

National Exams December 2014

04-SOFT-B17: Programming Language Paradigm

3 hours duration

NOTES:

- If doubt exists as to the interpretation of any question, the candidate is urged to submit with the answer paper, a clear statement of any assumptions made.
- This is an OPEN BOOK EXAM however only one textbook allowed plus notes and any non-communicating calculator is permitted.
- This exam consists of 10 questions, and each question has equal value.
- Most questions require an answer in essay format.
- Clarity and organization of the answer are important.

1. A **class** is the standard approach to defining a datatype in any object-oriented programming language. What is a data type? Provide a definition of a data type with an example.
2. Pick any object-oriented programming language that you are familiar with and discuss how that language handles object equality by default. What considerations should be made when deciding whether two objects are equal? What properties should be satisfied by a method that checks if two objects are equal
3. **Subclassing** is an approach to **subtyping** in an object-oriented language. What is the substitution principle for **subtyping**? (This is sometimes known as the Liskov Substitution Principle.) Provide an example where subclassing can violate the substitution principle. (Hint: If we have a class **Rectangle** and we create a subclass **Square**, is **Square** guaranteed to be a subtype of **Rectangle**.)
4. What is the difference between method **overriding** and method **overloading**? Provide an example to explain your answer.
5. When is it preferable to use object composition and not object inheritance?
6. Why are **exceptions** a better error handling mechanism than returning special values from a method/procedure? Are there any disadvantages to using exceptions? Use examples to explain your answer.
7. What is the most significant difference between a functional programming language and a procedural programming language? How does this difference become significant in today's context with big data analysis through programming frameworks such as Hadoop?
8. How is actor-oriented programming different from object-oriented programming? Provide an example for a situation when actor-oriented programming is preferable to object-oriented programming.
9. What is a lambda expression?
10. What are immutable objects? How does mutability/immutability affect the development of thread-safe programs?