#### 1. What is Software Engineering?

**Answer:**  
Software Engineering is a process of developing a software product in a well-defined systematic approach. In other words, developing a software by using scientific principles, methods, and procedures.

#### 2. What is the need to learn Software Engineering Concepts?

**Answer:**  
Imagine a person, who is good at building a wall may not be good at constructing a house. In a similar way, a person who can write programs does not have the knowledge to develop and implement the software in a well-defined systematic approach. Hence, there is a need for programmers to adhere to software engineering concepts such as requirements gathering, planning, development, testing, and documentation.

#### 3. What is SDLC OR Software Development Life Cycle?

**Answer:**  
SDLC defines a set of guidelines to develop a software product. SDLC has different phases namely: Gathering Requirements, Analysis, Planning, Development, Testing, Implementation, Maintenance, and Documentation. The order of the phases mentioned in SDLC may vary depending upon the model chosen to implement.

#### 4. What are the different types of models available in SDLC?

**Answer:**  
Many models have been proposed, to carry out the software implementation efficiently. Some of them include [the Waterfall Model](https://www.educba.com/course/waterfall-modeling/), [Agile Model](https://www.educba.com/course/agile-project-management/), Spiral Model, [Iterative Model](https://www.educba.com/iterative-model/), V-Model etc.

#### 5. Explain the role of a Software Project Manager?

**Answer:**  
This is the common software Engineering Interview Questions asked in an interview. Project Manager is responsible for driving the [software project](https://www.educba.com/course/online-software-project-estimation-training/) in a systematic approach. Some of the key roles & responsibilities of a software project manager include project planning, tracking the progress of the project, risk management, resource management, execution of development activities, delivering the project under cost, time and quality constraints.

#### 6. What is a Software Project Scope?

**Answer:**  
A scope is utilized to outline the activities performed to design, develop and deliver a software product. In other words, scope contains information on what project is intended to deliver and what it does not intend to. The scope also outlines information on what software product developed contains and what it does not contain.

#### 7. What is Software Project Estimation?

**Answer:**  
Project Estimation is a process utilized to calculate the development costs such as effort, time and resources required to deliver a project. [Project Estimations](https://www.educba.com/project-estimation-techniques/) are derived through past project experiences or with the help of consulting experts or with the help of standard predefined business formulas.

Let us move to the next software Engineering Interview Questions.

#### 8. Explain Functional Points?

**Answer:**  
Functional points are used to measure the size of the software product. In some business, scenarios play a key role in tracking and estimating the project delivery.

#### 9. What is a Baseline?

**Answer:**  
Baselines are put forth by the project managers to track the overall project delivery. Baselines are usually placed to track the overall tasks listed under a phase or stage. Baselines help project managers to track and monitor the overall execution of a project.

#### 10. What is Software Configuration Management?

**Answer:**  
Software Configuration Management helps users to track the overall changes made in a software product delivery. Updates or changes made to the software are tracked in terms of development and requirements gathering.

Let us move to the next software Engineering Interview Questions.

#### 11. What is Change Control?

**Answer:**  
Change control tracks the changes made in a software to ensure consistency and updates are incorporated as per the enterprise standards.

#### 12.Mention few project management tools?

**Answer:**  
Many project management tools are utilized as per the enterprise standards some of them include: [Gantt Charts](https://www.educba.com/course/gantt-chart-tutorials/), PERT Charts, Milestone Checklists, Histograms, MS project, Status reports etc.

#### 13. What is a Software requirement?

**Answer:**  
Requirements play a key role in providing a detailed description of the software product being developed. Software requirements help the developers and other support teams associated with project delivery, to understand the proposed target system and their expectations on it.

#### 14. Explain the Feasibility Study?

**Answer:**  
Feasibility Study is performed to assess the beneficial and practical attributes of a [software development](https://www.educba.com/course/sdlc-training-software-development-life-cycle/)Thorough analysis is performed by an organization with the help of feasibility study to understand the economic, operational and technical aspects involved in a software project delivery.  
•**Economic:** Economic study involves costs related to [resource management](https://www.educba.com/course/strategic-human-resource-management/), training costs, tools utilized and project estimation costs  
•**Technical:** Technical study helps the business to analyze the technical aspects involved in software delivery such as machines, operating systems, knowledge, and skills of resource allocated, tools utilized and training.  
•**Operational:** Operational study help business to study the change management and issues involved depending on the project needs.

#### 15. What are functional and non- functional requirements?

**Answer:**  
Functional requirements are utilized to specify the functional features as per the business requirements. For Example, adding a payment option to buy content from a website. Whereas Non- functional requirements provide insights into security, performance, user interface, interoperability costs etc.

#### 16. What are Software Metrics?

**Answer:**  
Metrics are utilized to guide the software product delivery as per the business standards. Metrics can also be used to measure few features of software product delivery. Metrics are divided into requirement metrics, product metrics, performance metrics, and process metrics.

Let us move to the next software Engineering Interview Questions.

#### 17. What is Modularization?

**Answer:**  
Modularization divides the software system tasks in multiple modules. These modules are independent to other modules and tasks invoked in each module are executed independently.

#### 18. Explain Concurrency and how is it achieved during the software product delivery?

**Answer:**  
This is the advanced software Engineering Interview Questions asked in an interview. Concurrency is a process of executing multiple events or tasks simultaneously. Concurrency can be achieved with the help of modules, events, and tasks associated with the software project delivery.

#### 19. What is Cohesion?

**Answer:**  
Cohesion is utilized to measure the intra-dependability among various attributes defined in a module.

#### 20. What is coupling?

**Answer:**  
Coupling is utilized to measure the inter-dependability of various elements defined in a module.

#### 21. Mention a few software analysis & Design tools?

**Answer:**  
Some of the key software analysis & design tools are Data flow Diagrams (DFD), Structured Charts, Data Dictionary, UML (Unified Modeling Languages) diagrams, ER (Entity Relationship) Diagrams etc.

Let us move to the next software Engineering Interview Questions.

#### 22. What is DFD Level 0?

**Answer:**  
DFD (Data Flow Diagrams) Level 0 depict the entire data flow along with all abstract details within a software information system. This type of DFD is also known as Context level DFD.

#### 23. What is Data Dictionary?

**Answer:**  
A data dictionary is also known as metadata. Data Dictionary is utilized to capture the information related to naming conventions of objects and files utilized in the software project.

#### 24. What is black box testing and white box testing?

**Answer:**  
**Black Box Testing:** [Black box testing](https://www.educba.com/black-box-testing/) is performed to validate the outputs along with valid inputs given. But, it does not test the implementation part of the program.

**White Box Testing:** [White Box testing](https://www.educba.com/white-box-testing/) is performed to validate the inputs, outputs and program implementation involved in its execution.

Let us move to the next software Engineering Interview Questions.

#### 25. What are the various types of software maintenance?

**Answer:**  
Maintenance types are corrective, adaptive, perfective and preventive.

**Corrective:** This type of maintenance is used to remove the errors spotted by business users.

**Adaptive:** This maintenance activity is performed to check the changes made in the hardware and software environment.

**Perfective:** This type of maintenance is used to implement changes in existing or new user requirements

**Preventive:** This maintenance activity is performed to avoid any issues in future implementations.

#### 26. Explain CASE tools?

**Answer:**  
CASE (Computer Aided Software Engineering tools) are utilized to implement, support, and accelerate various SDLC activities involved in a software project.

**1) What are the important categories of software?**

* System software
* Application software
* Embedded software
* Web Applications
* Artificial Intelligence software
* Scientific software.

**2) What is the main difference between a computer program and computer software?**

A computer program is a piece of programming code. It performs a well-defined task.  On the other hand, the software includes programming code, documentation and user guide.

**3) What is software re-engineering?**

It is a process of software development which is done to improve the maintainability of a software system.

**4) Describe the software development process in brief:**

The software development is a life cycle is composed of the following stages:

* Requirement analysis
* Specification
* Software architecture
* Implementation
* Testing
* Documentation
* Training and support
* Maintenance

**5) What are SDLC models available?**

Waterfall Model, Spiral Model, Big-bag model, Iterative Model, and V- Model are some of the famous SDLC models.

**6) What is verification and validation?**

**Verification:**

Verification is a term that refers to the set of activities which ensure that software implements a specific function.

**Validation:**

It refers to the set of activities which ensure that software that has been built according to the need of clients.

**7) In software development process what is the meaning of debugging?**

Debugging is the process that results in the removal of error. It is very important part of the successful testing.

**8) How can you make sure that your code is both safe and fast?**

In the software, development security is always first. So if the execution of the program is slow then, I will try to identify the reason out ways to its time complexity.

**9) Name two tools which are used for keeping track of software requirements?**

There many l ways to keep track of requirements.

**Two commonly used are:**

* Make a requirements specifications document to list all of the requirements.
* Create an excel sheet the list down the requirement, type, dependency, priority, etc.

**10) What is the main difference between a stubs, a mock?**

A stub is a minimal implementation of an interface which generally returns hardcoded data while mock usually verifies outputs against expectations. Those expectations are set in the test.

**11) What language do you like to write programming algorithms?**

Every developer has their views when it comes to the programming language choices. Though, one should prefer high-level languages because they are dynamic. Like C and C++ languages.

**12) What is computer software?**

Computer software is a package which includes a software program, its documentation, and user guide on how to use the software.

**13) According to you which SDLC model is the best?**

There, is no such ranking, as SDLC Models are adopted as per the need for the development process. It may differ software-to-software.

**14) Who is software project manager? What is his role?**

A software project manager is a person responsible for managing the software development project.

The project manager is doing the project planning, monitoring the progress, communication. He or she also manages risks and resources to deliver the project within time, cost, and quality constraints.

**15) What is mean by software scope?**

Software scope is a well-defined boundary. It includes all kind of activities that are done to develop and deliver the software product.

The software scope defines all functionalities and artifacts to be delivered as a part of the software. The scope also identifies what the product will do? What is not the part of the project? What is project estimation?

This process is helpful to estimate various aspects of the software product. This estimation can be decided either consulting experts or by using pre-defined formulas.

**16) How to find the size of a software product?**

The size of software product can be calculated using by following two methods

* Counting the lines of delivered code
* Counting delivered function points

**17) What are function points?**

Function points are the features which are provided by the software product. It is considered as a most important measurement for software size.

**18) What are software project estimation techniques available?**

Most widely used estimation techniques are:

* Decomposition technique
* Empirical technique

**19) What is Software configuration management?**

Software configuration management is a process of tracking and controlling changes that happen in the software.

Change control is a function which ensures that all changes made into the software system are consistent and created using organizational rules and regulations.

**20) How can you measure project execution?**

We can measure project execution using Activity Monitoring, Status Reports, and Milestone Checklists.

**21) Tell me about some project management tools.**

There are many types of management tools used as per the need for a software project. Some of them are Pert Chart, Gantt Chart, Resource Histogram, Status Reports, etc.

**22) What are software requirements?**

Software requirements are a functional description of a proposed software system. It is assumed to be the description of the target system, its functionalities, and features.

**23) What is feasibility study?**

It is a measure to find out how practical and beneficial the software project development will prove to the organization. The software analyzer conducts a study to know the economic, technical and operational feasibility of the project.

1. **Economic:**It includes the cost of training, cost of additional and tools and overall estimation of costs and benefits of the project.
2. **Technical:**It evaluate technical aspect. Is it possible to develop this system? Assessing the suitability of machine(s) and OS on which software will execute, knowledge of the software development and tools available for this project.
3. **Operational:**Here the analyst need to assess that the organization will able to adjust smoothly to the changes done as per the demand for the project. Is the problem worth solving at the estimated cost?

After, studying all this the final feasibility report is created.

**24) What are functional and non-functional requirements?**

Functional requirements are functional features which are expected by users from the proposed software product.

Non-functional requirements are related to security, performance, look, and feel of the user interface.

**25) What is software metric?**

Software Metrics offers measures for various aspects of software process which are divided into:

1. Requirement metrics: Length requirements, completeness
2. Product metrics: Number of coding Lines, Object-oriented metrics, design and test metrics.

**26) What is modularization?**

Modularization is a technique which is used for dividing a software system into various discreet modules. That is expected to carry out the tasks independently.

**27) What is cohesion?**

Cohesion is a measure that defines the intra-dependability among the elements of the module.

**28) Mentions some software analysis & design tools?**

Some of the most important software analysis and designing tools are:

* Data Flow Diagrams
* Structured Charts
* Structured English
* Data Dictionary
* Hierarchical Input Process Output diagrams
* Entity Relationship Diagrams and Decision tables

**29) What is mean by level-0 Data flow diagram?**

Highest abstraction level is called Level 0 of DFD. It is also called context level DFD. It portrays the entire information system as one diagram.

**30) What is the major difference between structured English and Pseudo Code?**

Structured English is native English language. It is used to write the structure of a program module. It uses programming language keywords. On the other hand, Pseudo Code is more like to the programming language without syntax of any specific language.

**31) What is structured design?**

Structured design is a conceptualization of problem. It also called solution design and which is based on ‘divide and conquer’ strategy.

**32) What is functional programming?**

It is a programming method, which uses the concepts of a mathematical function. It provides means of computation as mathematical functions, which also produces results irrespective of program state.

**33) What is Quality Assurance vs. Quality Control?**

Quality Assurance checks if proper process is followed while developing the software while Quality Control deals with maintaining the quality of software product.

**34) What are CASE tools?**

CASE means Computer Aided Software Engineering. They are set of automated software application programs, which are used to support, enhance and strengthen the SDLC activities.

**35) Which process model removes defects before software get into trouble?**

Clean room software engineering method removes defects before software gets into trouble.

**36) Solve this problem**

There are twenty different socks of two types in a drawer in one dark room. What is the minimum number of socks you need to take to ensure you have a matching pair?”

If you pick up three socks, they may be of the same type even if the odds are 50%. Odds never an equal reality. Therefore, the only way to ‘ensure you have a matching pair’ is to pick up at least 11 number of shocks.

**37) How you can make sure that your written code which can handle various kinds of error situation?**

I can write tests that define the expected error situations.

**38) Explain the differences between a Thread and a Process?**

A process is instance of  the computer program.In a single program it is possible to have one or more threads.

**39) Tell me the difference between an EXE and a DLL?**

An exe is an executable program while a DLL is a file that can be loaded and executed by programs dynamically. It is an external code repository for programs.  As both are different programs, reuse the same DLL instead of having that code in their file. It also reduces required storage space.

**40) What is strong-typing and weak-typing? Which is preferred? Why?**

Strong typing checks the types of variables at compile time. On the other hand,  weak typing checks the types of the system at run-time. Among them, Strong typing is always preferred because it minimizes the bugs.

**41) Describe the difference between Interface-oriented, Object-oriented and Aspect-oriented programming.**

* Interface programming is contract based.
* Object-oriented is a way to write granular objects which have a single purpose.
* Aspect Oriented Programming is to segregate the code in such a manner that various objects carry the main tasks, and the subsidiary tasks are carried by independent objects.

**42) Why using catch (exception) is always a bad idea?**  
It is a bad idea because:

* As there is no variable defined, it is not possible to read the exception
* It’s good to use an exception when you have known exception types.

**43) What type of data is passed via HTTP Headers?**

Script and metadata passed via HTTP headers.

**44) How do you prioritize requirements?**

First, you need to design a system by evaluating data structure. Then you should move on to the code structure needed to support it.

**45) Give me differences between object-oriented and component-based design?**

Object-oriented design can easily be encapsulated to some degree in component-based design.

**46) When do you use polymorphism?**

Polymorphism is used when there is a need for override functionality when inheriting class. It’s about shared classes and shared contracts.

**47) What is the difference between stack and queue?**

* Queue is always First In, First Out
* Stack is always Last In, First Out

**48) What is essential for testing the quality of the code?**

According to me, the unit testing framework is essential for testing the quality of the code.

**49) Do you think that the maintenance of software is expensive?**

According to me, maintenances of software will never be expensive if we are using proper development process.

**50) Give me differences between tags and branches?**

Tags are for versioning releases which are temporary holding places for doing such thing. However, branches are deleted when those changes are merged into the trunk.

**51) Where is a protected class-level variable available?**

Protected class-level variables are available to any sub-class derived from the base class.

**52) Is it possible to execute multiple catch blocks for a single try statement?**

Yes. Multiple catch blocks can be executed for a single try statement.

**53) When do you need to declare a class as abstract?**  
We should declare a class as abstract in the following situations:

1. When the class is inherited from an abstract class, but not all the abstract methods have been overridden.
2. In the case when minimum one of the methods in the class is declared as an abstract.

**Q.What is computer software?**

**A.** Computer software is a complete package, which includes software program, its documentation and user guide on how to use the software.

**Q.Can you differentiate computer software and computer program?**

**A.** A computer program is piece of programming code which performs a well defined task where as software includes programming code, its documentation and user guide.

**Q.What is software engineering?**

**A.** Software engineering is an engineering branch associated with software system development.

**Q.When you know programming, what is the need to learn software engineering concepts?**

**A.** A person who knows how to build a wall may not be good at building an entire house. Likewise, a person who can write programs may not have knowledge of other concepts of Software Engineering. The software engineering concepts guide programmers on how to assess requirements of end user, design the algorithms before actual coding starts, create programs by coding, testing the code and its documentation.

**Q.What is software process or Software Development Life Cycle (SDLC)?**

**A.**Software Development Life Cycle, or software process is the systematic development of software by following every stage in the development process namely, Requirement Gathering, System Analysis, Design, Coding, Testing, Maintenance and Documentation in that order.

**Q.What are SDLC models available?**

**A.** There are several SDLC models available such as Waterfall Model, Iterative Model, Spiral model, V-model and Big-bang Model etc.

**Q.What are various phases of SDLC?**

**A.** The generic phases of SDLC are: Requirement Gathering, System Analysis and Design, Coding, Testing and implementation. The phases depend upon the model we choose to develop software.

**Q.Which SDLC model is the best?**

**A.** SDLC Models are adopted as per requirements of development process. It may very software-to-software to ensuring which model is suitable.

We can select the best SDLC model if following answers are satisfied -

* Is SDLC suitable for selected technology to implement the software ?
* Is SDLC appropriate for client’s requirements and priorities ?
* Is SDLC model suitable for size and complexity of the software ?
* Is the SDLC model suitable for type of projects and engineering we do ?
* Is the SDLC appropriate for the geographically co-located or dispersed developers ?

**Q.What is software project management?**

**A.** Software project management is process of managing all activities like time, cost and quality management involved in software development.

**Q.Who is software project manager?**

**A.**A software project manager is a person who undertakes the responsibility of carrying out the software project.

**Q.What does software project manager do?**

**A.**Software project manager is engaged with software management activities. He is responsible for project planning, monitoring the progress, communication among stakeholders, managing risks and resources, smooth execution of development and delivering the project within time, cost and quality contraints.

**Q.What is software scope?**

**A.** Software scope is a well-defined boundary, which encompasses all the activities that are done to develop and deliver the software product.

The software scope clearly defines all functionalities and artifacts to be delivered as a part of the software. The scope identifies what the product will do and what it will not do, what the end product will contain and what it will not contain.

**Q.What is project estimation?**

**A.** It is a process to estimate various aspects of software product in order to calculate the cost of development in terms of efforts, time and resources. This estimation can be derived from past experience, by consulting experts or by using pre-defined formulas.

**Q.How can we derive the size of software product?**

**A.** Size of software product can be calculated using either of two methods -

* Counting the lines of delivered code
* Counting delivered function points

**Q.What are function points?**

**A.** Function points are the various features provided by the software product. It is considered as a unit of measurement for software size.

**Q.What are software project estimation techniques available?**

**A.** There are many estimation techniques available.The most widely used are -

* Decomposition technique (Counting Lines of Code and Function Points)
* Empirical technique (Putnam and COCOMO).

**Q.What is baseline?**

**A.** Baseline is a measurement that defines completeness of a phase. After all activities associated with a particular phase are accomplished, the phase is complete and acts as a baseline for next phase.

**Q.What is Software configuration management?**

**A.** Software Configuration management is a process of tracking and controlling the changes in software in terms of the requirements, design, functions and development of the product.

**Q.What is change control?**

**A.** Change control is function of configuration management, which ensures that all changes made to software system are consistent and made as per organizational rules and regulations.

**Q.How can you measure project execution?**

**A.** We can measure project execution by means of Activity Monitoring, Status Reports and Milestone Checklists.

**Q.Mention some project management tools.**

**A.** There are various project management tools used as per the requirements of software project and organization policies. They include Gantt Chart, PERT Chart, Resource Histogram, Critical Path Analysis, Status Reports, Milestone Checklists etc.

**Q.What are software requirements?**

**A.** Software requirements are functional description of proposed software system. Requirements are assumed to be the description of target system, its functionalities and features. Requirements convey the expectations of users from the system.

**Q.What is feasibility study?**

**A.** It is a measure to assess how practical and beneficial the software project development will be for an organization. The software analyzer conducts a thorough study to understand economic, technical and operational feasibility of the project.

* **Economic**- Resource transportation, cost for training, cost of additional utilities and tools and overall estimation of costs and benefits of the project.
* **Technical**- Is it possible to develop this system ? Assessing suitability of machine(s) and operating system(s) on which software will execute, existing developers’ knowledge and skills, training, utilities or tools for project.
* **Operational**- Can the organization adjust smoothly to the changes done as per the demand of project ? Is the problem worth solving ?

**Q.How can you gather requirements?**

**A.** Requirements can be gathered from users via interviews, surveys, task analysis, brainstorming, domain analysis, prototyping, studying existing usable version of software, and by observation.

**Q.What is SRS?**

**A.** SRS or Software Requirement Specification is a document produced at the time of requirement gathering process. It can be also seen as a process of refining requirements and documenting them.

**Q.What are functional requirements?**

**A.** Functional requirements are functional features and specifications expected by users from the proposed software product.

**Q.What are non-functional requirements?**

**A.** Non-functional requirements are implicit and are related to security, performance, look and feel of user interface, interoperability, cost etc.

**Q.What is software measure?**

**A.** Software Measures can be understood as a process of quantifying and symbolizing various attributes and aspects of software.

**Q.What is software metric?**

**A.** Software Metrics provide measures for various aspects of software process and software product. They are divided into –

* Requirement metrics : Length requirements, completeness
* Product metrics :Lines of Code, Object oriented metrics, design and test metrics
* Process metrics: Evaluate and track budget, schedule, human resource.

**Q.What is modularization?**

**A.** Modularization is a technique to divide a software system into multiple discreet modules, which are expected to carry out task(s) independently.

**Q.What is concurrency and how it is achieved in software?**

**A.** Concurrency is the tendency of events or actions to happen simultaneously. In software, when two or more processes execute simultaneously, they are called concurrent processes.

### **Example**

While you initiate print command and printing starts, you can open a new application.

Concurrency, is implemented by splitting the software into multiple independent units of execution namely processes and threads, and executing them in parallel.

**Q.What is cohesion?**

**A.** Cohesion is a measure that defines the degree of intra-dependability among the elements of the module.

**Q.What is coupling?**

**A.** Coupling is a measure that defines the level of inter-dependability among modules of a program.

**Q.Mentions some software analysis & design tools?**

**A.** These can be: DFDs (Data Flow Diagrams), Structured Charts, Structured English, Data Dictionary, HIPO (Hierarchical Input Process Output) diagrams, ER (Entity Relationship) Diagrams and Decision tables.

**Q.What is level-0 DFD?**

**A.** Highest abstraction level DFD is known as Level 0 DFD also called a context level DFD, which depicts the entire information system as one diagram concealing all the underlying details.

**Q.What is the difference between structured English and Pseudo Code?**

**A.** Structured English is native English language used to write the structure of a program module by using programming language keywords, whereas, Pseudo Code is more close to programming language and uses native English language words or sentences to write parts of code.

**Q.What is data dictionary?**

**A.** Data dictionary is referred to as meta-data. Meaning, it is a repository of data about data. Data dictionary is used to organize the names and their references used in system such as objects and files along with their naming conventions.

**Q.What is structured design?**

**A.** Structured design is a conceptualization of problem into several well-organized elements of solution. It is concern with the solution design and based on ‘divide and conquer’ strategy.

**Q.What is the difference between function oriented and object oriented design?**

**A.** Function-oriented design is comprised of many smaller sub-systems known as functions. Each function is capable of performing significant task in the system. Object oriented design works around the real world objects (entities), their classes (categories) and methods operating on objects (functions).

**Q.Briefly define top-down and bottom-up design model.**

**A.** Top-down model starts with generalized view of system and decomposes it to more specific ones, whereas bottom-up model starts with most specific and basic components first and keeps composing the components to get higher level of abstraction.

**Q.What is the basis of Halstead’s complexity measure?**

**A.** Halstead’s complexity measure depends up on the actual implementation of the program and it considers tokens used in the program as basis of measure.

**Q.Mention the formula to calculate Cyclomatic complexity of a program?**

**A.** Cyclomatic complexity uses graph theory’s formula: V(G) = e – n + 2

**Q.What is functional programming?**

**A.** Functional programming is style of programming language, which uses the concepts of mathematical function. It provides means of computation as mathematical functions, which produces results irrespective of program state.

**Q.Differentiate validation and verification?**

**A.** Validation checks if the product is made as per user requirements whereas verification checks if proper steps are followed to develop the product.

Validation confirms the right product and verification confirms if the product is built in a right way.

**Q.What is black-box and white-box testing?**

**A.** Black-box testing checks if the desired outputs are produced when valid input values are given. It does not verify the actual implementation of the program.

White-box testing not only checks for desired and valid output when valid input is provided but also it checks if the code is implemented correctly.

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Black Box Testing** | **White Box Testing** |
| Knowledge of software program, design and structure essential | No | Yes |
| Knowledge of Software Implementation essential | No | Yes |
| Who conducts this test on software | Software Testing Employee | Software Developer |
| baseline reference for tester | Requirements specifications | Design and structure details |

**Q.Quality assurance vs. Quality Control?**

**A.** Quality Assurance monitors to check if proper process is followed while software developing the software.

Quality Control deals with maintaining the quality of software product.

**Q.What are various types of software maintenance?**

**A.** Maintenance types are: corrective, adaptive, perfective and preventive.

* **Corrective**

Removing errors spotted by users

* **Adaptive**

tackling the changes in the hardware and software environment where the software works

* **Perfective maintenance**

implementing changes in existing or new requirements of user

* **Preventive maintenance**

taking appropriate measures to avoid future problems

**Q.What is software re-engineering?**

**A.** Software re-engineering is process to upgrade the technology on which the software is built without changing the functionality of the software. This is done in order to keep the software tuned with the latest technology.

**Q.What are CASE tools?**

**A.** CASE stands for Computer Aided Software Engineering. CASE tools are set of automated software application programs, which are used to support, accelerate and smoothen the SDLC activities.