

(<https://hackmd.io> /  DASS Qui...

/@redfox0/  
 [Edit](#) [By3ca49I-x/edit?](#)  
both)     

# DASS Quiz-1 Answer Key

---

## TA List

- Set A: Narain, Aniket, Nikhilesh
- Set B: Yashaswinee, Agrim, George
- Set C: Neel, Divijh, Agyeya

## Set A

---

### Question 1 (5 Marks)

1.) Explain which development model the team should adopt and why. (1.5)

- Correct Ans: Prototyping model combined with evolutionary design (1.5). If someone has written spiral they get (1). If someone has written one of prototype/evolutionary/iterative- incremental, they get (0.75). No marks will be awarded for Agile Dev Model. Agile is not a dev model, but rather a philosophy (read the agile manifesto), if someone has written scrum, they get 0, scrum is a process and not a model.

2. Describe how design and code quality should be managed over time as new features are added. (1.5)

- Correct Ans: As the system grows, code quality should be maintained through refactoring (0.75) and adherence to the open closed principle (0.75). The open closed principle ensures that new functionality can be added through extension rather than modification, reducing the risk of breaking existing code. If someone has written SOLID, they also get (0.75). Open Close Principle is open for extension, closed for modification.

3. Explain how testing and integration practices can reduce risk when changes are made frequently. (1)

- Correct Ans: To manage frequent changes safely, the team should rely on continuous integration (0.5) and regression testing (0.5)

OR

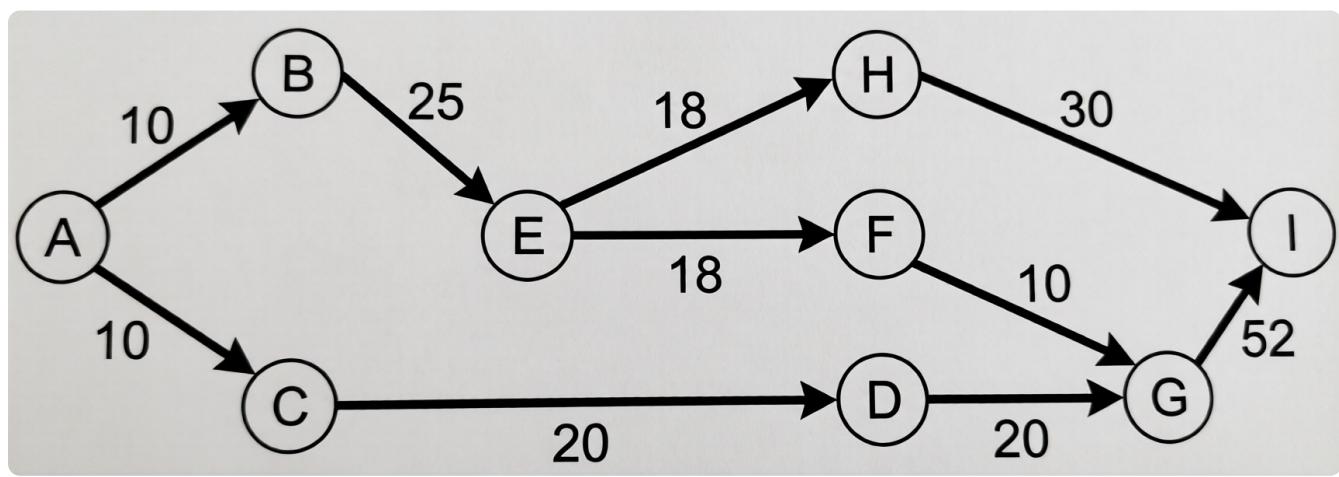
early & continuous feedback; ensures new code doesn't break old functions; automated build (1), 0.25 for each point individually and 0.25 if all 3 are written.

4. Discuss how scheduling concepts can help ensure the project is not delayed, even when some tasks change. (1)

- Correct Ans: From a scheduling perspective, the team should use the critical path method (0.5) to identify tasks that directly affect project completion. Understanding slack time (0.5) allows non-critical tasks to be delayed without impacting the overall timeline, providing flexibility when requirements change.

## Question 2 (6 Marks)

a) (1 mark)



b) (1 marks)

Critical Path: A → B → E → F → G → I

c) (2 marks)

Slack Time for D = 13 hours

### Calculation:

Expected Time for Critical Path = 115 hours

Time taken by path A → C → D = 30 hours

∴ Available time for path D → G → I =  $115 - 30 = 85$  hours

Let, time taken by D = t,

Then,  $t + 52 = 85$

$$\Rightarrow t = 33 \text{ hours}$$

Expected Time of D = 20 hours

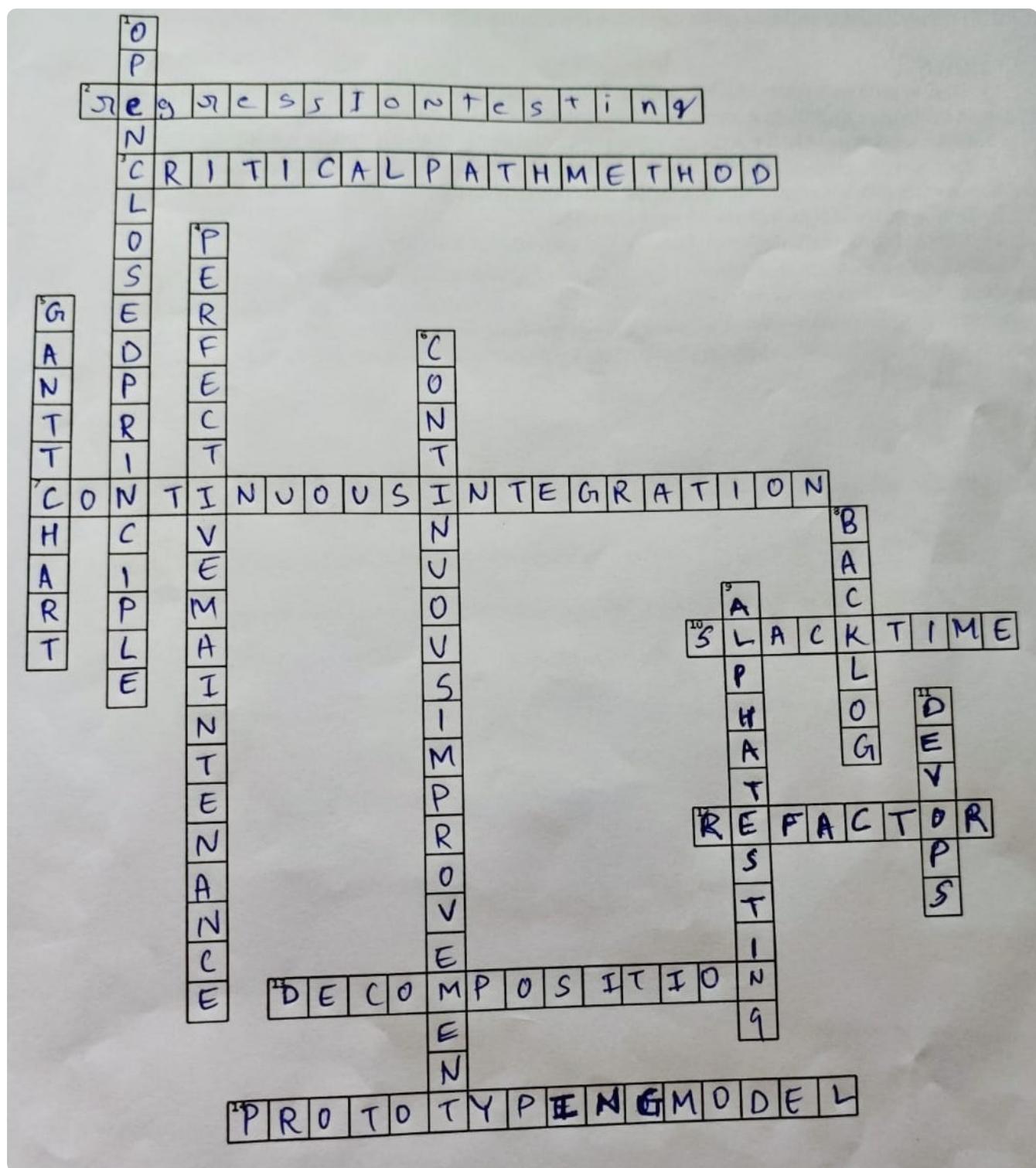
∴ Slack Time for D =  $33 - 20 = 13$  hours

d) (2 mark)

Slack Time for E = 0 hours

**Reason:** As E is on the critical path, there is no slack time for E.

### Question 3 (14 Marks)



#### ACROSS

2. REGRESSIONTESTING
3. CRITICALPATHMETHOD
4. CONTINUOUSINTEGRATION

5. SLACKTIME
6. REFACTOR
7. DECOMPOSITION
8. PROTOTYPINGMODEL

## DOWN

1. OPENCLOSEDPRINCIPLE
2. PERFECTIVE MAINTENANCE
3. GANTTCHART
4. CONTINOUSIMPROVEMENT
5. BACKLOG
6. ALPHATESTING
7. DEVOPS

## Set B

---

### Question 1 (5 Marks)

#### a. (1.5 marks).

Main issue lies in inefficient processes rather than effort, highlighting the need for process optimization.

Current process is iterative, needs continuous feedback loop and lacks visibility [0.5]. To address these problems, adopting process models like prototype [0.5] allows early feedbacks and exposes problems early[0.5].

#### b. (1.5 marks)

Current system breakage indicates rigid design and tightly coupled design [0.5]. Refactoring[0.5], along with Agile practices can make the system flexible and easier to adapt to changes. [0.5]

[Design Practices like SOLID, GRASP can make the system more resilient allowing new features to be added without much breakage.]

#### c. (1 mark)

Early detection of problems can be achieved through continuous integration[0.5]

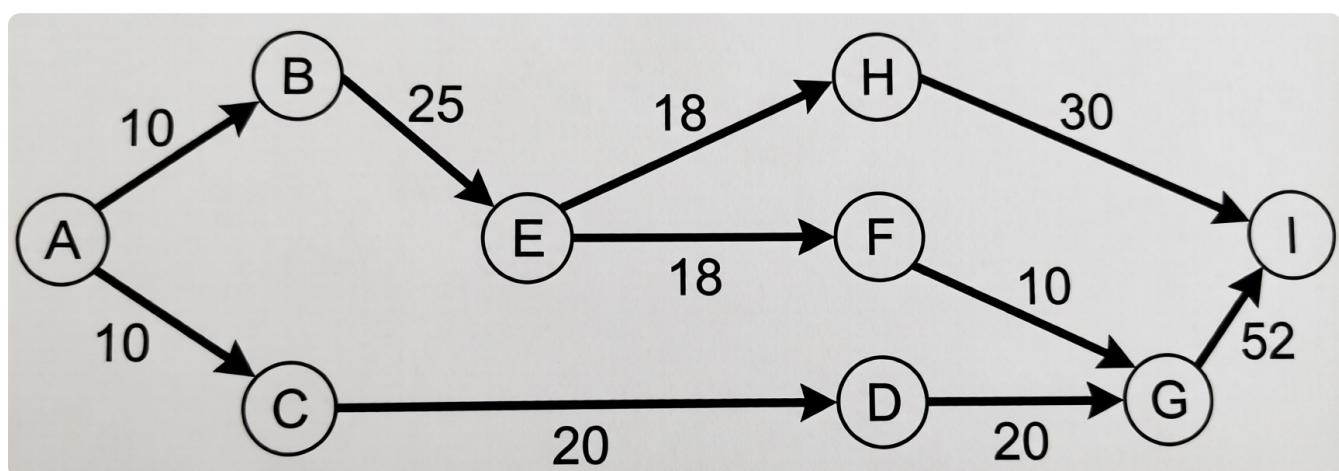
combined with regression testing prevents any old functionality from breaking[0.5].

#### d. (1 mark)

Improved scheduling visibility can be achieved using a Gantt chart together with the critical path method. While the Gantt chart provides a clear visual overview of progress[0.5], the critical path method helps identify tasks that directly impact the project's completion date[0.5]. Understanding slack time allows the team to focus attention on critical tasks while using flexibility in non-critical ones to manage delays more effectively.

### Question 2 (6 Marks)

a) (1 mark)



b) (1 mark)

Critical Path:  $A \rightarrow B \rightarrow E \rightarrow F \rightarrow G \rightarrow I$

c) (2 marks)

Slack Time for D = 13 hours

#### Calculation:

Expected Time for Critical Path = 115 hours

Time taken by path  $A \rightarrow C \rightarrow D$  = 30 hours

$\therefore$  Available time for path  $D \rightarrow G \rightarrow I$  =  $115 - 30 = 85$  hours

Let, time taken by D = t,

Then,  $t + 52 = 85$

$\Rightarrow t = 33$  hours

Expected Time of D = 20 hours

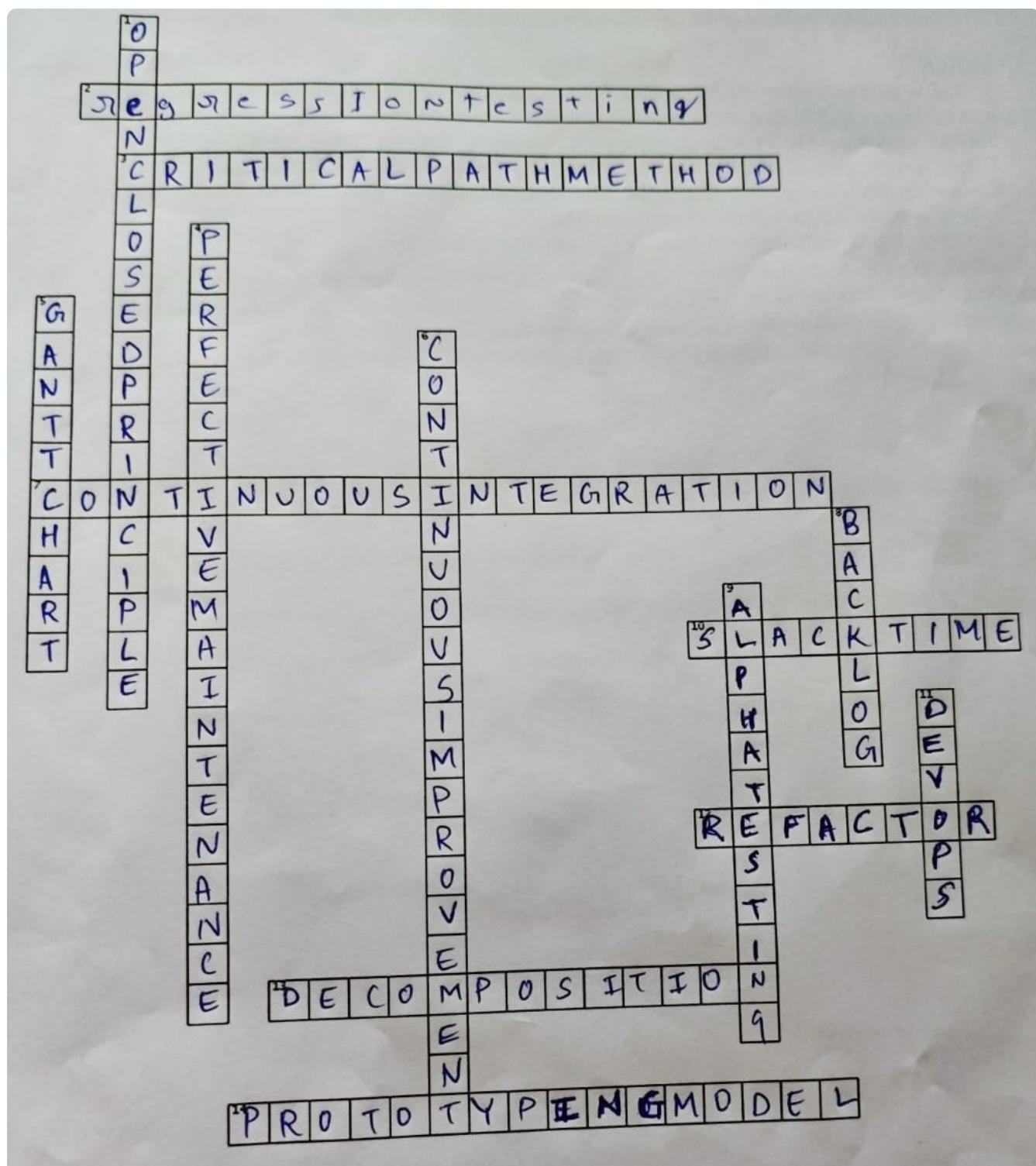
$\therefore$  Slack Time for D =  $33 - 20 = 13$  hours

d) (2 marks)

Slack Time for E = 0 hours

**Reason:** As E is on the critical path, there is no slack time for E.

### Question 3 (14 Marks)



#### ACROSS

2. REGRESSION TESTING

3. CRITICALPATHMETHOD
4. CONTINUOUSINTEGRATION
5. SLACKTIME
6. REFACTOR
7. DECOMPOSITION
8. PROTOTYPINGMODEL

## DOWN

1. OPENCLOSEDPRINCIPLE
2. PERFECTIVE MAINTENANCE
3. GANTTCHART
4. CONTINOUSIMPROVEMENT
5. BACKLOG
6. ALPHATESTING
7. DEVOPS

## Set C

---

### Question 1 (5 Marks)

a) (1.5 marks)

Explain which design and maintenance practices can help the team add new features safely as the system evolves.

To add new features safely as the system grows, the team should adopt evolutionary design supported by regular refactoring and the open closed principle. Evolutionary design allows the system structure to improve gradually instead of relying on a rigid upfront design. Refactoring helps keep the internal structure clean and understandable, while the open closed principle ensures that new functionality can be added by extending existing components rather than modifying them, reducing the likelihood of introducing defects. They should also implement preventive maintenance to stop errors before they occur and perfective maintenance to enhance features carefully, rather than just fixing bugs as they appear.

b) (1 mark)

Describe how integration and testing practices can reduce the risk of late-stage failures.

The risk of late-stage failures can be reduced through continuous integration combined with regression testing. Integration and testing practices to reduce late-stage failures Risk is reduced by performing rigorous unit testing on individual modules before combining them, followed by step-by-step integration where the system is tested at each stage rather than all at once.

c) (1.5 marks)

Discuss how process and workflow improvements can help the team deliver faster without increasing errors.

Faster and safer delivery can be achieved by focusing on process optimization and adopting DevOps practices. Process optimization emphasizes improving workflow efficiency rather than simply increasing effort. DevOps improves coordination between development and operations, making deployments more predictable and reducing release-related issues.

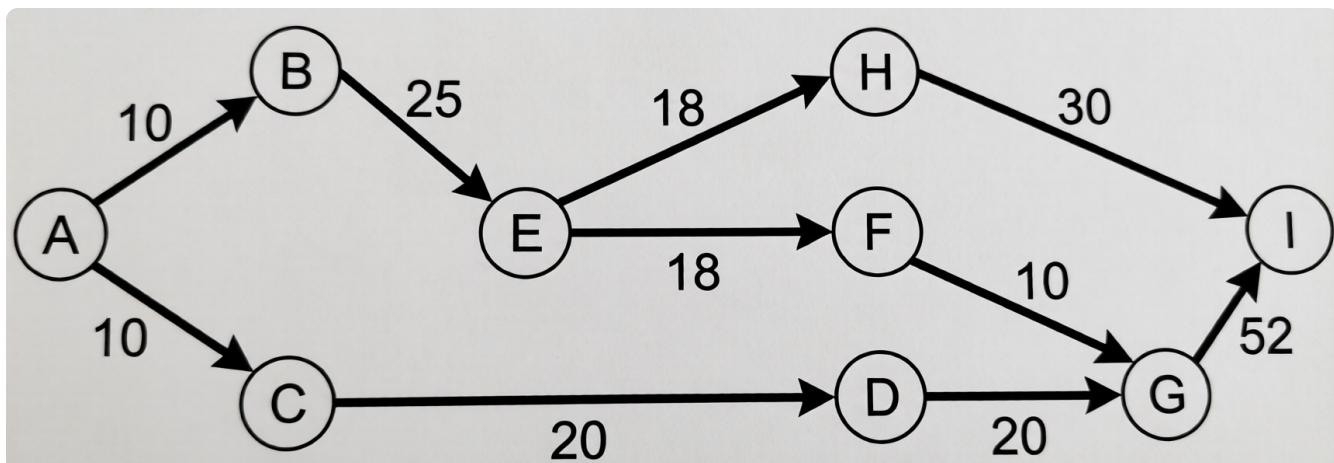
d) (1 mark)

Explain how project scheduling concepts can be used to manage delivery risk when deadlines are tight.

From a scheduling perspective, the team should apply the critical path method to identify tasks that directly affect the project completion date. Understanding slack time allows the team to prioritize critical tasks while using flexibility in non-critical activities to absorb changes. Project scheduling concepts to manage delivery risk Using iterative planning allows estimates to become more accurate over time (convergence) compared to a single upfront session.

## **Question 2 (6 Marks)**

a) (1 mark)



b) (1 mark)

Critical Path:  $A \rightarrow B \rightarrow E \rightarrow F \rightarrow G \rightarrow I$

c) (2 marks)

Slack Time for D = 13 hours

**Calculation:**

Expected Time for Critical Path = 115 hours

Time taken by path  $A \rightarrow C \rightarrow D$  = 30 hours

$\therefore$  Available time for path  $D \rightarrow G \rightarrow I$  =  $115 - 30 = 85$  hours

Let, time taken by D = t,

Then,  $t + 52 = 85$

$$\Rightarrow t = 33 \text{ hours}$$

Expected Time of D = 20 hours

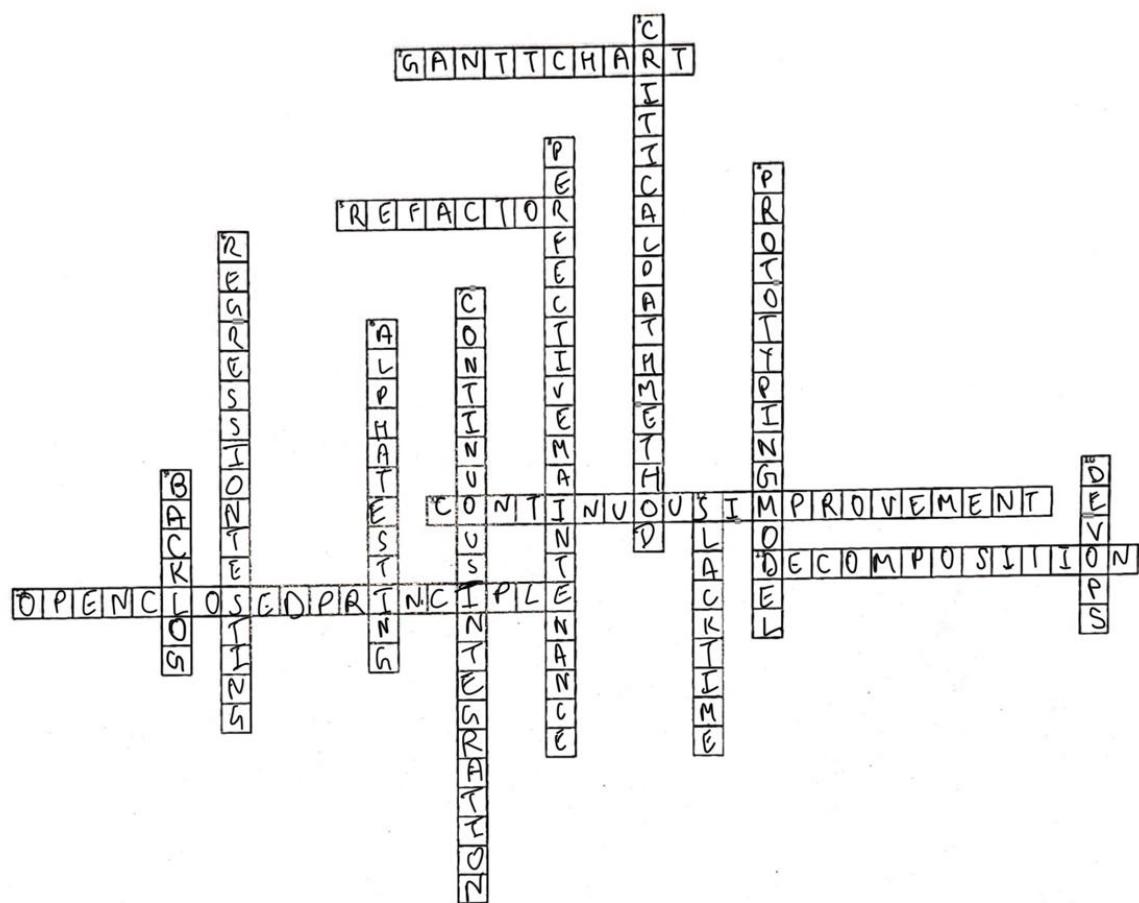
$\therefore$  Slack Time for D =  $33 - 20 = 13$  hours

d) (2 marks)

Slack Time for E = 0 hours

**Reason:** As E is on the critical path, there is no slack time for E.

### Question 3 (14 Marks)

**ACROSS**

2. GANTTCHART
3. REFACTOR
11. CONTINUOUSIMPROVEMENT
13. DECOMPOSITION
14. OPENCLOSEDPRINCIPLE

**DOWN**

1. CRITICALPATHMETHOD
2. PERFECTIVEMAINTENANCE
3. PROTOTYPINGMODEL
4. REGRESSIONTESTING
5. CONTINUOUSINTEGRATION
6. ALPHATESTING
7. BACKLOG
8. DEVOPS
9. SLACKTIME

