SLIATE

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, First Semester Examination – 2018

HNDIT2312-Principles of Software Engineering

Marking scheme

Instructions for Candidates:	No. of questions : 05
Answer Only **Four** questions	No. of pages : 02
All questions carry equal marks.	Time : **Two hours**

Q1.

i. Define “Software Engineering”. (03 marks)

Use of systematic, engineering approach in all stages of software development and project management to develop high quality and economical software using appropriate software tools

ii. What is the difference between Software engineering and the system engineering?

(04 marks)

System engineering is concerned with all aspects of computer-based system development, including hardware, software and processor engineering. Software engineering is a part of this process

(02 +0 2=04Marks)

iii. What are the two types of Software products name and explain. (04 marks)

1. *Generic products : These are produced by development company and sold on the open ,market to any customer.*
2. *Customized product : These are developed for a particular customer. Which is useful only for that customer who had given contract to the development company.*

(02 +0 2=04Marks)

iv. List out the six (06) Software Product Attribute. (06 marks)

1. *Maintainability*
2. *Dependability*
3. *Efficiency*
4. *Usability*
5. *Portability*
6. *Correctness*

(01 * 06 =06 Marks)

v. Briefly explain four (04) Software characteristics. (08 marks)

1. *Software is logical rather than a physical system element*. The characteristics of software is different than those of hardware.
2. *Software is developed or engineered, it is not manufactured in the classical sense.* Even though there exist similarities between software development and hardware manufacturers, the two activities are functionally different. Both activities requires good design, and people to work. Skills required to work on the software development is different than skills required to work a hardware manufacture. Software design are easily changed compares to the hardware.
3. *Software doesnot "wear out".* In hardware manufacture it exhibits relatively high failure in its early of SDLC defects are rectified and the failure rate drops to a steady-state level. Failure rate will be in steady s level for some period of time. As time passes ,the failure rate rises again, because components suffer from the dust, vibration, high tempaersture high voltage and etc. the hardware begins to ware out. Software is not affected to the environmental effects like temperature, vibrasion, dust etc.
4. *Generally software is custom – built,rather than beign assembled from existing components.* A software component should be designed and implemented son that it can be reused in many different programmes.

(02*04 = 08 Marks)

(Total 25 Marks)

Q2.

i. List three (03) process models which are providing good visibility. (03 marks)

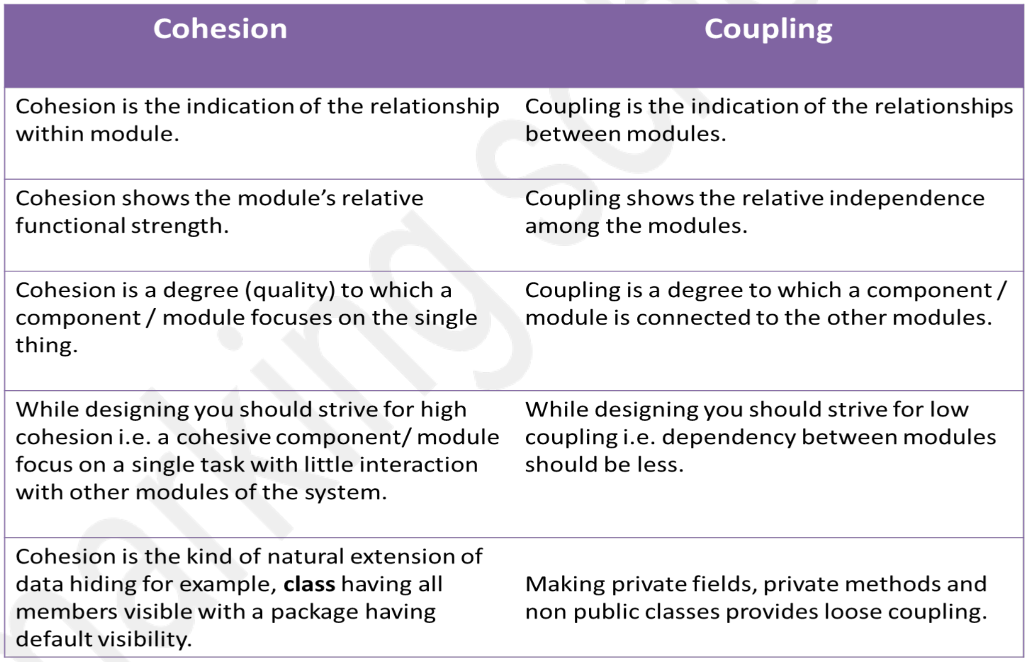
Spiral model
RAD
Prototyping
(01* 03 = 03 marks)

ii. Briefly explain why user requirements should be written using natural language.

(04 marks)

Requirement are written as natural language sentences supplemented by diagrams and tables. Used for writing requirements because it is expressive, intuitive and universal. This means that the requirement can be understood by users and customers.

iii. Explain briefly coupling and cohesion in modular design. (04 marks)



Cohesion	Coupling
Cohesion is the indication of the relationship within module.	Coupling is the indication of the relationships between modules.
Cohesion shows the module's relative functional strength.	Coupling shows the relative independence among the modules.
Cohesion is a degree (quality) to which a component / module focuses on the single thing.	Coupling is a degree to which a component / module is connected to the other modules.
While designing you should strive for high cohesion i.e. a cohesive component/ module focus on a single task with little interaction with other modules of the system.	While designing you should strive for low coupling i.e. dependency between modules should be less.
Cohesion is the kind of natural extension of data hiding for example, **class** having all members visible with a package having default visibility.	Making private fields, private methods and non public classes provides loose coupling.

(02 + 02 = 04 marks)

iv. Explain the use of coupling and cohesion in effective modular design. (06 marks)

Reduce coupling and minimize cohesion to built systems that are scalable, manageable and can be extended over time. These are terms that are used to indicate the qualitative analysis of the modularity in a system , and they helps to identify and measure the design complexity of object oriented systems.

v. List four (04) differences of functional and non-functional requirements of a system.

1. Functional requirement is specified by User, while non-functional requirement is specified by technical peoples e.g. Architect, Technical leaders and software developers.

2. Functional requirement is also the activity System must perform, on other hand non-functional are depending upon criticality of application. For example, if your application is not critical and you can live with downtime, you may not need to develop complex failover and disaster recovery code, reducing your application total development time.

3. Functional requirements defines a software's functionality i.e. what can they do, while non-functional requirements defines, other things which is not required by user but requirement by service provider or software itself e.g. logging is a non-functional requirement to support an application, not directly used by user but essential to troubleshoot any issue in production environment.

4. Non-functional requirements are sometimes defined in terms of metrics (something that can be measured about the system) to make them more tangible.

5. Non-functional requirements may also describe aspects of the system that don't relate to its execution, but rather to its evolution over time (e.g. maintainability, extensibility, documentation, etc.).

(02 * 04= 08 Marks) (08 marks)

(Total 25 Marks)

Q3

i. Writ three(03) advantages of software architecture (03 marks)

Stakeholder communication

System analysis

Large-scale reuse

(01 * 03 = 03 Marks)

ii. Give any four(04) architectural patterns (04 marks)

- Layered pattern
- Client-server pattern
- Model-view-controller pattern
- Master-slave pattern

- Pipe-filter pattern
- Broker pattern
- Peer-to-peer pattern
- Event-bus pattern

(01 * 04 = 04 Marks)

iii. List five (04) stages in testing process. (04 marks)

- *Unit testing*
- *Integration Testing*
- *System testing*
- *Acceptance testing*
- *Alpha /beta testing*
- *Regression test*

iv. Explain the difference between Process quality and Product quality. (06 marks)

Product quality is focusing on meeting tolerances in the end results of the manufacturing activities. The end result is measures on a standard of "good enough".

Process quality focuses on each activity and forces the activities on achieve maximum tolerances irrespective of the end result.

(03 * 02 = 06 Marks)

v. Explain the difference between verification and validation in software checking process.
(08 marks)

- *Both verification and validation are essential and balancing to each other.*
- *Different error filters are provided by each of them.*
- *Both are used to finds defect in different way, verification is used to find the defects in the implemented software application*

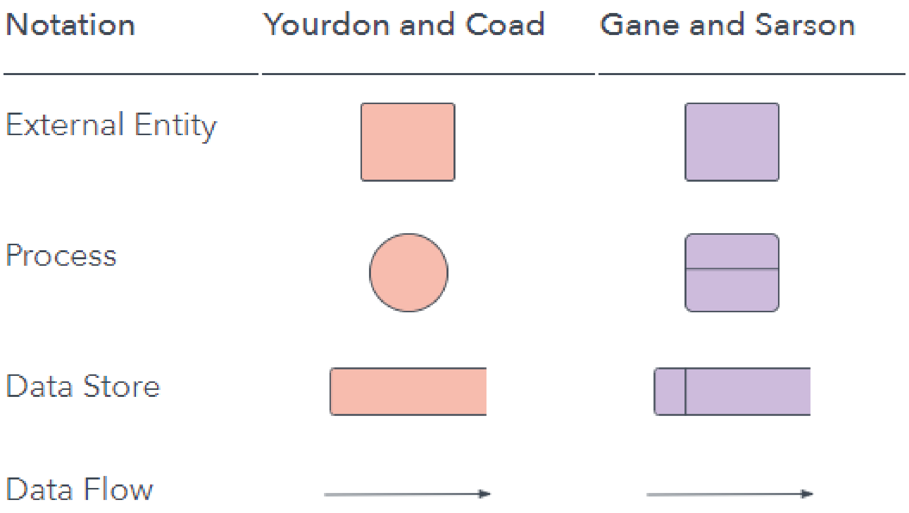
(Total 25 Marks)

Q4.

i. Briefly explain UML (03 marks)

- A UML diagram is a partial graphical representation (view) of a model of a system under design, implementation, or already in existence.
- UML specification defines two major kinds of UML diagram:
  - Structure diagrams
  - Behavior diagrams.

ii. Draw and name four symbols in DFD Diagram (04 marks)



Notation	Yourdon and Coad	Gane and Sarson
External Entity	■	■
Process	●	▤
Data Store	▭	▭
Data Flow	→	→

(01 * 04 = 04 marks)

iii. When drawing DFD you have to follow some rules and tips, Write four (04)

(04 Marks)

- Each process should have at least one input and an output.
- Each data store should have at least one data flow in and one data flow out.
- Data stored in a system must go through a process.
- All processes in a DFD go to another process or a data store.

(01 * 04 = 04 marks)

iv. What is Use case Diagram? Name two (04) components in a Use case Diagram
(06 marks)

Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors).

- Subject or System boundary
- Actors
- Use cases
- Relationship(Links)

(02 + (01 * 04) = 06 Marks)

v. Describe the four (04) Best Practices for Designing an Interface (08 marks)

- **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.

(02 * 04 = 08 Marks)

(Total 25 Marks)

Q5.

i. What is Project management? (03 marks)

ii. Describe the importance of change management in software maintenance.

(04 marks)

Software systems are subject to continual change request from user, developer and market forces. Change management is concerned with keeping managing of these changes and ensuring that they are implemented in the cost effective way

iii. Explain the followings. (04 marks)

a. Release management :

This involves preparing software for external release and keeping track of the system versions that have been released for customer use.

b. Configuration management

Discipline for evaluating, co-ordination ,approving or disapproving , and implementing changes in artifacts that are used to construct and maintain software system s. Development and application of standards and procedures for managing an evolving software product.

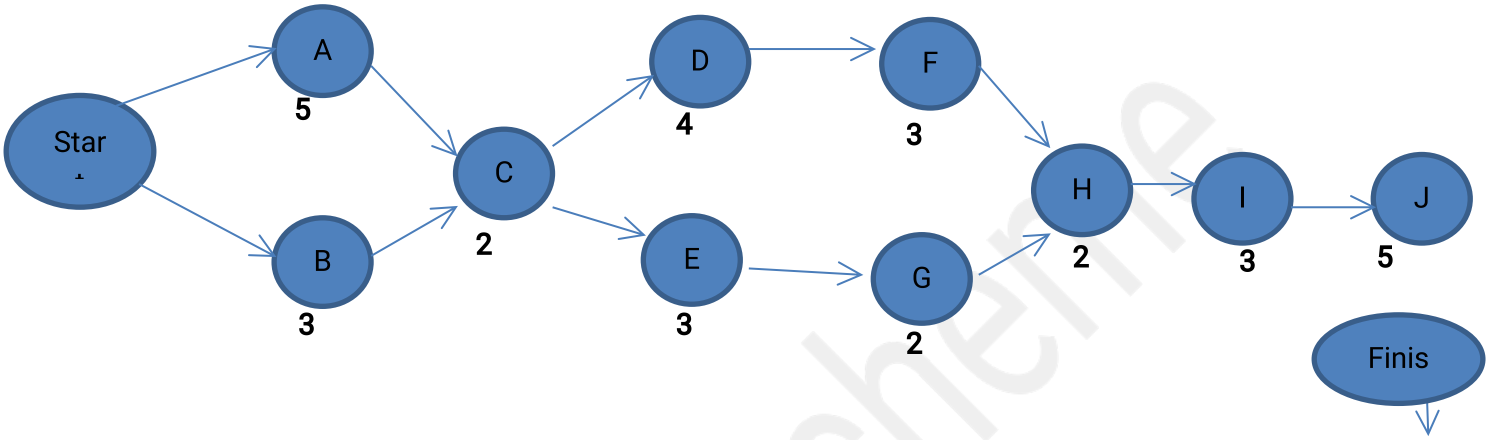
(02 * 02 = 04 marks)

iv. Use below given table

Activity	Dependencies	Duration
A	-	5
B	-	3
C	A, B	2
D	C	4
E	C	3
F	D	3
G	E	2

H	**F, G**	**2**
I	**H**	**3**
J	**I**	**5**

a. Draw a network diagram (08 Marks)



b. Find critical path and minimum completion time of project (04 marks)

Critical path : A,C,D,F,H,I,J

Duration: 24 Days

c. If task “E” delay from three (03) Days what will happen to the critical path of project? (02 marks)

Nothing happen to critical path

(Total 25 Marks)