1. Software doesn't only refer to code. Computer programs, procedures, and possibly associated documentation and data pertaining to the operation of a computer system.
2. Because after adding extra personnel, they have to spend time to understand the entire project before they can help on the project tasks. It usually will take longer time especially for huge and complex projects.
3. Software functional quality reflects how well it complies with or conforms to a given design, based on functional requirements or specifications.
   Example: 1.A bank system should provide all transactions for customers.
   2.A store should provide customers receipts for each purchase.
   3. Internet service providers should provide customers their bills before each charge.

   Software structural quality refers to how software meets non-functional requirements that support the delivery of the functional requirements, such as robustness or maintainability.
   Example: 1.A bank system categorize customers' transactions in different categories, such as education, groceries.
   2.A store provide customers receipts for each purchase according to customers' preferred way, such as email, text.
   3. Internet service providers provide customers their bills a week before they stop providing services.

4. line 1 is a fault, line 2 and 4 are errors when x=100, the entire program is a failure when x=100.

5. region 1: tested part whose behaviors programmed behaviors meet customers' specifications. Testers should try their best to maximize this part as possible.

region 2: untested part whose behaviors programmed behaviors meet customers' specifications. As the complement of region 1, it may contain unknown bugs.

region 3: tested part whose behaviors programmed by developers while unspecified by customers. The extra functionalities pass the test while not desired by customers.

region 4: tested part whose behaviors specified by customers while not implemented by developers. It refers to missing functionalities under testing.

region 5: untested part whose behaviors specified by customers while not implemented by developers.

region 6: untested part whose behaviors programmed by developers while unspecified by customers.

region 7: tested part whose behaviors is not implemented by developers and not specified by customers.

region 8: All other possible behaviors can have some bugs.

6. The IntelliTest contains traditional unit test functionalities while implements input automatically generations and test information collections which allow programmers to save efforts on writing test inputs and test oracles. The IntelliTest will automatically generate precise test input values without any user interventions. Certainly, programmers can have some guidance on input generations to fit their requirements. After collecting all information, including correctness assertions, the test cases will be generated and judge automatically without programmers' efforts on writing traditional unit test. What's more, all testing information will be collected by IDE, such as input, output, and automatically generated test cases, so that programmers can run the test repeatedly and fix bugs.