Cambridge International AS and A Level Computer Science

9608

For examination from 2017

Topic 4.4.3 Project management

PERT and GANTT charts

Contents

Introduction

How to use this guide

The aim of this guide is to facilitate your teaching of the objectives relating to Program Evaluation Review Technique (PERT) and GANTT charts in Cambridge International AS and A Level Computer Science, syllabus topic 4.4.3 Project management. This is part of topic 4.4 Software development. The guidance and activities in this resource are designed to help teachers devise programmes of study which provide teaching time devoted to theory work as well as activities that consolidate learning.

Section 1, and lists some key terms used in this topic and their definitions. Section 2 introduces PERT and GANTT charts, including some worked examples. Section 3 provides links to online resources. Section 4 gives ideas for class and homework activities.

Learning objectives

Using this document should help you guide learners in the following syllabus learning objectives:

- show understanding of project planning techniques including the use of GANTT and Program Evaluation Review Technique (PERT) charts
- describe the information that GANTT and PERT charts provide
- construct and edit GANTT and PERT charts

Prior knowledge

Before you begin teaching this topic you should:

- understand how tools available for project management (PERT and GANTT) can assist a project manager in the delivery of a project on-time and on-budget
- understand how a PERT chart displays project tasks, durations and dependencies and its advantages and disadvantages
- understand how a GANTT chart displays project tasks, durations and dependencies and its advantages and disadvantages
- understand the different task durations used as inputs to GANTT charts
- be able to calculate the estimated time
- know how to construct GANTT and PERT charts
- understand how to identify the critical path and calculate slack time for a project

1. Key terms

Word/phrase	Meaning
backward pass	The process to determine the late start or late finish times for activities in the critical path method
critical path method	A mathematical modelling technique used to schedule a set of tasks. The model allows the longest path (the critical path) to be determined which gives the minimum time required for project completion
earliest finish time (EF)	The earliest start time for the activity plus the time required to complete the activity (the earliest an activity can be completed)
earliest start time (ES)	The earliest time an activity can start once the previous dependent activities are completed
expected time, T_E	Calculated using O, M and P, the expected time is: $T_E = \frac{O+4M+P}{6}$
forward pass	The process to determine the early start or early finish times for activities in the critical path method
GANTT chart	A diagram that shows each task in a project as a block of calendar time. Each block is labelled with the task description and the amount of time the block represents.
latest finish time (LF)	The latest time an activity can be completed without delaying the entire project.
latest start time (LS)	The latest finish time minus the time required to complete the activity
most likely time (M)	The project manager should have a high degree of certainty that a task will be completed within this time
optimistic time (O)	The shortest time that an activity is likely to take – there will be a small probability (less than 5%) of this happening
pessimistic time (P)	The project manager should be almost certain (e.g. 99%) that the task will be completed within this time
Program Evaluation Review Technique (PERT) chart	An implementation of the Critical Path Method where the time for each task is estimated by using a weighted average based on a probabilistic estimation of the optimal, most likely and a pessimistic time to complete the task
slack	The number of spare / float weeks

2. Theory and worked examples

2.1. What are PERT charts?

Program Evaluation Review Technique (PERT) charts show each task in a project as a node. Dependencies between tasks (e.g. where one task requires another one to be completed before it can start) are clearly shown by interconnections between the task nodes. PERT charts also show timing information for each task. PERT charts are similar to the critical path method (CPM) which identifies the longest path through the project, and therefore the minimum time for the project to be completed. In the example below, the critical path is shown by the red arrows.

PERT charts try to make a realistic estimate by considering three scenarios and then generating an Expected Time as a weighted average:
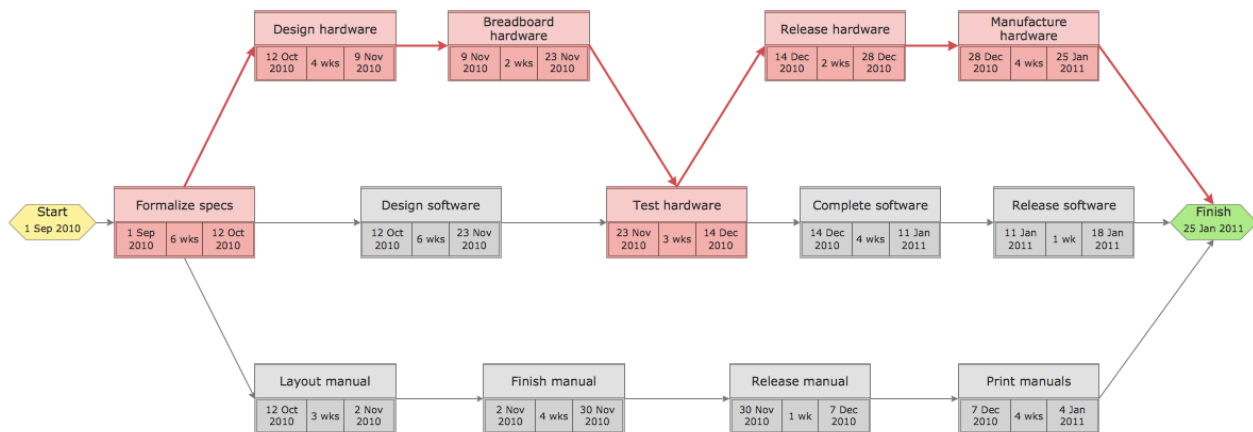
Project managers can use PERT charts to:

- set a realistic timetable for project completion
- make sure focus is maintained on the most critical tasks for the critical path – since the path leads to the minimum time the project requires, any delays to these tasks will result in a delay to the overall project
- identify tasks that need to be shortened if the overall project time needs to be reduced
- identify tasks that can be carried out simultaneously
- identify slack time where certain tasks are not as time-critical to the overall deadline.

Whereas CPM uses a single estimate for task duration, the PERT method requires the project manager to supply three estimates to make the estimates as accurate as possible. The three estimates are:

- Optimistic Time (O) – the shortest time that the activity is likely to take. There will be a small probability (less than 5%) of this happening
- Most Likely Time (M) – the project manager should have a high degree of certainty that the task will be completed within this time
- Pessimistic Time (P) – the project manager should be almost certain (e.g. 99%) that the task will be completed within this time

The 'expected time' T_E is then calculated as $T_E = \frac{O+4M+P}{6}$.

2.2. PERT chart – worked example

2.2.1 Inputs

The following table shows the tasks, dependencies, and estimated times a project manager might input to a PERT chart for a software development project (estimates are used for illustrative purposes only), and the calculated Expected Times:

Task Identifier	Task Description	Predecessor Task(s)	Optimistic Time (O)	Most Likely Time (M)	Pessimistic Time (P)	Expected Time (T_E)
A	Establish project		4	5	12	**6**
B	Establish customer requirements	A	2	3	4	**3**
C	Produce software specification documents	B	6	8	22	**10**
D	Write test plans	C	4	6	8	**6**
E	Write code	C	3	4	5	**4**
F	Developer testing	E	2	4	6	**4**
G	System testing	D, F	2	3	4	**3**
H	Write manuals	C	5	7	15	**8**

2.2.2 Network diagram

The network diagram is based on the tasks and their dependencies (predecessor tasks). Task A has no predecessor, and therefore starts the project on the left. Task B has only task A as a predecessor, and is therefore the next task. The diagram shows that tasks D, E and H (writing test plans, code and manuals) all have task C (specification) as a predecessor, and can therefore be carried out simultaneously.
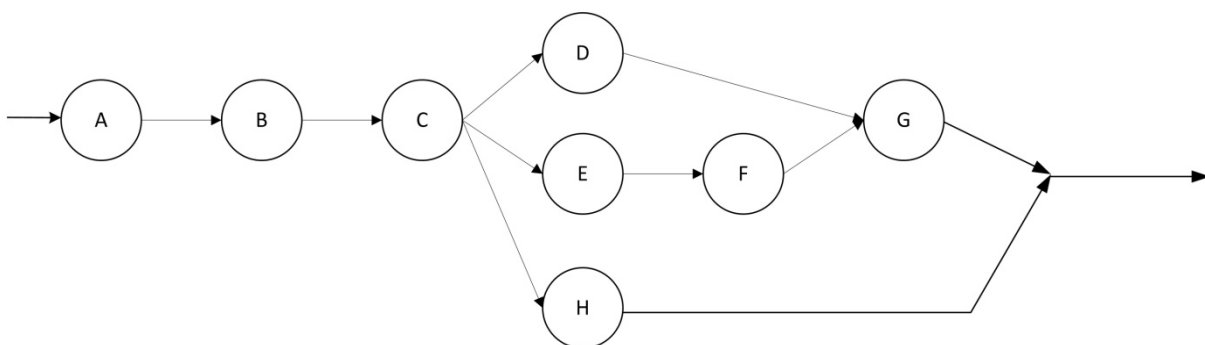
2.2.3 Identify the critical path

The critical path is determined by using the estimated times to work out the earliest start (ES) and finish (EF) times (forward pass), and latest start (LS) and finish (LF) times (backward pass), and identifying the tasks where ES and LS are equal.
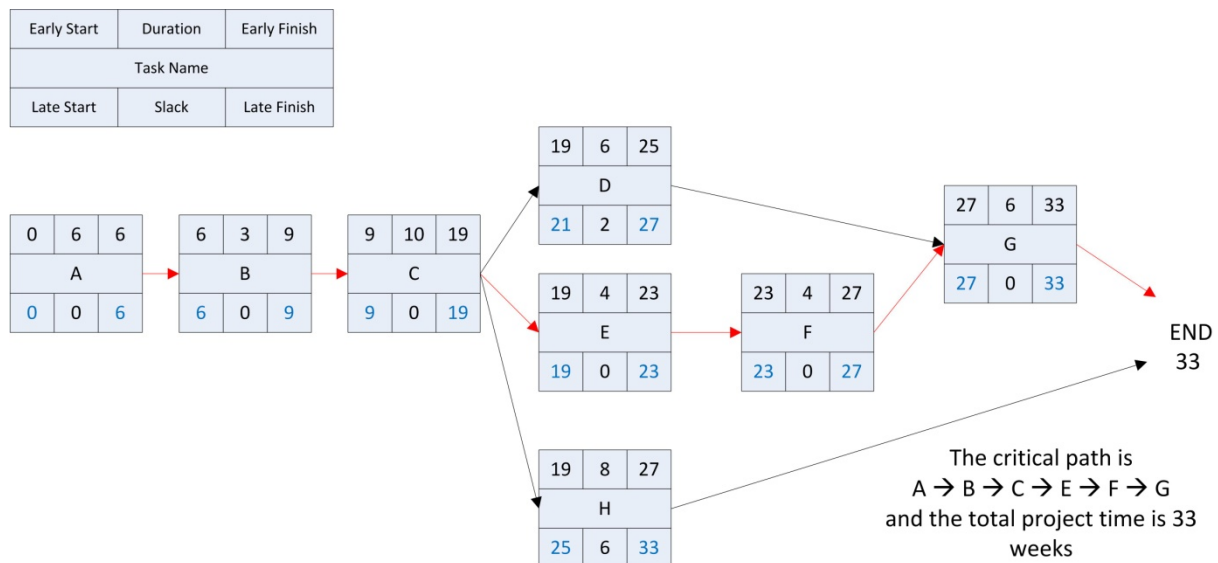
Step1: Forward pass

This is done by working from left to right across the chart. The ES of task A is week 0, and the EF is 6 since this is the ES plus the duration (estimated time). The earliest that task B can begin is week 6 (since task A must be complete) so its ES is 6, and EF is 6 + 3 = 9 . Note that task G has an ES of 27 because it is dependent on both task D and task F being complete first, and the earliest that both tasks will be completed is week 27.

Early Start	Duration	Early Finish
	Task Name	
Late Start	Slack	Late Finish

19	6	25
	D	
21	2	27

0	6	6
	A	
0	0	6

6	3	9
	B	
6	0	9

9	10	19
	C	
9	0	19

19	4	23
	E	
19	0	23

23	4	27
	F	
23	0	27

27	6	33
	G	
27	0	33

19	8	27
	H	
25	6	33

END
33

The critical path is
A → B → C → E → F → G
and the total project time is 33 weeks

Step 2: Backward pass

Working from right to left, the Latest Finish Time (LF) and Latest Start Time (LS) can be filled in. The final time is 33 weeks, so the LF of task H must be 33. Because this task has a duration of eight weeks it means it has an LS of 33 – 8 = week 25.

Note that for some tasks, such as Task C, there are different possibilities. To calculate the LF of Task C we need to look at the latest start times for Tasks D, E and H which are dependent on it. The latest time task C can finish (LF) must be 19 weeks because Task E has to start at this point.

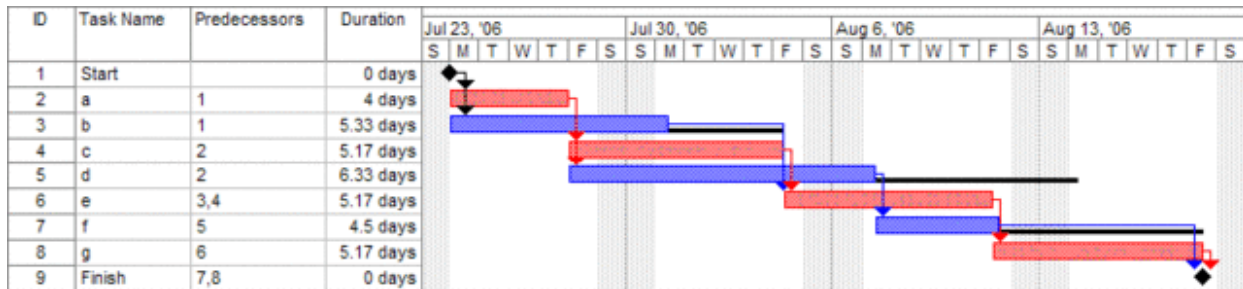Step 3: Critical path and slack

The critical path is identified by finding the nodes where ES = LS. These are nodes for which there is no 'slack' or spare time for a task in the project.

The slack is calculated as LS – ES, e.g. Task H has 6 weeks' slack, and could be scheduled to start sometime between week 19 and week 25 depending on factors such as the resources available to a task on the critical path.

2.3. What are GANTT charts?

GANTT charts display the tasks in a project as a box or line showing the calendar duration of the task on the horizontal axis (the horizontal length of the task box is proportional to the task duration). Tasks are normally arranged in date order on the vertical axis. The time relation of all tasks to each other (for example, tasks carried out simultaneously) is therefore clearly apparent in a GANTT chart. The project status can be easily determined at intermediate dates in the project, and progress of individual tasks can be shown by filling in the task boxes. Unlike PERT charts, GANTT charts do not show the critical path, however, dependencies between tasks can be indicated by lines linking tasks.

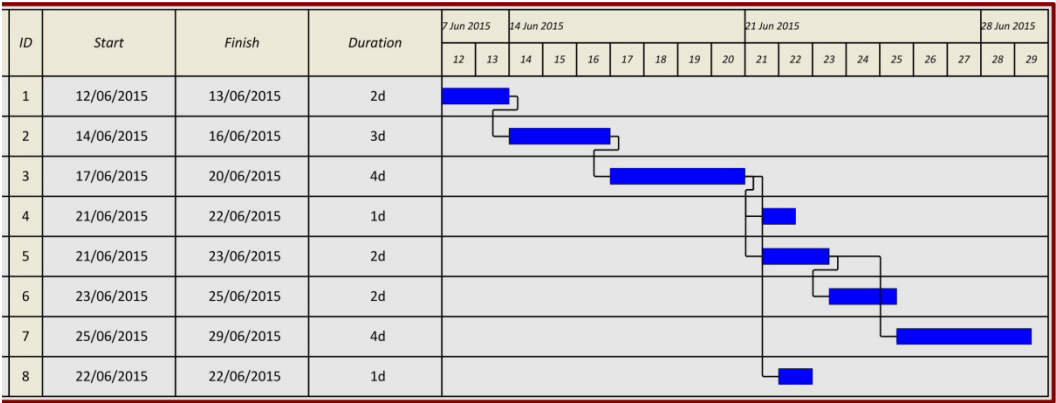| ID	Task Name	Predecessors	Duration	Jul 23, '06	Jul 30, '06	Aug 6, '06	Aug 13, '06
1	Start		0 days				
2	a	1	4 days				
3	b	1	5.33 days				
4	c	2	5.17 days				
5	d	2	6.33 days				
6	e	3,4	5.17 days				
7	f	5	4.5 days				
8	g	6	5.17 days				
9	Finish	7,8	0 days				

2.4. GANTT charts – worked example

The following table shows the tasks, dependencies, and estimated times a project manager might input to a basic GANTT chart for a software development project.

Project start date: 12 June 2015			
Task Identifier	Task Description	Predecessor Task(s)	Time (days)
1	Establish project	-	2
2	Establish customer requirements	1	3
3	Produce software specification documents	2	4
4	Write test plans	3	1
5	Write code	3	2
6	Developer testing	5	2
7	System testing	4, 6	4
8	Write customer documentation	3	3

Task 1 has no predecessors, and can thus start on 12 June. The GANTT chart shows the task as a box starting on 12 June and finishing on 13 June on the horizontal access. Task 2 requires Task 1 to be completed, and the duration is three days, so the box covers the dates 14 to 16 June. The line from the finish of Task 1 to the start of Task 2 indicates the dependency. Note that Tasks 4, 5 and 8 all require Task 3 to be completed, and have no other dependencies, so these all start on the same date. The chart below show all seven days of the week, but often, weekend days are excluded.

ID	Start	Finish	Duration	7 Jun 2015		14 Jun 2015							21 Jun 2015							28 Jun 2015	
				12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
1	12/06/2015	13/06/2015	2d																		
2	14/06/2015	16/06/2015	3d																		
3	17/06/2015	20/06/2015	4d																		
4	21/06/2015	22/06/2015	1d																		
5	21/06/2015	23/06/2015	2d																		
6	23/06/2015	25/06/2015	2d																		
7	25/06/2015	29/06/2015	4d																		
8	22/06/2015	22/06/2015	1d																		

3. Online resources

There are numerous resources available that introduce and contain examples of GANTT and PERT charts. The following resources are ones that you and your users should consider using.

The content of websites is dynamic and constantly changing. Schools are strongly advised to check each site for content and accessibility prior to using it with learners. Cambridge International Examinations is not responsible for the accuracy or content of information contained in these sites. The inclusion of a link to an external website should not be understood to be an endorsement of that website or the site's owners (or their products/services).

3.1. Websites

3.1.1 General

Overview of PERT, GANTT and CPM: why they are needed, advantages and disadvantages of each.

http://www.southampton.ac.uk/~jps7/Aircraft%20Design%20Resources/project%20management/ganttv1.pdf

Covers most of the topic – Gantt charts, PERT charts, and includes a worked example of forward and backward pass to find the critical path, although not in the context of software development projects.

http://www.teach-ict.com/as_a2_ict_new/ocr/A2_G063/331_systems_cycle/planning_tools/miniweb/pg9.htm

Explanation of GANTT and PERT charts, including step-by-step instructions on how to construct them and the benefits of each.

http://www.umsl.edu/~sauterv/analysis/488_f02_papers/ProjMgmt.html

3.1.2 GANTT

What a Gantt chart is, and how to create them.

http://www.gantt.com/index.htm

3.1.3 PERT

Comprehensive look at critical path and PERT.

https://www.smartsheet.com/critical-path-method

3.2. Videos

3.2.1 General

Project management (eight minutes) – how a project manager can use PERT and GANTT charts for project management. This is a generic video, but gives a good overview of the responsibilities of a project manager.

https://www.youtube.com/watch?v=qkuUBcmmBpk&feature=youtu.be&feature=iv&src_vid=Es4IKP_pQh0&annotation_id=annotation_2969299389

3.2.2 GANTT

Brief summary of GANTT chart purpose (three minutes)

https://www.youtube.com/watch?v=VIm8yWpBxFA

3.2.3 PERT

The following videos are from an hour-long lecture showing how to construct a PERT chart. It goes through the principles of CPM and PERT. Part 4 is beyond A Level and starts to look at the statistical variables / expectations that can be applied to the models created.

- Part 1 https://www.youtube.com/watch?v=LdRZN5o08eM
- Part 2 https://www.youtube.com/watch?v=d6-bb9oDsSA
- Part 3 https://www.youtube.com/watch?v=Os1XRKt8ABA
- Part 4 https://www.youtube.com/watch?v=wW1uU_UsARY

3.3. Software

Microsoft Visio has a number of templates including PERT and GANTT tools

GanttProject is a free package that allows professional quality GANTT charts to be developed

http://www.ganttproject.biz/

4. Class and homework activities

To cover the learning objectives in this topic, learners will need to practise the following:

- how to construct a PERT / GANTT chart for a set of tasks given the tasks, their relationship (e.g. successor / predecessor) and the time taken for each task
- how to interpret PERT / GANTT charts to determine the total time for completion, potential blockages and slack periods

4.1. Homework questions

The following activities are suggested to consolidate your students' learning after they have studied this topic:

1. The probable times (in days) for a number of tasks in a software development project have been estimated. Calculate the weighted expected time, T_E, that would be used for estimating the overall task time in a PERT chart.

Task	Description	Optimistic O	Most Likely M	Pessimistic P	Time Expected T_E
A	Produce software specification	4	7	10	
B	Write test plans	7	23	27	
C	Software design	11	14	17	
D	Software code	10	22	28	
E	Developer test	7	23	27	
F	Software test	22	29	30	

2. The expected duration of each task has been calculated. The task dependencies are shown in the table below.

Task	Description	Order	Duration
A	Produce software specification	To be completed first	1 week

B	Write test plans	Start when A is complete	3 weeks
C	Software design	Start when A is complete	2 weeks
D	Software code	Start when B is complete	3 weeks
E	Developer test	Start when C is complete	3 weeks
F	Software test	Start when D and E are complete	4 weeks

    a. Construct a PERT chart for the project.

    b. Identify the critical path and hence the minimum time to completion.

    c. Describe where there is slack time in the system.

3. Construct a GANTT chart for the software development project. Assume that the first task starts on 17th October 2015.
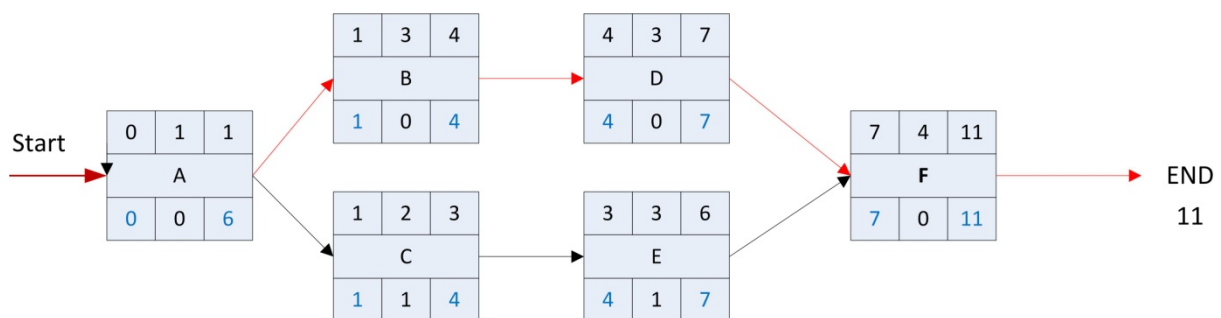
4.2. Homework solutions

1. The time expected is calculated as $T_E = \frac{O+4M+P}{6}$

Task	Description	Optimistic O	Most Likely M	Pessimistic P	Time Expected T_E
A	Produce software specification	4	7	10	**7**
B	Write test plans	7	23	27	**21**
C	Software design	11	14	17	**14**
D	Software code	10	22	28	**21**
E	Developer test	7	23	27	**21**
F	Software test	22	29	30	**28**

2. PERT Chart
   a.

   b. The Critical Path is A → B → D → F and it will take a time of 11 weeks to complete the project at best.
   c. Tasks C and E both have 1 week of slack time since the Latest Start Times are both one week more than the Earliest Start Times.

3. (Note – the task description should also be displayed, as well as the task ID.)

ID	Start	Finish	Duration	Jun 2015			Jul 2015				Aug 2015				
				14/6	21/6	28/6	5/7	12/7	19/7	26/7	2/8	9/8	16/8	23/8	30/8
1	12/06/2015	18/06/2015	1w												
2	19/06/2015	09/07/2015	3w												
3	19/06/2015	02/07/2015	2w												
4	03/07/2015	23/07/2015	3w												
5	10/07/2015	30/07/2015	3w												
6	31/07/2015	27/08/2015	4w												