Directed Acyclic Graphs (DAGs)

Madhavan Mukund

https://www.cmi.ac.in/~madhavan

Programming, Data Structures and Algorithms using Python

Week 4

Tasks and dependencies

- Startup moving into new office space
- Major tasks for completing the interiors
  - Lay floor tiles
  - Plaster the walls
  - Paint the walls
  - Lay conduits (pipes) for electrical wires
  - Do electrical wiring
  - Install electrical fittings
  - Lay telecom conduits
  - Do phone and network cabling

Tasks and dependencies

- Startup moving into new office space
- Major tasks for completing the interiors
  - Lay floor tiles
  - Plaster the walls
  - Paint the walls
  - Lay conduits (pipes) for electrical wires
  - Do electrical wiring
  - Install electrical fittings
  - Lay telecom conduits
  - Do phone and network cabling

- Constraints on the sequence
  - Lay conduits before tiles and plastering
  - Lay tiles, plaster wall before painting
  - Finish painting before any cabling/wiring work
  - Electrical wiring before installing fittings

Tasks and dependencies

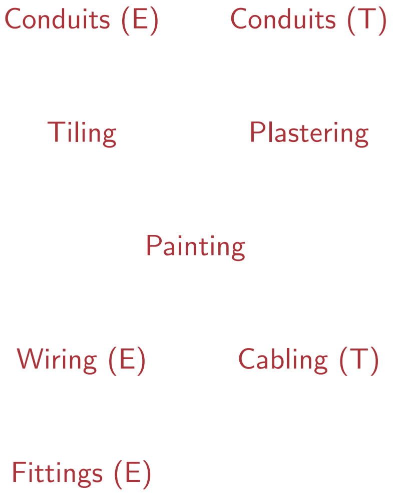
- Startup moving into new office space
- Major tasks for completing the interiors
  - Lay floor tiles
  - Plaster the walls
  - Paint the walls
  - Lay conduits (pipes) for electrical wires
  - Do electrical wiring
  - Install electrical fittings
  - Lay telecom conduits
  - Do phone and network cabling
- Constraints on the sequence
  - Lay conduits before tiles and plastering
  - Lay tiles, plaster wall before painting
  - Finish painting before any cabling/wiring work
  - Electrical wiring before installing fittings
- Represent constraints as a directed graph
  - Vertices are tasks
  - Edge (t, u) if task t has to be completed before task u

Tasks and dependencies

- Constraints on the sequence
  - Lay conduits before tiles and plastering
  - Lay tiles, plaster wall before painting
  - Finish painting before any cabling/wiring work
  - Electrical wiring before installing fittings
- Represent constraints as a directed graph
  - Vertices are tasks
  - Edge (t, u) if task t has to be completed before task u

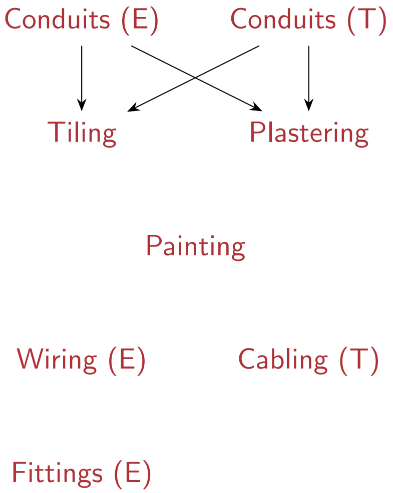
Tasks and dependencies

- Constraints on the sequence
  - Lay conduits before tiles and plastering
  - Lay tiles, plaster wall before painting
  - Finish painting before any cabling/wiring work
  - Electrical wiring before installing fittings
- Represent constraints as a directed graph
  - Vertices are tasks
  - Edge (t, u) if task t has to be completed before task u



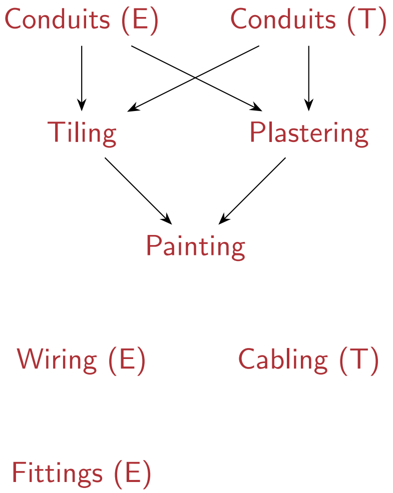
Tasks and dependencies

- Constraints on the sequence
    - Lay conduits before tiles and plastering
    - Lay tiles, plaster wall before painting
    - Finish painting before any cabling/wiring work
    - Electrical wiring before installing fittings
- Represent constraints as a directed graph
    - Vertices are tasks
    - Edge (t, u) if task t has to be completed before task u



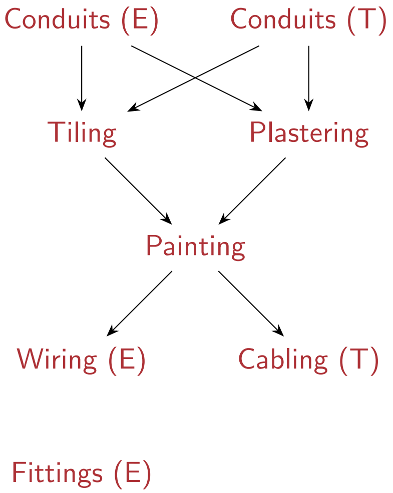
Tasks and dependencies

- Constraints on the sequence
  - Lay conduits before tiles and plastering
  - Lay tiles, plaster wall before painting
  - Finish painting before any cabling/wiring work
  - Electrical wiring before installing fittings
- Represent constraints as a directed graph
  - Vertices are tasks
  - Edge (t, u) if task t has to be completed before task u



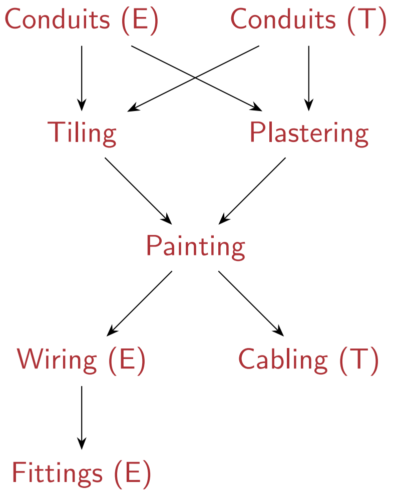
Tasks and dependencies

- Constraints on the sequence
  - Lay conduits before tiles and plastering
  - Lay tiles, plaster wall before painting
  - Finish painting before any cabling/wiring work
  - Electrical wiring before installing fittings
- Represent constraints as a directed graph
  - Vertices are tasks
  - Edge (t, u) if task t has to be completed before task u



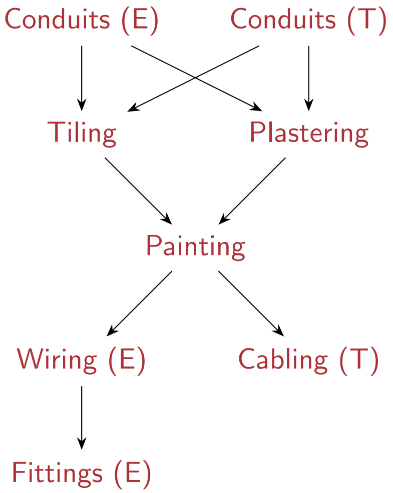
Tasks and dependencies

- Constraints on the sequence
  - Lay conduits before tiles and plastering
  - Lay tiles, plaster wall before painting
  - Finish painting before any cabling/wiring work
  - Electrical wiring before installing fittings
- Represent constraints as a directed graph
  - Vertices are tasks
  - Edge (t, u) if task t has to be completed before task u



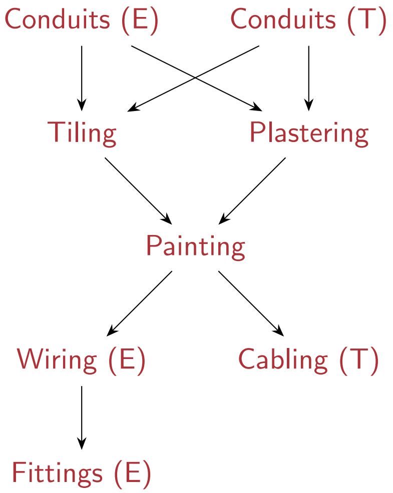
Typical questions

- Schedule the tasks respecting the dependencies



Typical questions

- Schedule the tasks respecting the dependencies
    - Conduits (E) – Conduits (T) – Tiling – Plastering – Painting – Wiring (E) – Cabling (T) – Fittings (E)



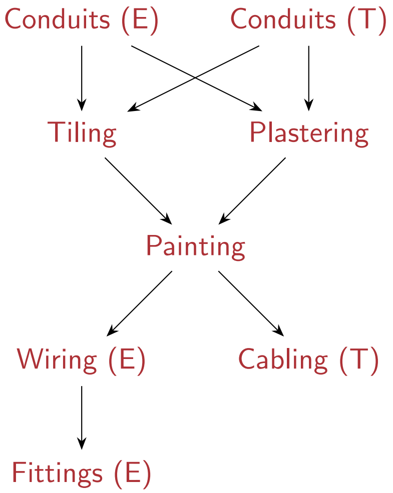
Typical questions

- Schedule the tasks respecting the dependencies
  - Conduits (E) – Conduits (T) – Tiling – Plastering – Painting – Wiring (E) – Cabling (T) – Fittings (E)
  - Conduits (T) – Conduits (E) – Plastering – Tiling – Painting – Wiring (E) – Fittings (E) – Cabling (T)
  - . . .

$$\begin{tikzcd}
\text{Conduits (E)} \arrow[d] \arrow[drr] & & \text{Conduits (T)} \arrow[dll] \arrow[d] \\
\text{Tiling} \arrow[dr] & & \text{Plastering} \arrow[dl] \\
& \text{Painting} \arrow[dl] \arrow[dr] & \\
\text{Wiring (E)} \arrow[d] & & \text{Cabling (T)} \\
\text{Fittings (E)} & &
\end{tikzcd}$$

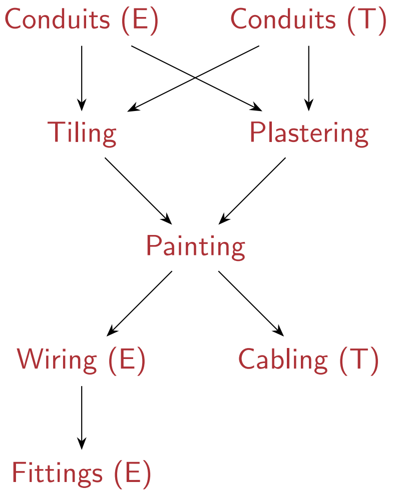
Typical questions

- Schedule the tasks respecting the dependencies
  - Conduits (E) – Conduits (T) – Tiling – Plastering – Painting – Wiring (E) – Cabling (T) – Fittings (E)
  - Conduits (T) – Conduits (E) – Plastering – Tiling – Painting – Wiring (E) – Fittings (E) – Cabling (T)
  - . . .
- How long will the work take?



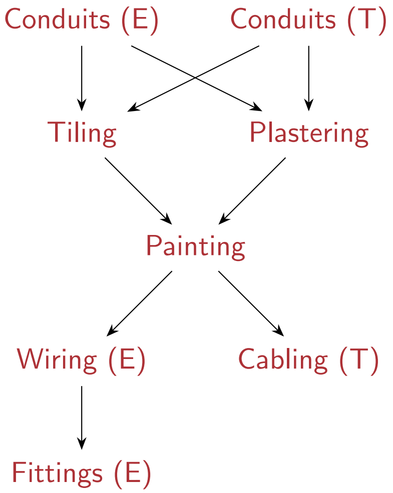
Directed Acyclic Graphs

- Formally, we have a directed acyclic graph (DAG)
- $G = (V, E)$, a directed graph without directed cycles



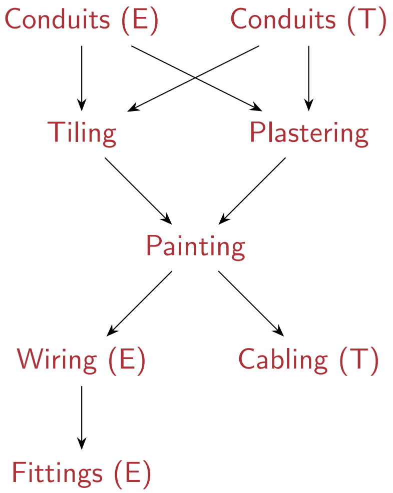
Directed Acyclic Graphs

- Formally, we have a directed acyclic graph (DAG)
- $G = (V, E)$, a directed graph without directed cycles
- Find a schedule
  - Enumerate $V = \{0, 1, \ldots, n-1\}$ such that for any $(i, j) \in E$, i appears before j



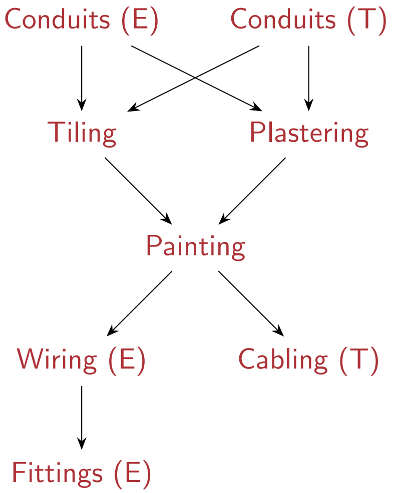
Directed Acyclic Graphs

- Formally, we have a directed acyclic graph (DAG)
- $G = (V, E)$, a directed graph without directed cycles
- Find a schedule
  - Enumerate $V = \{0, 1, \ldots, n-1\}$ such that for any $(i,j) \in E$, i appears before j
  - Topological sorting



Directed Acyclic Graphs

- Formally, we have a directed acyclic graph (DAG)
- $G = (V, E)$, a directed graph without directed cycles
- Find a schedule
  - Enumerate $V = \{0, 1, \dots, n-1\}$ such that for any $(i, j) \in E$, i appears before j
  - Topological sorting
- How long with the work take?
  - Find the longest path in the DAG



Summary

- Directed acyclic graphs are a natural way to represent dependencies
- Arise in many contexts
  - Pre-requisites between courses for completing a degree
  - Recipe for cooking
  - Construction projects
  - ...
- Problems to be solved on DAGS
  - Topological sorting
  - Longest paths