1. What are the fundamental activities that are common to all software

processes?

Software specification (or requirements engineering): Define the main functionalities of the software and the constrains around them.

Software design and implementation: The software is to be designed and programmed.

Software verification and validation: The software must conform to its specification and meets the

customer needs.Software evolution (software maintenance): The software is being modified to meet customer and market requirements changes.

2. Explain why the waterfall model of the software process is not an accurate reflection of software development activities.

Few business systems have stable requirements. Hard to respond to change. The SDLC is not linear but require iteration.

Customer must have patience system is delivered at the end of the project.

Document heavy is the necessary on every project.

3. Why are iterations usually limited when the waterfall model is used?

Document driven

Sequential phases

The cost of iterations is costly requiring a lot of rework.

4. Give an example of a software project that would benefit from using the waterfall model. Explain your reasoning.

Project: Stable requirements, Unlikely to change, contractual obligations, Large projects need structure.

Rewrite of an existing system or enhancement of an existing.

5. When is it appropriate to use prototyping in the SDLC?

6. For throwaway prototyping:

Outline the strengths and and. problems.

Describe the sort of project that would be an ideal application. Now

describe the type of situation where the model may lead to

difficulties.

7. Explain the difference between iterative and incremental

development.

8. For incremental:

Outline the advantages and disadvantages.

Describe the sort of project that would be an ideal application.

Now describe the type of situation where the model may lead to difficulties.

9. Give an example of a software project that would benefit from using

the spiral model. Explain your reasoning.

Accelerated delivery to the custom – Can see progress.

Early increments act as a prototype to help capture requirements. Highest priority requirements are delivered first – most tested.

Lower risk of project frailer.

Reduced maintenance costs.

Hard to map functionality to increments.

Some systems need a common set of facilities.

Functionality must be delivered quickly tight framework.

Partial functionality can deliver and expanded in subsequent releases.

Unsuitable for tightly coupled systems where all functionality must work.

Controlling software.

10.Giving reasons for your answer based on the type of system being developed, suggest the most appropriate software process model which might be used as a basis for developing the following systems:

* A virtual reality system to support software maintenance. Requirements will change and there will be an extensive user interface component;
* A College accounting system that replaces an existing system Requirements are fairly well-known and are stable;
* An interactive system for railway passengers that finds train times from terminals installed in stations complex user interface but which must be stable and reliable.

Incremental + prototyping agile

Waterfall

Incremental + prototyping agile

11.You work in an organisation which develops diverse software applications ranging from traditional well documented business applications, to highly interactive products and other products which the customer wants delivered to the market as soon as possible with additional functionality being delivered in subsequent releases. Your manager has decided to standardise on one process model the waterfall model, how would you respond?

http://www.cnrpark.it/