12 Conclusion - exam questions

You should know the answers to these questions

Name 3 items from the code of ethics and provide a one-line explanation.

- **Public Interest**: Act consistently with the public interest to ensure societal welfare.
- **Client and Employer**: Prioritize the best interests of clients and employers within the bounds of the public good.
- **Product**: Strive to produce high-quality software that meets professional standards.

If you are an independent consultant, how can you ensure that you will not have to act against the code of ethics?

Include a clause in contracts that allows termination if compliance with the code of ethics is compromised.

What would be a possible metric for measuring the amount of innovation of a manufacturing company?

The number of products in the portfolio that are younger than five years.

Explain the 2 main steps of test amplification: input amplification and assertion amplification

- **Input Amplification**: Modifies the test setup by injecting boundary conditions and new states to explore untested paths.
- **Assertion Amplification**: Generates assertions by observing runtime behavior to verify more properties of the object.

When you chose the "No Silver Bullet" paper

What's the distinction between essential and accidental complexity?

- **Essential Complexity**: Inherent to the nature of software, involving abstract, precise concepts like algorithms and data.
- **Accidental Complexity**: Arises from current tools and techniques used to implement software.

Name 3 reasons why the building of software is essentially a hard task? Provide a one-line explanation.

- **Complexity**: High interdependence and numerous states in software.
- **Conformity**: Interfaces must align with arbitrary, changing external systems.
- **Changeability**: Software must frequently evolve with new demands.

Why is "object-oriented programming" no silver bullet?

While it improves design, it primarily removes accidental complexities without addressing the inherent complexity of software.

Why is "program verification" no silver bullet?

It can ensure correctness relative to specifications but does not eliminate specification errors or the cost of verification itself.

Why are "components" a potential silver bullet?

Reusable components can significantly reduce development time by allowing pre-built, tested modules to be assembled rather than coded from scratch.

When you chose the "Killer Robot" paper

Which regression tests would you have written to prevent the "killer robot"?

…

Was code reviewing applied as part of the QA process? Why (not)?

…

Why was the waterfall process disastrous in this particular case?

…

Why was the user-interface design flawed?

…