## Software engineering Questions and Answers:

### 1: What is software engineering, anyway?

Some basic definitions of software engineering are:

- The application of a disciplined engineering approach to the development of software systems.
- A body of knowledge and experience in software development practice and process.

### 2: I like computers. Is Software Engineering for me?

Software Engineering is all about using engineering principles for the production of software. If you like computers, like solving challenging problems and would like to make an impact on the world in which we all live, you should consider software engineering (SE).

Computer systems are an integral part of today's society. Software is a critical component of all computer systems, including the "embedded systems" used in communication networks, vehicles, consumer electronics, and medical devices. Software engineers have the knowledge and skills needed to produce high-quality, effective software on which all these computer systems depend.

BSSE program is one of the first four ABET accredited programs in the United States. Our curriculum provides a strong foundation upon which to build a successful SE career.

### 3: What is the difference between computer engineering (CE), computer science (CS), and software engineering (SE)?

While computer engineering programs include courses in software, SE programs incorporate much more detail in software development practice and process, including advanced areas of software architecture, requirements management, quality assurance, and process improvement.

Software engineering programs do not stress computer hardware and electronics as much as computer engineering programs do.

On the other hand, software engineering is based on computer science, as other engineering disciplines are based on natural or life sciences. However, software engineering adds an emphasis on issues of process, design, measurement, analysis and verification. In general, scientists seek new knowledge, while engineers want to build things, solve problems, and help people. Both roles are important.

### 4: Why are software engineers needed?

Computer systems are pervasive and have a major impact on society. Software is a critical component of all computer systems, including the "embedded systems" used in communication networks, vehicles, consumer electronics and medical devices. Software engineers have the knowledge and skills needed to produce high-quality, effective software on which all these computer systems depend.

### 5: I have no previous programming experience. Can I still be successful Software Engineer?

Any previous programming experience that you may have is certainly an asset, but it is not a requirement. It is a very common myth that SE is mainly programming. In fact, SE programs put a lot more emphasis on software development practice and process, including advanced areas of...
6: **What are the characteristics of good design? Name some Design Tools?**

- **Object Oriented Design Tools** - Rational Rose, Rhapsody, Telelogic Tau G2
- **UI Design Tools** - Rapid/Paper prototypes, Simple
- **SUMI (Software Usability Measurement Inventory)** - It will tell you how your product compares to this standardization base: whether you are about average for the market, below, or above.

7: **What is SDLC? What are the various SDLC models? Explain them?**

- **What is SDLC**
  - A software cycle deals with various parts and phases from planning to testing and deploying. All these activities are carried out in different ways, as per the needs. Each way is known as a Software Development Lifecycle Model (SDLC).

- **Some SDLC models**
  - *The Linear model (Waterfall)*
    - Separate and distinct phases of specification and development
    - All activities in linear fashion
    - Next phase starts only when first one is complete
  - *Evolutionary development*
    - Specification and development are interleaved (Spiral, incremental, prototype based, Rapid Application development)
    - Incremental Model (Waterfall in iteration)
    - RAD (Rapid Application Development) - Focus is on developing quality product in less time
    - Spiral Model - We start from smaller module and keeps on building it like a spiral. It is also called Component based development.
  - *Formal systems development*
  - *Agile Methods*
    - Inducing flexibility into development
  - *Reuse-based development*
    - The system is assembled from existing components

8: **What is RUP and PSP?**

- **RUP stands for Rational Unified Process**
  - It deals with -
    - Iterative software development process.
    - Visual Modeling of Systems
    - Quality Management
    - Change Control Management
    - Deals with the role, the activity, and the artifact
    - For managing OO Software Development

- **Some Features of RUP are** -
  - Online Repository of Process Information and Description
  - Templates for all major artifacts, including:
    - Requirements tracking (RequisitePro templates)
– Use Cases (Word Templates )
– Project Management (Project Templates)
• Process Manuals describing key processes

PSP stands for Personal Software Process

PSP Objectives are -
– To introduce individuals to a process-based approach to developing software
– To show individuals how to measure, estimate, schedule, and track their work
– To show individuals how to improve the quality of their programs
– In general PSP improves quality and productivity
- The time saved in testing because of better quality reduces time across the project by 20-40%

Advantage of PSP -
– To manage your work & assess/build your talents/skills
– To plan better
– To track your performance precisely
– To measure the quality of your software products

9: What are the Different types of Testing in software engineering? How to design a Test Case?

What is Testing in Software Engineering
Testing is running the program(or product) under various circumstances and conditions to find errors and bugs in it. This is important as releasing a faulty product will not only cause serious problems to the end user, it will also harm the companies reputation. There are various kind of Testing conditions and which one to use depends on type of product.

Types of Testing in Software Engineering
As per Test Target, there is * Unit Testing * Integration Testing * System Testing As per Test Objective, there is * User Acceptance Testing * Installation Testing * Functional * Alpha / Beta testing * Regression * Performance * Stress * Usability * Configuration * Smoke (Sanity Test)

Let’s see them one by one -

Unit Testing
- The Tools used in Unit Testing are debuggers, tracers and is Done by Programmers. Unit testing verifies the functioning in isolation of software pieces which are separately testable.

Integration Testing
- According to IEEE, Integration Testing is An orderly progression of testing in which software elements, hardware elements, or both are combined and tested, to evaluate their interactions, until the entire system has been integrated. It Test against system design and Focuses on communication between modules start with one module, then add incrementally. Various Types of Integration Testing are: * big bang approach - Integrate Everything at once * top-down approach - Keep on breaking the system in parts one by one and than test each part. * bottom-up approach - Test the small parts first and than keep on integrating the system and keep on testing the bigger module of it. * mixed approach - Done by help of stubs (Dummy modules)

Sysem Testing
- IEEE Defines it as The process of testing an integrated hardware and software system to verify that the system meets its specified requirements. It is tested against system specification. May test manual procedures, restart and recovery, user interface, stress, performance. In System Testing, real data is used and sometimes users participations is also used.
Objective Based Testing
Some of the testing under this category and what they mean are as follows - *

User Acceptance Testing
- Testing conducted to determine whether or not a system satisfies its acceptance criteria and to enable the customer to determine whether or not to accept the system. It is Done against requirements and is done by actual users. *

Installation Testing
- System testing conducted once again according to hardware configuration requirements. Installation procedures may also be verified *

Functional Testing
- It checks that the functional specifications are correctly implemented. Can also check if Non Functional behavior is as per expectations. *

Stress testing (aka endurance testing)
- We impose abnormal input to stress the capabilities of the software. Input data volume, input data rate, processing time, utilization of memory, etc. are tested beyond the designed capacity. *

Regression Testing
- Regression Testing according to IEEE is "selective retesting of a system or component to verify that modifications have not caused unintended effects". It is repetition of tests intended to show that the software’s behavior is unchanged, except insofar as required. It Can be done at each test level. *

Performance Testing
- It is verifying that the software meets the specified performance requirements (response time, volume ..)

Alpha / Beta testing
- Probably one term which you must be aware of as we often hears this software is in Alpha phase and in Beta phase. Here Testing is done by representative set of potential users for trial use. Please Note - in-house (alpha testing) - external (beta testing)

Usability Testing
- It evaluate the Human Computer Interface. Verifies for ease of use by end-users. Verifies ease of learning the software, including user documentation. Checks how effectively the software functions in supporting user tasks. Checks the ability to recover from user errors.

Configuration Testing
- It is Used when software meant for different types of users. It also check that whether the software performs for all users.

Smoke (Sanity Test)
- It is used to Verify whether the build is ready for feature/requirement based testing).

Recovery Testing
- It is used in verifying software restart capabilities after a “disaster”

Security Testing
- It is used to verify proper controls have been designed.

How to Design a Test Case
This Question is often asked in microsoft of from any other company who is looking to hire you for testing work. A test case will have 5 section -

10: What is SRS?
SRS stands for Software Requirement Specification. It establishes the basis for agreement between customers and contractors or suppliers on what the software product is expected to do, as well as what it is not expected to do. Some of the features of SRS are -
- It sets permits a rigorous assessment of requirements before design can begin.
- It sets the basis for software design, test, deployment, training etc. It also sets pre-requisite for a good design though it is not enough.
- It sets basis for software enhancement and maintenance.

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>11: What are Functional and Non Functional Requirements in Software Engineering?</strong></td>
<td>Functional Requirements are the Expected functionality or services from the system. Non - Functional Requirements are System property and constraints. Now Constraints can be on requirements itself. Some examples are of Response time, 24X7 availability of the system etc.</td>
</tr>
<tr>
<td><strong>12: What is Requirement Engineering?</strong></td>
<td>Software Requirements are Descriptions and specifications of a system – A requirement is a condition or capability to which the system must conform and Define. A system is designed with certain functionality in mind and to come up with that functionality and its specification right at the beginning is Requirement Engineering. Requirements Engineering – Eliciting, organizing, and documenting the requirements of the system – The process of establishing the services that the customer requires from a system and the constraints under which it operates and is developed Software requirements should be •Clear •complete •Unambiguous •Quantitative</td>
</tr>
</tbody>
</table>
Bibliography

www.google.com

www.interviewquestion.com