Short Answers:

1. What are the essential attributes of good software?

- Required functionality.
- Performance to the user.
- Maintainable.
- Dependable.
- Usable.

2. What are the two fundamental types of software product?

- Generic Products (stand-alone)
- Customized (bespoke) products (made for certain situation/company).

3. What is software engineering?

Software engineering is an engineering discipline that is involves all aspects of software production.

4. What are the four fundamental activities in software processes?

- Software specification.
- Software development.
- Software validation.
- Software evolution.

5. What is the distinction between computer science and software engineering?

- Computer Sciences is about the theories and underlining methods for computer systems and software.
- Software Engineering is about is about the actually practices involved in producing good software.

6. What are the 3 general issues that affect many different types of software?

- Heterogeneity.
- Business and social change.
- Security and trust.

7. What software engineering fundamentals apply to all types of software systems?

- Which development process to use.
- Dependability and performance.
- Understanding and managing specification and requirements.
- Reuse resources that have already been developed.

8. What are three key characteristics of the engineering of web-based software engineering?

- Reusing software has become prominent, so modular and usage of pre-existing software.
- Web –based systems change quickly over time and must be designed incrementally
- User interfaces are held back by web-browser, and difficulty of technologies to be integrated.

9. List the 3 generic process mode is that are used in software engineering?

- The Waterfall model
- Incremental development
- Reuse-oriented software engineering

10. What are the three benefits of incremental development, compared to the waterfall model?

- Cost of accommodating change, is reduced.
- Easier to get customer feedback.
- Quicker production and deployment of software to customers.

11. What are the development stages in reuse-based development?

- Component analysis.
- Requirements modification.
- System design and reuse
- Development and integration

12. What are the principal requirements engineering activities?

- Feasibility study
- Requirements elicitation and analysis.
- Requirements specification.
- Requirements validation.

13. What are the advantages of using incremental development and delivery?

- Customers have early access to prototypes, allowing for better requirements idea.
- Customers don't have to wait for the entire system.
- It maintains the benefits of incremental development.
- Highest-priority services are first, allowing for the must testing over development.

14. What are the 4 sectors in each loop in Boehm's spiral model?

- Objective setting. (determine objectives, alternatives, and constraints)
- Risk assessment and reduction. (evaluate alternatives, identify, resolve risks)
- Development and validation. (develop, verify next-level product)
- Planning. (plan next phase)

15. What are the six fundamental best practices in the RUP?

- Develop software iteratively.
- Manage Requirements.
- Use Component-based architectures.
- Visually model software.
- Verify software quality.
- Control changes to software.

16. What are the shared characteristics of different approaches to rapid software development?

- Software specification, design, and implementation are intertwined.
- The system is produced in iterations.
- System user interfaces are made using interactive development system.

17. For what types of system are agile approaches to development particularly likely to be successful?

- The software company is developing a small or medium-sized product.
- Custom system development for the intern company.

18. List the 5 principles of agile methods.

- Customer involvement.
- Incremental delivery.
- People not process.
- Embrace change.
- Maintain simplicity.

19. List 4 questions that should be asked when deciding whether or not to adopt an agile method of software development.

- Is incremental delivery strategy possible, and realistic?
- What technologies are available to support agile development?
- What type of system is being produced?
- How large is the system being produced?

20. What are three important characteristics of extreme programming?

- User requirements as scenarios (called user stories).
- Pair programming.
- Test-first development.

21. What is test-first development?

- Test first development uses automated testing framework, to write tests before the actually code is implemented.

22. What are the possible problems of test-first development?

- Programmers sometimes take shortcuts when writing tests.
- Some tests are difficulty to implement, when designing incrementally.
- It is unsure the exact amount of code coverage your tests actually have.

23. Briefly describe the advantages of pair programming.

- Supports collective ownership and responsibility for a system.
- Works as an informal review since two people always look over the code.
- Helps constant refactoring of the code.

24. What is a Scrum sprint?

- A Scrum sprint is a short fixed time where certain features are assessed, selected, developed and reviewed.

25. What are the barriers to introducing agile methods into large companies?

- Project managers might not have experience with agile.
- Large organizations usually have procedures and standards that get in the way of agile.
- Agile methods work best when all members are of a high level, which might not happen at a large organization.
- The culture of the organization might get in the way.

<u>Questions</u>:

1. What are the situations where waterfall model is most appropriate?

- Requirements are well understood.
- Requirements are unlikely to change during production.
- When documentation is important.
- Long life cycle.
- Common management model for the whole project is required.

2. Provide an example of a software project that would be amenable to the waterfall model.

- Military systems, requirements are known well in advance and do not change.
- Documentation is important.
- The military uses a common management model.

3. What are the situations where waterfall model is least appropriate?

- The customer doesn't fully know what he wants.
- Development must react to change.
- Short life cycle.

4. Briefly describe at least four good things about the waterfall model.

- Good documentation, allows new members to catch up, and makes maintenance easier.
- Common management, and deliverables, allows easier management and visibility for the company.
- Known requirements, allow the final product to perfectly match the user's expectations.
- Requirements unlikely to change, allows a better system design from the beginning that won't suffer from constantly integrating new things.

5. The waterfall model is sometimes described as a "document-driven' model, whereas the spiral model is described as a "risk-driven" model. Answer the following two questions.

a. What determines when to progress to the next phase or iteration in each model?

- Waterfall: The next phase does not start until the previous phase has been finished.
- Spiral: The next phase starts only after the planning for it is complete, i.e after it has been reviewed and decided to continue to the next loop.

b. How are changes in requirements accommodated in each model?

- Waterfall: Changes are not accommodated after a few iterations; if they are it will be very costly requiring a restart of the process.
- Spiral: Is a change tolerant process, during the planning phase of the next loop, new requirements or changes can be accommodated.

6. When to use incremental development model?

- When the requirements might change.
- When customer feedback is important to craft the final result.
- The customer could use a partially complete product if given to them.

7. You are currently working for a software company that is working on smartphone apps, which is the hottest, fastest growing area. You are a software engineering hired to write the software for the smart phones. Unfortunately, the requirements are constantly being changed and will not be finalized until a few weeks before shipping. What process model would you use and why?

- Agile method. (incremental with small releases every couple of weeks)
    - You can use customer experience to design a more intuitive user interface, and experience.
    - As new requirements are given to you, you can adopt your design to incorporate them.

    - An interactive development system allows quick development of user interfaces very important in smartphone app production.

8. Which software process model best supports development of applications with sophisticated graphical user interface, and why?

- Incremental delivery.
    - Customers can use early increments as prototypes, and help refine the user interface.
    - The process maintains the benefits of incremental development so its relatively easy to incorporate changes.
    - The important GUI features will be implemented first, and thus will have the most testing, ensuring their smooth running.