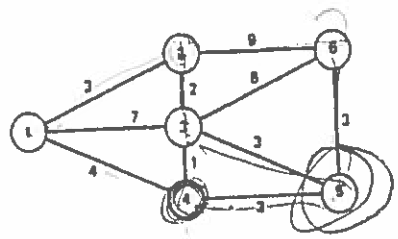
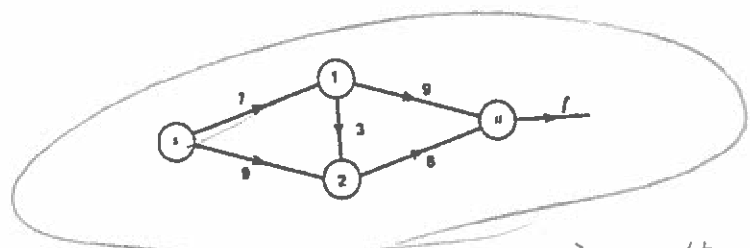
Additional Examples and Algorithms

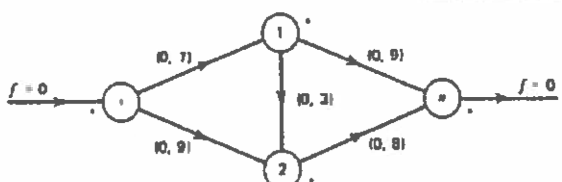
Example 1. Shortest Path

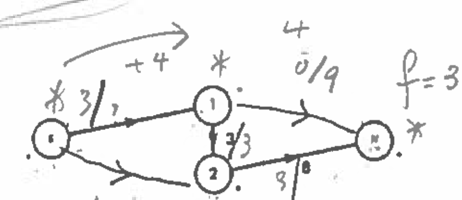


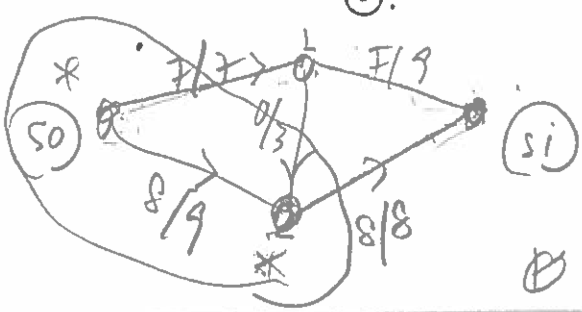
0. $[0^*, 3, 7, 4, \infty, \infty]$
1. $[0^*, \underline{3^*}, 7, 4, \infty, \infty]$
2. $[0^*, \underline{3^*}, 5, 4^*, \infty, 12]$
3. $[0^*, 3^*, 5^*, 4^*, 7, 12]$
4. $[0^*, 3^*, 5^*, 4^*, 7^*, 11]$
5. $\boxed{[0^*, 3^*, 5^*, 4^*, 7^*, 10^*]}$

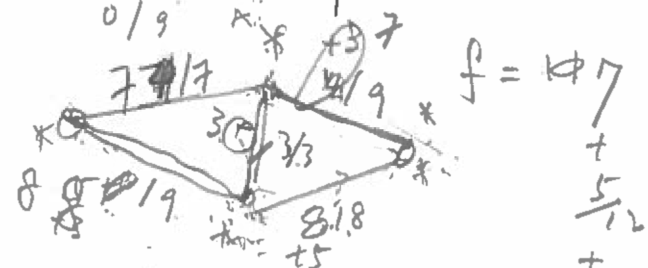
Example 2. A maximal flow problem







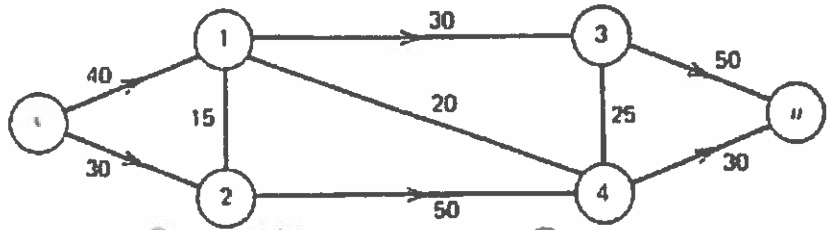


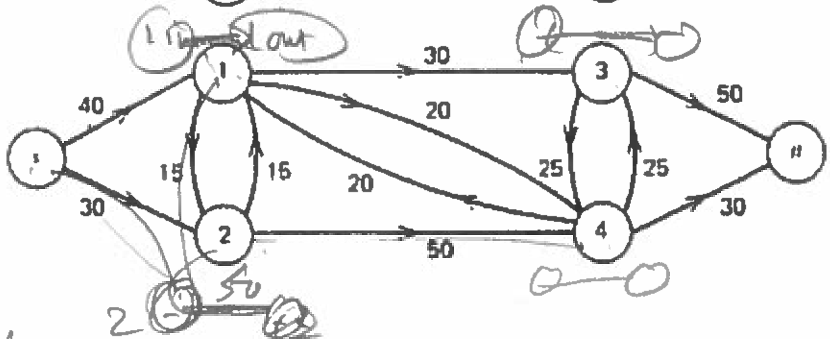


$S = \{so, 2\}, \; K(S, \bar{S}) = 7 + 8 = 15$

Quiz on Thursday will cover material in Homework 8,
(Make sure you understand the solution [Sample])
and also the shortest path algorithm.

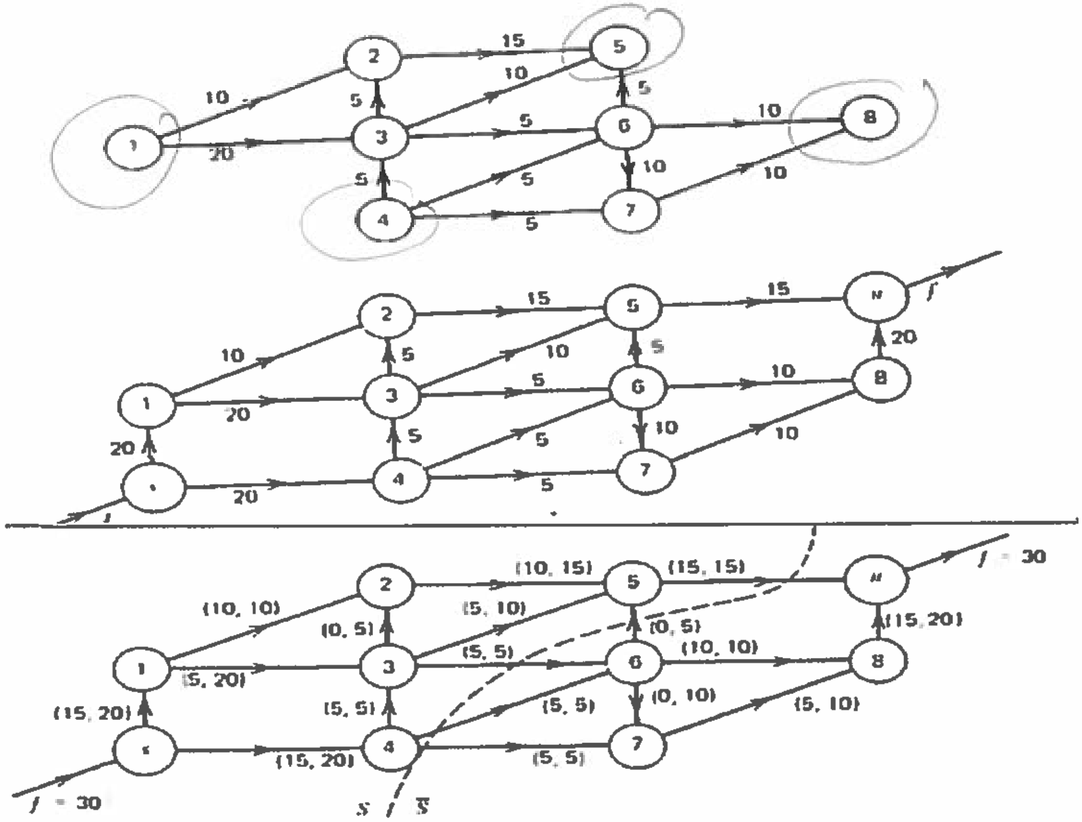
Example 3. Another maximal flow example with an undirected arc





Example: If we impose 50 unit vertex constraints on each vertex, we can split each ~~vertices~~ vertex, not equal to the source & sink into an incoming and outgoing vertex. Then set the capacity constraint of the vertex to be a regular arc capacity constraint between $n_{in} \to n_{out}$.

Example 4. A transshipment example with multiple sources and sinks



8.4 Critical Path Method (CPM), Project Evaluation and Review Techniques (PERT)

One can use network models to deal with scheduling problem of large complex projects with many activities - construction, building, manufacturing, launching (scientific, commercial, industrial) projects, etc.

Basic set up

1. Node 1 represents the start of the project. An arc should lead from node 1 to represent each activity that has no predecessors.
2. A node (called the finish node) representing the completion of the project should be included in the network.
3. Number the nodes in the network so that the node representing the completion of an activity always has a larger number than the node representing the beginning of an activity (there may be more than one numbering scheme that satisfies rule 3).
4. 4 An activity should not be represented by more than one arc in the network.
5. Two nodes can be connected by at most one arc.

FIGURE 26
Activity A Must Be Completed Before Activity B Can Begin

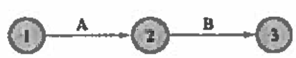


FIGURE 27
Activities A and B Must Be Completed Before Activity C Can Begin

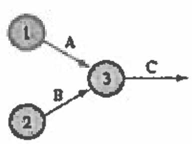


FIGURE 28
Activity A Must Be Completed Before Activities B and C Can Begin

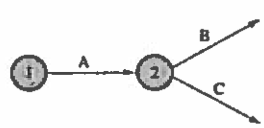


FIGURE 29
Violation of Rule 5

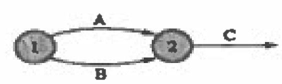
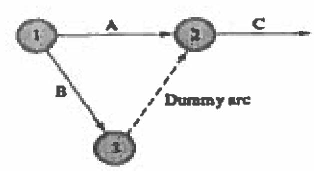


FIGURE 30
Use of Dummy Activity



Remark To avoid violating rules 4 and 5, it is sometimes necessary to utilize a dummy activity that takes zero time.

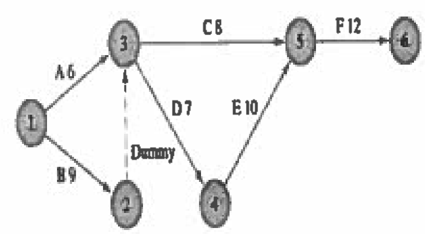
Example Suppose activities A and B are both predecessors of activity C and can begin at the same time. In the absence of rule 5, we could represent this by Figure 29. However, because nodes 1 and 2 are connected by more than one arc, Figure 29 violates rule 5. By using a dummy activity (indicated by a dotted arc), as in Figure 30, we may represent the fact that A and B are both predecessors of C. Figure 30 ensures that activity C cannot begin until both A and B are completed, but it does not violate rule 5.

Example Widgetco is about to introduce a new product (product 3).

- One unit of product 3 is produced by assembling 1 unit of product 1 and 1 unit of product 2.
- Before production begins on either product 1 or 2, raw materials must be purchased and workers must be trained.
- Before products 1 and 2 can be assembled into product 3, the finished product 2 must be inspected. A list of activities and their predecessors and of the duration of each activity is given in Table 12. Draw a project diagram for this project.

TABLE 12
Duration of Activities and Predecessor Relationships for Widgetco

Activity	Predecessors	Duration (Days)
A = train workers	—	6
B = purchase raw materials	—	9
C = produce product 1	A, B	8
D = produce product 2	A, B	7
E = test product 2	D	10
F = assemble products 1 and 2	C, E	12



Node 1 = starting node
Node 6 = finish node

Definition The early event time for node i, represented by $ET(i)$, is the earliest time at which the event corresponding to node i can occur.

The late event time for node i, represented by $LT(i)$, is the latest time at which the event corresponding to node i can occur without delaying the completion of the project.

Computation of early event time

Set $ET(1) = 0$. In gneral, if $ET(j)$ is known for $j < i$, we can find $LT(i)$ as follows.

Step 1 Find each prior event to node i that is connected by an arc to node i. These events are the immediate predecessors of node i.

Step 2 To the ET for each immediate predecessor of the node i add the duration of the activity connecting the immediate predecessor to node i.

Step 3 $ET(i)$ equals the maximum of the sums computed in step 2.

Computation of late event time

Set $LT(n)$ to be the finish time. In general, if $LT(j)$ is known for $j > i$, we can find $LT(i)$ as follows:

Step 1 Find each node that occurs after node i and is connected to node i by an arc.
These events are the immediate successors of node i.

Step 2 From the LT for each immediate successor to node i, subtract the duration of the activity joining the successor the node i.

Step 3 $LT(i)$ is the smallest of the differences determined in step 2.

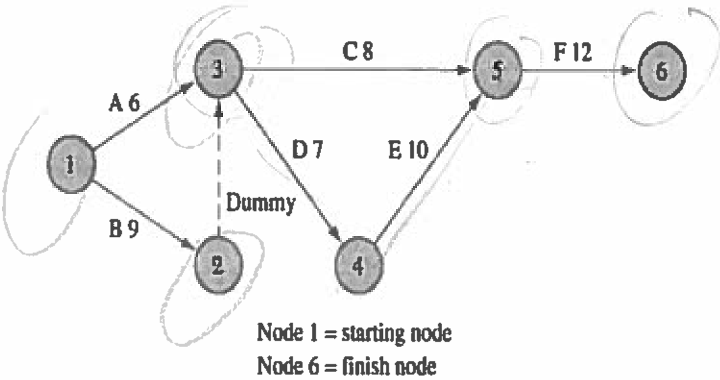


TABLE 13
ET and LT for Widgetco

Node	ET(i)	LT(i)
1	0	0
2	9	9
3	9	9
4	16	16
5	26	26
6	38	38

Total flow

For an arbitrary arc representing activity (i, j), the total float, represented by $TF(i, j)$, of the activity represented by (i, j) is the amount by which the starting time of activity (i, j) could be delayed beyond its earliest possible starting time without delaying the completion of the project (assuming no other activities are delayed). So, $TF(i, j) + ET(i) + t_{ij} \le LT(j)$, and hence

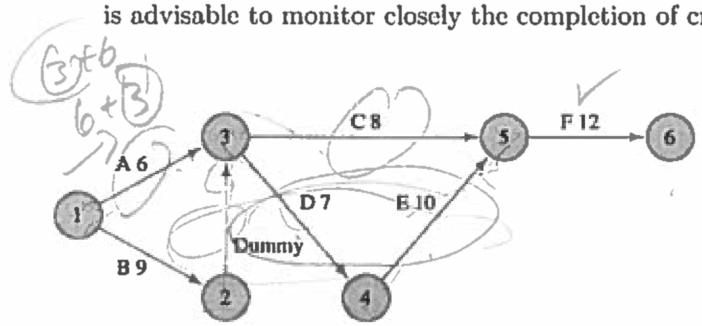
$$TF(i, j) = LT(j) - ET(i) - t_{ij}.$$

Activity B: $TF(1, 2) = LT(2) - ET(1) - 9 = 0$
Activity A: $TF(1, 3) = LT(3) - ET(1) - 6 = 3$
Activity D: $TF(3, 4) = LT(4) - ET(3) - 7 = 0$
Activity C: $TF(3, 5) = LT(5) - ET(3) - 8 = 9$
Activity E: $TF(4, 5) = LT(5) - ET(4) - 10 = 0$
Activity F: $TF(5, 6) = LT(6) - ET(5) - 12 = 0$
Dummy activity: $TF(2, 3) = LT(3) - ET(2) - 0 = 0$

Find a Critical Path Definitions Any activity with a total float of zero is a critical activity.

A path from node 1 to the finish node that consists entirely of critical activities is called a critical path.

- In our example, activities B, D, E, F, and the dummy activity are critical activities and the path 1-2--3-4-5-6 is the critical path.
- We can use the minimal path algorithm to find the critical path with $TF(i, j)$ as the distance.
- It is possible for a network to have more than one critical path.
- A critical path in any project network is the longest path from the start node to the finish node.
- Any delay in the duration of a critical activity will delay the completion of the project, so it is advisable to monitor closely the completion of critical activities.



Node 1 = starting node
Node 6 = finish node

Free float

The free float of the activity corresponding to arc (i, j), denoted by $FF(i, j)$, is the amount by which the starting time of the activity corresponding to arc (i, j) (or the duration of the activity) can be delayed without delaying the start of any later activity beyond its earliest possible starting time. So, $ET(i) + t_{ij} + FF(i, j) \leq ET(j)$, and hence

$$FF(i, j) = ET(j) - ET(i) - t_{ij}.$$

In our example, we have the following.

Activity B: $FF(1, 2) = 9 - 0 - 9 = 0$
Activity A: $FF(1, 3) = 9 - 0 - 6 = 3$
Activity D: $FF(3, 4) = 16 - 9 - 7 = 0$
Activity C: $FF(3, 5) = 26 - 9 - 8 = 9$
Activity E: $FF(4, 5) = 26 - 16 - 10 = 0$
Activity F: $FF(5, 6) = 38 - 26 - 12 = 0$

Using Linear Programming to Find a Critical Path

$$\begin{array}{lll}
\min z = x_6 - x_1 & & \\
\text{s.t.} & x_3 \geq x_1 + 6 & \text{(Arc (1, 3) constraint)} \\
& x_2 \geq x_1 + 9 & \text{(Arc (1, 2) constraint)} \\
& x_5 \geq x_3 + 8 & \text{(Arc (3, 5) constraint)} \\
& x_4 \geq x_3 + 7 & \text{(Arc (3, 4) constraint)} \\
& x_5 \geq x_4 + 10 & \text{(Arc (4, 5) constraint)} \\
& x_6 \geq x_5 + 12 & \text{(Arc (5, 6) constraint)} \\
& x_3 \geq x_2 & \text{(Arc (2, 3) constraint)} \\
\text{All variables urs} & &
\end{array}$$

An optimal solution to this LP is $z = 38$, $x_1 = 0$, $x_2 = 9$, $x_3 = 9$, $x_4 = 16$, $x_5 = 26$, and $x_6 = 38$. This indicates that the project can be completed in 38 days.