Software Engineering - Assignment 1.

1) Explain the needs of Software Engineering:

A)= The need of Software Engineering arises because of higher rate of change in user requirements and environment on which the software is working:

• Large Software: It is easier to build a wall than to a house or building, likewise, as the size of software become large, engineering has to step to give it a scientific process.

• Scalability: If the software process were not based on scientific and engineering concepts, it is easier to re-create new software than to scale an existing one.

• Cost: As hardware industry has shown it's skills and huge manufacturing has lower down the price of computer and electronic hardware. But the cost of software remains high if proper process is not adapted.
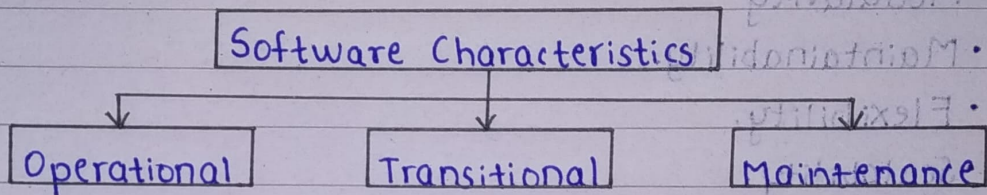
• Dynamic Nature: The always growing and adapting nature of software hugely depends upon the environment in which the user works. If the nature of software is always changing, new enhancements need to be made in existing one. This is where software Engineering plays a good role.

• Quality Management: Better process of software development provides better and quality software product.

2) Explain characteristics of good software.

A) = Software is treated as good software by the means of different factors. A software product is concluded as a good software by what it offers and how well it can be used. The factors that decide the software properties are divided into 3 categories:

```
          ┌─────────────────────────┐
          │  Software Characteristics │
          └─────────────────────────┘
       ↓                ↓                ↓
┌────────────┐   ┌──────────────┐   ┌──────────────┐
│ Operational│   │ Transitional │   │ Maintenance  │
└────────────┘   └──────────────┘   └──────────────┘
```

Well engineered and crafted software is expected to have following characteristics:

1. Operational

This tells us how well software works in operations. It can be measured on:

• Budget   • Functionality.
• Usability   • Dependability.
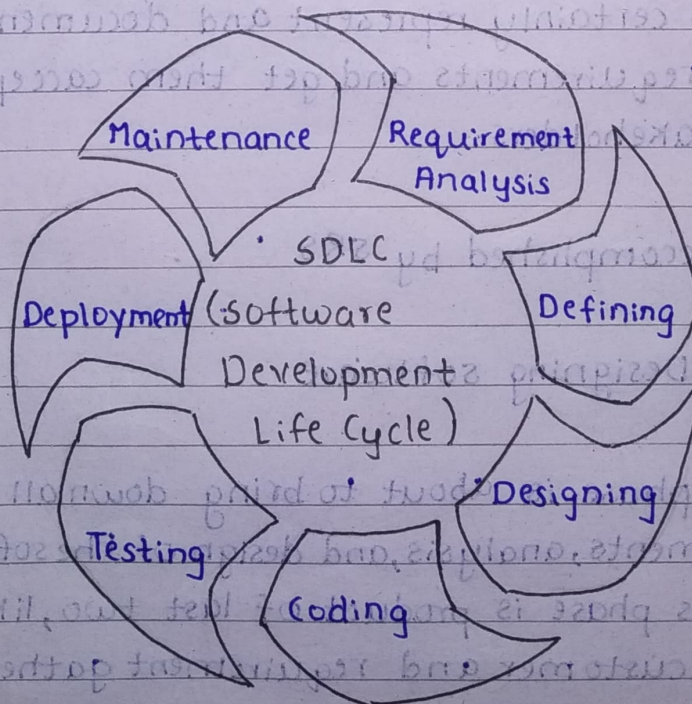• Efficiency   • Security.
• Correctness   • Safety.

2. Transitional

This aspect is important when the software is moved from one platform to another:

• Portability.
• Interoperability.
• Reusability.
• Adaptability.

3. Maintenance

This aspect briefs about how well a software has the capabilities to maintain itself in the ever-changing environment:

- Modularity.
- Maintainability.
- Flexibility.
- Scalability.

In short, Software Engineering is a branch of computer science, which uses well-defined engineering concepts required to produce efficient, durable, scalable, in-budget and on-time software products.

3) Explain SDLC model in detail.

A):- ① A software life cycle model (also termed process model) is a pictorial and diagrammatic representation of the software life cycle.

② A life cycle model represents all methods required to make a software product transit through it's life cycle stages. It also captures the structures in which these methods are to be undertaken.

③ In other words, a life cycle model maps the various activities performed on a software product from its inception to retirement. During any life cycle stage, more than one activity may also be carried out.

④ Without a SDLC, the entry and exit criteria cannot be recognized. without SDLC, it becomes tough for software project managers to monitor progress of the project.

⑤ SDLC includes following steps:

Maintenance | Requirement Analysis
SDLC (software Development Life Cycle)
Deployment | Defining
Testing | Designing
Coding

⑥ Stage 1: Planning and requirement analysis.

- Requirement Analysis is the most important and necessary stage in SDLC.

- The senior members of team perform it with inputs from all stakeholders and domain experts

- Planning of quality assurance requirements and identification of risks associated with project is also done at this stage.

- SRS (software Requirement Specification) document is created. The developers should throughly follow this document.

⑦ Stage 2: Defining Requirements.

- Once the requirements analysis is done, the next stage is to certainly represent and document the software requirements and get them caccepted from project stakeholders.

- This is accomplished by SRS

⑧ Stage 3: Designing software.

- The next phase is about to bring down all the knowledge of requirements, analysis, and design of the software project. This phase is product of last two, like inputs from the customer and requirement gathering.

⑨ Stage 4: Developing project.

- In this phase of SDLC, actual development begins, and programming is built. The implementation of design becomes concerning writing code.
- Developers have to follow the coding guidelines described by their management and programming tools like compilers, interpreters, debuggers, etc. are used to develop and implement code.

⑩ Stage 5: Testing.

- After code is generated, it is tested against the requirements to make sure that the products are solving the needs addressed and gathered during the requirements stage.
- During the stage, unit testing, integration testing, system testing, acceptance testing are done.

⑪ Stage 6: Deployment.

- Once software is certified, and no bugs or errors are stated, then it is deployed.
- Then based on assessment, the software may be released as it is or with suggested enhancement in object segment. After it is deployed, maintenance begins.

(12) Stage 7: Maintenance.

- Once the client starts using the develop systems,
  then the real issues come up and requirements
  to be solved from time to time.

This procedure where care is taken for the development
product is known as maintenance.