



SLIATE

SRI LANKA INSTITUTE OF ADVANCED TECHNOLOGICAL EDUCATION

(Established in the Ministry of Higher Education, vide in Act No. 29 of 1995)

Higher National Diploma in Information Technology

Second Year, Second Semester Examination – 2019

HNDIT 2412- Principles of Software Engineering (Developer Track)

Answer Script

Part A is Compulsory

PART-A

Q1)

- i. What are Generic software products?

Stand-alone systems that are marketed and sold to any customer who wishes to buy them. Examples – PC software such as graphics programs, project management tools.

(03 marks)

- ii. Define the term Software Engineering.

Software engineering is an engineering discipline that is concerned with all aspects of software production from the early stages of system specification through to maintaining the system after it has gone into use

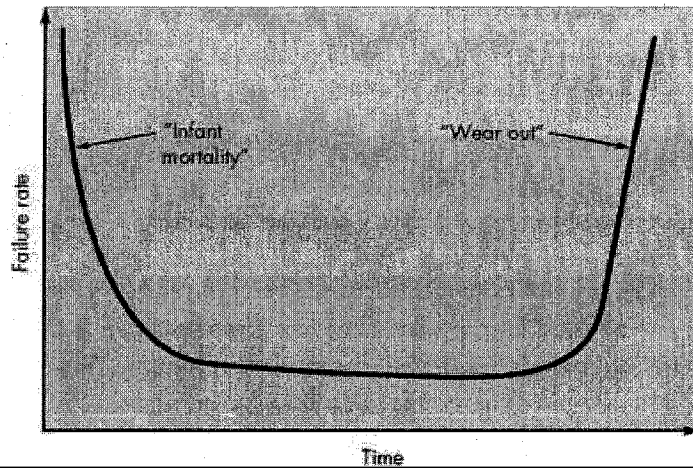
(04 marks)

- iii. Name four types of application?

Stand-alone applications, Data collection system
Interactive transaction-based applications, System of system
Embedded control systems, System for modeling and simulations
Batch processing systems.
Entertainment systems

(Any 04 answers, 04 M arks)

iv. Briefly explain following graph?



This is bathtub curve explain hardware cost of a System

There are three stages of this curve

First stage is decreasing the failures rates

Second stage is the normal life of the product

Third stage is increasing the failure rate

(Any 03 answers, 06 marks)

v. Briefly explain four essential characteristic of good software?

Maintainability-Software should be written in such a way so that it can evolve to meet the changing needs of customers.

Dependability and security-Dependable software should not cause physical or economic damage in the event of system failure.

Efficiency-Software should not make wasteful use of system resources such as memory and processor cycles.

Acceptability-This means that it must be understandable, usable and compatible with other systems that they use.

(Each answer 02 marks, $02 \times 4 = 08$ marks)

(Total 25 marks)

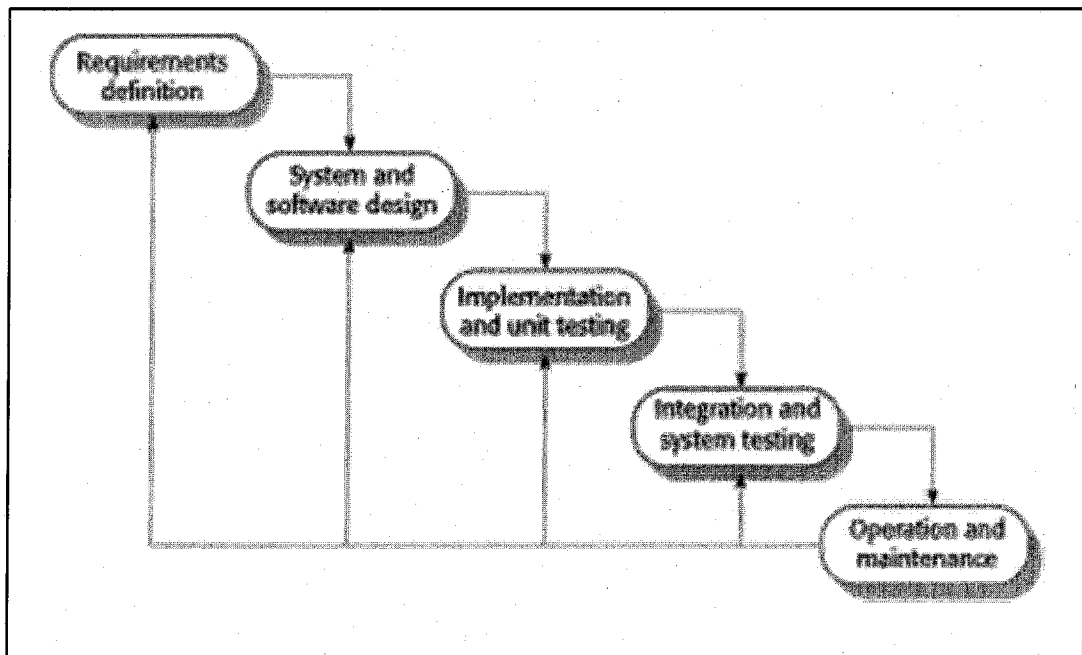
Q2)

I. What is Plan driven software process?

Plan-driven processes are processes where all of the process activities are planned in advance and progress is measured against this plan

(03 marks)

II. Draw the waterfall model with all the steps.



(All the steps 04 marks, partial answer 02 Marks)

III. Name four sectors of Spiral model?

Objective setting
Risk assessment and reduction
Development and validation
Planning

(Each answer 01 marks, 01×4=04 Marks)

IV. Give three benefits of Software prototyping?

Improved system usability
A closer match to users' real needs
Improved design quality
Improved maintainability
Reduced development effort

(Any 03 answers, 06 marks)

V. Briefly explain following activities of software process?

a. Specification

The process of establishing what services are required and the constraints on the system's operation and development

b. Design

Design and implementation processes are concerned with transforming a requirements specification into an executable software system.

c. Validation

Software validation is the process of checking that the system conforms to its specification and that it meets the real needs of the users of the system.

d. Evolution

Software evolution takes place when you change existing software systems to meet new requirements.

(Each answer 02 marks, 02×4=08 Marks)

(Total 25 Marks)

PART-B

Q3)

- I. What is meant nonfunctional requirements? Give three examples for that.
Define system properties and constraints. **(01 Mark)**

Accuracy

Modifiability

Portability

Reliability

Security

Usability

(Any 03 answers, 03 Marks)

- II. Briefly explain requirement elicitation and analysis.
Sometimes called requirements elicitation or requirements discovery.
Involves technical staff working with customers to find out about the application domain, the services that the system should provide and the system's operational constraints. May involve end-users, managers, engineers involved in maintenance, domain experts, trade unions, etc. These are called *stakeholders*. **(04 Marks)**

- III. One of the most important goals of object oriented design is to have high cohesion classes and loose coupling between these classes. What is the purpose of this?
- High Cohesion and low coupling give us better designed code that is easier to maintain.
 - Low coupling is in the context of two or many modules. If a change in one module results in many changes in other module then they are said to be highly coupled.

(04 Marks)

- IV. Identify the functionalities of the following system and write down them.

An Online Public Access Catalog (OPAC) is an e-Library website which is part of Integrated Library System (ILS), also known as a Library Management System (LMS), and managed by a library or group of libraries. Patrons of the library can search library catalog online to locate various resources - books, periodicals, audio and visual materials, or other items under control of the library. Patrons may reserve or renew item, provide feedback, and manage their account.

Manage Account

Search Catalog

Reserve Item

Renew Item

Provide Feedback

(05 Marks)

- V. Briefly explain four best practices for designing user interface (UI).

Keep the interface simple-The best interfaces are almost invisible to the user. They avoid unnecessary elements and are clear in the language they use on labels and in messaging

Create consistency and use common UI elements. By using common elements in your UI, users feel more comfortable and are able to get things done more quickly.

Strategically use color and texture. You can direct attention toward or redirect attention away from items using color, light, contrast, and texture to your advantage.

Use typography to create hierarchy and clarity. Carefully consider how you use typeface. Different sizes, fonts, and arrangement of the text to help increase scanability, legibility and readability.

(Each answer 02 marks, $02 \times 4 = 08$ marks)

(Total 25 marks)

Q4)

- I. Define the term software architecture? Give three architectural patterns.
The design process for identifying the sub-systems making up a system and the framework for sub-system control and communication is architectural design.

(02 mark for definition)

Layered pattern

Client-server pattern

Model-view-controller pattern

(03 marks)

- II. You are recruited as a project manager to develop LMS (Learning Management System) to SLIATE. This LMS is used by the student, lecturers and administrators. Lecturers are uploading learning materials as slide shows, PDF and video lessons. Students are logging to the system and download the learning materials. What is best software architectural pattern for the above problem? Justify your answer.

Client Server model or MVC pattern

(02 marks)

This solution is web based solution.

Need to maintain central server to upload learning materials.

This service used by the student as the clients.

Easy to manage

(02 marks for justification)

III. Compare unit testing and integration testing.

Unit testing

Level of software testing where individual units/ components of software are tested
The purpose is to validate that each unit of the software performs as designed.

(02 marks)

Integration testing

Level of software testing where individual units are combined and tested as a group
The purpose of this level of testing is to expose faults in the interaction between integrated units.

(02 marks)

IV. Briefly explain following terms.

a. **Black box testing**

- The technique of testing without having any knowledge of the interior workings of the application is called black-box testing.
- Equivalence Partitioning
- Boundary Value Analysis

(03 marks)

b. **White box testing**

- White Box Testing is defined as the testing of a software solution's internal structure, design, and coding.
- In this type of testing, the code is visible to the tester.
- Procedure to derive and/or select test cases

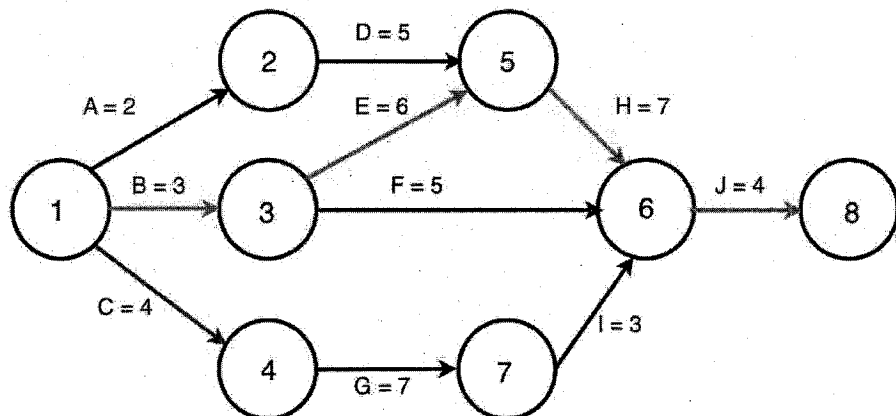
(03 marks)

V. The following table explains the activities, duration and dependencies.

Task	Predecessors Tasks (Dependencies)	Time (Weeks)
A	-	2
B	-	3
C	-	4
D	A	5
E	B	6
F	B	5
G	C	7
H	D,E	7
I	G	3
J	F,H,I	4

a. Draw the Activity on Arrow diagram.

(04 Marks)



b. Find the critical path of the project.

(02 Marks)

Critical path=B+E+H+J
=3+6+7+4
=20 weeks

(Total 25 Marks)

Q5)

I. What is the purpose of configuration management?

- Configuration Management helps organizations to systematically manage, organize, and control the changes in the documents, codes, and other entities during the Software Development Life Cycle.

(02 Marks)

II. Name four reasons for software maintenance.

- Market Conditions
- Client Requirements
- Host Modifications
- Organization Changes

(Each answer 01 marks, 01×4=04 marks)

III. Briefly describe three types maintenance?

Corrective maintenance

This includes modifications and updates done in order to correct or fix problems, which are either discovered by user or concluded by user error reports.

Adaptive Maintenance

This includes modifications and updates applied to keep the software product up-to-date and tuned to the ever changing world of technology and business environment.

Perfective Maintenance

This includes modifications and updates done in order to keep the software usable over long period of time.

Preventive Maintenance

It aims to attend problems, which are not significant at this moment but may cause serious issues in future.

(Any 03 answers, 02×3=06 marks)

IV. What is meant Program restructuring?

It is a process to re-structure and re-construct the existing software. It is all about re-arranging the source code, either in same programming language or from one programming language to a different one.

(03 Marks)

V. Briefly explain Software forward-engineering and Re-engineering.

Forward-engineering

Forward engineering is a process of obtaining desired software from the specifications in hand which were brought down by means of reverse engineering.

(02 Marks)

Re-engineering

When we need to update the software to keep it to the current market, without impacting its functionality, it is called software re-engineering.

(02 Marks)

VI. Briefly explain following terms.

a. Codeline and baseline

- A codeline is a sequence of versions of source code with later versions in the sequence derived from earlier versions.
- A baseline is a definition of a specific system

(02 marks)

b. System building

- The process of assembling program components, data and libraries, then compiling these to create an executable system

(02 marks)

c. Release management

- Preparing software for external release and keeping track of the system versions that have been released for customer use.
- Major releases which deliver significant new functionality
- Minor releases, which repair bugs and fix customer problems that have been reported

(02 marks)

(Total 25 marks)