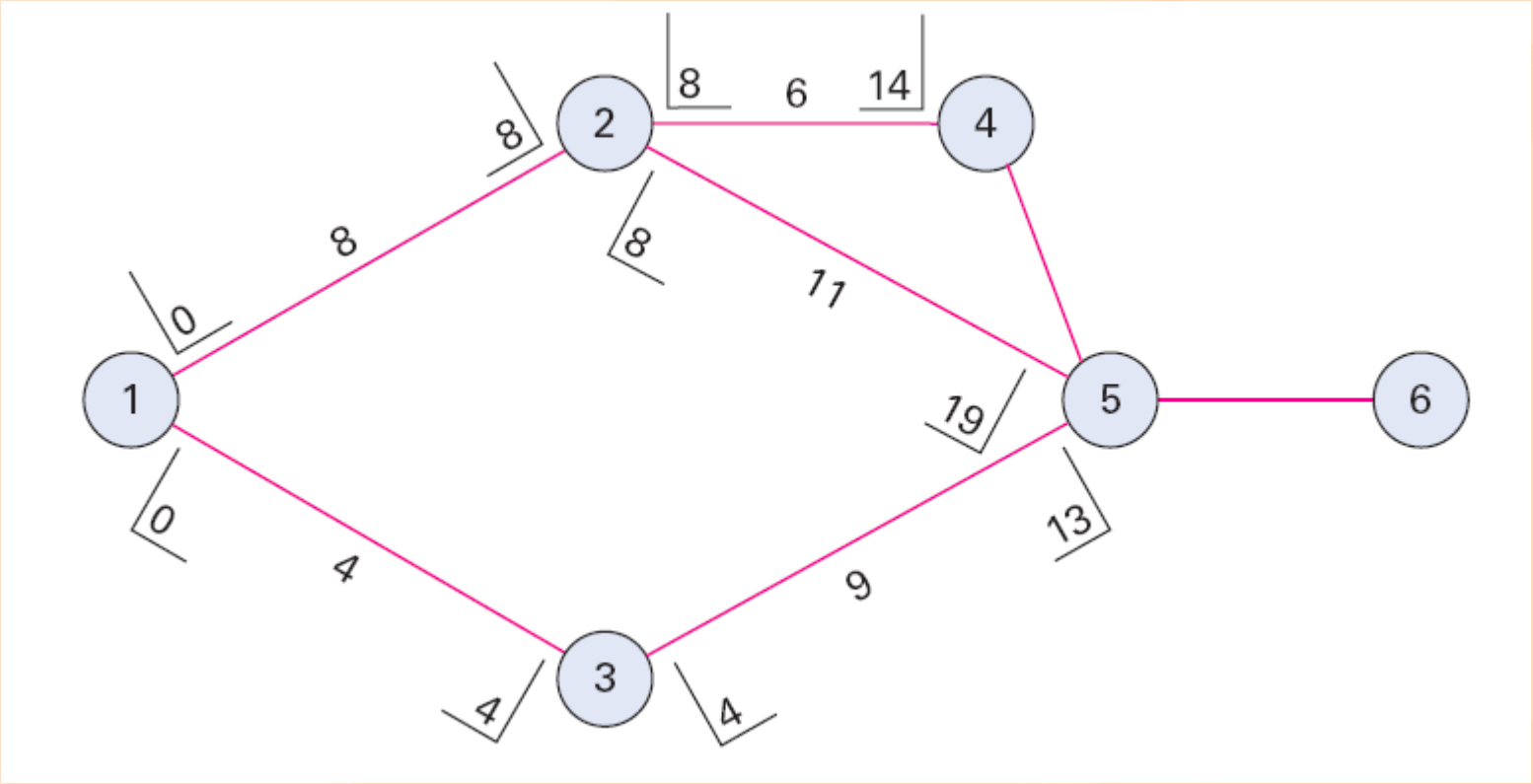



$EF_{1\text{-}2} = 0 + 8 = 8$ and $EF_{1\text{-}3} = 0 + 4 = 4$

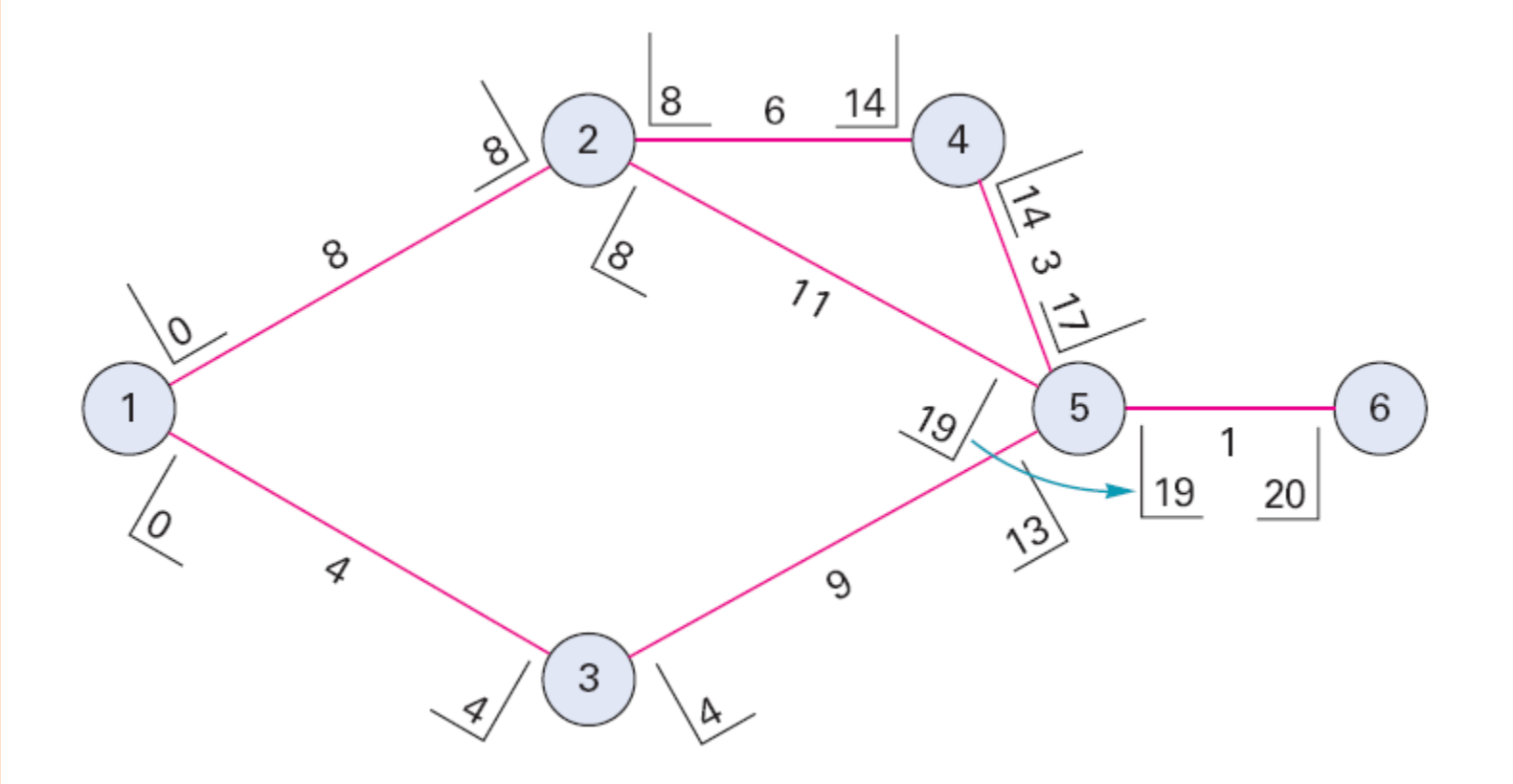
# ES and EF



# ES and EF



# ES and EF



# Slack and the Critical Path

- Slack can be computed one of two ways:
  - Slack = LS – ES
  - Slack = LF – EF
- Critical path
  - The critical path is indicated by the activities with zero slack

# Slack and the Critical Path

- Slack can be computed one of two ways:
  - Slack = LS – ES
  - Slack = LF – EF

| Activity | LS | ES | (LS – ES) Slack |
|---|---|---|---|
| 1-2 | 0 | 0 | 0 |
| 1-3 | 6 | 0 | 6 |
| 2-4 | 10 | 8 | 2 |
| 2-5 | 8 | 8 | 0 |
| 3-5 | 10 | 4 | 6 |
| 4-5 | 16 | 14 | 2 |
| 5-6 | 19 | 19 | 0 |

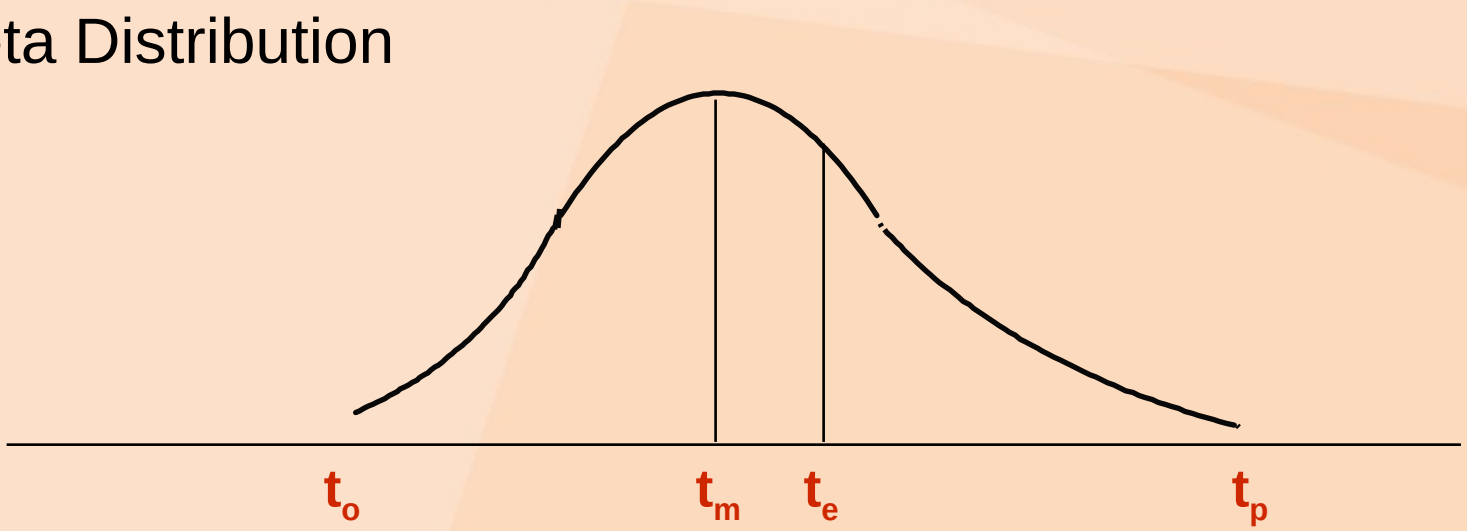
- Critical path
  - The critical path is indicated by the activities with zero slack

# Probabilistic Time Estimates

- The beta distribution is generally used to describe the inherent variability in time estimates
- The probabilistic approach involves three time estimates:
  - **Optimistic time, ($t_o$)**
    - The length of time required under optimal conditions
  - **Pessimistic time, ($t_p$)**
    - The length of time required under the worst conditions
  - **Most likely time, ($t_m$)**
    - The most probable length of time required

# Probabilistic Estimates

Figure 17.8



# Expected Time

$$t_e = \frac{t_o + 4t_m + t_p}{6}$$

$t_e$ = expected time

$t_o$ = optimistic time

$t_m$ = most likely time

$t_p$ = pessimistic time

# Variance

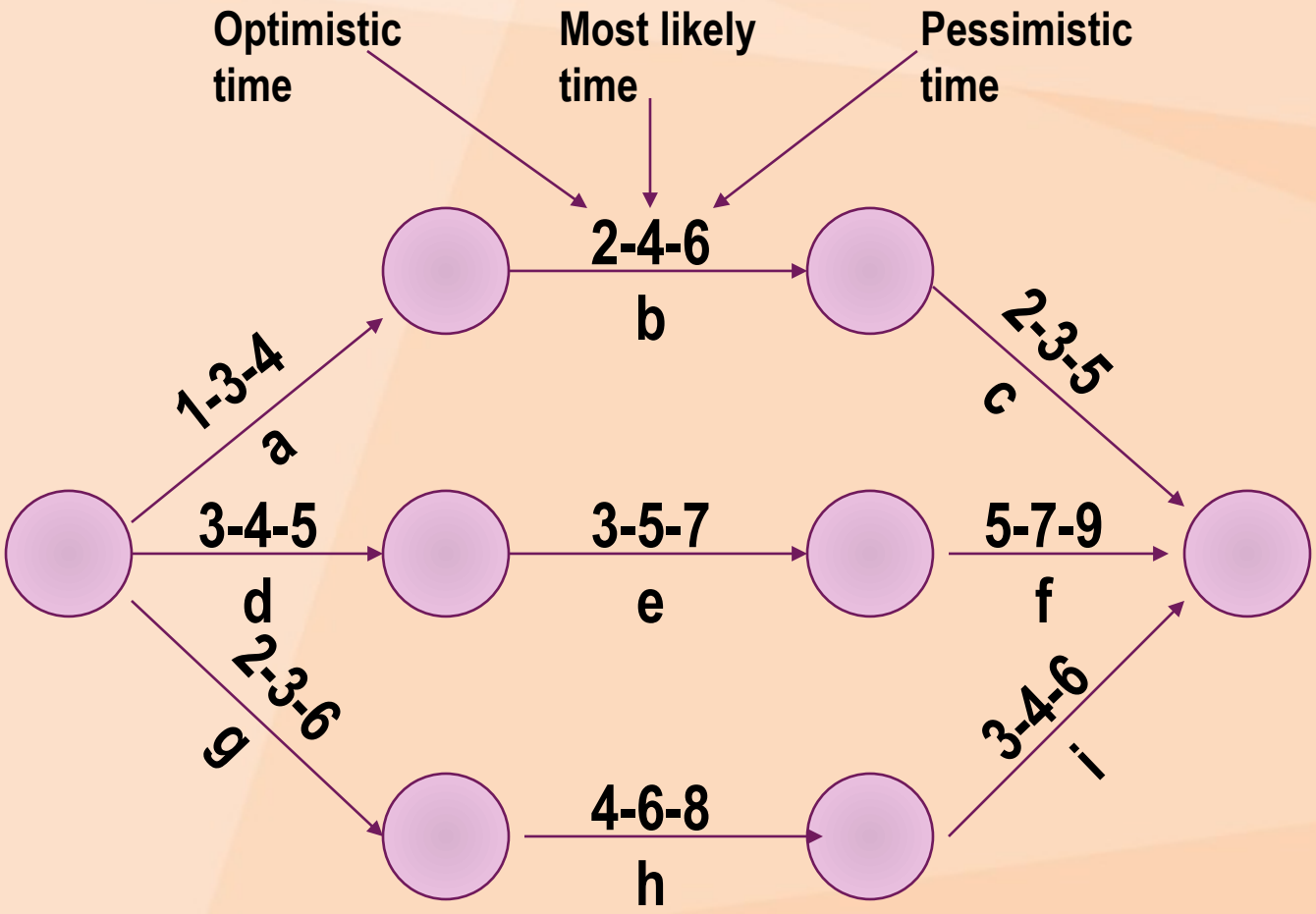
$$\sigma^2 = \frac{(t_p - t_o)^2}{36}$$

$\sigma^2$ = variance

$t_o$ = optimistic time

$t_p$ = pessimistic time

# Example 5

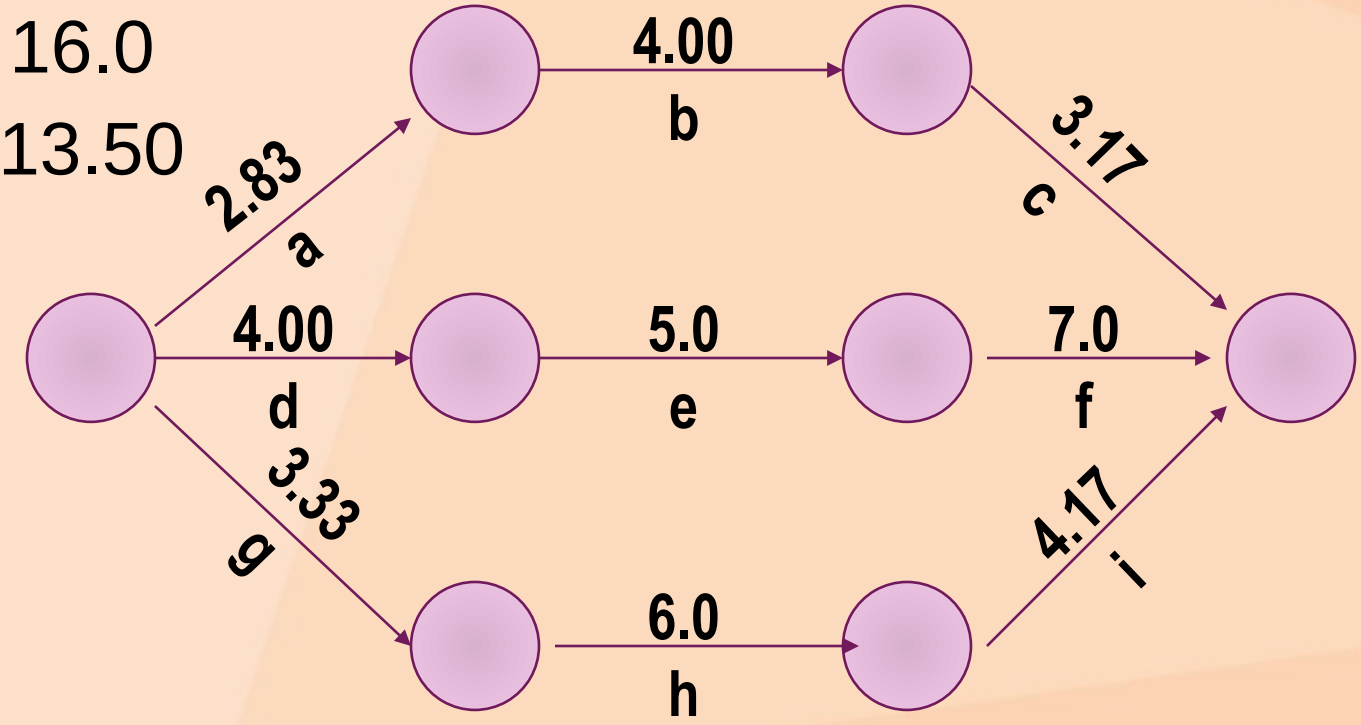

Optimistic time
Most likely time
Pessimistic time
2-4-6
b
1-3-4
a
2-3-5
c
3-4-5
d
3-5-7
e
5-7-9
f
2-3-6
g
4-6-8
h
3-4-6
i

# Example 5 Time Estimates

$T_{abc} = 10.0$

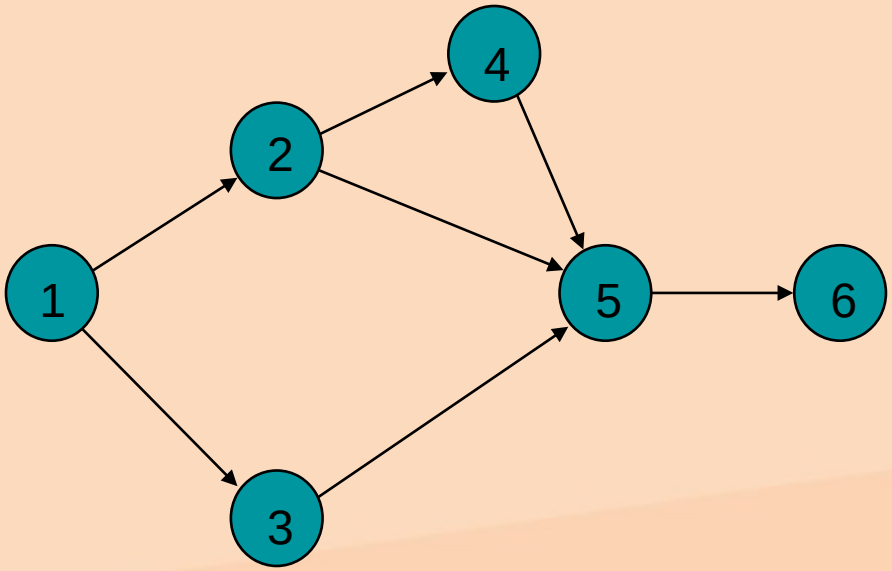
$T_{def} = 16.0$

$T_{ghi} = 13.50$



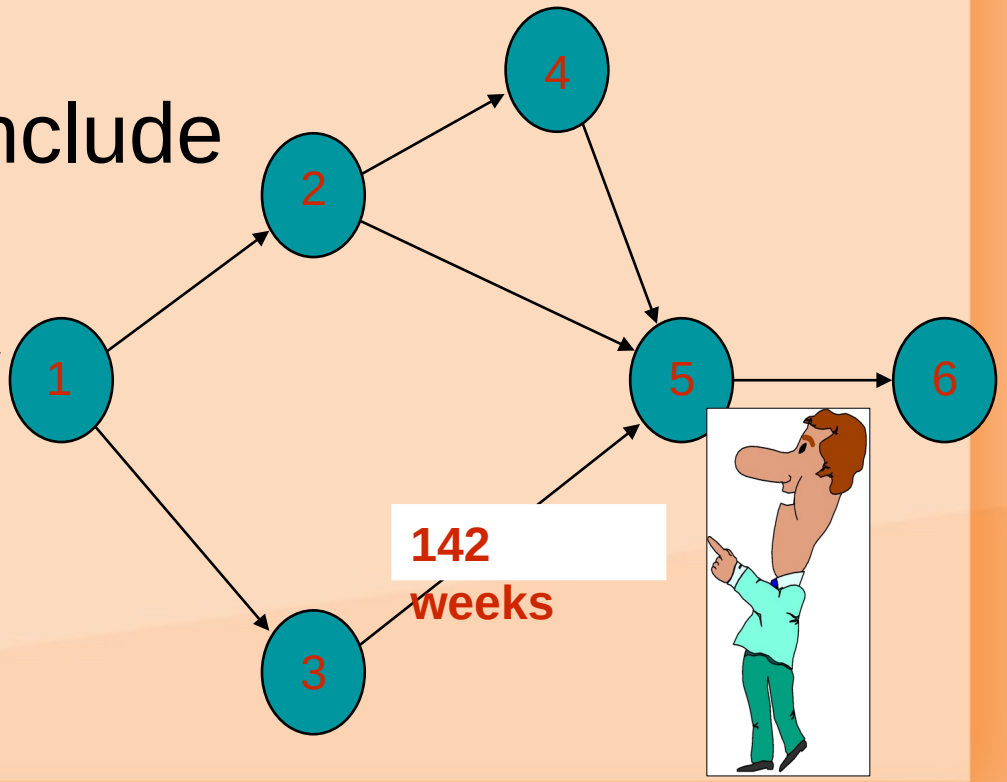
# Advantages of PERT

- Forces managers to organize
- Provides graphic display of activities
- Identifies
  - Critical activities
  - Slack activities



# Limitations of PERT

- Important activities may be omitted
- Precedence relationships may not be correct
- Estimates may include a fudge factor
- May focus solely on critical path



# Project Management Software

- Computer aided design (CAD)
- Groupware (Lotus Notes)
- CA Super Project
- Harvard Total Manager
- MS Project
- Sure Track Project Manager
- Time Line

# Advantages of PM Software

- Imposes a methodology
- Provides logical planning structure
- Enhances team communication
- Flag constraint violations
- Automatic report formats
- Multiple levels of reports
- Enables what-if scenarios
- Generates various chart types

# Project Risk Management

- Risk: occurrence of events that have undesirable consequences
  - Delays
  - Increased costs
  - Inability to meet specifications
  - Project termination

# Risk Management

- Identify potential risks
- Analyze and assess risks
- Work to minimize occurrence of risk
- Establish contingency plans