Project Planning and Control Methods

Lecture #7-P1

# Schedule Constraints (Time)


Amin Alvanchi, PhD

Construction Engineering and Management


LinkedIn Instagram WebPage


Department of Civil Engineering, Sharif University of Technology

# Outline



- Introduction
- Time constraints
- Time crashing
- Fast tracking
- Line of balance (LOB)

# Introduction



- From our “initial schedule development” to this stage, the only constraints we have looked into are dependencies among different activities.
- In reality before project schedules become practical they need to address different project constraints and concerns including the ones related to the project resources, time and uncertainty!
- In next several sessions we will discuss different project constraints (that need to be addressed in the schedule) and their prevalent solutions!

# Time constraint



- In our initial schedule development and CPM calculations, the "schedule" we developed might violate time constraints set for the project. Time constraints such as project start time limitations, finish time deadline and other project milestones (events that receive special attention in the project such as delivering main deliverables).
- We usually need to compress our schedule from the initial schedule developed.
- In addition, when a project goes behind the schedule during project execution, to accelerate the project we need to compress the schedule of the rest of activities.
- Two main project schedule compression methods are used when initial project exceeds time constraints set for the project:
  - Time crashing
  - Fast tracking

# Time crashing



- Time crashing is a method to compress the project schedule by reducing activity duration by using additional resources.

**Question:** Which activities are you going to pick when you want to do the time crashing?

# Time crashing



- Time crashing is a method to compress the project schedule by reducing activity duration by using additional resources.

**Question:** Which activities are you going to pick when you want to do the time crashing?

- ***Activity Crashing Guidelines:***
  - Activities on the critical chain are the candidates for time crashing!
  - Among critical activities choose activities with minimum cost increase per day (or time unit) of required activity duration reduction.

  **Question**: Which activity you are going to choose first for crashing?
    - Activity A (normal duration: 8 days, normal cost: 10 MToman, crashed duration: 6 days, crashed cost: 13 MToman)
    - Activity B (normal duration: 12 days, normal cost: 1 MToman, crashed duration: 11 days, crashed cost: 2 MToman)

# Time crashing



- Cost Increase per day of reduction for activity A = (13-10)/2 = 1.5 MToman
- Cost Increase per day of reduction for activity B = (2-1)/1 = **1 Mtoman**

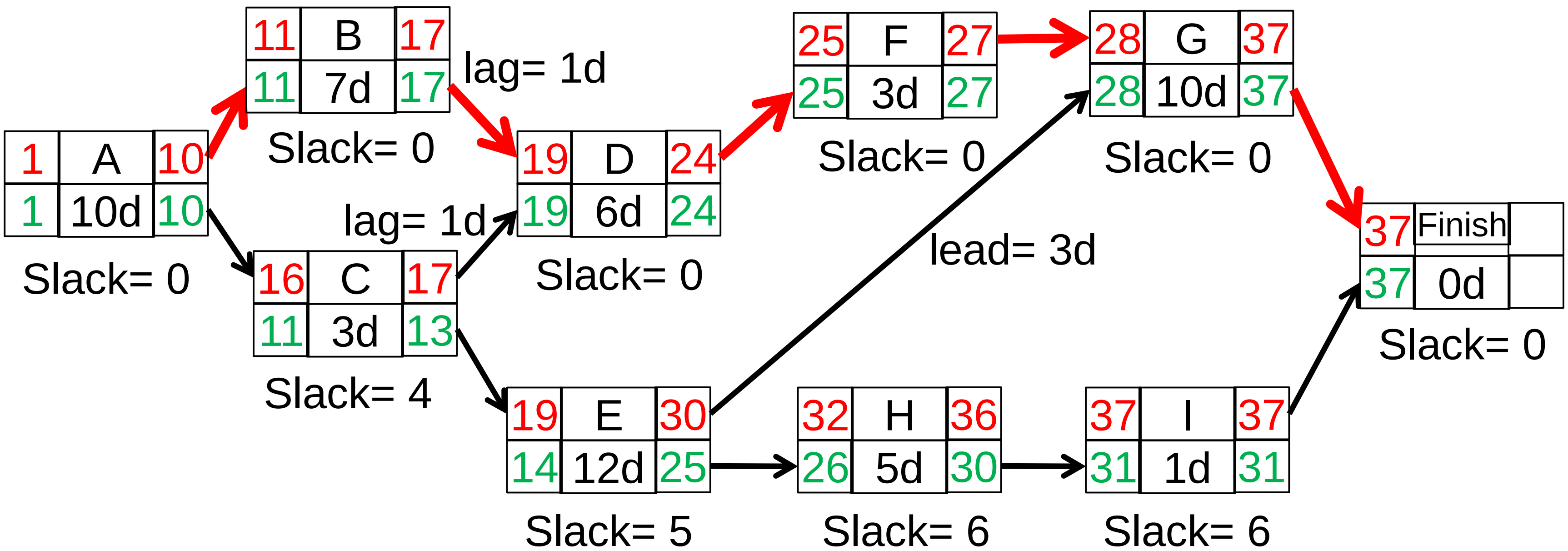
- If there are several critical chains in the project, we need to reduce the duration of all critical chains when compressing the schedule!

By crashing activities on the critical chain, gradually the duration of the critical chains gets decreased and new chains are added to the critical chains!

# Time crashing



- **Example 1**: A contractor has signed a contract to complete a project within 32 days. However, CPM calculation of the initial schedule results in 37 days of project duration as presented in below:



# Time crashing



- **Example 1 (cont'd)**

1. Before any crashing decision you need to calculate duration of different activity chains

| Chain | Duration |
|---|---|
| ABDFG | 37 |
| ACDFG | 33 |
| ACEG | 32 |
| ACEHI | 31 |

Duration of different chains shows that for reducing first 4 days of the schedule duration we just need to reduce activities on the chain ABDFG (crashing round 1), for reducing the 5$^{th}$ day (to reach our goal of 32 days) we need to reduce both ABDFG and ACDFG chains (crashing round 2).

# Time crashing



- **Example 1 (cont'd)**

2. Estimate the normal duration cost and the crash duration cost of the candidate activities (i.e., activities on ABDFG and ACDFG chains).

| Activity | Normal duration | Normal Cost | Crash 1 duration | Crash 1 cost | Crash 1 rate | Crash 2 duration | Crash 2 cost | Crash 2 rate |
|---|---|---|---|---|---|---|---|---|
| A | 10 | 100 | 8 | 130 | 15 | 7 | 160 | 30 |
| B | 7 | 140 | 6 | 150 | 10 | 5 | 160 | 10 |
| D | 6 | 90 | 4 | 110 | 10 | | | |
| F | 3 | 40 | *Some activities might not be crashed! (e.g., shipping steel)* | | | | | |
| G | 10 | 50 | 7 | 100 | 17 | 6 | 130 | 30 |
| C | 3 | 25 | 2 | 35 | 10 | | | |

# Time crashing



- **Example 1 (cont'd)**

3. Start crashing with the activities which force minimum cost, and continue until you reach reduction required!

***Round 1:*** 4 days need to be reduced from chain ABDFG.

| Activity | Normal duration | Normal Cost | Crash 1 duration | Crash 1 cost | Crash 1 rate | Crash 2 duration | Crash 2 cost | Crash 2 rate |
|---|---|---|---|---|---|---|---|---|
| A | 10 | 100 | 8 | 130 | 15 | 7 | 160 | 30 |
| B | 7 | 140 | ***6*** | ***150*** | ***10*** | ***5*** | ***160*** | ***10*** |
| D | 6 | 90 | ***4*** | ***110*** | ***10*** | | | |
| F | 3 | 40 | | | | | | |
| G | 10 | 50 | 7 | 100 | 17 | 6 | 130 | 30 |
| C | 3 | 25 | 2 | 35 | 10 | | | |

Activities B with 2 days reduction and total additional cost of 20 and D with 2 days reduction with total additional cost of 20 are crashed!

# Time crashing



- **Example 1 (cont'd)**

***Round 2:*** 1 day needs to be reduced from chains ABDFG and ACDFG.

| Activity | Normal duration | Normal Cost | Crash 1 duration | Crash 1 cost | Crash 1 rate | Crash 2 duration | Crash 2 cost | Crash 2 rate |
|---|---|---|---|---|---|---|---|---|
| A | 10 | 100 | 8 | 130 | 15 | 7 | 160 | 30 |
| B | 7 | 140 | ***6*** | ***150*** | ***10*** | ***5*** | ***160*** | ***10*** |
| D | 6 | 90 | ***4*** | ***110*** | ***10*** | | | |
| F | 3 | 40 | | | | | | |
| G | 10 | 50 | 7 | 100 | 17 | 6 | 130 | 30 |
| C | 3 | 25 | 2 | 35 | 10 | | | |

Activity A with 2 days reduction and total additional cost of 30 is crashed!

**Question:** Why we did not select Activity **C** in 2$^{nd}$ cycle of crashing regardless of its minimum cost?

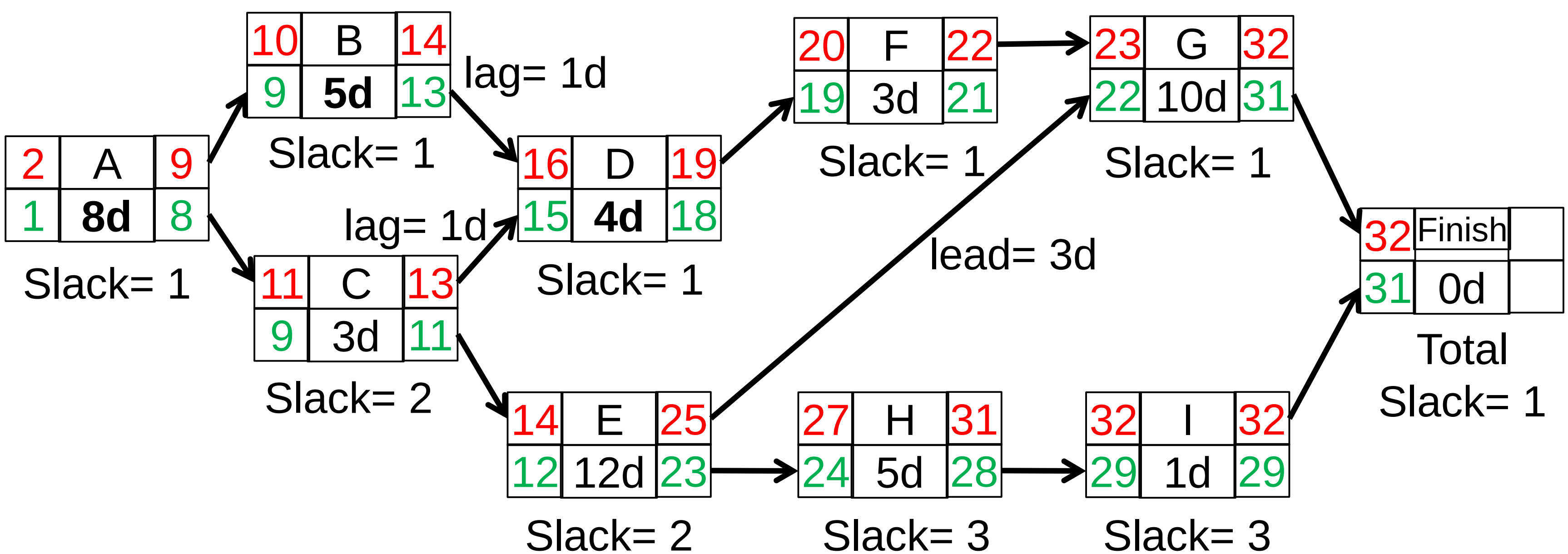
# Time crashing



- In the first cycle, we reduced duration of activity D by 2 days. So duration of chain ACDFG is already reduced from 33 to 31.
- **Total Slack:** In similar situations when we need to finish the project within an specific time but base on our schedule we are going to finish it earlier, we have a total slack for the project. E.g., in our example we have:

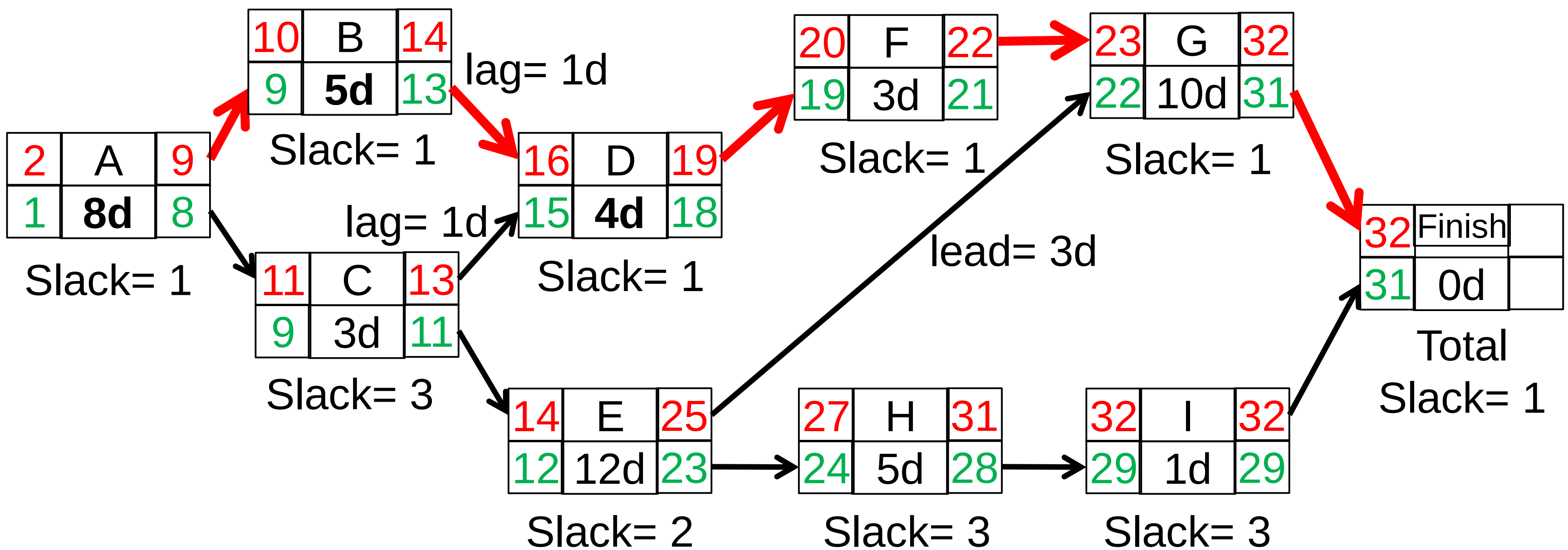
  ***32-31=1 total slack time***

  For calculating *critical path*, we need to deduct this slack time from calculated slack time of each activity! E.g. in our example we have:



# Time crashing






- After using mentioned method for crashing do the final check if the duration goes below what is expected (e.g., in our case final duration is 31 days which is below the expected time of 32 days)!

# Time crashing



- We might be able to fill the *total slack* time by returning previous crashed activities and save the cost! E.g., in our case we start looking into the last crashed activities of B and D and check if returning their crash time will keep us still within expected time. By reversing the second crash of activity B we are going to increase duration of activity B from 5 days to 6 days, eliminate the total slack and save $10!

| Activity | Normal duration | Normal Cost | Crash 1 duration | Crash 1 cost | Crash 1 rate | Crash 2 duration | Crash 2 cost | Crash 2 rate |
|---|---|---|---|---|---|---|---|---|
| A | 10 | 100 | **8** | **130** | **15** | 7 | 160 | 30 |
| B | 7 | 140 | ***6*** | ***150*** | ***10*** | *5* | *160* | *10* |
| D | 6 | 90 | ***4*** | ***110*** | ***10*** | | | |
| F | 3 | 40 | | | | | | |
| G | 10 | 50 | 7 | 100 | 17 | 6 | 130 | 30 |
| C | 5 | 25 | 4 | 35 | 10 | | | |

# Time crashing



- Use of ***overtime*** for reducing the total duration of the project is a special type of applying schedule crashing to the project activities! Overtime is widely used in projects. In overtime we are going to use extra time of current project employees instead of hiring new workers! However, it should be considered that continual use of Overtime in the project will usually have an inverse effect (i.e., will increases the duration of the activities) in long term as a result of fatigue created in the workers!
- Time crashing is done by additional resources. Adding additional resources to activities is done in a discrete manner (e.g., from 2 workers to 3 workers). So, there is no guaranty that we can crash an activity day by day (e.g., duration can be reduces from 9 days to 6 days)

# In class exercise-1



- For the following project, client has asked a duration of 24 days to complete the project. What will be the lowest cost if we do time crashing to reduce the project duration.

| Task | Predecessor | Normal Duration | Normal Cost | Crash 1 duration | Crash 1 cost k$ | Crash 2 duration | Crash 2 cost k$ |
|---|---|---|---|---|---|---|---|
| A | | 10 | 10 | 8 | 13 | 6 | 20 |
| B | A | 5 | 8 | 4 | 10 | | |
| C | B | 8 | 9 | 6 | 13 | | |
| D | C | 5 | 4 | 4 | 8 | | |

# In class exercise-1



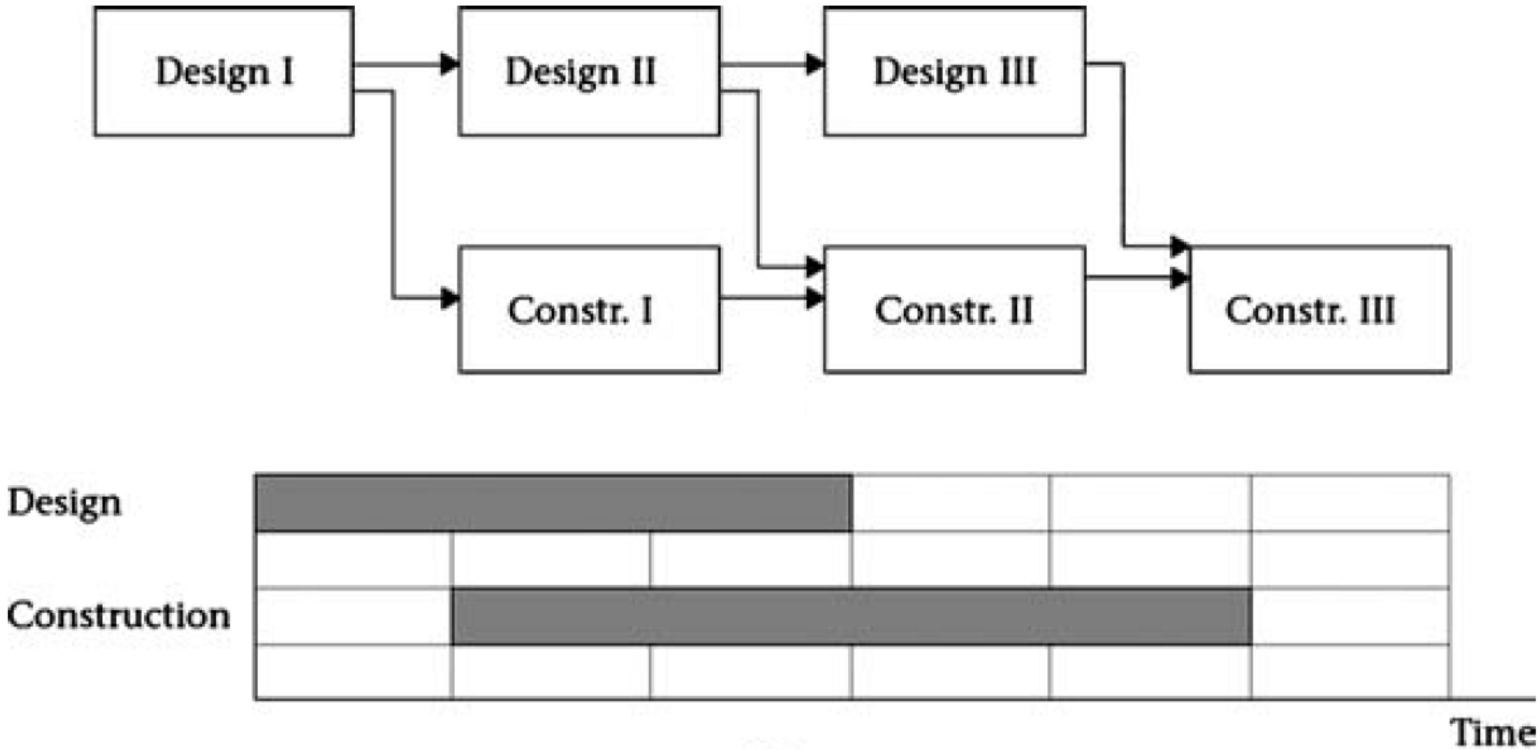
For the following project, client has asked a duration of 24 days to complete the project. What will be the lowest cost if we do time crashing to reduce the project duration.

| Task | Predecessor | Normal Duration | Normal Cost | Crash 1 Reduction | Rate1 k$ | Crash 2 Reduction | Rate2 k$ |
|---|---|---|---|---|---|---|---|
| A | | 10 | 10 | 2 | 1.5 | 2 | 3.5 |
| B | A | 5 | 8 | 1 | 2 | | |
| C | B | 8 | 9 | 2 | 2 | | |
| D | C | 5 | 4 | 1 | 4 | | |
| | **Total** | **28** | **31** | **24** | **38** | | |

# Fast tacking



- The term fast-track in the construction world means starting the construction process while the design is still under development.

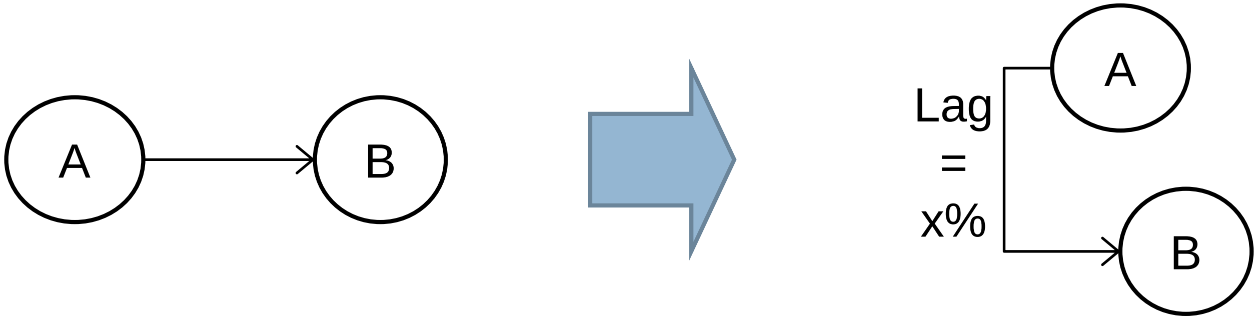



**Construction starts after 33% progress in design**

# Fast tacking



- Generalizing fast tracking concept to other phases and activities of projects; this means parallelizing activities by changing activity dependencies from FS to SS or FF usually by considering a percentage for the lag time when it is possible!




**Question**: By adopting fast-tracking technique we are compressing the schedule and expecting to accelerate the project by accepting a risk of rework (= extra costs). Why?

# Fast tacking



- By applying fast tracking technique, we are loosening FS dependencies between two activities (e.g., activity A and B) and change it to SS or FF dependencies .
- This means we are going to suppose works done for in-progress activity A are finalized and we will base our efforts in activity B up on that.
- Mistakes caught or deviations requested during progress of activity A may change basis of already completed portions for activity B and results in rework in both Activities A and B!
- ***Fast Tracking Guidelines:***
  - Activities on the critical chain are the candidates for fast tracking!
  - Among critical activities choose activities with minimum risk cost per day (or time unit) of required activity duration reduction.
    - **Example 2**: We are going to fast track our project for 5 days. Which two activity you are going to choose first for fast tracking?

# Fast tacking



- **Example 2 (cont'd):**
- Activity A has an FS dependency with activity B. Expectedly every day of fast tracking creates 2% of reworks in B. Activity B has a rework cost of 100 TToman per percent of rework.
- Activity C has an FS dependency with activity D. Expectedly every day of fast tracking creates 1% of reworks in D. Activity D has a rework cost of 150 TToman per percent of rework.

**Solution**:

- **Note:** Costs spent for reworks in activities A and C (predecessor activities) are not included in our fast tracking costs. Why?
- Risk cost for A-B fast tracking is: 2% * 5 * 100 TToman per percent = 1000 TToman
- Risk cost for C-D fast tracking is: 1% * 5 * 150 TToman per percent = **750 TToman**

# Fast tacking



- If there are several critical chains in the project, we need to reduce the duration of all critical chains when compressing the schedule!
- Similar approach to *Crashing* for compressing schedules will be followed in *Fast tracking.* Just calculation of extra cost is different!!!!!!
- In Actual projects, for compressing schedules usually a combined method of crashing and fast tracking is applied. This means we are going to reduce the duration of some activities by crashing and parallelize some other activities by loosening their dependencies from FS to SS or FF!

# In class exercise-2



- For the following project, client has asked a duration of 24 days to complete the project. What will be the lowest cost if we do fast tracking to reduce the project duration.

| Task | Predecessor | Normal Duration (d) | Normal Cost (k$) | Rework (% / d) | Rework cost (k$/%) |
|---|---|---|---|---|---|
| A | | 10 | 10 | 1% | 2 |
| B | A | 5 | 8 | 2% | 2 |
| C | B | 8 | 9 | 5% | 1 |
| D | C | 5 | 4 | 1% | 1 |

# In class exercise-2



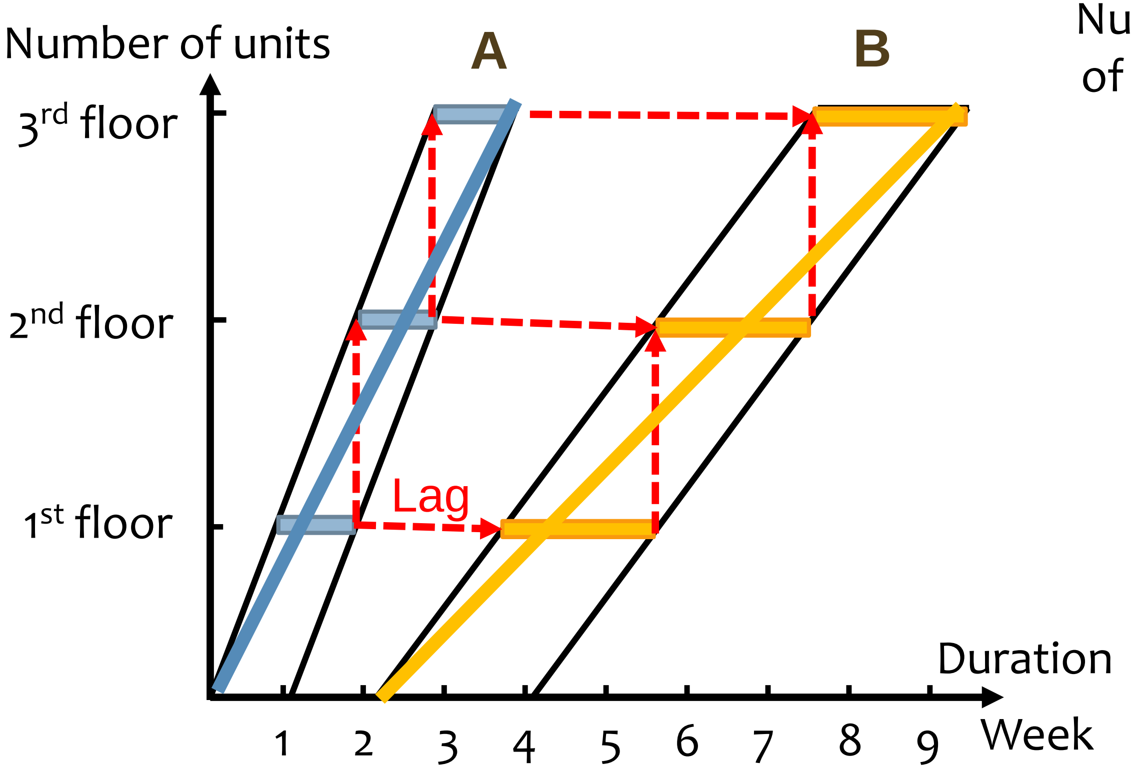
- For the following project, client has asked a duration of 24 days to complete the project. What will be the lowest cost if we do fast tracking to reduce the project duration.

| Task | Predecessor | Normal Duration (d) | Normal Cost (k$) | Rework (% / d) | Rework cost k$/% | Fast tracking rate (d/ k$) | Day to fast track (d) |
|---|---|---|---|---|---|---|---|
| A | | 10 | 10 | 1% | 2 | 2 | |
| B | A | 5 | 8 | 2% | 2 | 4 | |
| C | B | 8 | 9 | 5% | 1 | 5 | |
| D | C | 5 | 4 | 1% | 1 | 1 | 4 |
| | **Total** | **28** | **31** | | | **= 31 + 1 * 4 =** | **35 k$** |

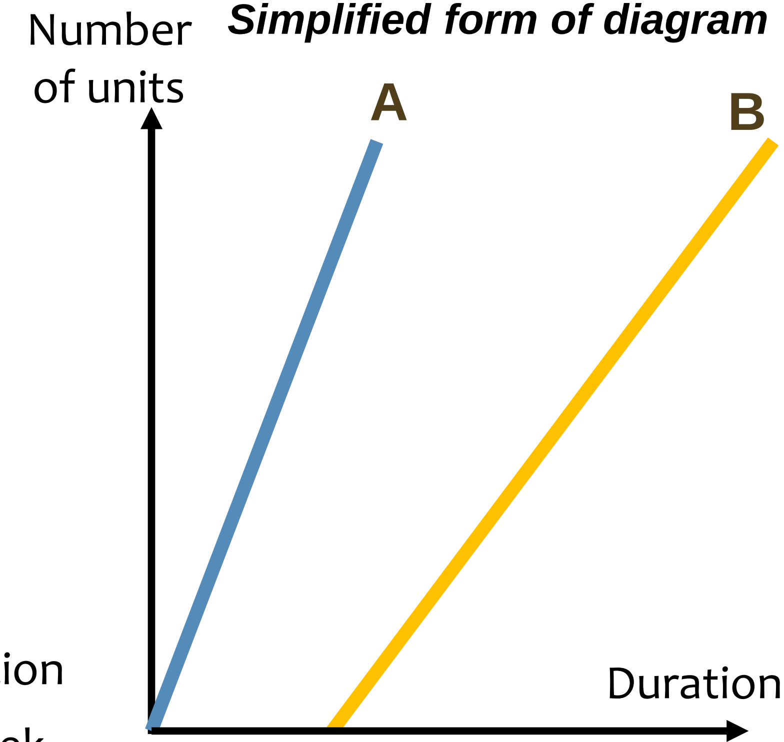
# Line of balance (LOB)



- LOB is a simple diagram to show location and time (duration) at which a certain crew(s) will be working on a given work package.




*Simplified form of diagram*



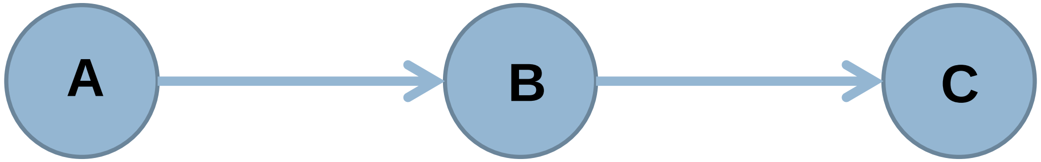

- LOB is categorized under group of methods called location base management (LBM) or location base scheduling (LBS) tools.

# Line of balance (LOB)



- LOB diagram is a useful tool for scheduling set of **work package** (or operation) with **repetitive serial activities** in **different locations** (e.g., Piling, bedrooms finishing, road construction operations, etc..)
- It is a ***Good Visual tool*** that lets us to see if a construction program can be achieved with the minimum waiting time between tasks and minimum waiting time for different working crews.
- It focuses on ***balancing*** the time of repetitive serial activities within work packages by either re-distribution of resources or reducing process waste.

**Example 1:** Suppose a network of 3 serial dependent work groups:

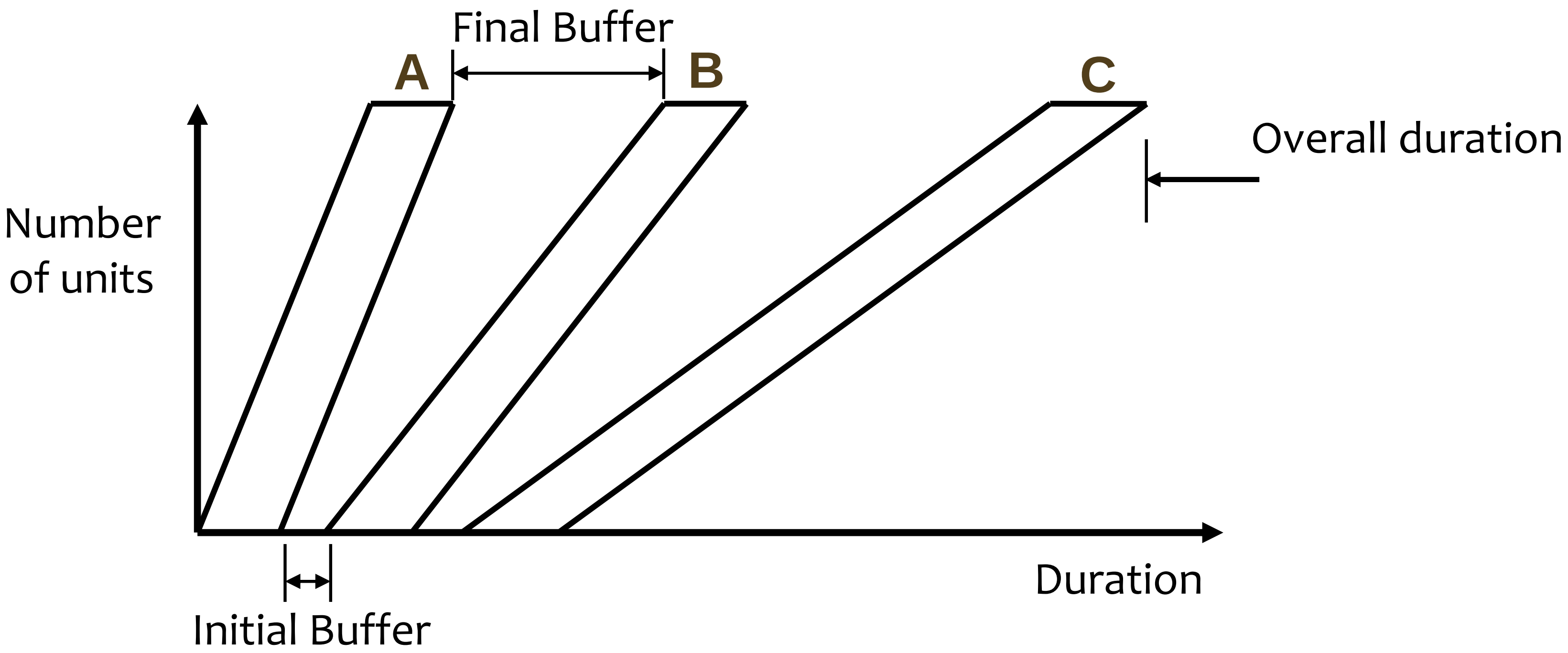



It is going to be presented in an LOB diagram as:

# Line of balance (LOB)



**Example 1 (cont'd):**

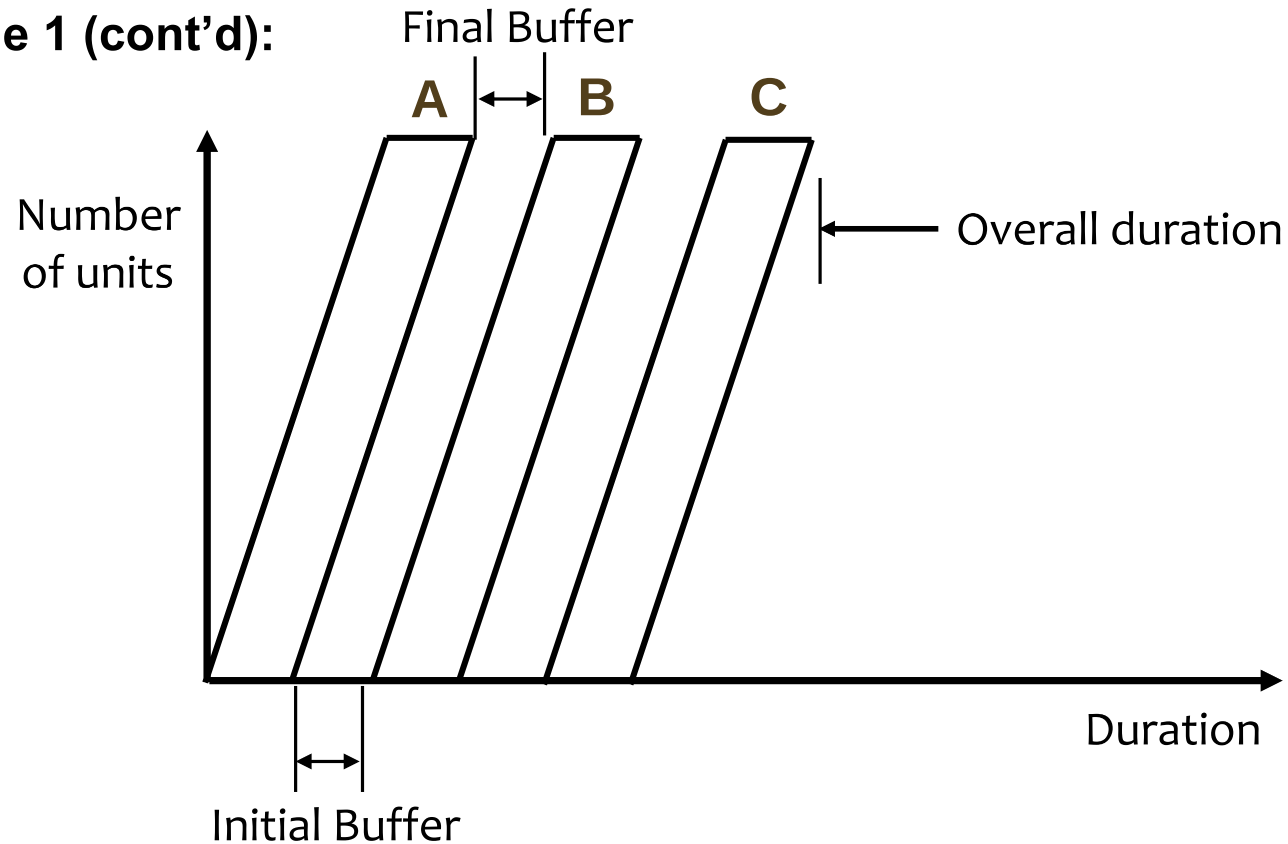



Work package B slower than A and work package C slower than B, we need to speed up activities done in work package B and C to reduce overall duration!

# Line of balance (LOB)



**Example 1 (cont'd):**

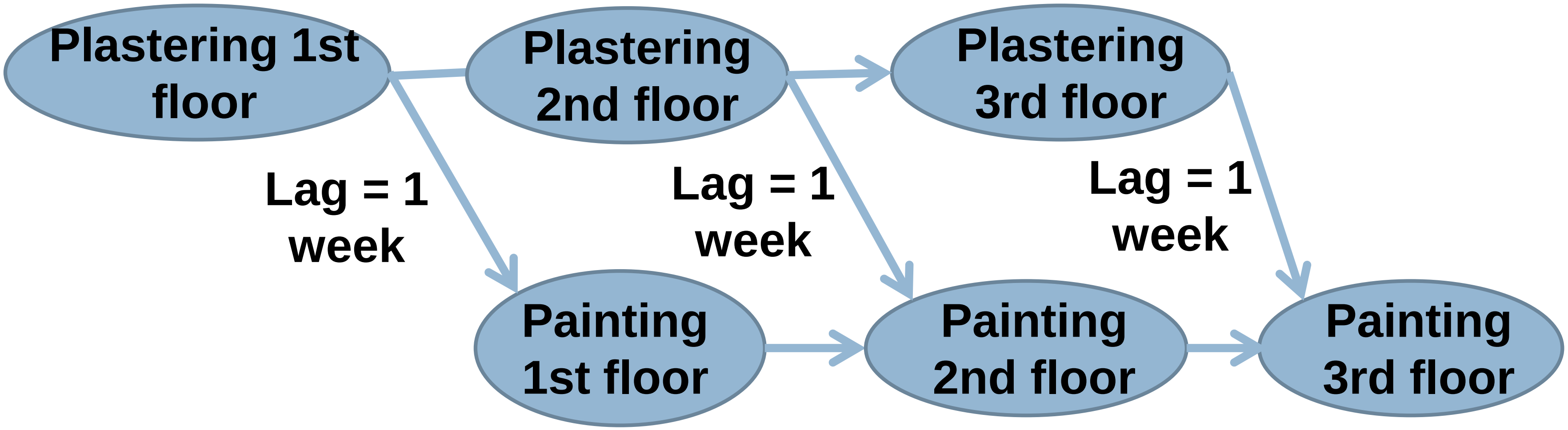



With the same rate of construction we are going to reach a balanced schedule!

# Line of balance (LOB)



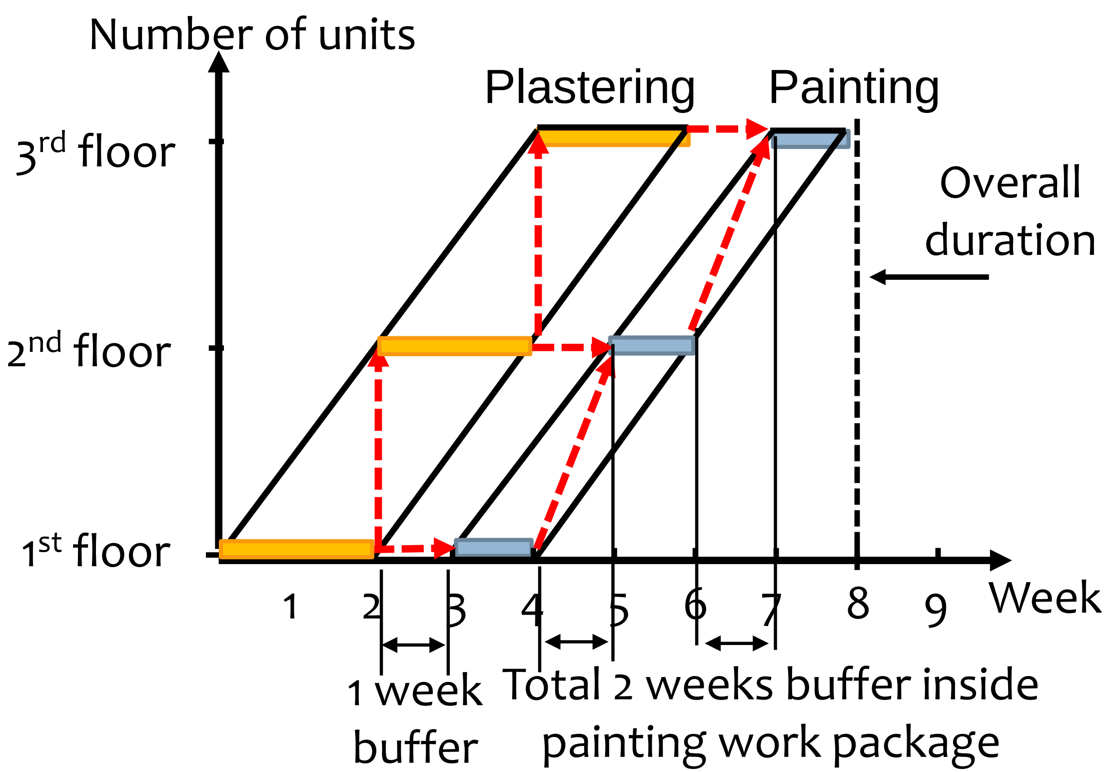
**Example 2:** In interior finishing operation in a 3 storey building, painting of each floor can only be started 1 week after plastering of the floor is done. Plastering of each floor lasts for two weeks and each floor is painted in one week. According to the "as soon as possible" policy for scheduling the work we have:



# Line of balance (LOB)



**Example 2:** In interior finishing operation in a 3 storey building, painting of each floor can only be started 1 week after plastering of the floor is done. Plastering of each floor lasts for two weeks and each floor is painted in one week. According to the "as soon as possible" policy for scheduling the work we have:



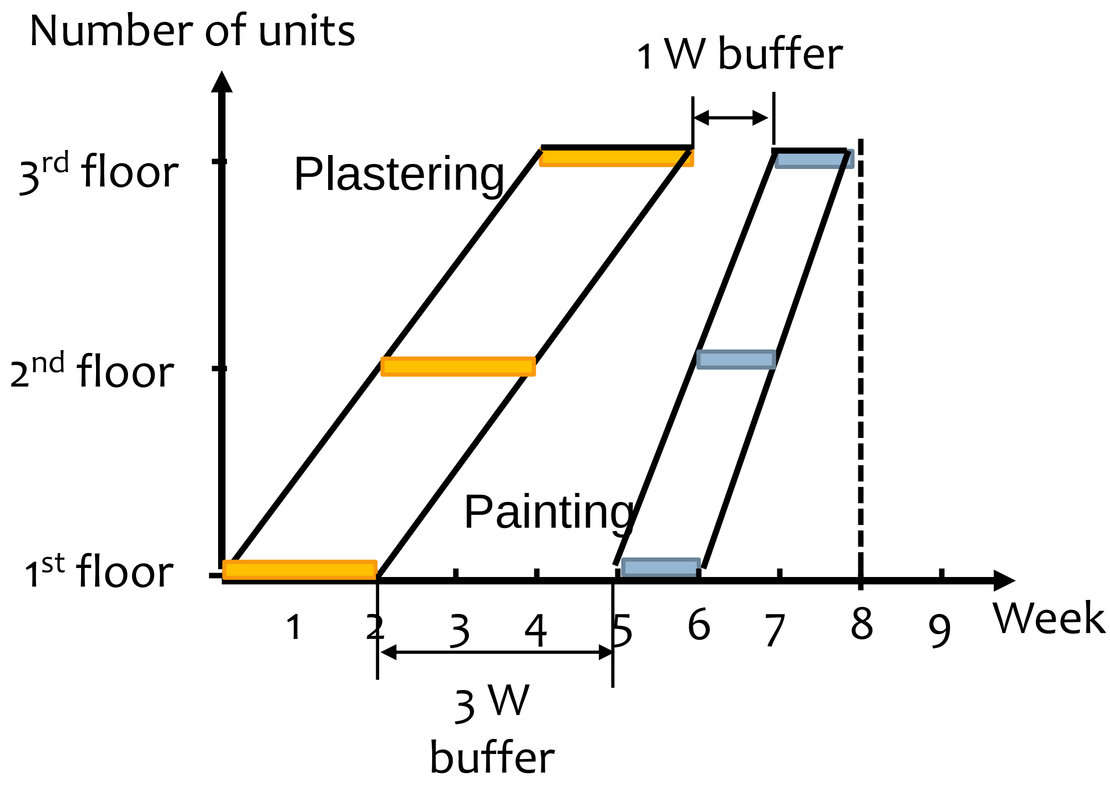

**What does 1 week buffer between plastering and painting mean?**

**What does total 2 weeks buffer inside painting task mean?**

# Line of balance (LOB)



**Example 2 (cont'd):** We can improve the schedule by removing inside buffers and not letting painters to wait after each floor done!



# Line of balance (LOB) - calculations



- The objective of using LOB is to achieve a resource-balanced schedule by determining suitable number of crews to employ in each repetitive activity. This is done such that:
  - The units (locations) are delivered with a rate that meets a pre-specified deadline;
  - The logical CPM network of each activity (task) in each working unit (location) is respected;
  - Crews' work continuity is maintained.

# Line of balance (LOB) - calculations



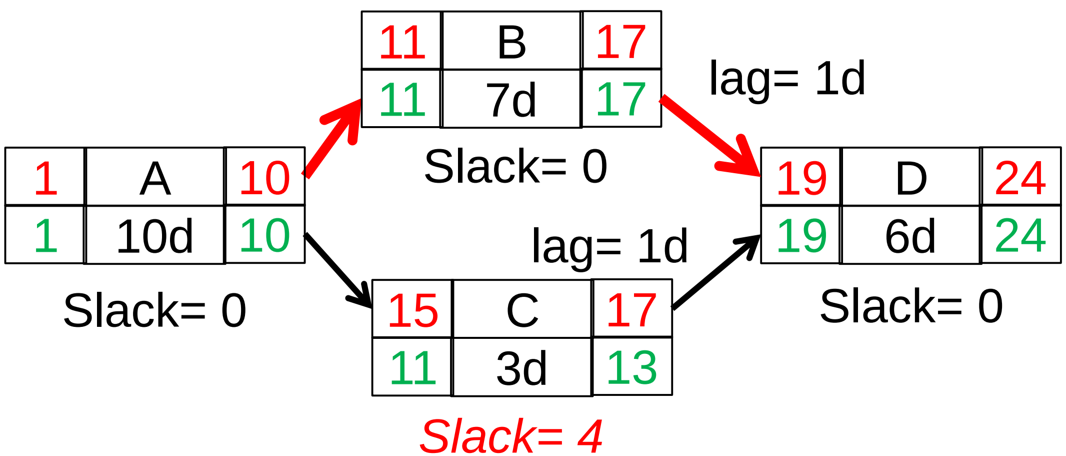
- The CPM-LOB formulation that achieves the above objective involves four main steps:
  1) Use CPM calculations to estimate activity durations deadlines
  2) Calculate desired activity delivery rates to meet CPM calculations
  3) Estimate number of crews to meet the desired delivery rate using LOB calculations
  4) Drawing the LOB schedule

# Line of balance (LOB) - calculations



1) Use CPM calculations to estimate activity durations deadlines:

**CPM Example**: CPM calculation for a repetitive operation with ***5 units*** is given as in below:




For non-critical activities, it is a ***choice*** to use the slack time to adjust activity duration deadlines and reduce number of resources required!

# Line of balance (LOB) - calculations



1) Use CPM calculations to estimate activity durations deadlines:

**CPM Example (cont'd)**: CPM calculation for a repetitive operation with ***5 units***

| | **AD (d)** | **Slack (d)** | | | | | |
|---|---|---|---|---|---|---|---|
| A | 10 | 0 | | | | | |
| B | 7 | 0 | | | | | |
| C | 3 | **4** | | | | | |
| D | 6 | 0 | | | | | |

# Line of balance (LOB) - calculations



2) Calculate desired activity **delivery rates** to meet CPM calculations:

If we have an estimated activity duration of AD (in the CPM calculation) to do in N number of units (or locations), desired delivery rate of R for delivering number of units per time period becomes:

R (delivery Rate) (unit / time) = N (unit) / AD (time)

**Example**: If it is required that a painting activity of a 5 storey building (N=5) to be done within 2 weeks (AD=2), delivery rate becomes:

R (unit / week) = 5 (unit) / 2 (week) = 2.5 (unit / week)

# Line of balance (LOB) - calculations



2) Calculate desired activity **delivery rates** to meet CPM calculations

**CPM Example (Cont'd)**: CPM calculation for a repetitive operation with ***5 units***

| | AD (d) | Slack (d) | R (unit/d) | R (use slack) (unit/d) | | | |
|---|---|---|---|---|---|---|---|
| A | 10 | 0 | 5 /10=0.50 | | | | |
| B | 7 | 0 | 5 / 7=0.71 | | | | |
| C | 3 | 4 | 5 / 3=1.67 | **5 / 7=0.71** | | | |
| D | 6 | 0 | 5 / 6=0.83 | | | | |

**7**

**What does 1.67 or 0.71 delivery rate mean for activity C?**

# Line of balance (LOB) - calculations



3) Estimate number of crews to meet the desired delivery rate using LOB calculations

R = desired delivery rate (unit / time)

D = duration that one crew finishes one location-unit (time x crew / unit)

C = estimated number of working crews (crew)

C (crew) = R (unit / time) * D (time x crew / unit)

**Example**: In the painting activity example with a 2 weeks deadline and desired rate (R) of 2.5 unit/week, if it takes 1 week (D) for one painting crew to complete one floor, we have number of required workers (C) as:

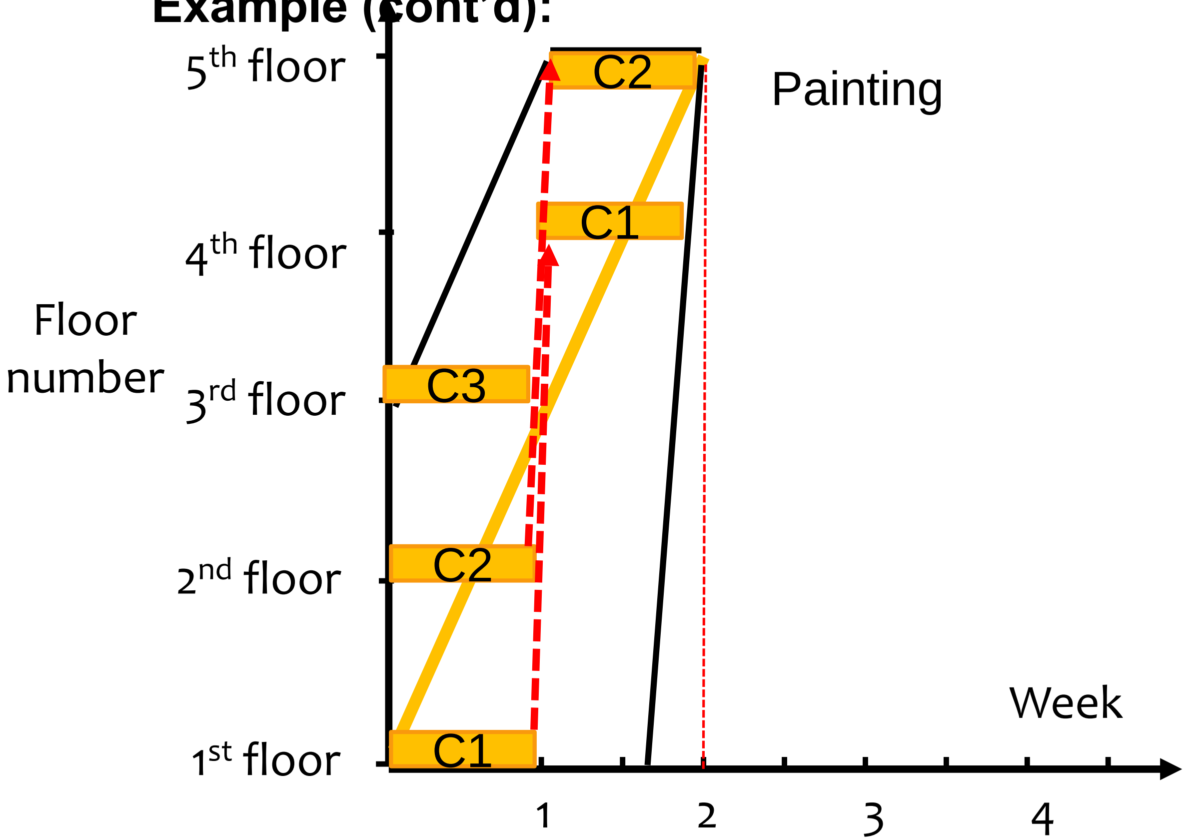
C = 2.5 (unit / week) * 1 (week x crew / unit) = 2.5 crews

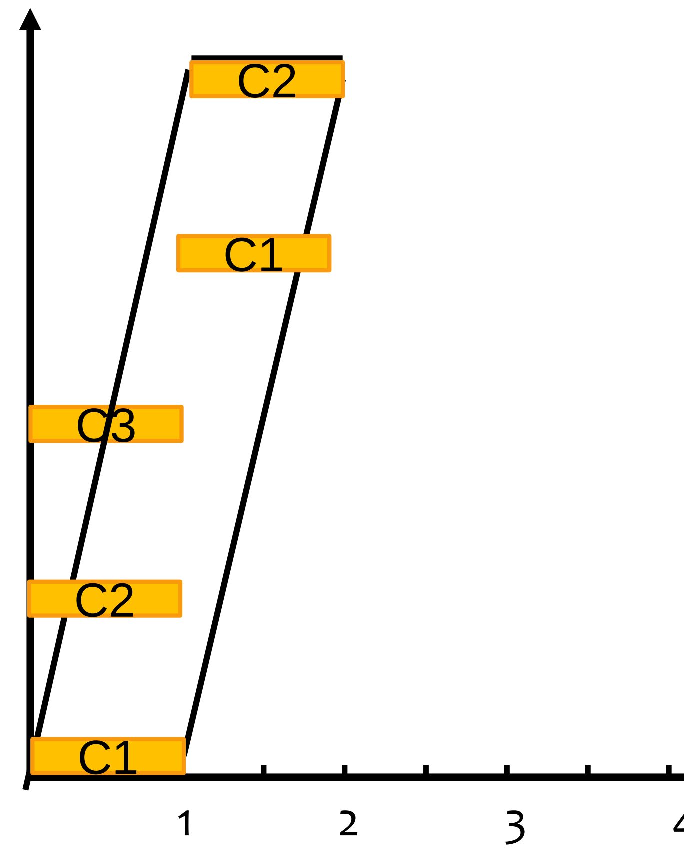
# Line of balance (LOB) - calculations

3) Estimate number of crews to meet the desired delivery rate using LOB calculations:

3 crews for the 1st and 2 crews for the 2nd week.

**Example (cont'd):**



# Line of balance (LOB) - calculations



3) Estimate number of crews to meet the desired delivery rate using LOB calculations:

**CPM Example (Cont'd)**: CPM calculation for a repetitive operation with ***5 units***

| | AD (d) | Slack (d) | R (unit/d) | R (use slack) (unit/d) | D (d.L/unit) | C (L) | C (use slack) (L) |
|---|---|---|---|---|---|---|---|
| A | 10 | 0 | 0.50 | | 4 | 0.5x4=2 | |
| B | 7 | 0 | 0.71 | | 1.4 | 0.71x1.4=1 | |
| C | 3 | **4** | 1.67 | **0.71** | 2 | 1.67x2=3.3 | **0.71x2=1.4** |
| D | 6 | 0 | 0.83 | | 2.4 | 0.83x2.4=2 | |

**What does 3.3 or 1.4 crews mean for activity C?**

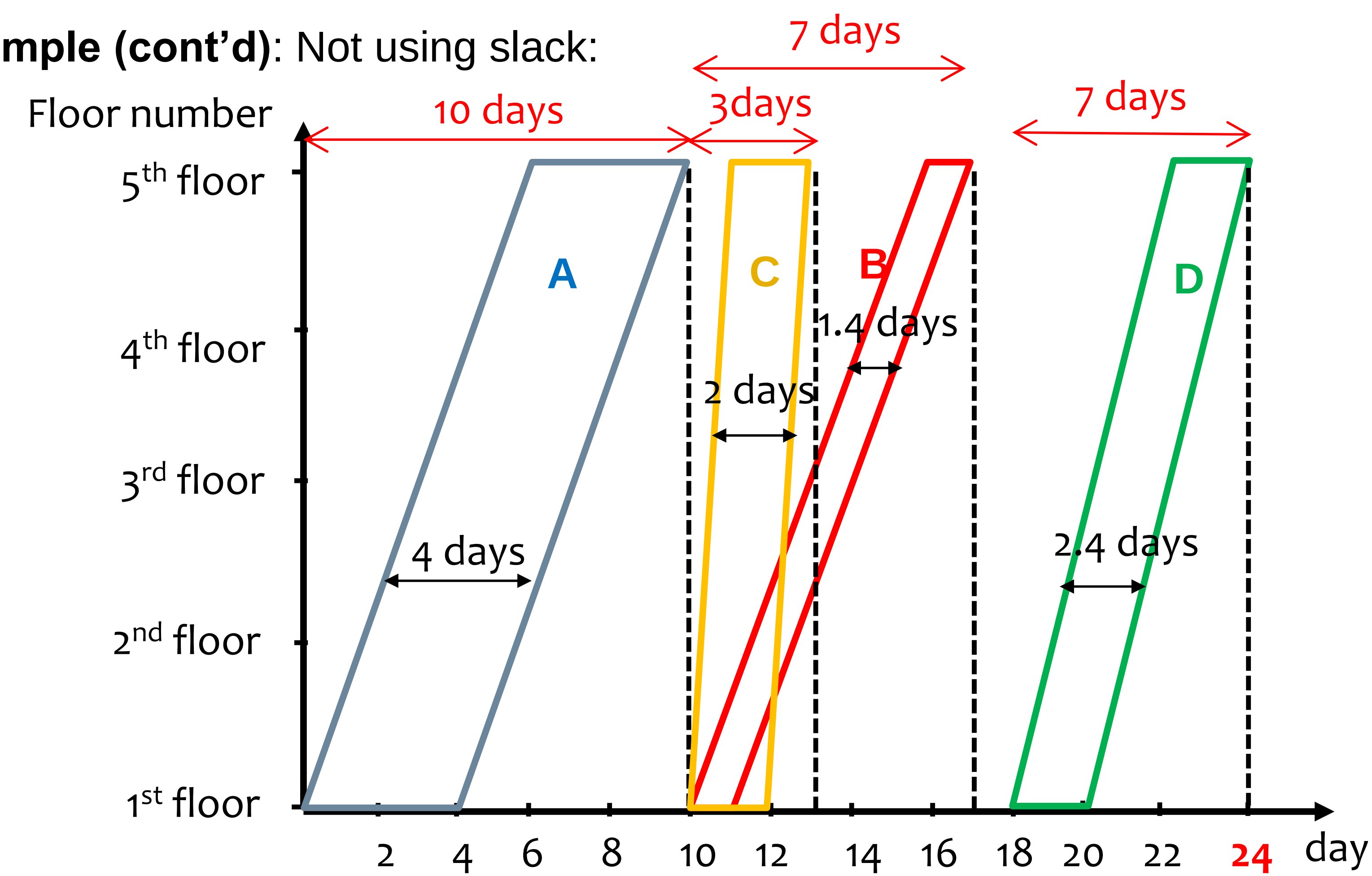
**How are crews distributed in each work package?**

# Line of balance (LOB) - calculations



4) Drawing the LOB schedule:

**CPM example (cont'd)**: Not using slack:



# Line of balance (LOB) - calculations



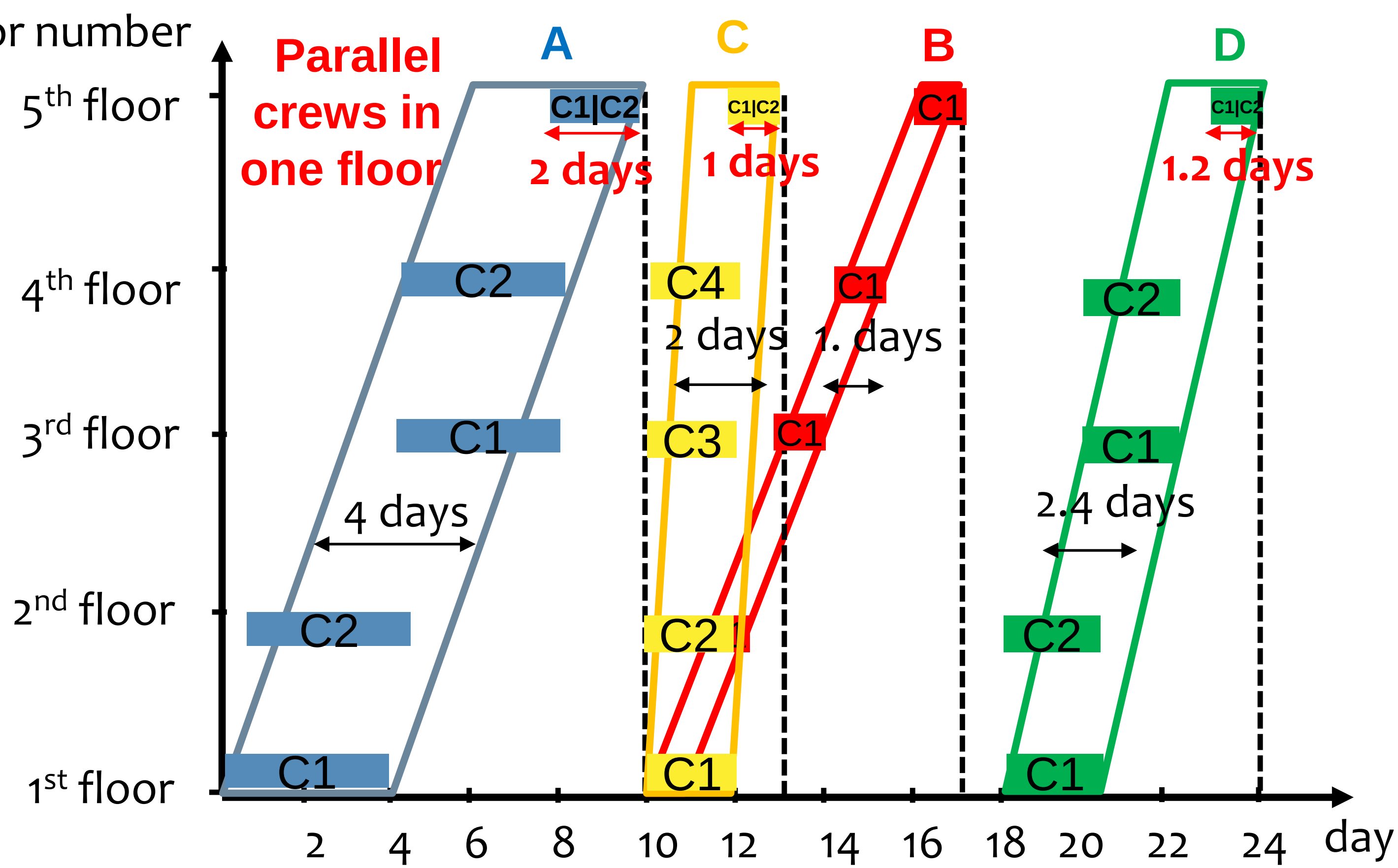
4) Drawing the LOB schedule:

**Example (cont'd)**: Not using slack:

Crew number in A
=(2x4+2x4+2*2) / 10
=2 crews

Crew number in C
=(4*2+2*1) / 3
= 3.3 crews

What if two parallel crews can not share the work in one floor?

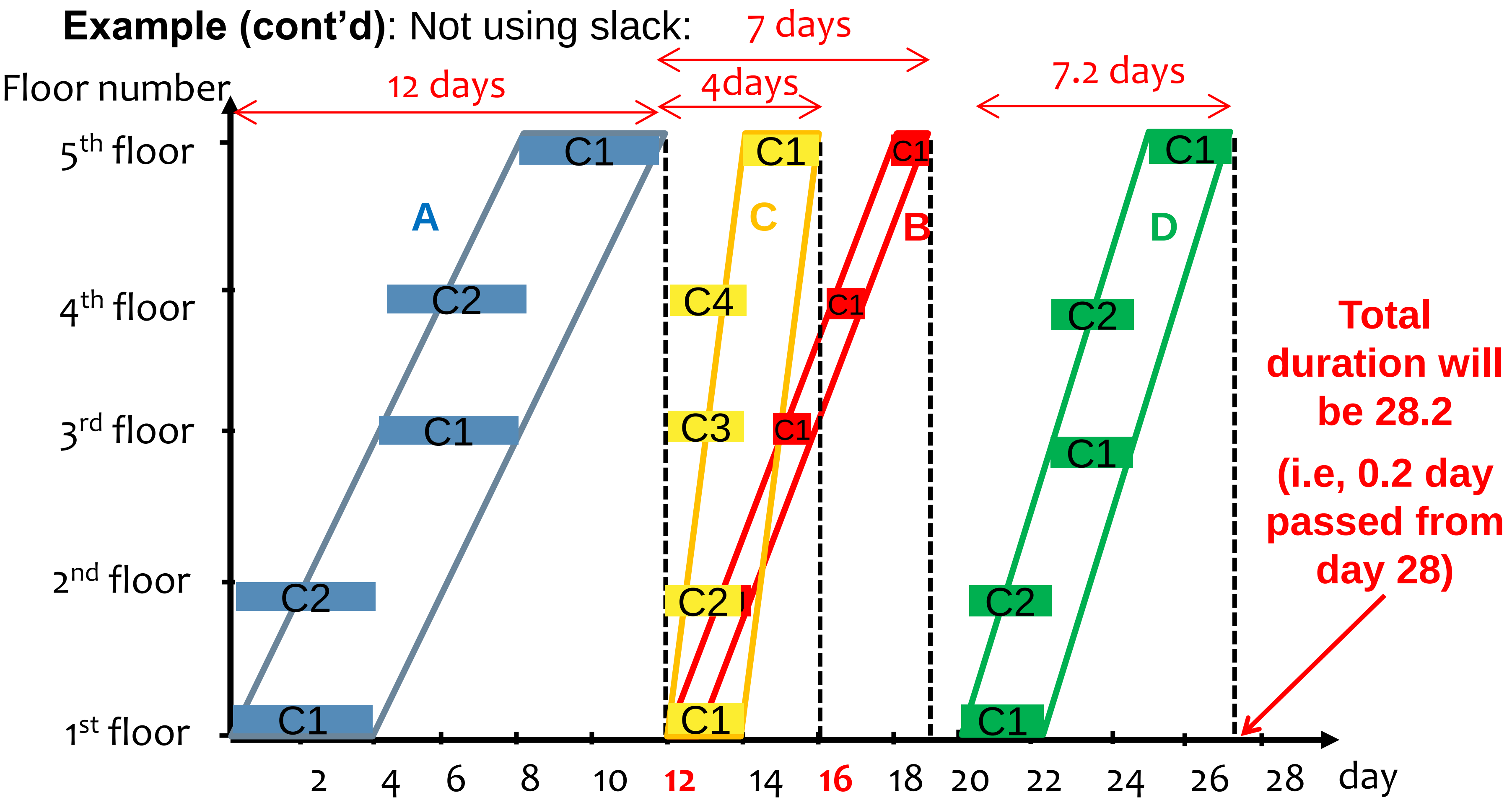


# Line of balance (LOB) - calculations



4) Drawing the LOB schedule:

**Example (cont'd)**: Not using slack:



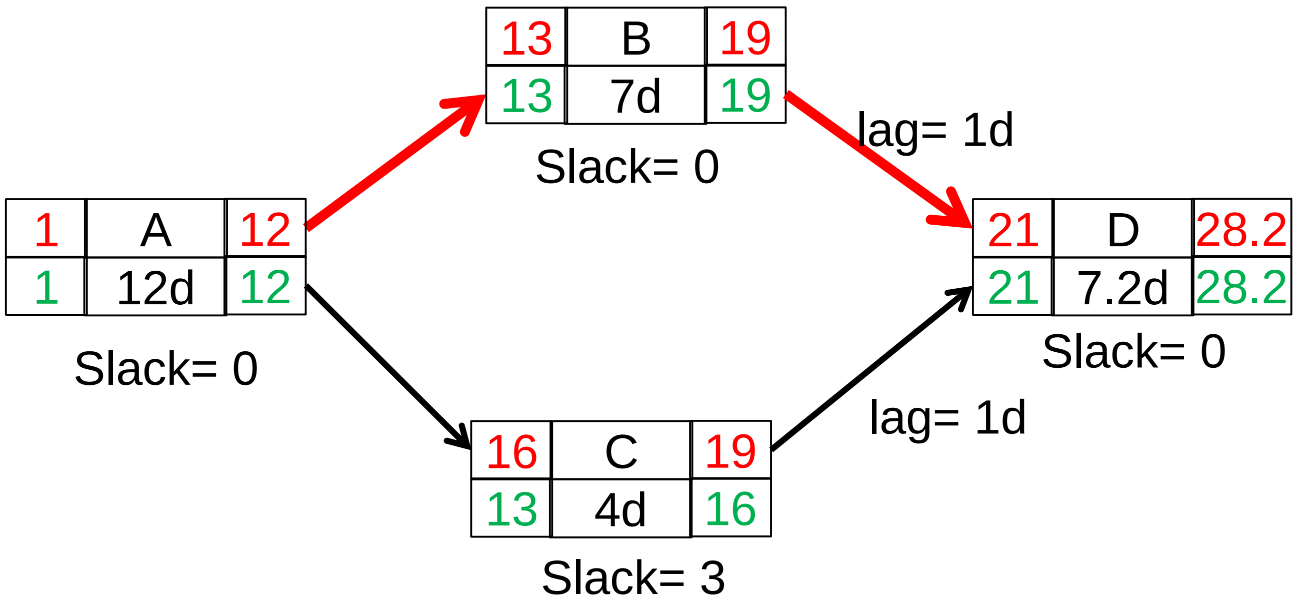
# Line of balance (LOB) - calculations



4) Drawing the LOB schedule:

**Example (cont’d)**:

CPM changes with new durations calculated!



# Line of balance (LOB) - calculations



4) Drawing the LOB schedule:

**Example (cont'd)**: LOB calculations update:

| | AD (d) | Slack (d) | R (unit/d) | R (use slack) (unit/d) | D (d.L/unit) | C (L) | C (use slack) (L) |
|---|---|---|---|---|---|---|---|
| A | **10** | 0 | 0.50 | | 4 | **2** | |
| B | 7 | 0 | 0.71 | | 1.4 | 1 | |
| C | **3** | **4** | 1.67 | 0.71 | 2 | **3.3** | 1.4 |
| D | **6** | 0 | 0.83 | | 2.4 | **2** | |

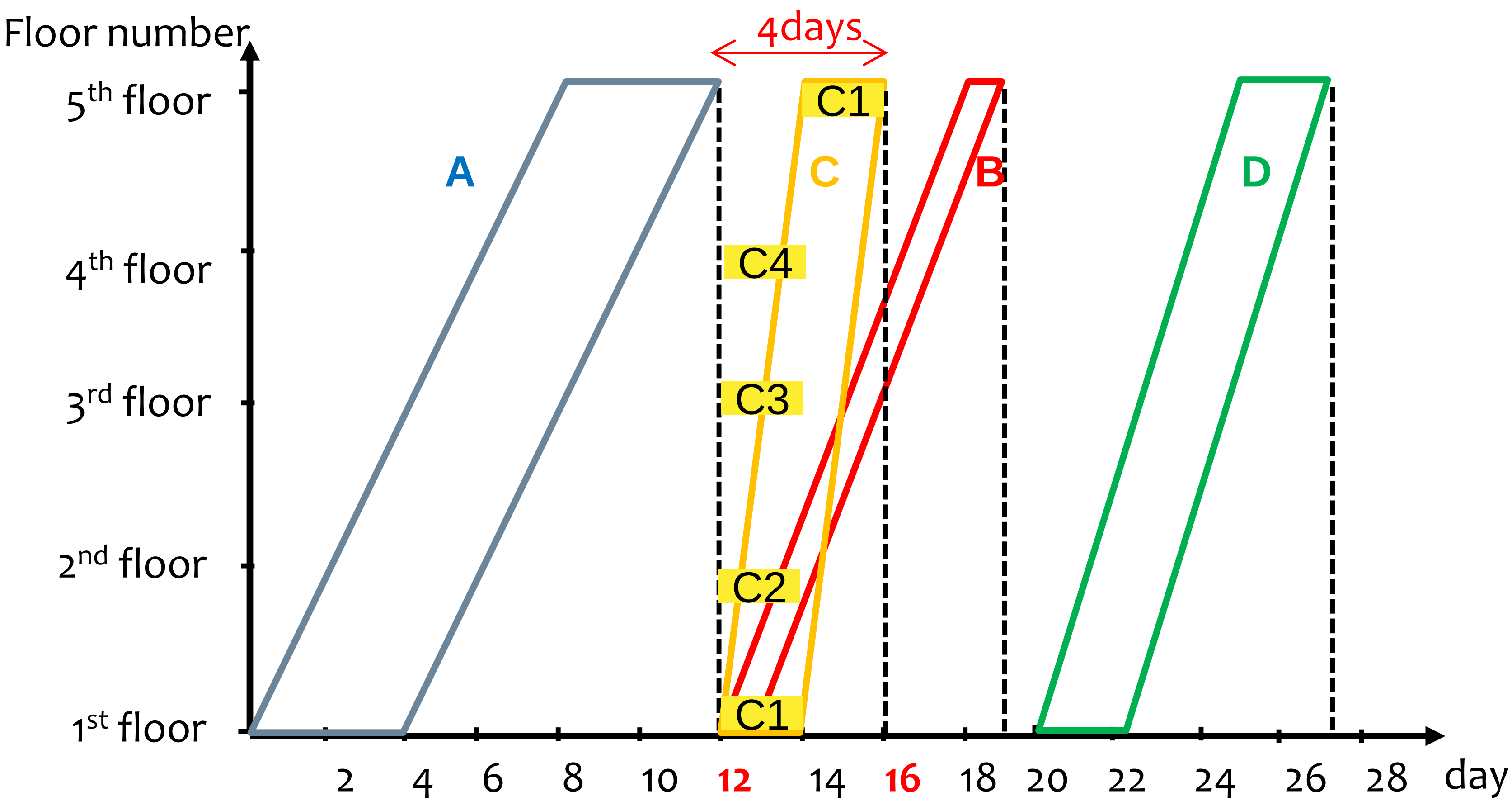
| | AD (d) | Slack (d) | R (unit/d) | R (use slack) (unit/d) | D (d.L/unit) | C (L) | C (use slack) (L) |
|---|---|---|---|---|---|---|---|
| A | **12** | 0 | 0.42 | | 4 | **1.7** | |
| B | 7 | 0 | 0.71 | | 1.4 | 1 | |
| C | **4** | **3** | 1.25 | 0.71 | 2 | **2.5** | 1.4 |
| D | **7.2** | 0 | 0.69 | | 2.4 | **1.7** | |

# Line of balance (LOB) - calculations



4) Drawing the LOB schedule:

**Example (cont'd)**: Use slack for activity C, just hire 2 crews instead of 4 (duration increase from 4 to 6 days):



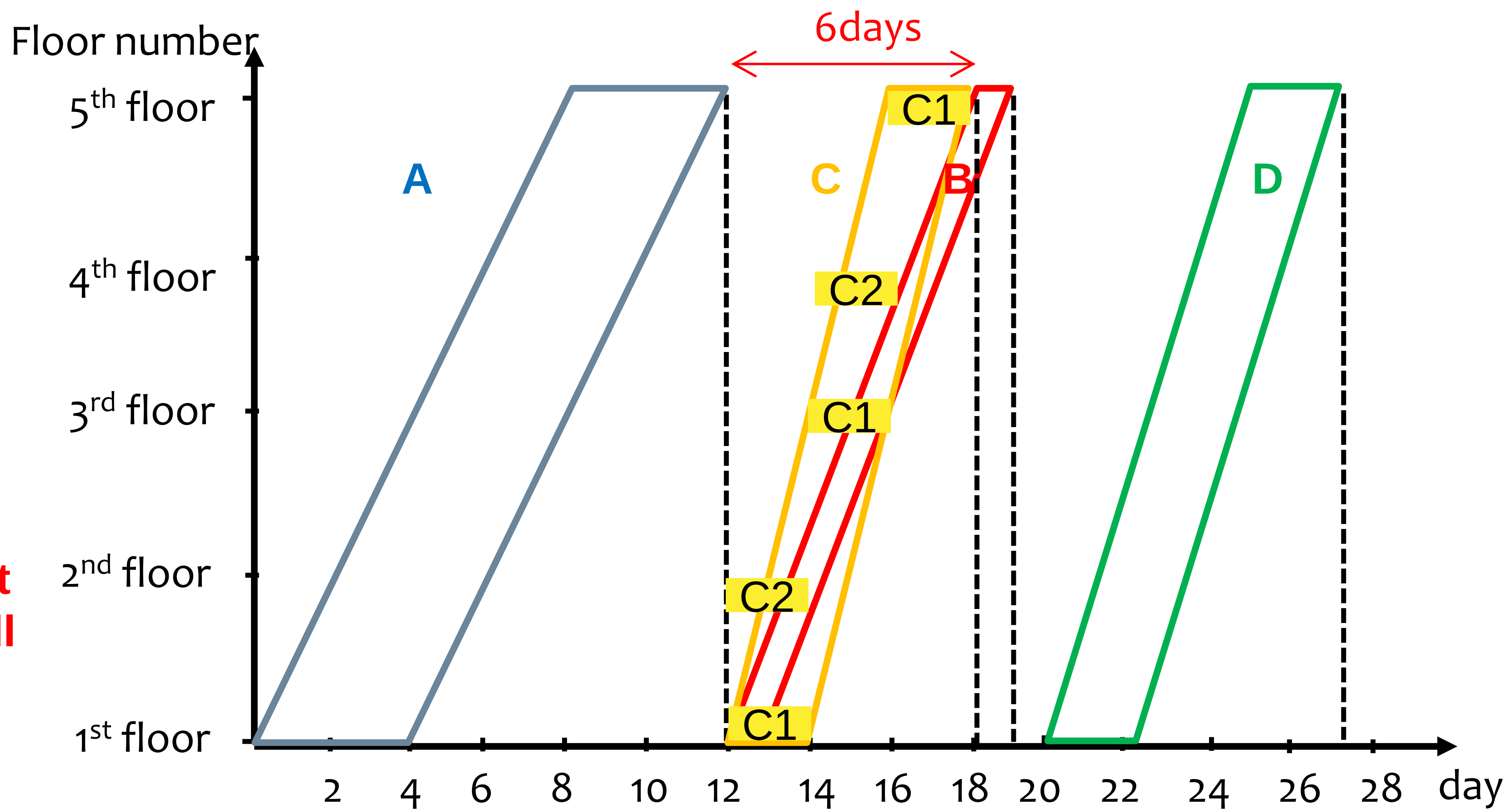
# Line of balance (LOB) - calculations



4) Drawing the LOB schedule:

**Example (cont'd)**: Use slack for activity C, just hire 2 crews instead of 4 (duration increase from 4 to 6 days):

**Why not using all 3 slack days?**

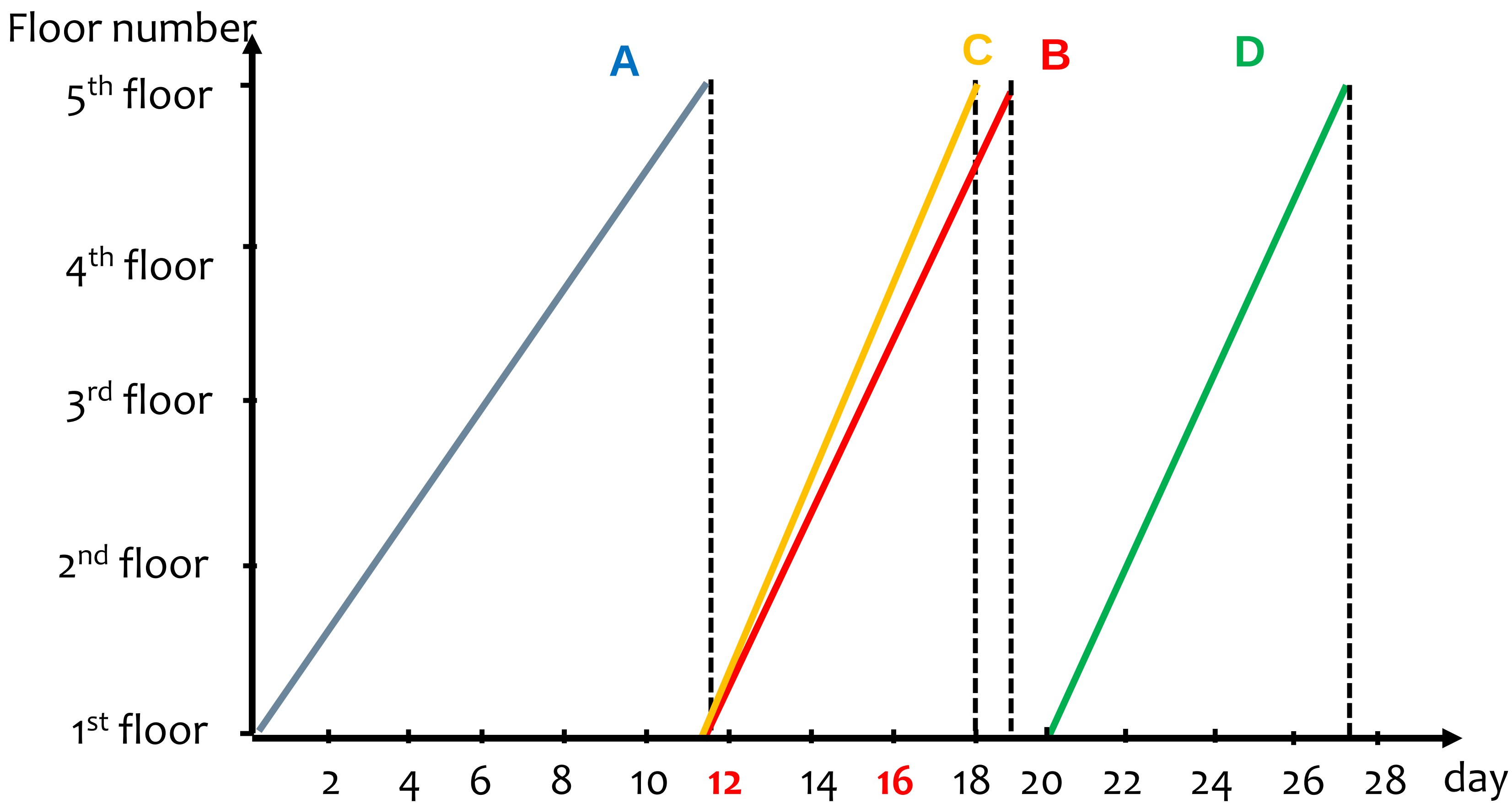


# Line of balance (LOB) - calculations



4) Drawing the LOB schedule:

**Example (cont'd)**: Simplified diagram



# Line of balance (LOB) - calculations



4) Drawing the LOB schedule:

**Example (cont'd)**: LOB calculations final update:

| | AD (d) | Slack (d) | D (d.L/unit) | R (unit/d) | C (L) |
|---|---|---|---|---|---|
| A | 12 | 0 | 4 | 0.42 | 1.7 |
| B | 7 | 0 | 1.4 | 0.71 | 1 |
| C | **6** | **1** | 2 | 0.83 | 1.7 |
| D | 7.2 | 0 | 2.4 | 0.69 | 1.7 |

In estimating work package (operation or activity) durations, specially in repetitive or location based jobs, make sure how different crews move from one location to the other!

# In class exercise-3



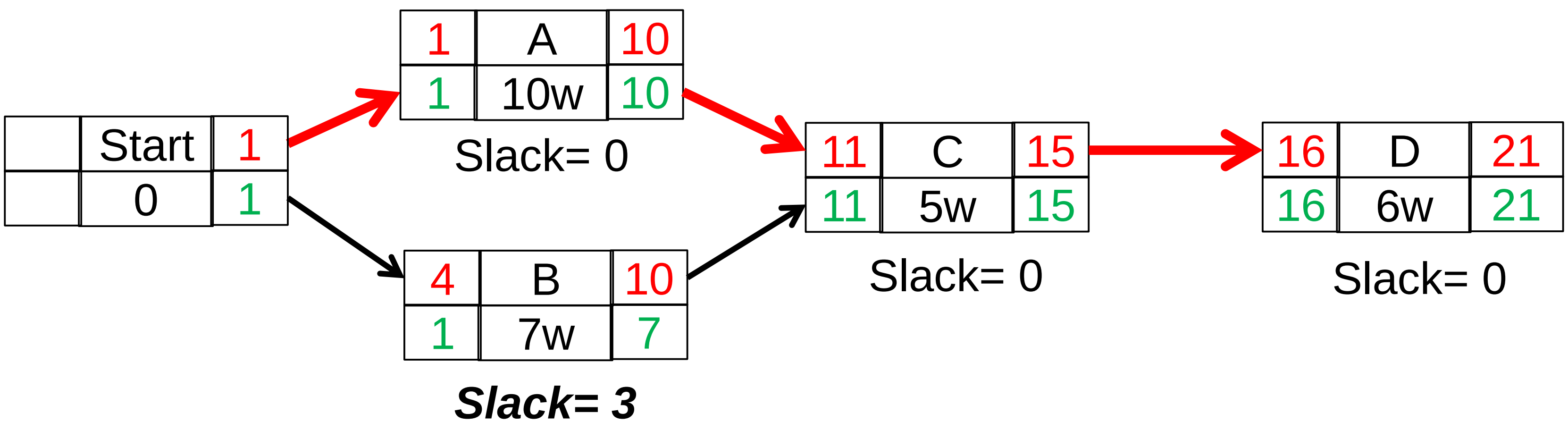
- Draw LOB diagram of a **4 storey** building with repetitive activities as presented in table below. Calculate number of crew required for each work package.

| | Predecessor | Duration (week) | D (w.L/unit) |
|---|---|---|---|
| A | | 10 | 5 |
| B | | 7 | 1.4 |
| C | A,B | 5 | 2 |
| D | C | 6 | 3.2 |

# In class exercise-3



- CPM calculation to determine critical and non-critical work packages (activities):



# In class exercise-3



Calculating crew needs:

| | TD (w) | Slack (w) | D (w.L/unit | R (unit/w) | R (use slack) (unit/w) | C (L) | C (use slack) (L) |
|---|---|---|---|---|---|---|---|
| A | 10 | 0 | 5 | 0.40 | 0.40 | 2 | 2 |
| B | 7 | 3 | 1.4 | 0.57 | 0.40 | 0.8 | 0.56 |
| C | 5 | 0 | 2 | 0.80 | 0.80 | 1.6 | 1.6 |
| D | 6 | 0 | 3.2 | 0.67 | 0.67 | 2.1 | 2.1 |

What does crew number below 1 mean?

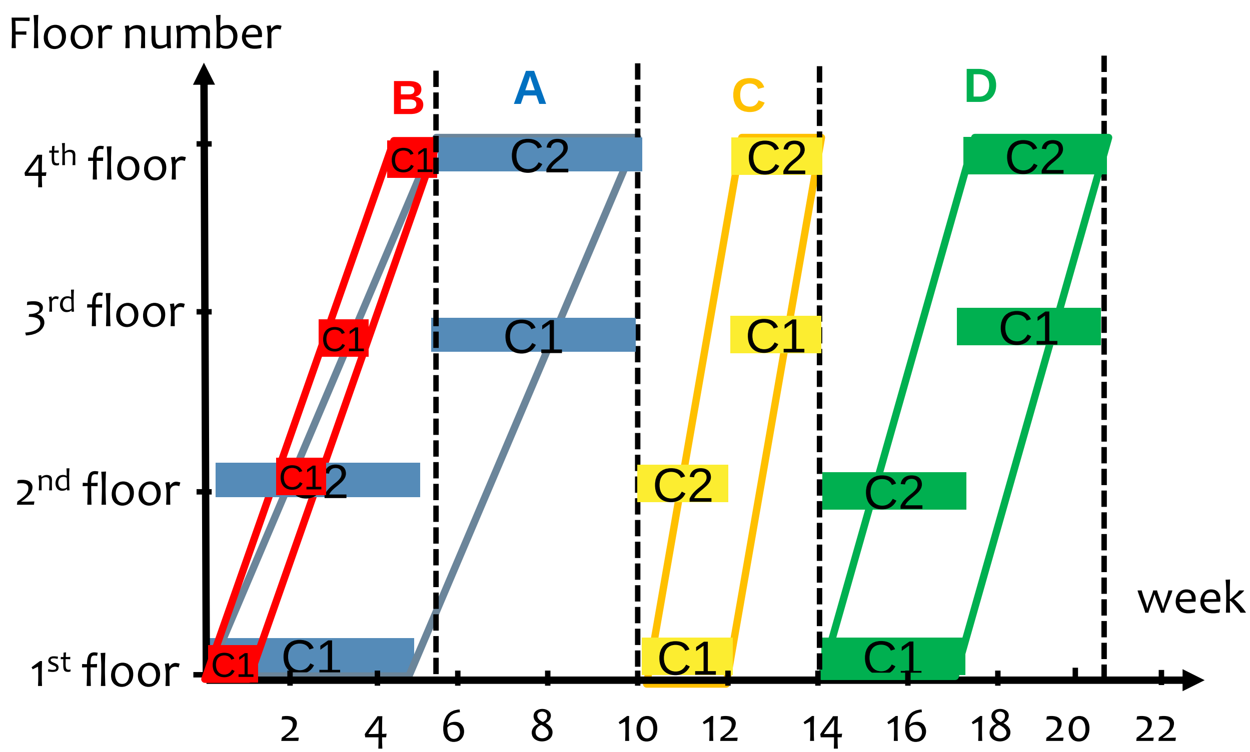
# In class exercise-3



Drawing LOB diagram (not using slacks):

**- We can not use slack time for B since possible longest duration is 5.6 w!**

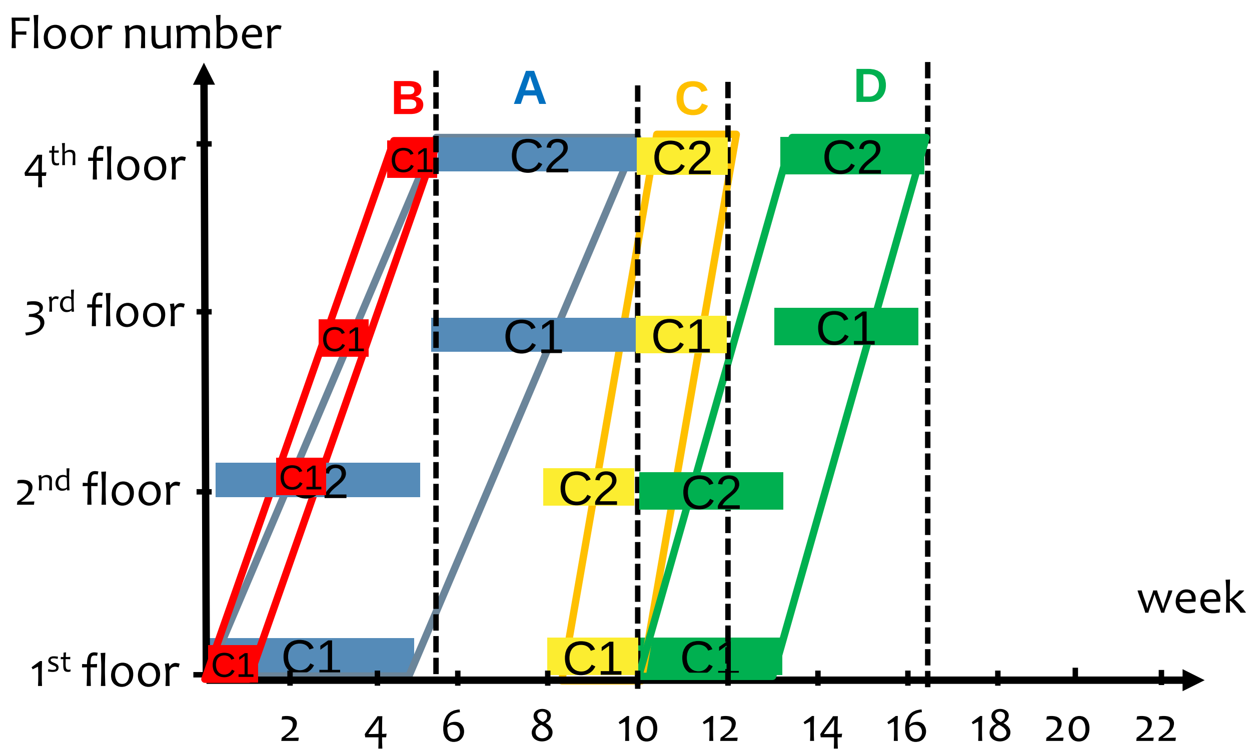
**- We can not end up with 5 w for duration of C, it can be either 4 w (using 2 crews for the entre work) or 8 w (using 1 crew to complete the job), this happens similarly for D.**



# In class exercise-3



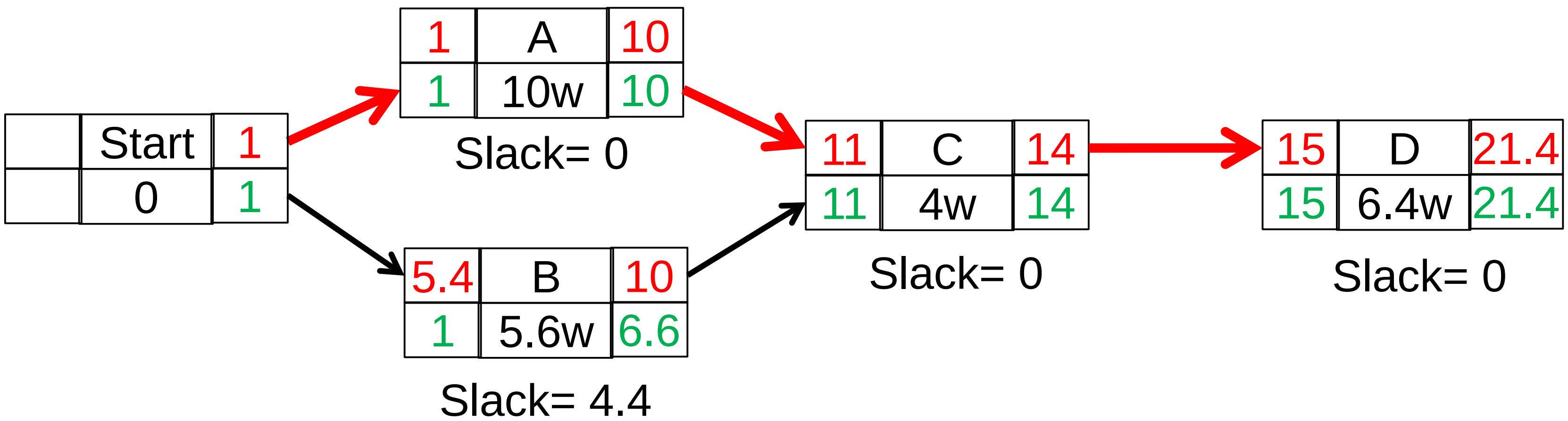
- Drawing LOB diagram (applying dependency at the unit level-No slack):



# In class exercise-3



Update calculations:

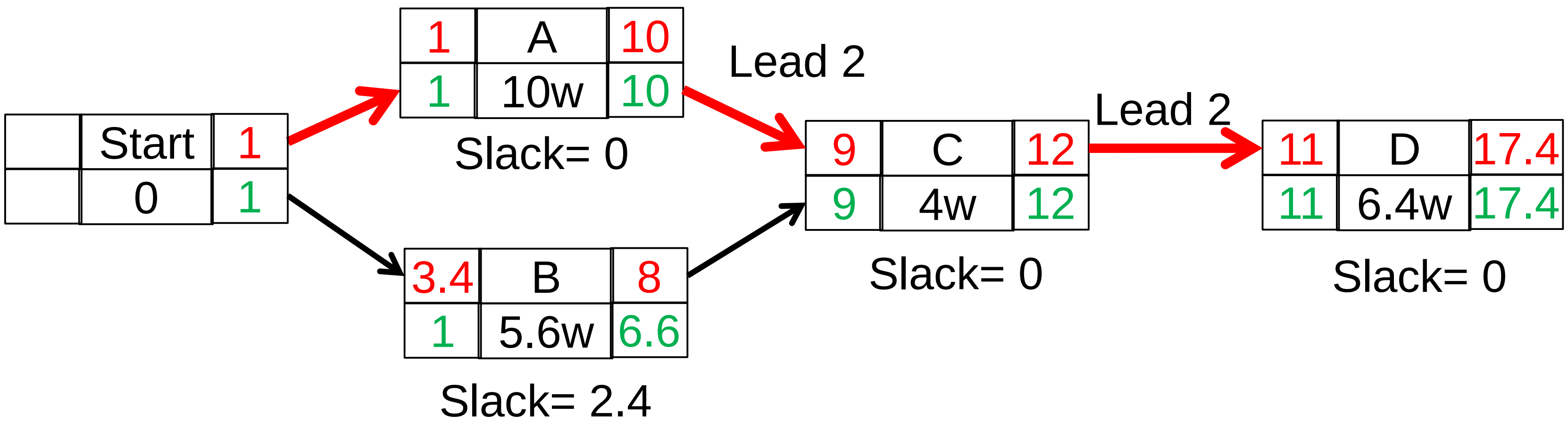



| | TD (W) | Slack (W) | D (w.L/unit) | R (unit/w) | C (L) |
|---|---|---|---|---|---|
| A | 10 | 0 | 5 | 0.4 | 2 |
| B | **5.6** | **4.4** | 1.4 | 0.7 | 1 |
| C | **4** | 0 | 2 | 1.0 | 2 |
| D | **6.4** | 0 | 3.2 | 0.6 | 2 |

# In class exercise-3



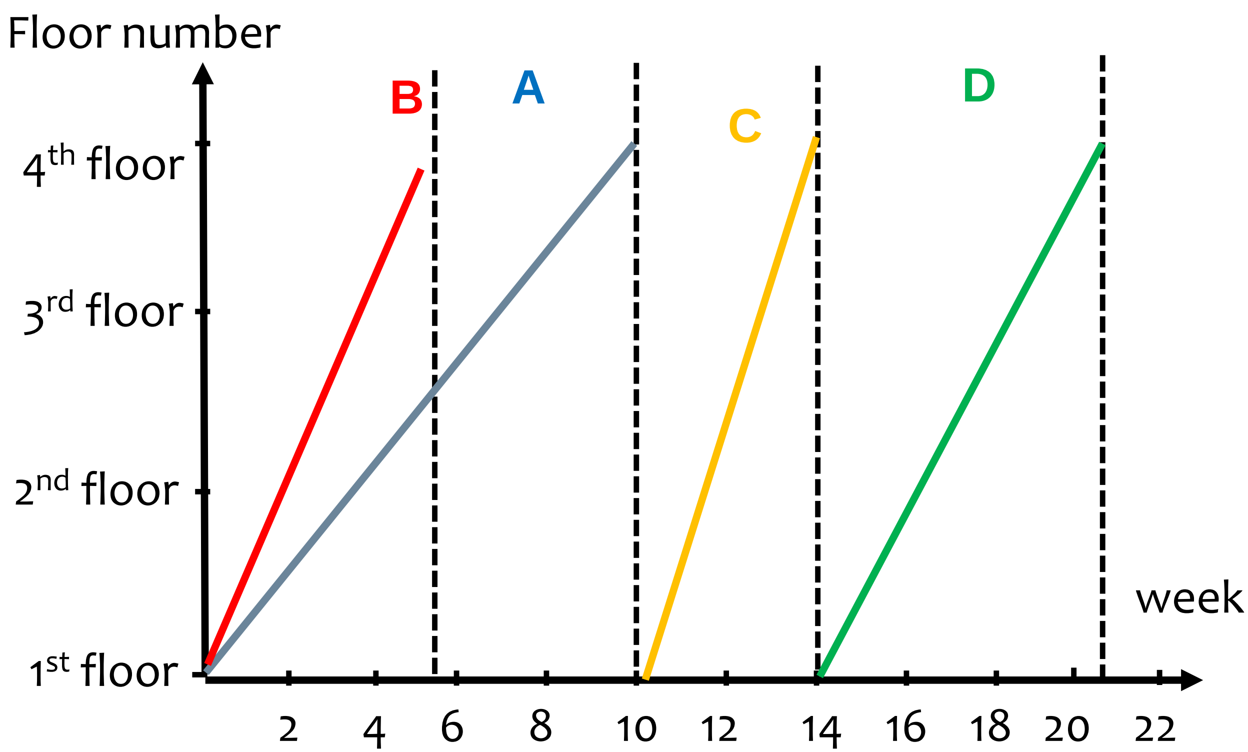
Update calculations:



# In class exercise-3



- Simplified LOB diagram:



# LBS Software

# Line of balance (LOB) - Summary



- LOB is a location based planning tool
- Use LOB diagrams for presenting repetitive jobs to high level management
- Use LOB calculations for balancing number of crews to be used in a job!
- You can use LOB calculations back and forth with time crashing (for identifying proper number of crews) and fast tracking (for identifying proper duration of activity buffers)!

# Assignment 5 – time constraint



a. (25 marks) A Client notifies the contractor that the project needs to be finished within 8 weeks and asks for any additional cost to be announced by the contractor. With the activity crashing information as in below, what will be new total cost of the project?

| WBS | Work packages | Predecessor | Normal duration (w) | Normal cost (MT) | Crash 1 duration (w) | Crash 1 cost (MT) |
|---|---|---|---|---|---|---|
| 1 | Window installation | | 3 | 3 | 1 | 5 |
| 2 | Plumbing | | 5 | 7 | 2 | 11 |
| 3 | Wall plastering | 1 | 5 | 5 | 1 | 8 |
| 4 | Floor isolation | 2 | 3 | 4 | 1 | 5 |
| 5 | Floor finishing | 2,3 | 3 | 5 | 2 | 7 |
| 6 | Door installation | 4 | 2 | 6 | 1 | 7 |
| 7 | Painting | 3 | 3 | 4 | 1 | 7 |

# Assignment 5 – time constraint



b. (25 marks) The contractor has also fast tracking option with the following information:

| WBS | Predecessor | Normal duration (w) | Normal cost (MT) | Chance of rework (% per week of fast tracking) | Rework cost (TT/%) |
|---|---|---|---|---|---|
| **1** | | 3 | 3 | | |
| **2** | | 5 | 7 | 10 | 60 |
| **3** | 1 | 5 | 5 | | |
| **4** | 2 | 3 | 4 | 15 | 70 |
| **5** | 2,3 | 3 | 5 | 15 | 75 |
| **6** | 4 | 2 | 6 | 30 | 30 |
| **7** | 3 | 3 | 4 | 20 | 90 |

Calculate additional cost of the project with using fast tracking for compressing the project to 8 weeks duration! Which method is more beneficial?

# Assignment 5 – time constraint



- c. (25 marks) Contractor was thinking to use a combined method of crashing and fast tracking to reduce to total additional cost for bringing total duration within 8 weeks! Can he find a combined method of project compressing to reduce the total additional cost? If yes, what will be the final additional cost!

# Assignment 5 – time constraint



- d. (25 marks) Information of interior construction activities for a 5 story building is given in table below. Use LOB diagram for synchronizing crews and more accurately calculating activity durations!

| WBS | Predecessor | Estimated Total Duration (w) | Crew Duration per unit (w.L/Unit) | Weekly cost of crew (KT) |
|---|---|---|---|---|
| **1** | | 3 | 1 | 1000 |
| **2** | | 5 | 2.5 | 1500 |
| **3** | 1 | 5 | 3 | 700 |
| **4** | 2 | 3 | 2 | 400 |
| **5** | 2,3 | 3 | 2 | 900 |
| **6** | 4 | 2 | 4 | 800 |
| **7** | 3 | 3 | 1 | 300 |

***Due date one week***

# Thank you!