Forward Pass and backward pass Techniques

Activity-on-node network:

Critical path method uses **activity-on-node networks** where activities are represented as nodes and the links between nodes represent precedence (or sequencing) requirements.

Labeling conventions

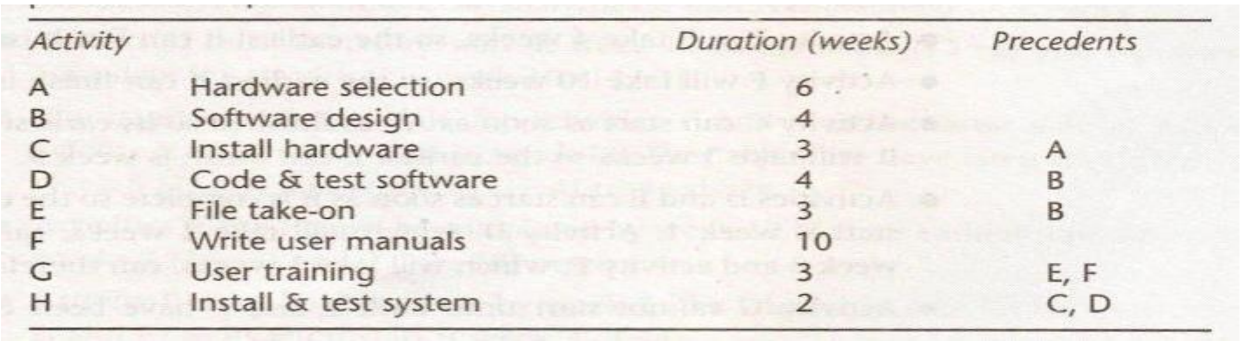
- There are a number of differing conventions that have been adopted for entering information on an activity-on-node network. One of the more common conventions for labelling nodes, and the one adopted here, is shown on the left.
- The activity label is usually a code developed to uniquely identify the activity and may incorporate a project code
- The activity description will normally be a brief activity name such as 'Test take-on module'.

Earliest start	Duration	Earliest finish
Activity label, activity description		
Latest start	Float	Latest finish

Adding the time dimension

- Having created the logical network model indicating what needs to be done and the interrelationships between those activities, we are now ready to start thinking about when each activity should be undertaken.
- The critical path approach is concerned with two primary objectives: planning the project in such a way that it is completed as quickly as possible: and identifying those activities where a delay in their execution is likely to affect the overall end date of the project or later activities' start dates.
- The method requires that for each activity we have an estimate of its duration. The network is then analyzed by carrying out a **forward pass**, to calculate the earliest dates at which activities may commence and the project be completed, and a **backward pass**, to calculate the latest start dates for activities and the **critical path**.
- In practice we would use a software application to carry out these calculations for anything but the smallest of projects. It is important, though, that we understand how the calculations are carried out in order to interpret the results correctly and understand the limitations of the method.

- The description and example that follow use the small example project outlined in Table — a project composed of eight activities whose durations have been estimated as shown in the table



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

Table: An Example project specification with estimated activity durations and precedence requirements

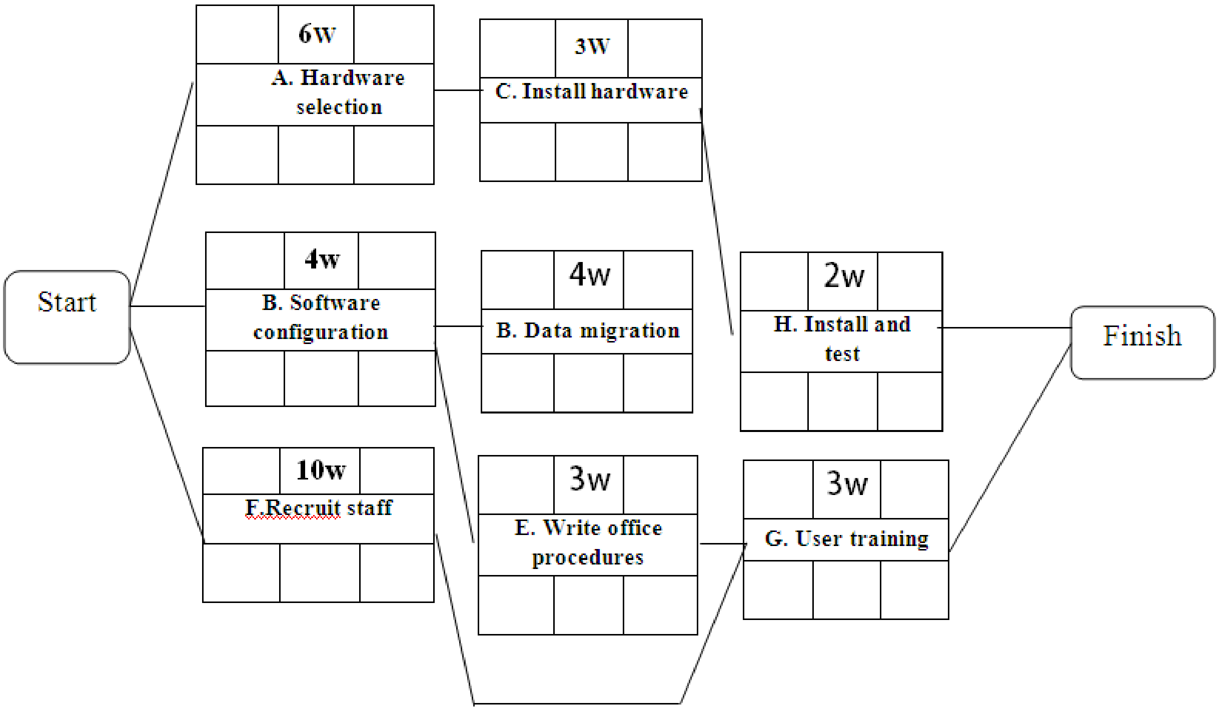


Figure: The precedence network for the example project

The forward pass

- The forward pass is carried out to calculate the earliest dates on which each activity may be started and completed.

- Where an actual start date is known, the calculations may be carried out using actual dates. Alternatively we can use day or week numbers and that is the approach we shall adopt here. By convention, dates indicate the end of a period and the project is therefore shown as starting at the end **of** week zero (or the beginning of week 1).
- The forward pass and the calculation of earliest start dates is calculated according to the following reasoning.
- Activities A, B and F may start immediately, so the earliest date for their start is zero.
- Activity A will take 6 weeks, so the earliest it can finish is week 6.
- Activity B will take 4 weeks, so the earliest it can finish is week 4.
- Activity F will take 10 weeks, so the earliest it can finish is week 10.
- Activity C can start as soon as A has finished so its earliest start date is week 6. It will take 3 weeks so the earliest it can finish is week 9.

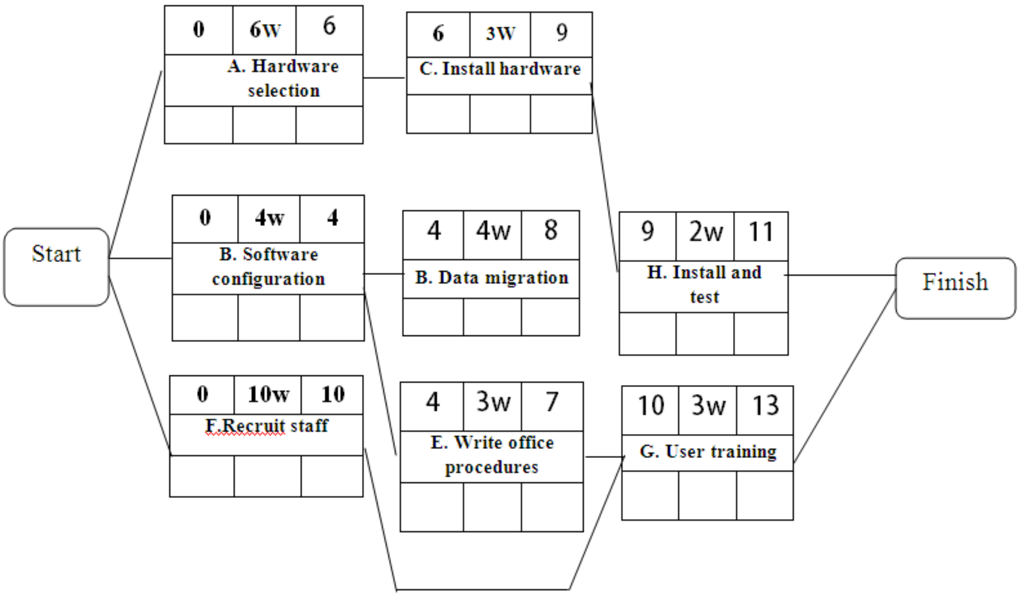


Figure: The network after the forward pass

- Activities D and E can start as soon as B is complete so the earliest they can each start is week 4. Activity D, which will take 4 weeks, can therefore finish by week 8 and activity E, which will take 3 weeks, can therefore finish by week 7.
- Activity G cannot start until both E and F have been completed. It cannot therefore start until week 10 — the later of weeks 7 (for activity E) and 10 (for activity F). It takes 3 weeks and finishes in week 13.
- Similarly, Activity H cannot start until week 9 — the later of the two earliest finished dates for the preceding activities C and a
- The project will be complete when both activities H and G have been completed. Thus the

earliest project completion date will be the later of weeks 11 and 13— that is, week 13. The results of the forward pass are shown in Figure.

The backward pass

- The second stage in the analysis of a critical path network is to carry out a backward pass to calculate the latest date at which each activity may be started and finished without delaying the end date of the project. In calculating the latest dates, we assume that the latest finish date for the project is the same as the earliest finish date — that is, we wish to complete the project as early as possible.
- Figure 6.16 illustrates our network alter carrying out the backward pass.
- The latest activity dates are calculated as follows.
- The latest completion date for activities G and 1-1 is assumed to be week 13.
- Activity H must therefore start at week 11 at the latest (13-2) and the lateststart date for activity G is week 10 (13-3).
- The latest completion date for activities C and D is the latest date at which
- activity H must start — that is. week 11. They therefore have latest start dates of week 8 (11- 3) and week 7 (11-4) respectively.
- Activities E and F must be completed by week 10 so their earliest start dates are weeks 7 (10-3) and 0 (10-10 respectively.
- Activity B must be completed by week 7 the latest start date for both activities D and El soits latest start is week 3 (7-4).
- Activity A must be completed by week 8 (the latest start date for activity C) so its latest startis week 2 (8-6).

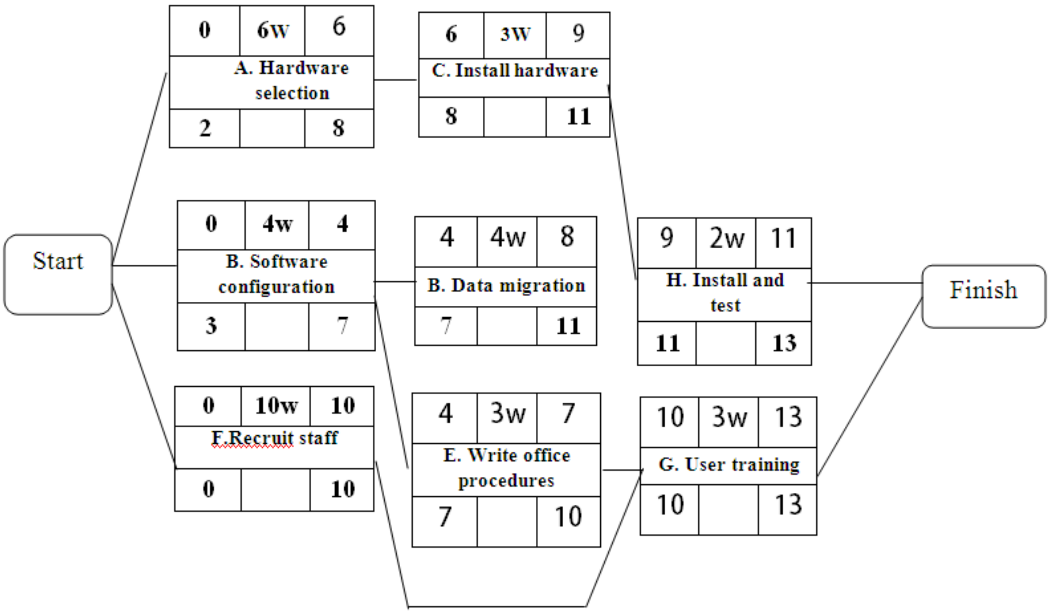


Figure: The network after the backward pass

The latest start date for the project start is the earliest of the latest start dates for activities A. B and F. This is week zero. This is, of course, not very surprising since **it** tells us that if the project does not start on time it won't finish on time.