Activity Planning

Introduction

- A detailed plan should include a schedule indicating the start and completion time for each activity. This enable us to:
  - Ensure appropriate resources are available when required.
  - Avoid different activities competing for the same resources at the same time.
  - Produce a detailed schedule showing which staff carry out each activity.
  - Produce time cash flow forecast.
  - Replan the project during its correct drift from the target.

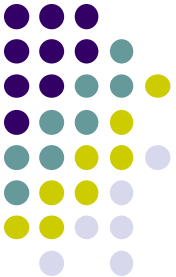
When to plan?

- Planning is an ongoing process of refinement.
- Each iteration become more accurate and detailed than the last.
- Purpose of planning during feasibility study and project start up:
  - To estimate timescales
  - Risk for not achieving target completion dates
  - Keeping within budget.
  - Ensuring resource availability.

0bjectives of Activity Planning

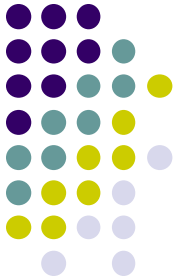
- **Feasibility assessment**: Is the project possible within required timescales and resource constraint ?
- **Resource allocation**: What are most effective ways of allocating resources and when they should be available.
- **Detailed costing**: How much will the project cost and when that expenditure will take place?
  - After activity plan and allocating resources we can obtain detailed estimates of costs and timing.
- **Motivation**: Providing targets and monitor the achievement.
- **Co-ordination**: When do the staff need to be available to work on a particular project.
  - When staff need to be transferred between projects?
  - Co-ordination among teams when involved in large projects.

Project Schedules



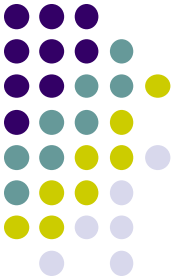
- Project plan must be developed to the level of showing dates when each activity should start and finish and how much of each resource will be required.
- Once plan has been refined in detail it become the ***project schedule***.
- Project schedule comprises of four main stages:
  1. Produce an ideal activity plan (when & in what order).
  2. Ideal activity plan then subject to activity risk analysis.
  3. Resource allocation, availability of resources.
  4. Schedule production, after allocation we can draw up and publish a project schedule indicating the start and end dates

Identifying activities



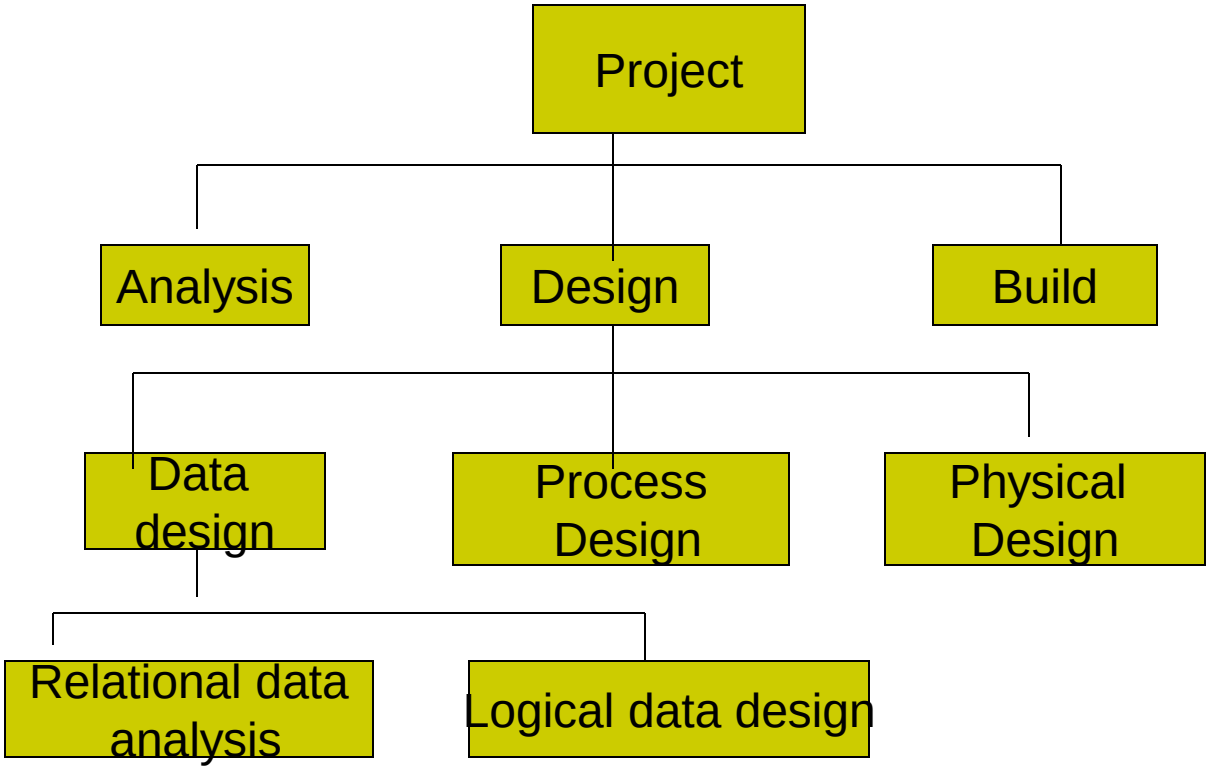
- Project composed of a number of interrelated activities.
- Three approaches to identify the activities:
  - Activity based approach
  - Product based approach
  - Hybrid approach

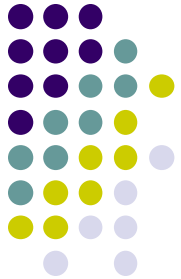
Activity Based Approach



- Creating a list of all activities that the project involve.
- Common way of generating a task list is to create a ***Work Breakdown Structure*** (WBS).
- **WBS** is an exhaustive, hierarchical (from general to specific) tree structure of deliverables and tasks that need to be performed to complete a project.
- I.e. identify the main task and breaking each of these into set of lower level tasks.
- A well-designed WBS describes planned outcomes instead of planned actions.
- Outcomes are the desired ends of the project, and can be predicted accurately; actions comprise the project plan and may be difficult to predict accurately.

Work Breakdown Structure





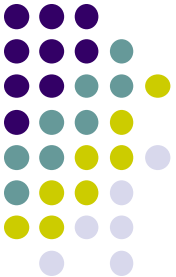
WBS Outline Example

0.0 Retail Web Site
1.0 Project Management
2.0 Requirements Gathering
3.0 Analysis & Design
4.0 Site Software Development
    4.1 HTML Design and Creation
    4.2 Backend Software
        4.2.1 Database Implementation
        4.2.2 Middleware Development
        4.2.3 Security Subsystems
        4.2.4 Catalog Engine
        4.2.5 Transaction Processing
    4.3 Graphics and Interface
    4.4 Content Creation
5.0 Testing and Production

Product Based Approach

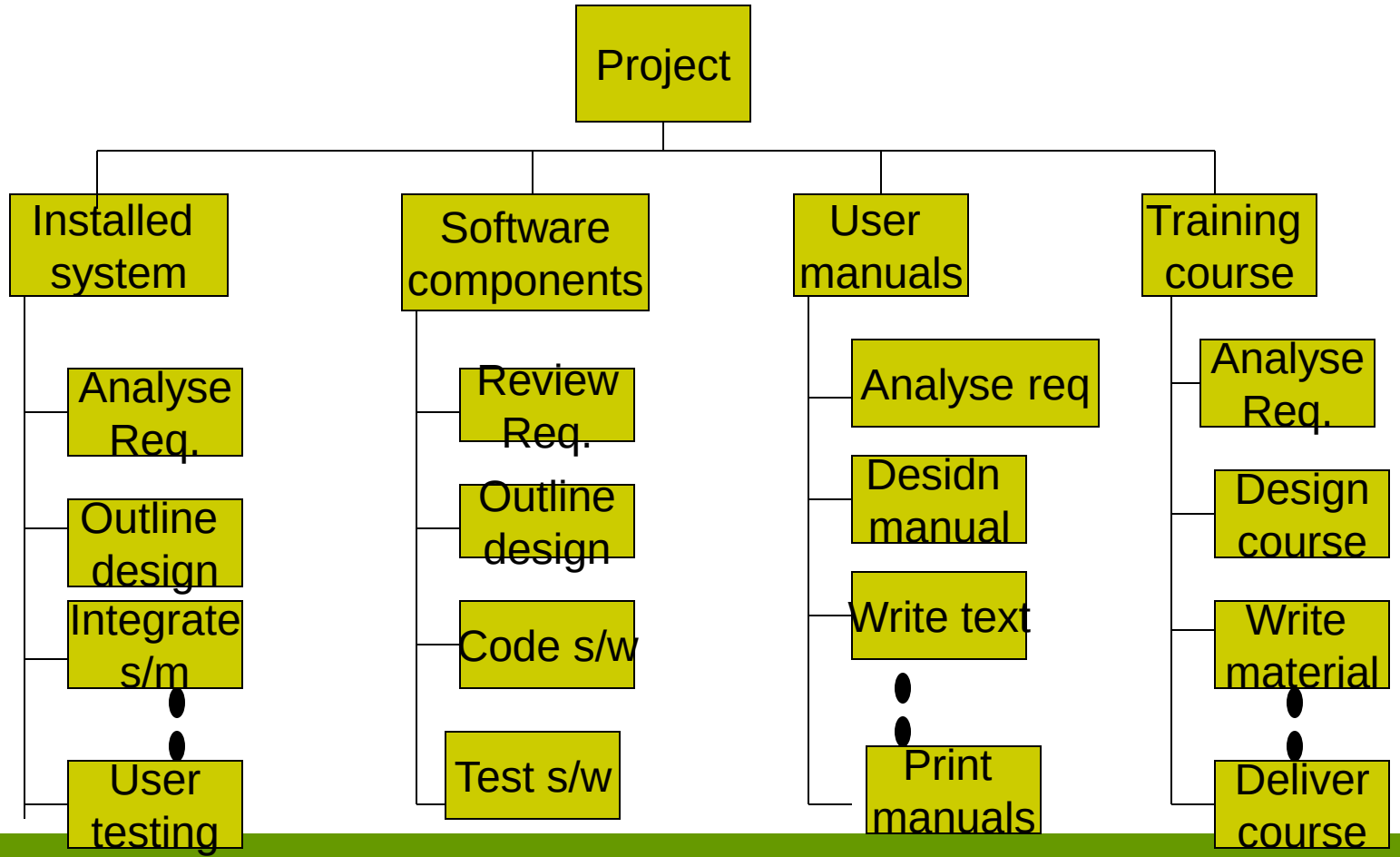
- It consists of producing PBS and PFD.
- PFD indicates for each product, which other products are required as input.

Hybrid Based Approach

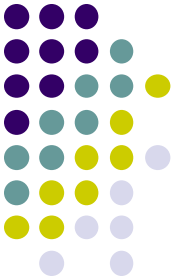


- WBS discussed based on structuring of activities.
- WBS can also be based on simple list of final deliverables.
- Consists of five levels:
  - Level 1: **Project**
  - Level 2: **Deliverables** such as software, manuals, training courses.
  - Level 3:**Components**,key items needed to produce deliverables.
  - Level 4; **Work packages**, major work items for making components.
  - Level 5: **Task**, responsibility of a single person.

WBS based on deliverables

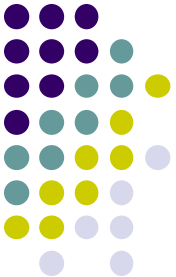


Project Management Methods



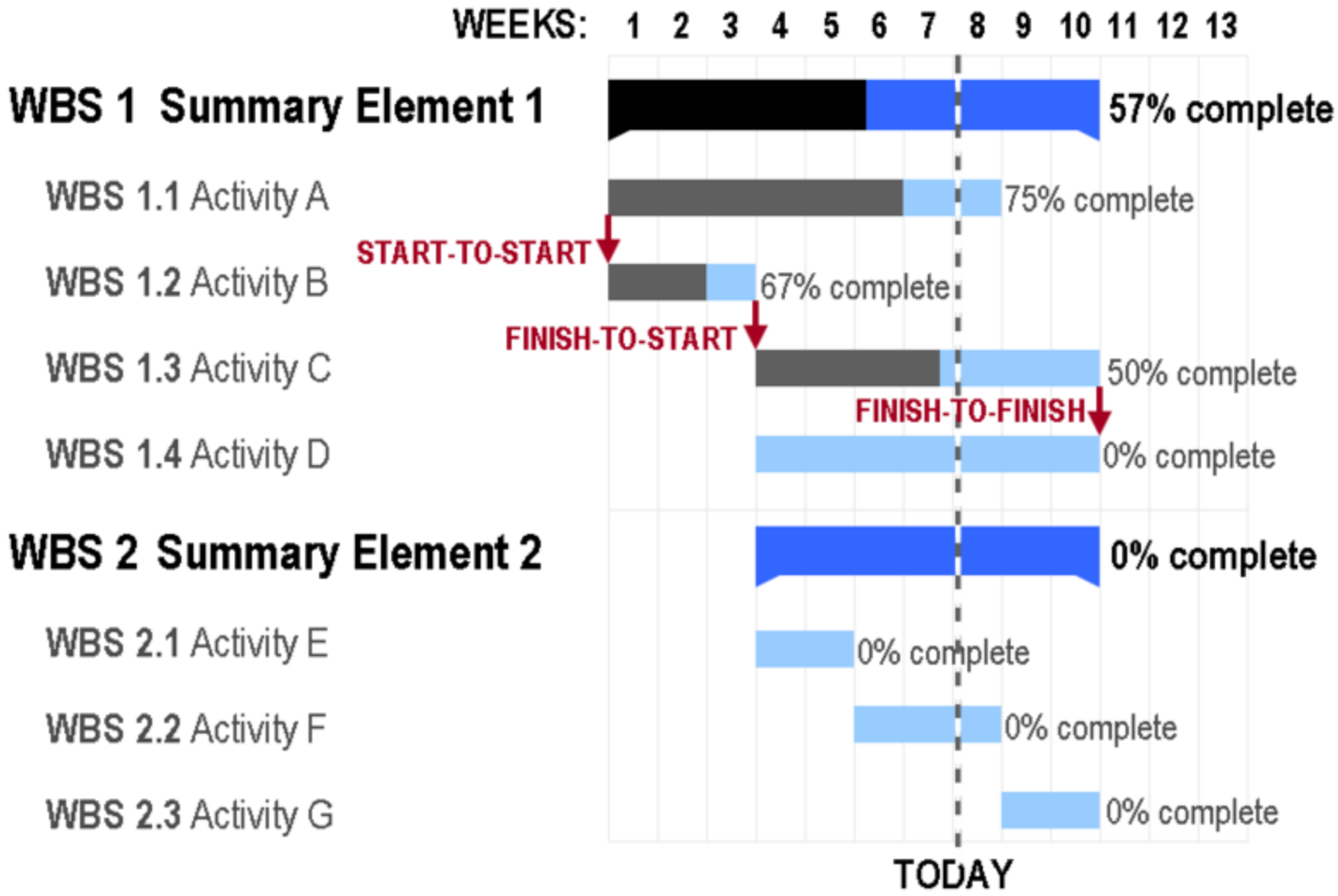
- Bar Chart / Gantt Chart
- Milestone Chart
- Pert
- CPM

Gantt Chart

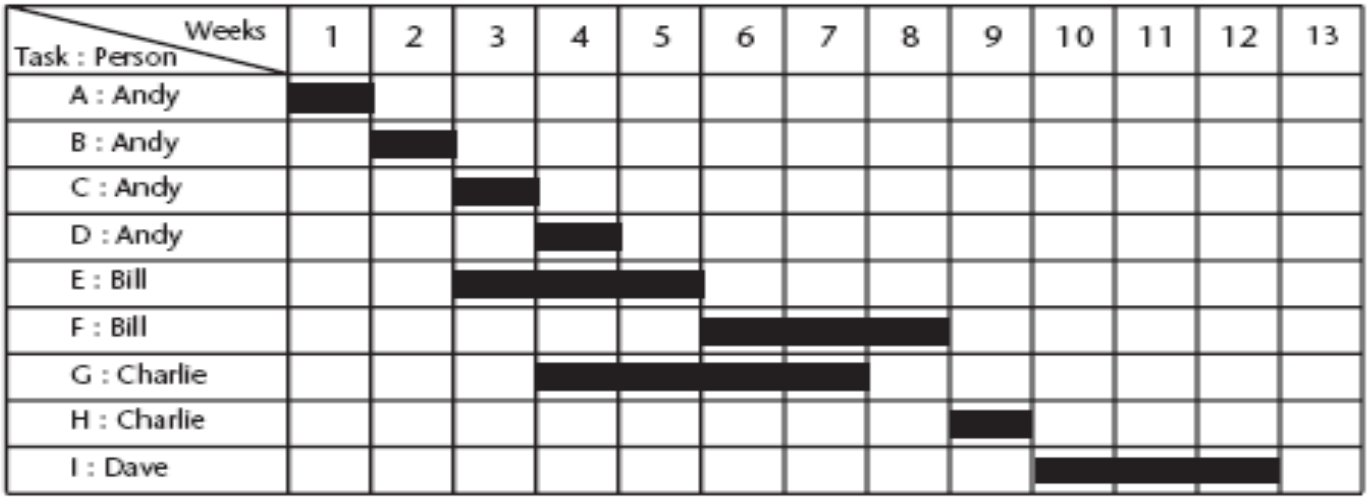


- A popular type of bar chart that illustrates a ***project schedule.***
- Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project.
- Terminal elements and summary elements comprise the WBS of the project.
- Also show the dependency (i.e., precedence network) relationships between activities.

Gantt Chart


WEEKS: 1 2 3 4 5 6 7 8 9 10 11 12 13
WBS 1 Summary Element 1
57% complete
WBS 1.1 Activity A
75% complete
START-TO-START
WBS 1.2 Activity B
67% complete
FINISH-TO-START
WBS 1.3 Activity C
50% complete
FINISH-TO-FINISH
WBS 1.4 Activity D
0% complete
WBS 2 Summary Element 2
0% complete
WBS 2.1 Activity E
0% complete
WBS 2.2 Activity F
0% complete
WBS 2.3 Activity G
0% complete
TODAY

Bar chart



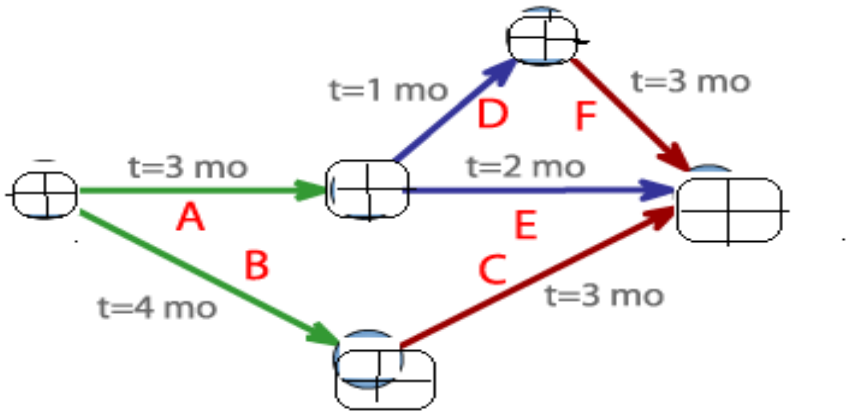
Activity key

A : Overall design
B : Specify module 1
C : Specify module 2
D : Specify module 3
E : Code module 1
F : Code module 3
G : Code module 2
H : Integration testing
I : Sytem tesing

PERT Chart

- ***Program Evaluation and Review Technique***, a model for project management to analyze and represent the tasks involved in completing a given project.
- Especially the time needed to complete each task, and identifying the minimum time needed to complete the total project.

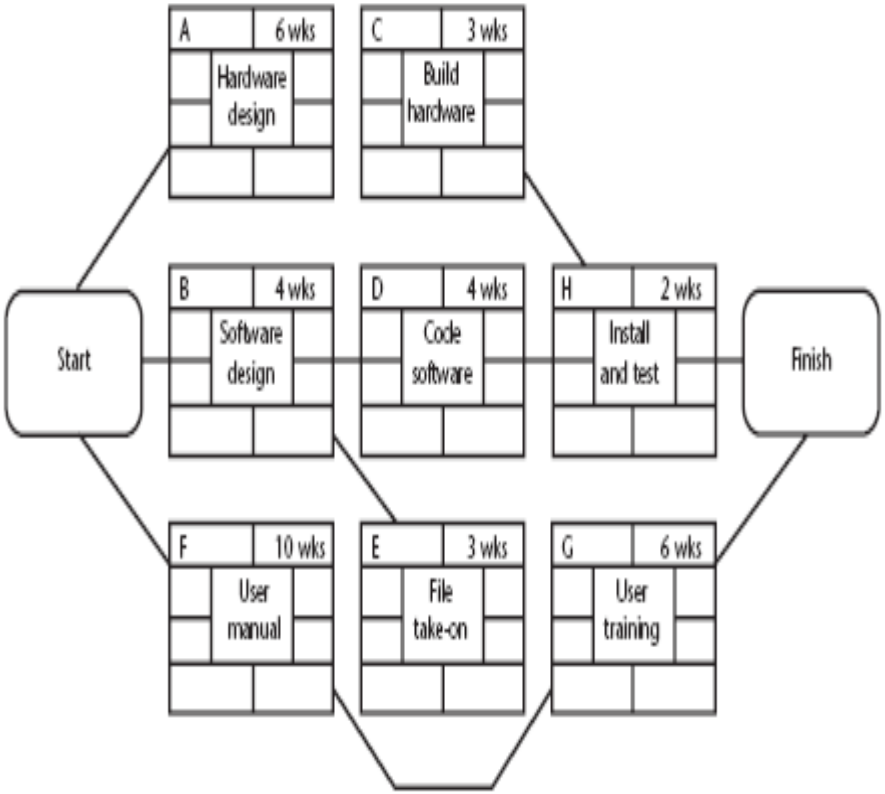
PERT chart



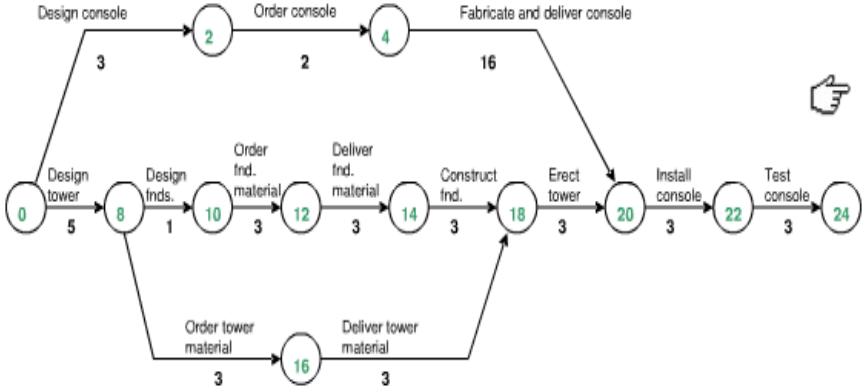
Activity Conventions

- Activity On Node Convention (AON)
  - Nodes are the project activities
  - links between nodes represent precedence relationships

- Activity On Arrow Convention (AOA)
  - Nodes are events: beginning or endings.
  - Arrows represents the activities.
  - CPM and PERT developers used AOA.



Activity on node



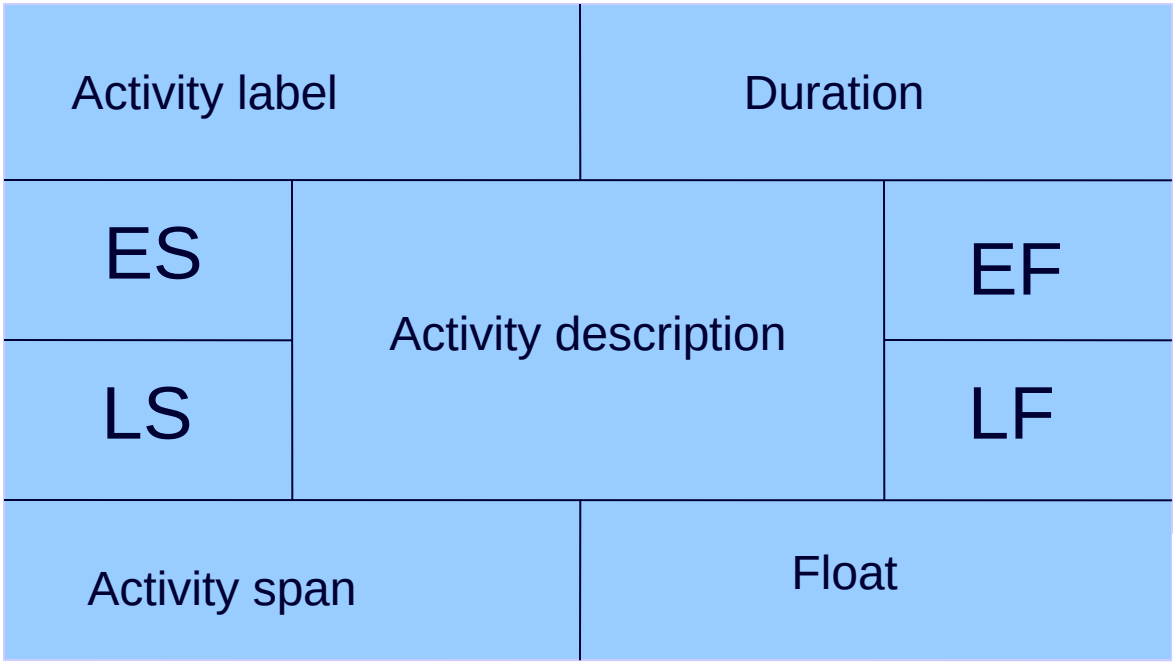
Activity on arrow

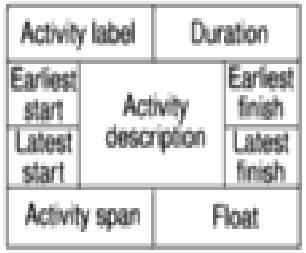
Constructing precedence network

Some rules for the construction of precedence network.

- A project network should have only one start node
- A project network should have only one end node
- A node has duration
- Links normally have no duration
- Precedents are the immediate preceding activities
- Times moves from left to right
- A network may not contain loops
- A network should not contain dangles

Notation


Activity label
Duration
ES
Activity description
EF
LS
LF
Activity span
Float


Activity label
Duration
Earliest start
Activity description
Earliest finish
Latest start
Latest finish
Activity span
Float

Start and finish times

- Activity ‘write report software’
- Earliest start (ES)
- Earliest finish (EF) = ES + duration
- Latest finish (LF) = latest task can be completed without affecting project end
  Latest start = LF - duration

Example

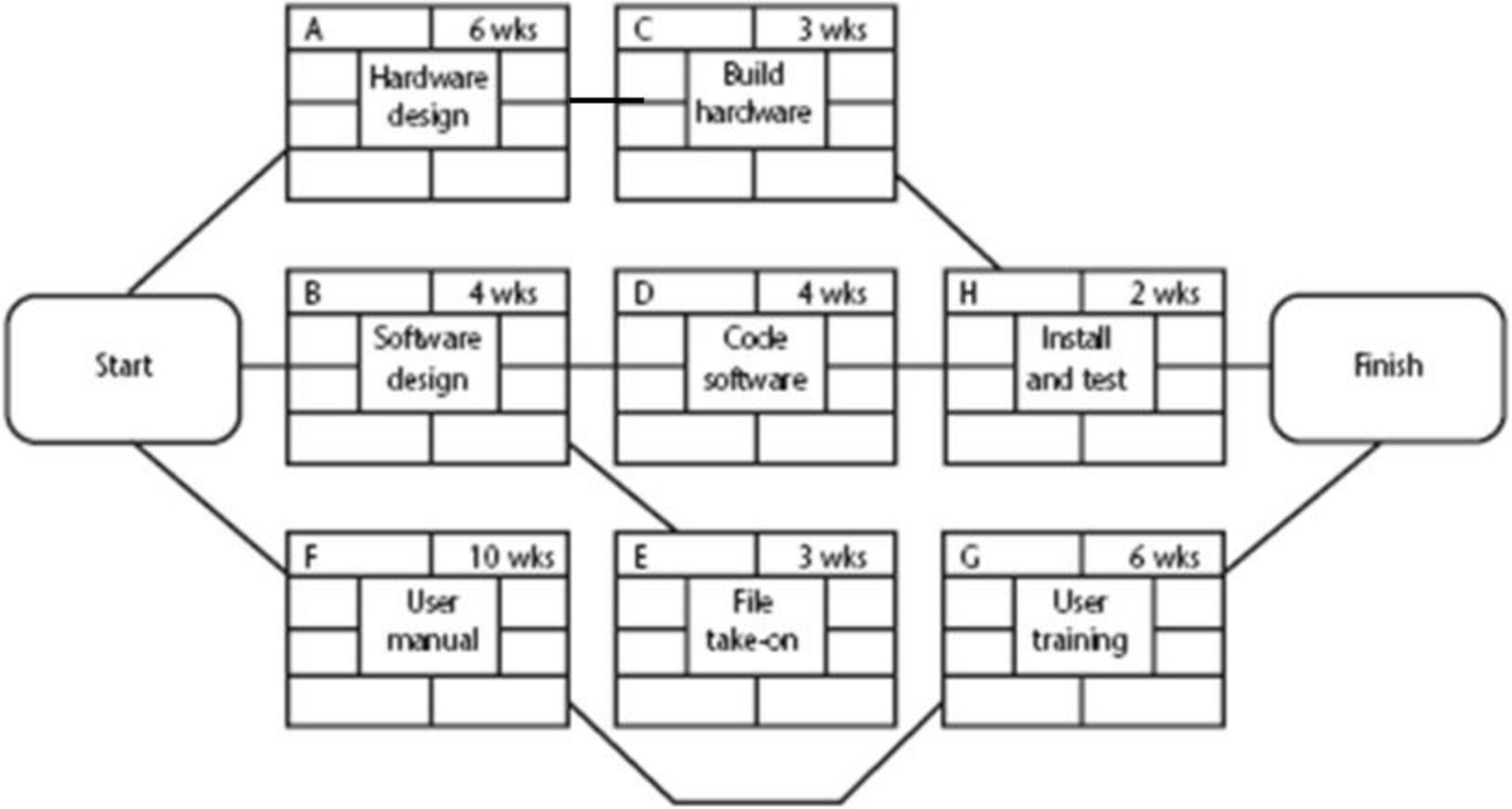
- earliest start = day 5
- latest finish = day 30
- duration = 10 days

- earliest finish = ?
- latest start = ?

Float = LF - ES - duration

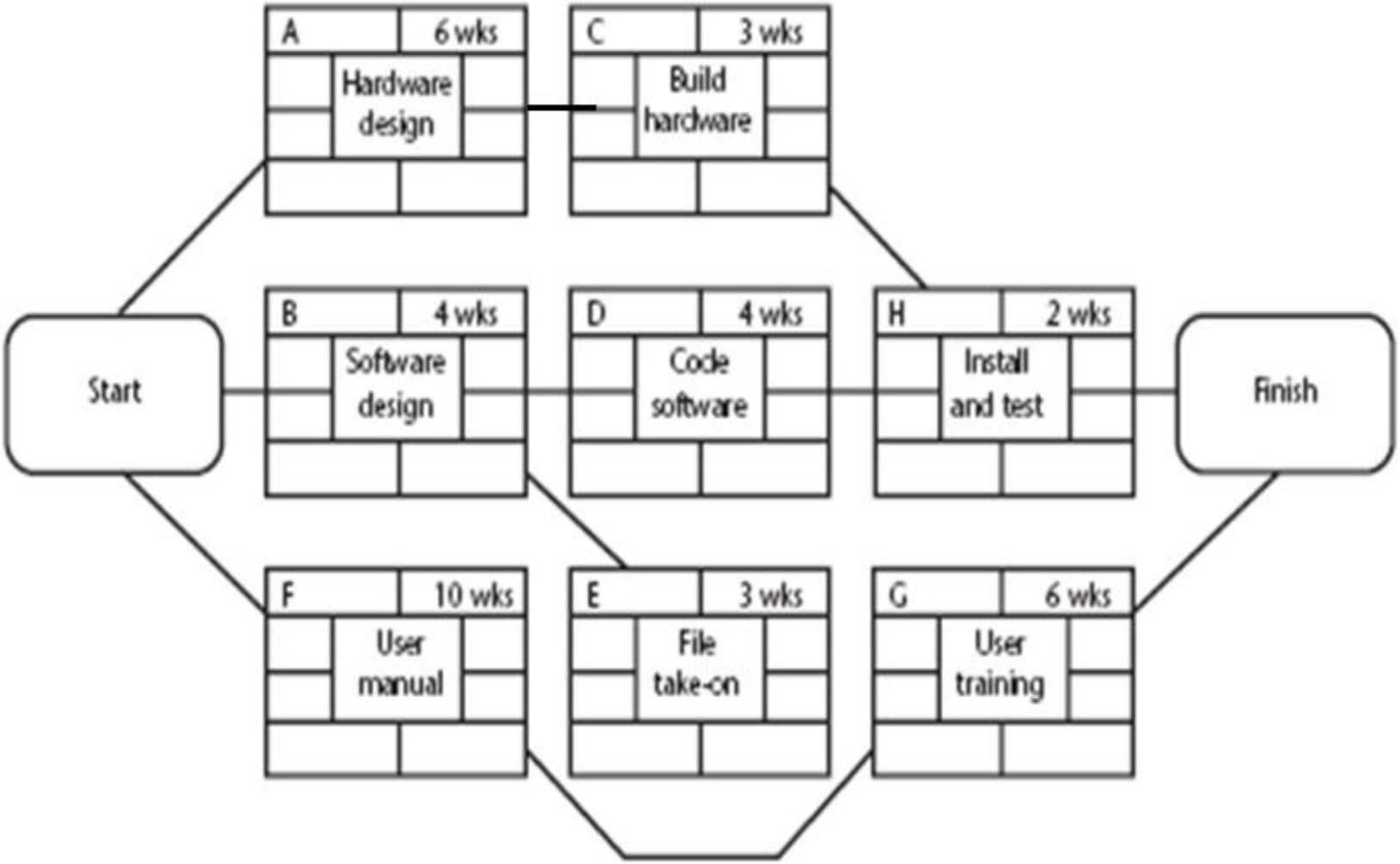
What is it in this case?

Example of an activity network



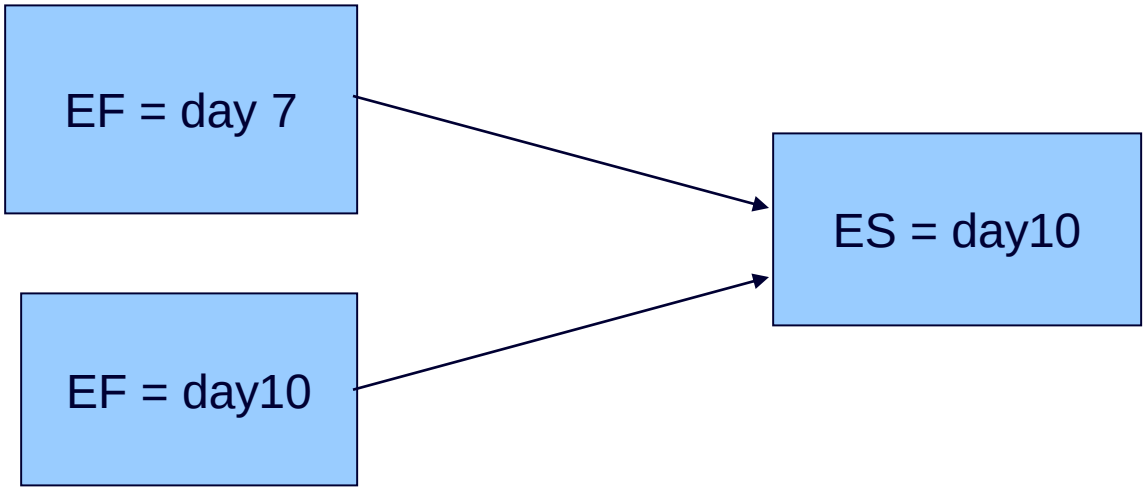
Example

Activity	Activity Description	Duration(weeks)	Precedents
A	Harware selection	6	
B	Software design	4	
C	Install hardware	3	A
D	code &test software	4	B
E	File take-on	3	B
F	Write user manuals	10	
G	User training	3	E,F
H	Install & test system	2	C,D


A
6 wks
Hardware design
C
3 wks
Build hardware
B
4 wks
Software design
D
4 wks
Code software
H
2 wks
Install and test
Start
Finish
F
10 wks
User manual
E
3 wks
File take-on
G
6 wks
User training

Earliest start date

- Earliest start date for the *current* activity = earliest finish date for the *previous*
- When there is more than one previous activity, take the *latest* earliest finish
- Note ‘day 7’ = end of work on day 7



Latest start dates

- Start from the *last* activity
- Latest finish (LF) for last activity = earliest finish (EF)
- work backwards
- Latest start (LS) = LF for activity – duration
- Latest finish for *current* activity = Latest start for the *following*
- More than one following activity - take the *earliest* LS

Forward Pass

Begin at starting event and work forward

Earliest Start Time Rule:

- If an activity has only a single immediate predecessor, its ES equals the EF of the predecessor
- If an activity has multiple immediate predecessors, its ES is the maximum of all the EF values of its predecessors

ES = Max {EF of all immediate predecessors}

Backward Pass

Begin with the last event and work backwards

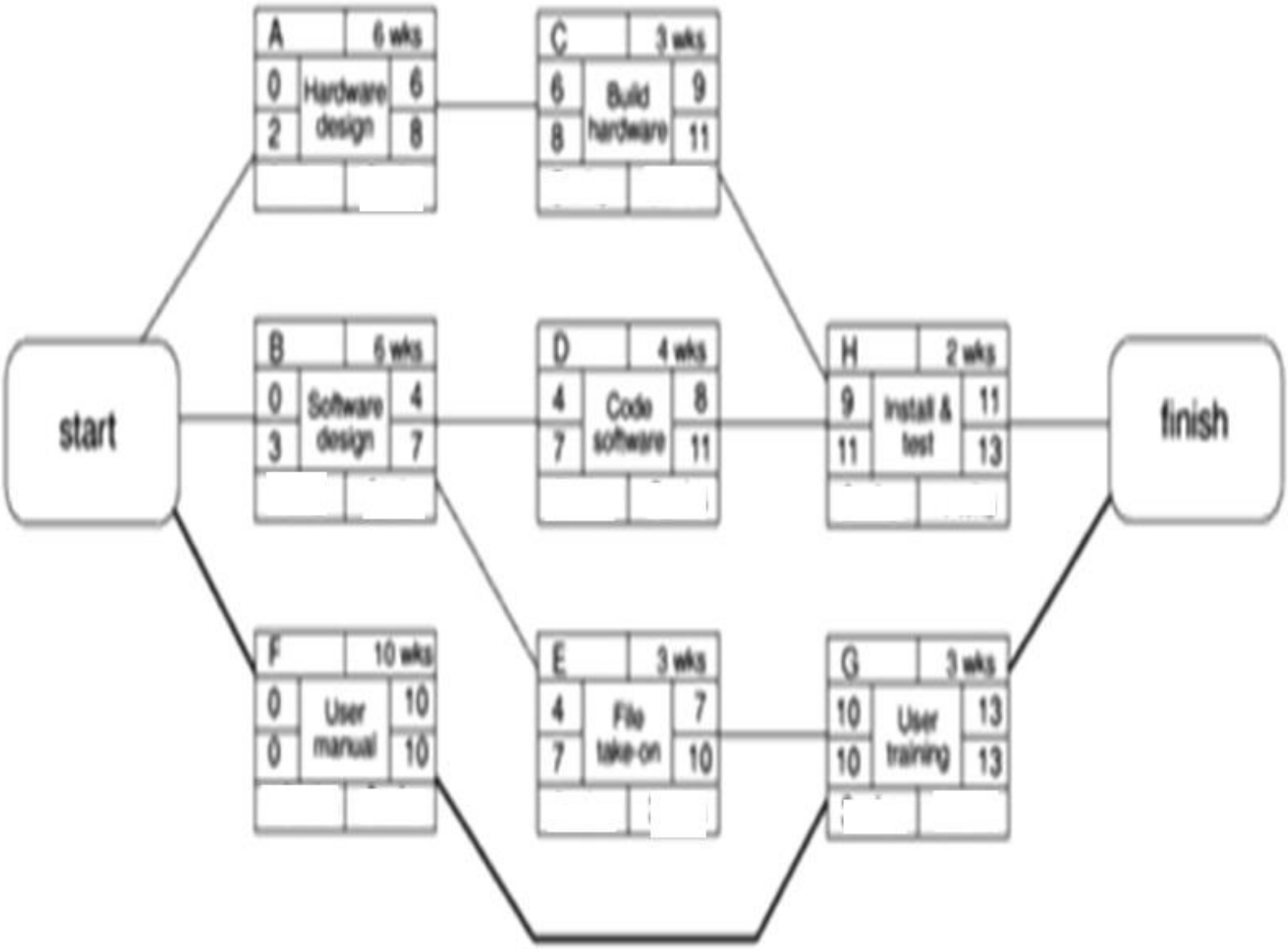
Latest Finish Time Rule:

- If an activity is an immediate predecessor for just a single activity, its LF equals the LS of the activity that immediately follows it
- If an activity is an immediate predecessor to more than one activity, its LF is the minimum of all LS values of all activities that immediately follow it

LF = Min {LS of all immediate following activities}

Table 6.4 *The activity schedule showing total float for each activity*

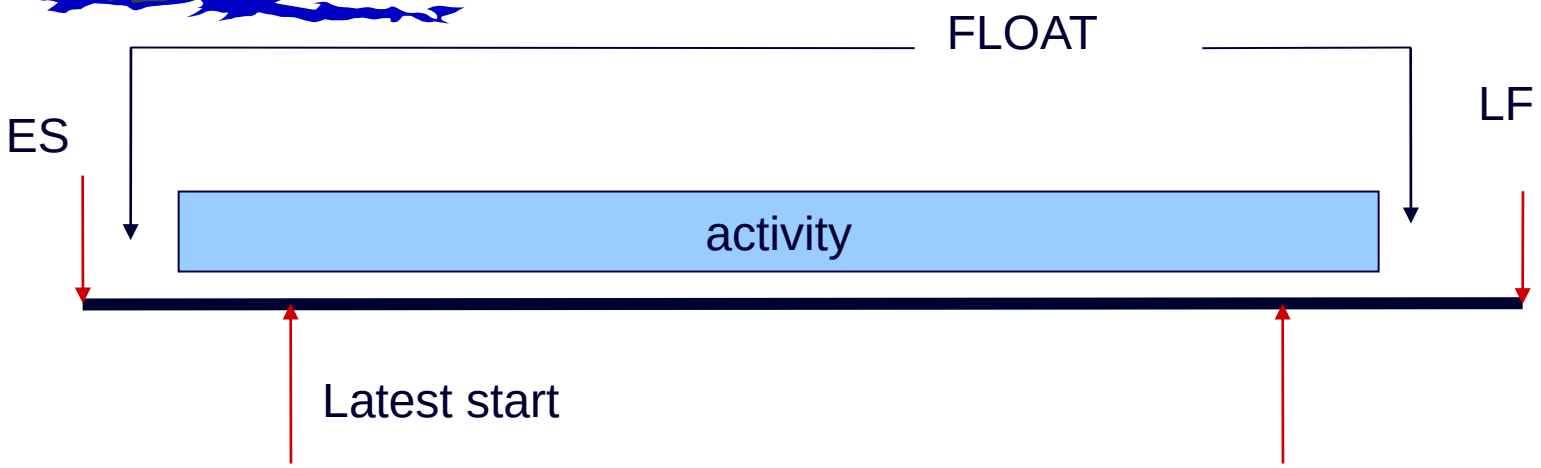
Activity	Precedents	Duration (weeks)	Earliest start date	Latest start date	Earliest finish fate	Latest finish date	Total float
A	_____	6	0	2	6	8	2
B	_____	4	0	3	4	7	3
C	A	3	6	8	9	11	2
D	B	4	4	7	8	11	3
E	B	3	4	7	7	10	3
F	_____	10	0	0	10	10	0
G	E,F	3	10	10	13	13	0
H	C,D	2	9	11	11	13	2


start
A
6 wks
0
2
Hardware design
6
8
B
6 wks
0
3
Software design
4
7
F
10 wks
0
0
User manual
10
10
C
3 wks
6
8
Build hardware
9
11
D
4 wks
4
7
Code software
8
11
E
3 wks
4
7
File take-on
7
10
H
2 wks
9
11
Install & test
11
13
G
3 wks
10
10
User training
13
13
finish

Float

Float = Latest finish -Earliest start - Duration

Float can also be calculated as the difference between the earliest and latest start dates for an activity *or* the difference between the earliest and latest finish dates.

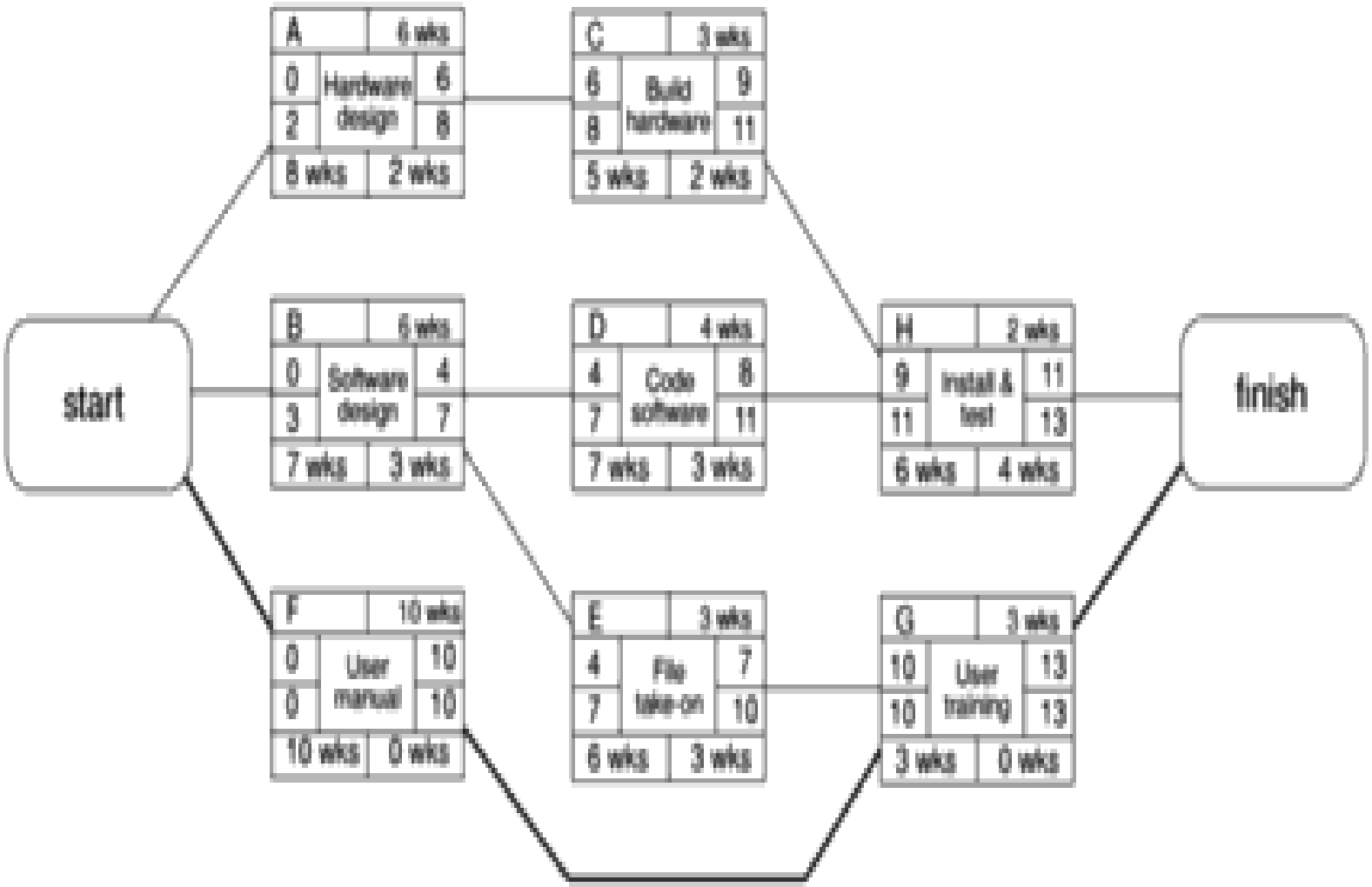


Activity Float =difference between the earliest and latest start dates for an activity *or* the difference between the earliest and latest finish dates.

Its a measure of how much the start or completion of an activity may be delayed without affecting the end date of the project.

Activity span= difference between the earliest start and latest finish dates.

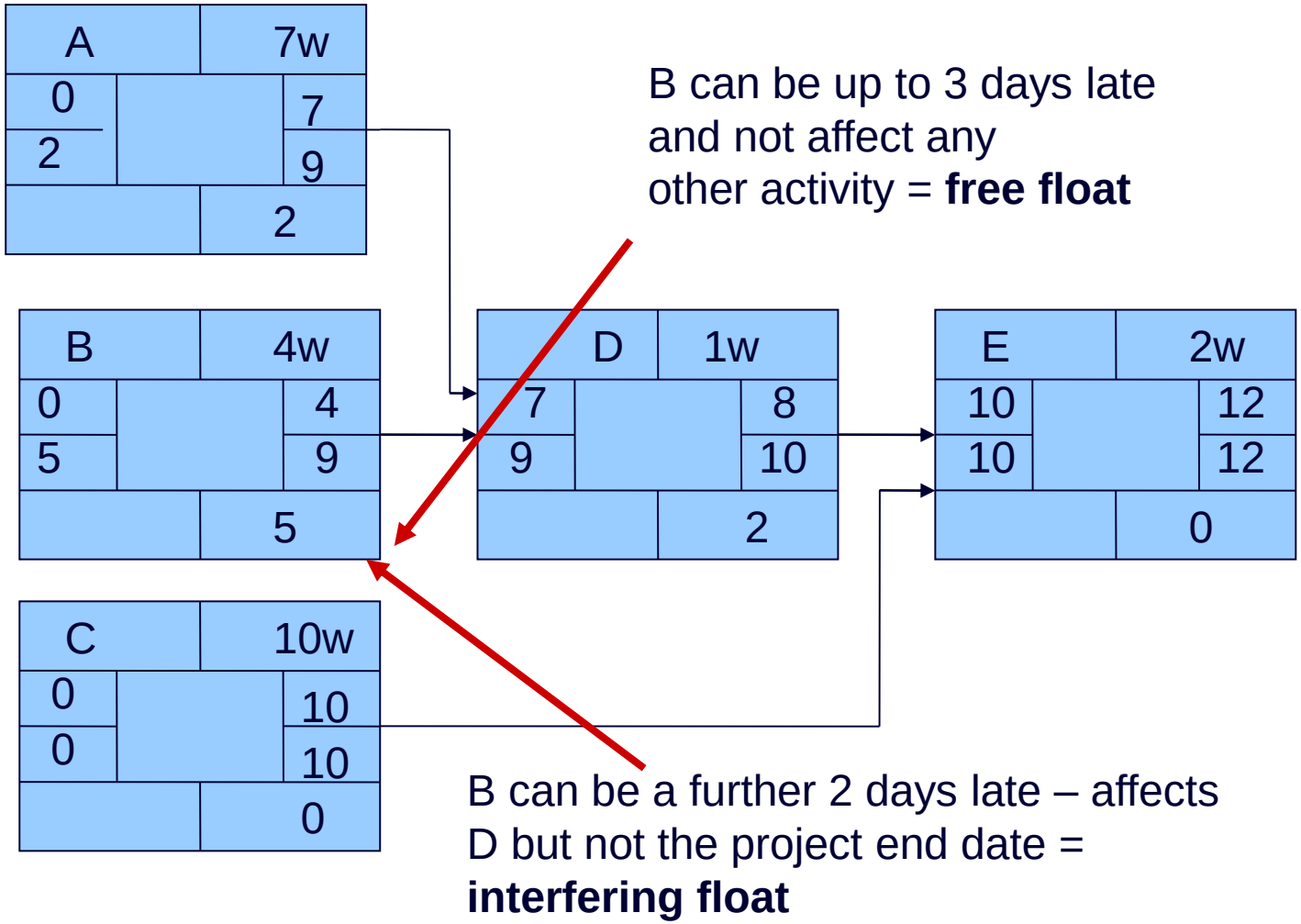
Its a measure of the maximum time allowable for the activity.



Critical path: F-G

Figure 6.21 *A precedence network.*

Free and interfering float



Free float: the time by which an activity may be delayed without affecting any subsequent activity.

It is calculated as the difference between the earliest completion date for the activity and earliest start date of the succeeding activity.

Interfering float: how much the activity may be delayed without delaying the project date-even though it will delay the start of subsequent activities.

It is calculated as the difference between total float and the free float

Critical path

- Note the path through network with zero floats
- Critical path: any delay in an activity on this path will delay whole project
- Can there be more than one critical path?
- Can there be no critical path?
- Sub-critical paths

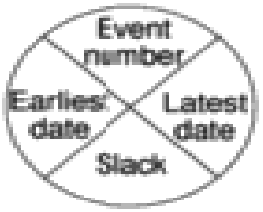
Table F.9 *Activity floats*

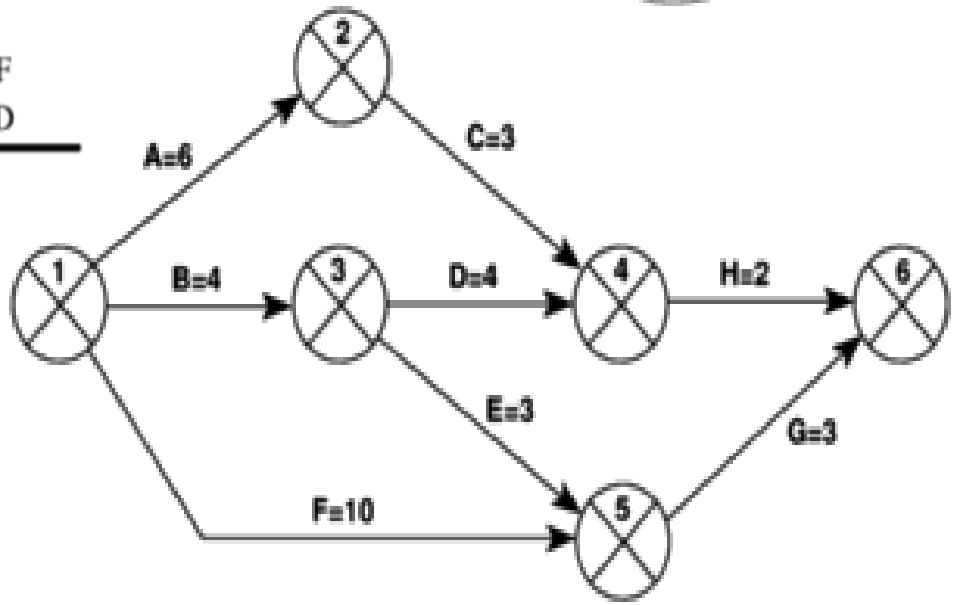
Activity	*Total float*	*Free float*	*Interfering float*
A	2	0	2
B	3	0	3
C	2	0	2
D	3	1	2
E	3	3	0
F	0	0	0
G	0	0	0
H	2	2	0

Activity on arrow network(CPM NETWORK)

Activity		Duration (weeks)	Precedents
A	Hardware selection	6	
B	Software design	4	
C	Install hardware	3	A
D	Code & test software	4	B
E	File take-on	3	B
F	Write user manuals	10	
G	User training	3	E, F
H	Install & test system	2	C, D

Labelling Convention





Forward Pass rule

During the forward pass, earliest dates are recorded as they are calculated. For events, they are recorded on the network diagram and for activities they are recorded on the activity table.

The forward pass rule: the earliest date for an event is the earliest finish date for all the activities terminating at that event. Where more than one activity terminates at a common event we take the latest of the earliest finish dates for those activities.

Table 6.2 *The activity table after the forward pass*

Activity	*Duration (weeks)*	*Earliest start date*	*Latest start date*	*Earliest finish fate*	*Latest finish date*	*Total float*
A	6	0		6		
B	4	0		4		
C	3	6		9		
D	4	4		8		
E	3	4		7		
F	10	0		10		
G	3	10		13		
H	2	9		11		

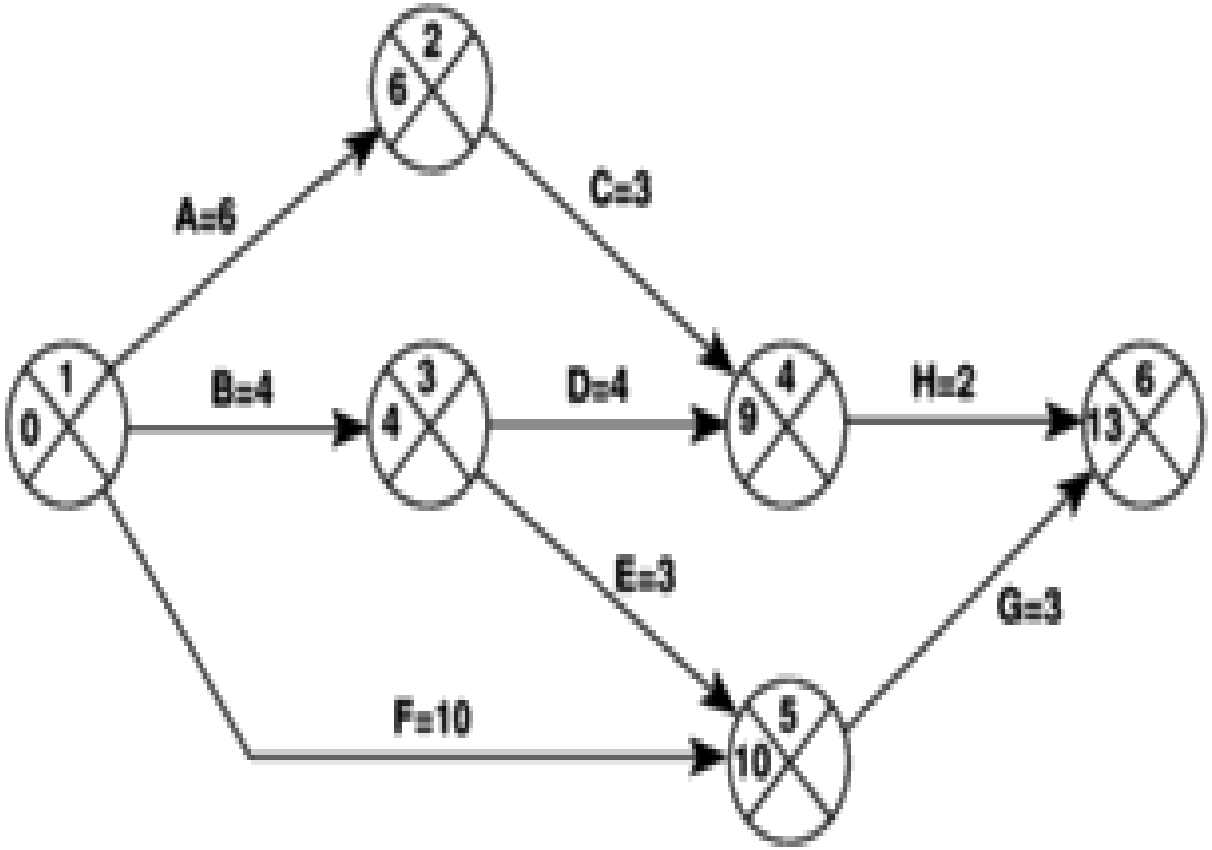


Figure 6.18 *A CPM network after the forward pass.*

Backward Pass Rule

The backward pass rule: the latest date for an event is the latest start date for all the activities that may commence from that event. Where more than one activity commences at a common event we take the earliest of the latest start dates for those activities.

Table 6.3 *The activity table following the backward pass*

Activity	*Duration (weeks)*	*Earliest start date*	*Latest start date*	*Earliest finish fate*	*Latest finish date*	*Total float*
A	6	0	2	6	8	
B	4	0	3	4	7	
C	3	6	8	9	11	
D	4	4	7	8	11	
E	3	4	7	7	10	
F	10	0	0	10	10	
G	3	10	10	13	13	
H	2	9	11	11	13	

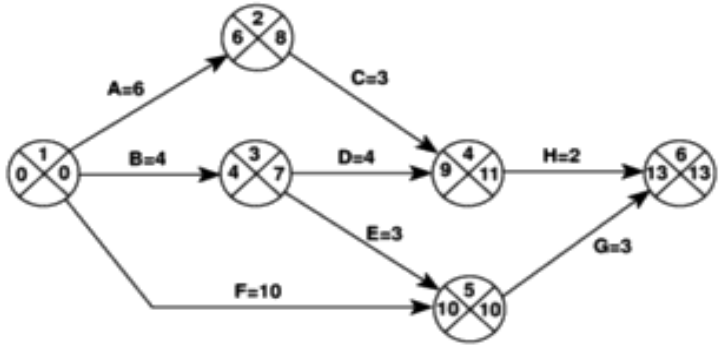


Figure 6.19 *The CPM network after the backward pass.*

S

Critical Path

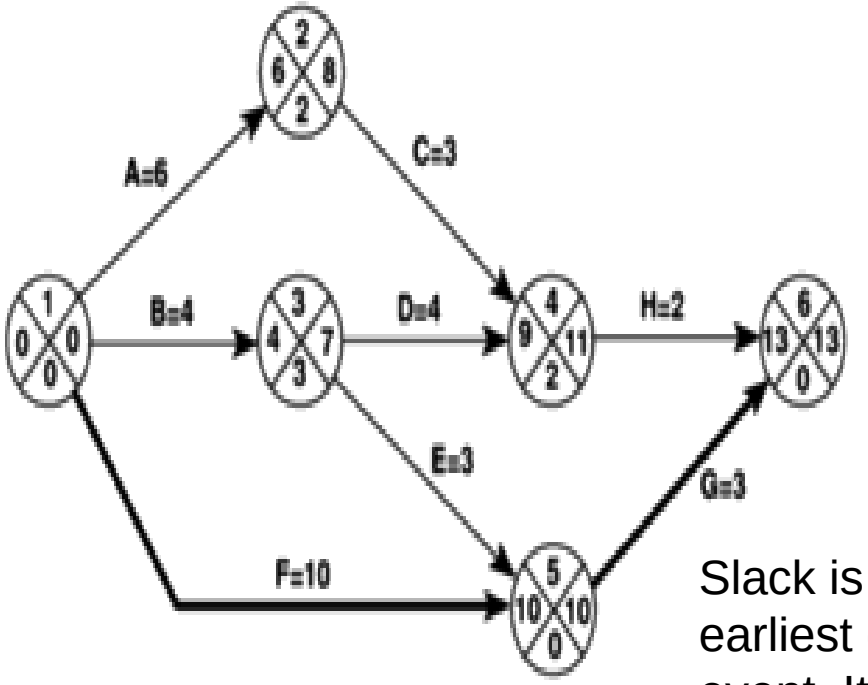


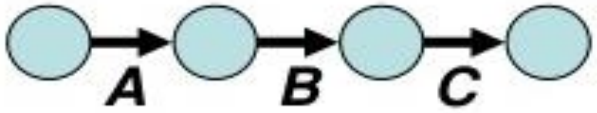
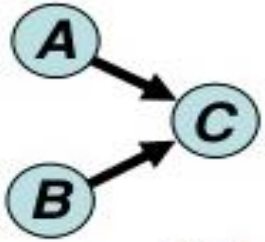
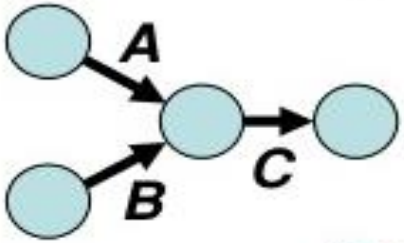
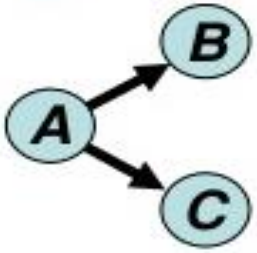
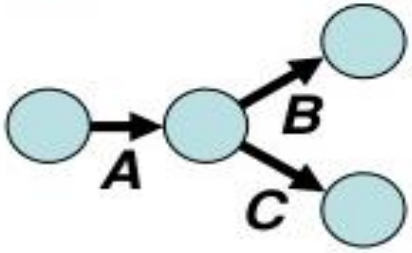
Figure 6.20 *The critical path.*

The critical path is the longest path through the network.

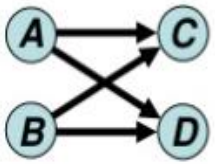
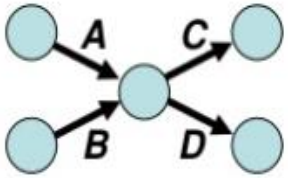
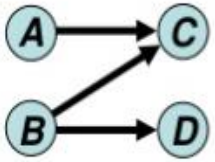
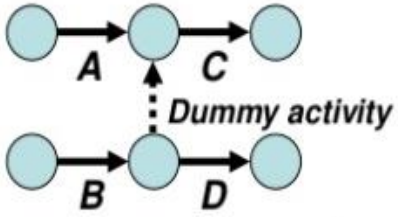
Slack is the difference between earliest date and latest date of an event. Its a measure of how late an event may be without delaying end date of the project

Critical path: F-G

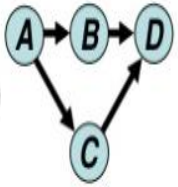
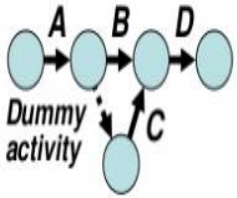
A Comparison of AON and AOA Network Conventions

Activity on Node (AON)	Activity Meaning	Activity on Arrow (AOA)
(i) 	A comes before B, which comes before C	
(ii) 	A and B must both be completed before C can start	
(iii) 	B and C cannot begin until A is completed	

A Comparison of AON and AOA Network Conventions

Activity on Node (AON)	Activity Meaning	Activity on Arrow (AOA)
(iv) 	C and D cannot begin until both A and B are completed	
(v) 	C cannot begin until both A and B are completed; D cannot begin until B is completed. A dummy activity is introduced in AOA	

A Comparison of AON and AOA Network Conventions

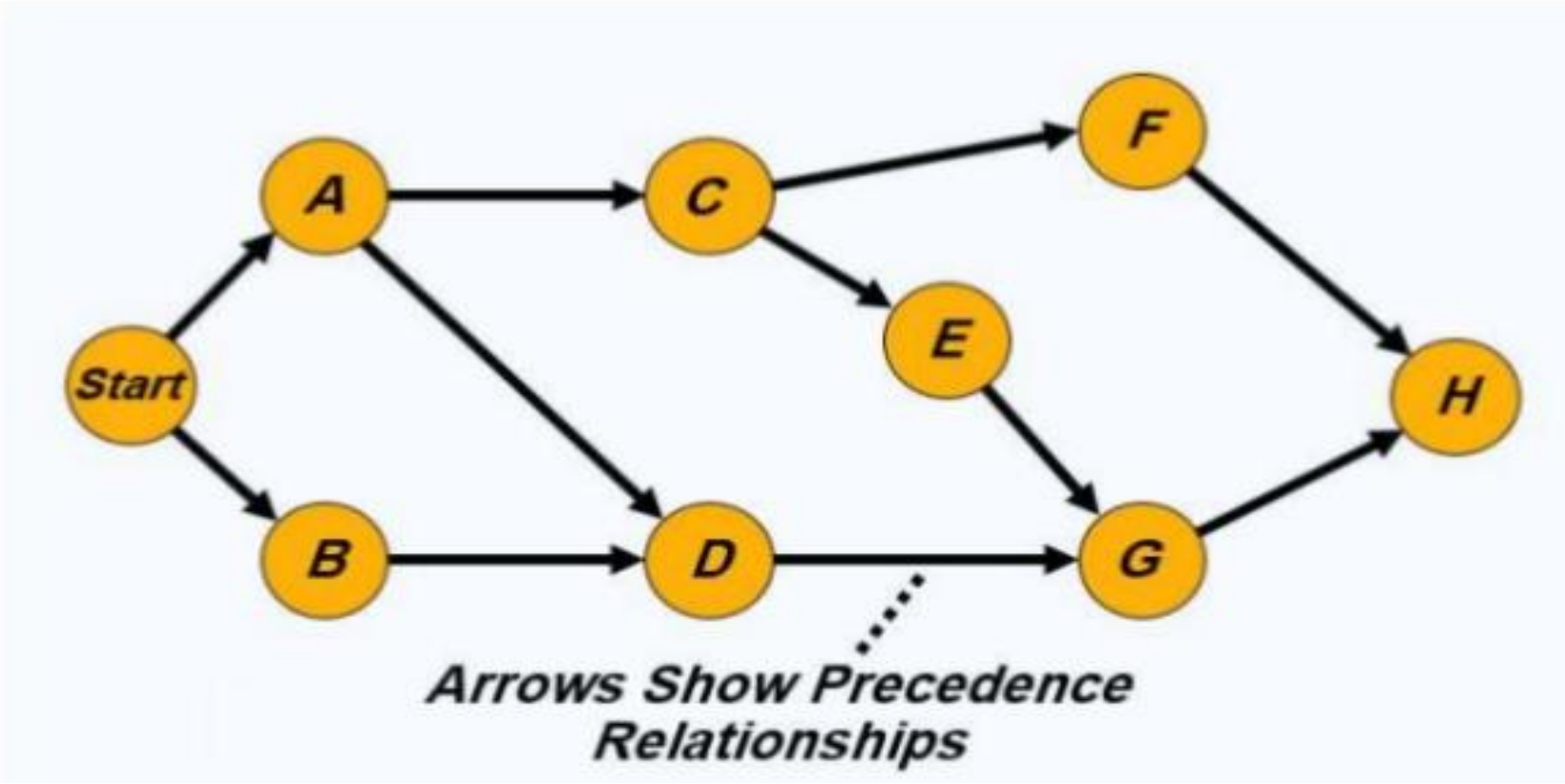
Activity on Node (AON)	Activity Meaning	Activity on Arrow (AOA)
(vi) 	B and C cannot begin until A is completed. D cannot begin until both B and C are completed. A dummy activity is again introduced in AOA.	

Example:

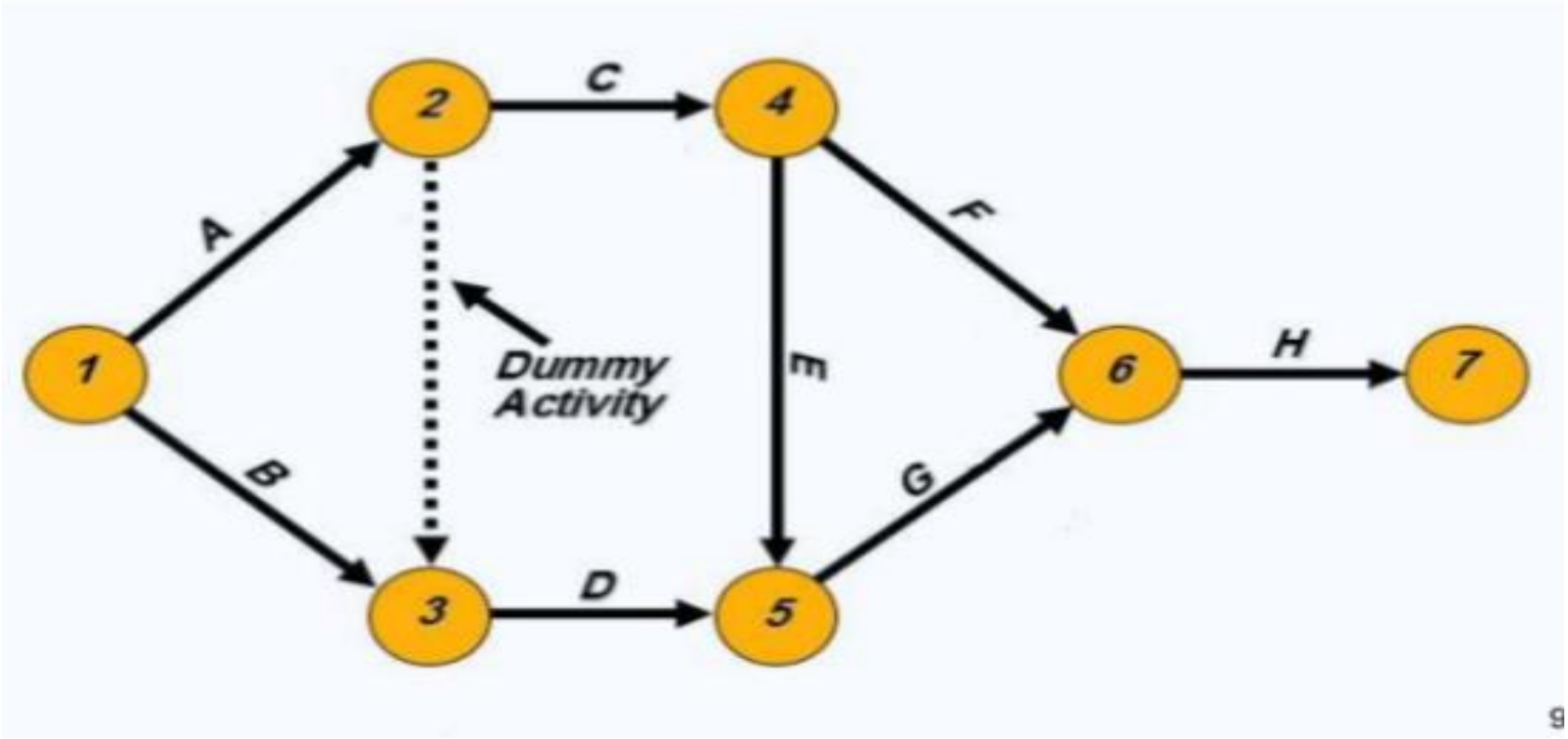
Activity	Immediate Predecessors
A	—
B	—
C	A
D	A, B
E	C
F	C
G	D, E
H	F, G

ACTIVITY	DESCRIPTION	IMMEDIATE PREDECESSORS	TIME (WEEKS)
A	Build internal components	—	2
B	Modify roof and floor	—	3
C	Construct collection stack	A	2
D	Pour concrete and install frame	A, B	4
E	Build high-temperature burner	C	4
F	Install pollution control system	C	3
G	Install air pollution device	D, E	5
H	Inspect and test	F, G	2

AON Network



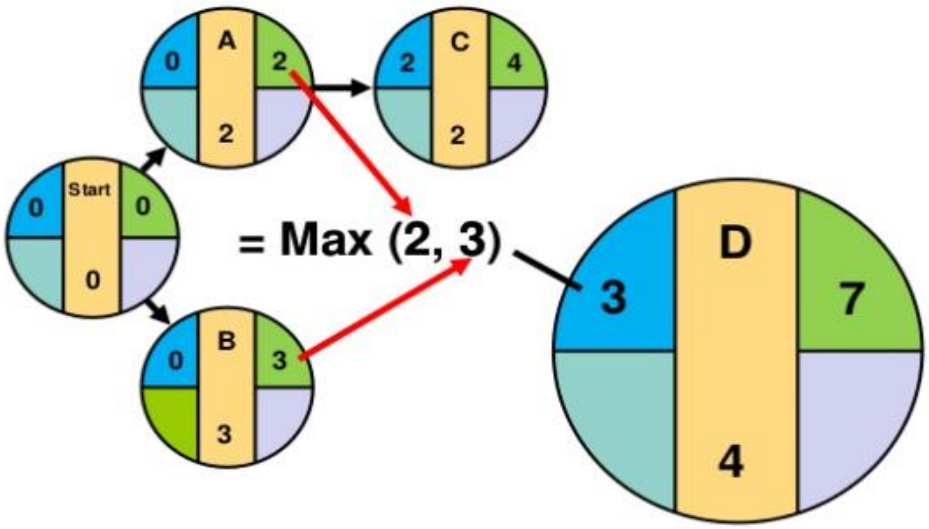
AOA Network



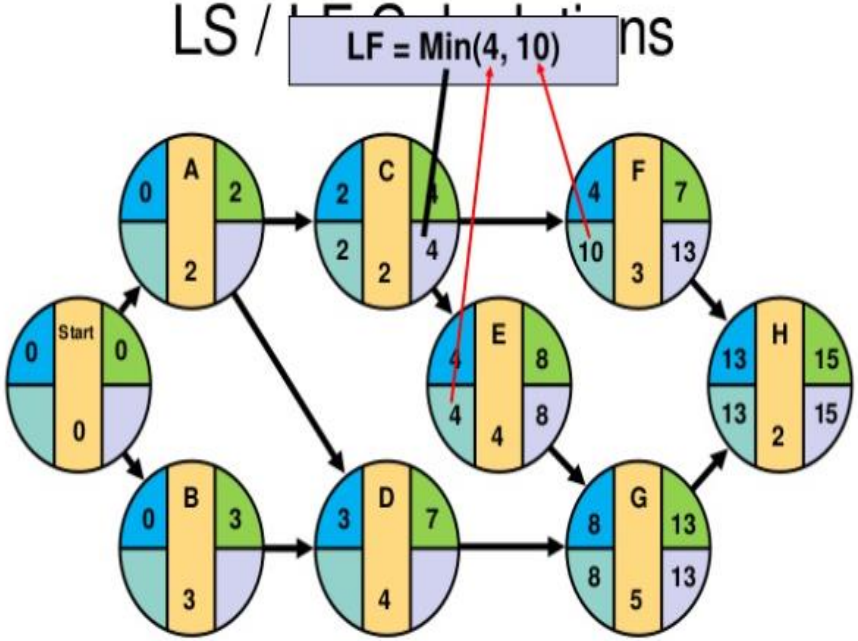
Computing Slack Time

Activity	Earliest Start ES	Earliest Finish EF	Latest Start LS	Latest Finish LF	Slack = (LS – ES) = (LF – EF)	On Critical Path
A	0	2	0	2	0	Yes
B	0	3	1	4	1	No
C	2	4	2	4	0	Yes
D	3	7	4	8	1	No
E	4	8	4	8	0	Yes
F	4	7	10	13	6	No
G	8	13	8	13	0	Yes
H	13	15	13	15	0	Yes

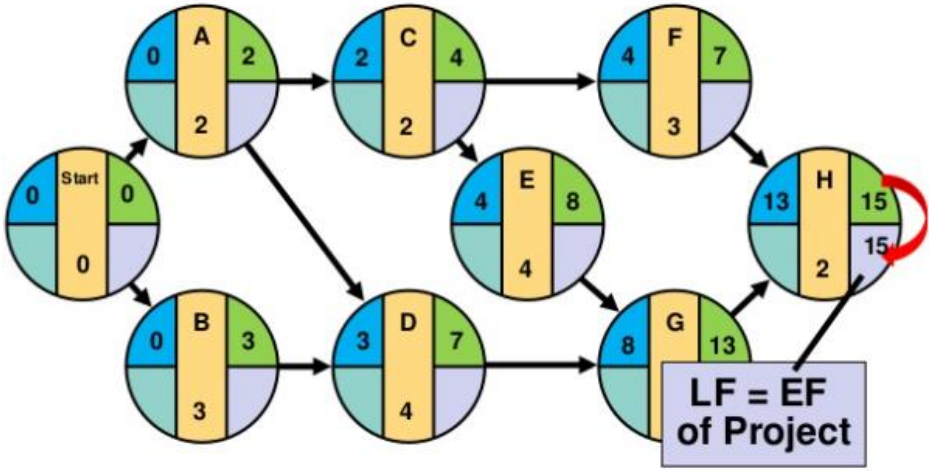
ES / EF Calculations



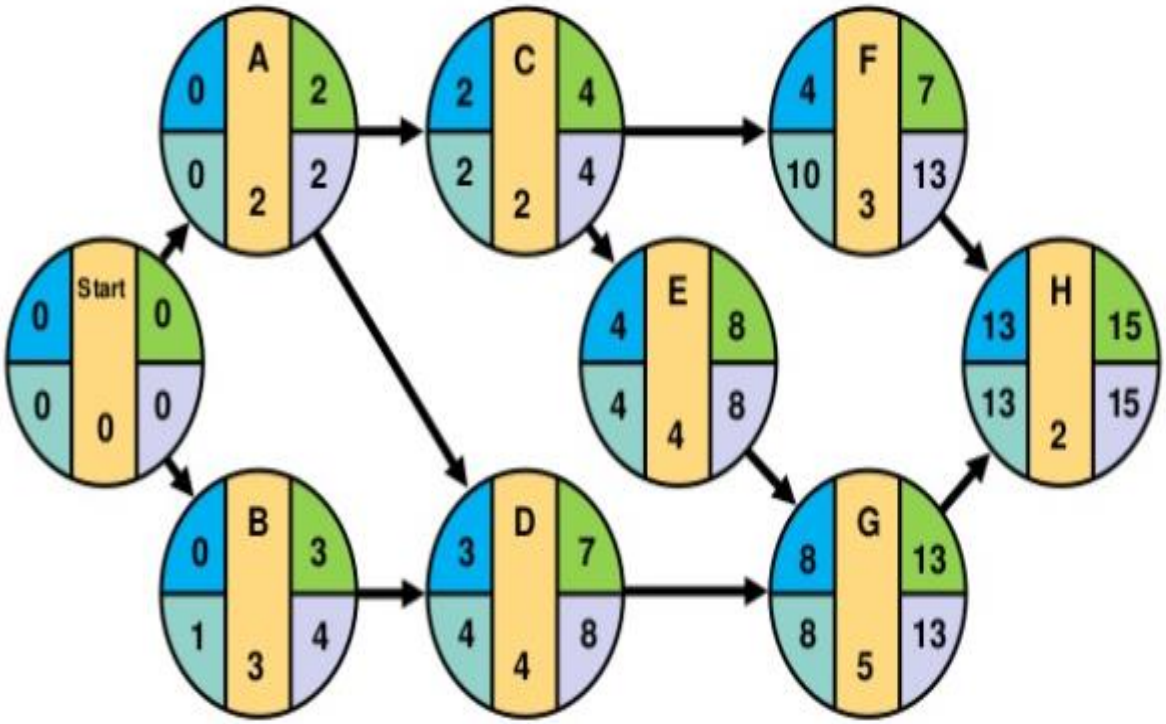
LS / LF Calculations

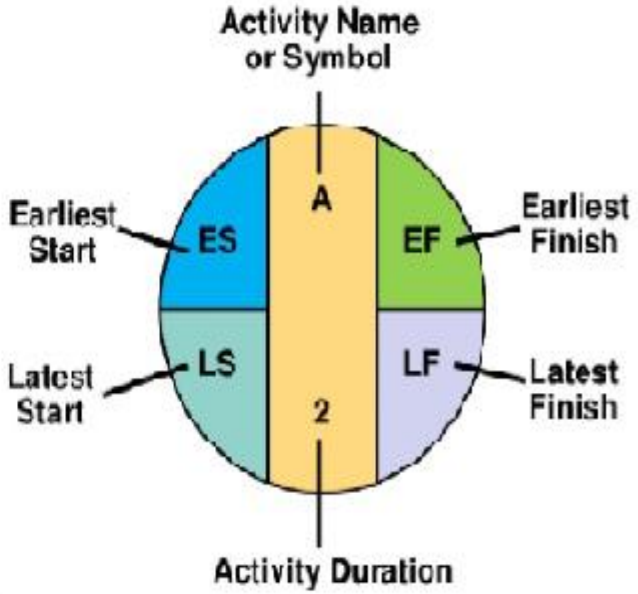


LS / LF Calculations

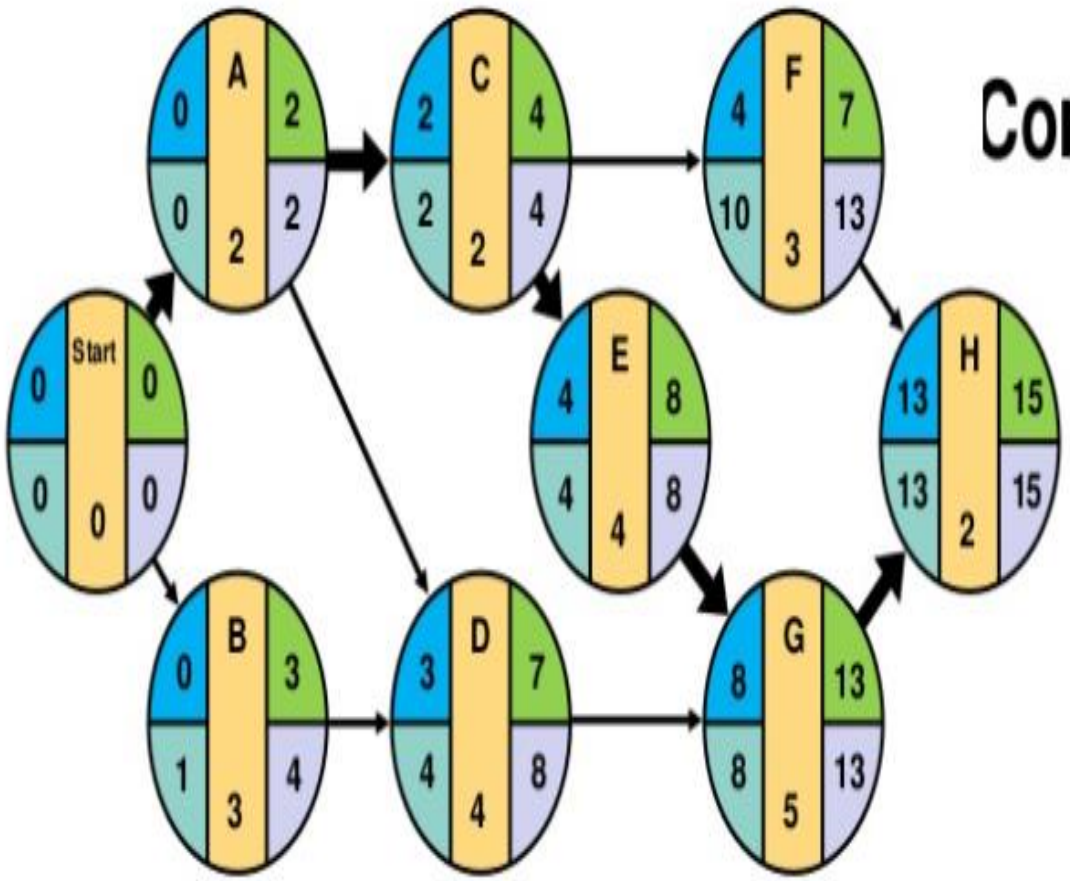


LS / LF Calculations





Critical Path



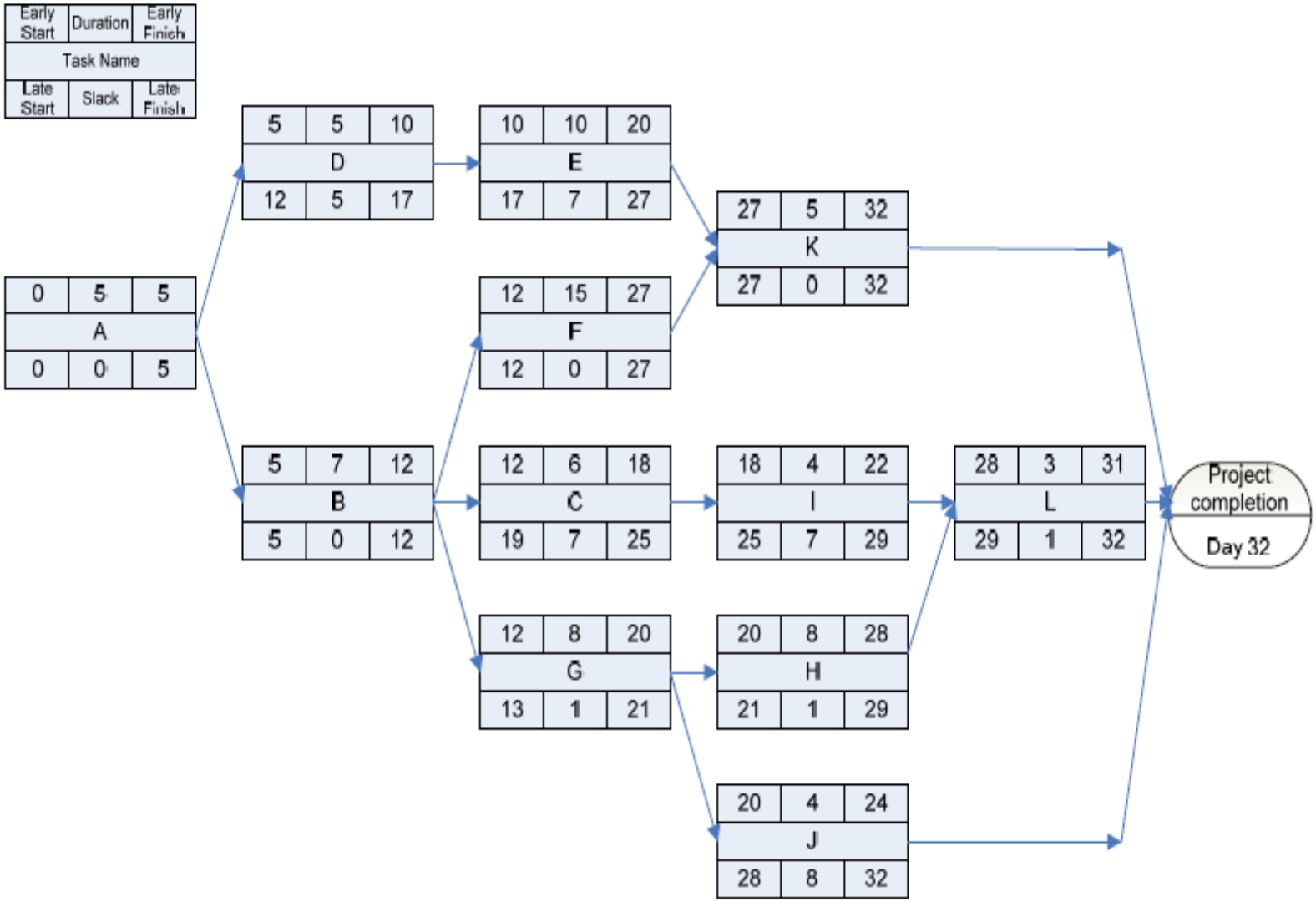
Critical Path

A – C – E – G – H

Completion Time = 15 weeks

Create a precedence activity network using the following details:

Activity	Depends on	Duration (days)
A		5
B	A	7
C	B	6
D	A	5
E	D	10
F	B	15
G	B	8
H	G	8
I	C	4
J	G	4
K	E,F	5
L	I,H	3



The critical path is A-B-F-K where all the floats are 0.