Assignment#5 (Kipshakbaev Sanzhar)

Question:

Suggest six reasons why software dependability is important in most sociotechnical systems.

Answer:

• To avoid the introduction of accidental errors into the system during software specification and development.

• To design verification and validation processes that are effective in discovering residual errors that affect the dependability of the system.

• To design protection mechanism that guard against external attacks that can compromise the availability or security of the system.

• To configure the deployed system and its supporting software correctly for its operating environment.

• System failure costs may be enormous.

• Users often reject systems that are unreliable, unsafe, or insecure.

Question:

Explain how programming language support of exception handling can contribute to the reliability of software systems

Answer:

The paper discusses the basic concepts underlying the issue of software reliability, and argues that programmed exception handling is inappropriate for dealing with suspected software errors. Instead it is shown, using an example program, how exception handling can be combined with the recovery block structure. The result

is to improve the effectiveness with which problems due to anticipated faulty input data, hardware components, etc., are dealt with, while continuing to provide means for recovering from unanticipated faults, including ones due to residual software design errors

Question:

Should software engineers working on the specification and development of safety-related systems be professionally certified or licensed in some way? Explain your reasoning.

Answer:

Software engineers should working on the specification and development of safely-related systems be professionally certified in some way because:

• Some applications contain millions of lines of code for making them comparable in complexity to the most complex modern machines with very high reliability expected.

• Software engineers advocate many different technologies and practices, with much disagreement.

• Software engineers use a wide variety of technologies: compilers, code repositories, text editors.

• They also use a wide variety of practices to carry out and coordinate their efforts: pair programming, code reviews and daily stand up meetings.

• Some organizations have specialists to perform each of the tasks in the software development process. Other organizations required software engineers to do many or all of them.

• In large projects, people may specialize in only one role. In small projects, people may fill several or all roles at the same time.

So, certification of software engineers is an important issue. Some see it as a tool to improve professional practice.