PCPF Question Bank 2022-23

1. List and explain different problem domains where we can make use of scripting languages.

2. Which are important factors to be considered, while making a choice of a programming language ?

3. What is pattern matching? How does scripting languages utilise the power of pattern matching?

4. What is Polymorphism? Explain different programming constructs that make use of the concept of polymorphism in any object oriented programming language.

5. What is currying? Define a haskell function "**add3**" that adds 3 inputs provided to it. Define a curried version of this function named "**sumplus1000"** that adds 1000 to its two inputs.

6. Explain synchronization. How can it be implemented by spinning and blocking?

7. Which principles of storage allocation mechanism used to manage an object's space?

8. Discuss six principal options used to create thread of control in concurrent programs.

9. What are clauses, terms, and structures in Prolog? What are facts, rules, and queries ? (Note: Give examples for each)

10. What are constructors and destructors? Explain with help of example the order of calling of constructors amongst inherited classes.

11. Describe different parameter passing modes for subroutines.

12. Define a haskell function named "addUs" that adds 2 input numbers. Using this function as a building block, define a Haskell function "multiplyUs" that multiplies two input numbers.

    The multiplyUs function should cater to following requirements:

    1. Inputs may be signed numbers e.g. "multiplyUs (-2) * (3)" should result in "-6"and "multiplyUs (-2) * (-6)" should result in "12"

    2. It should use guard expressions and recursion.

    3. No need to write the main function to do user interaction writing definition for "addUs" and "multiplyUs" is sufficient.

13. What are Scripting Languages? List common characteristics of scripting

languages.

14. Explain with example the difference between declarative and imperative programming paradigm.

15. Briefly describe the process of resolution and unification in logic programming with example.

16. What is Data Hiding in Object Oriented Programming Paradigm? Describe how data hiding is implemented in C++ or Java.

17. Define Haskell function that inputs one operator +,-,*,^ and two operands which may be Int, Integer, Float or Double. The function will perform the operation and computes the result. Clearly mention the type signature for the function.

18. Note: Students are not expected to write the main function and do user IO.

19. Explain the different communication and synchronization techniques in Concurrent Programming model.

20. What is type checking and type clash? What do you mean by statically typed and strongly typed programming language? List any two statically typed languages.

21. Explain following terms: Concurrent system, Parallel system, Distributed system, Race condition, Context switching.

22. What mathematical formalism underlies functional programming?
23. Write a note on Naming and Scoping rules for scripting languages.

24. Demonstrate in object oriented programming how to resolve a call to one of the multiple methods with the same name and signature in the superclass and subclass is made.

25. What is the role of an Exception Handler in a programming language ? Briefly explain important tasks it performs.

26. Explain how Prolog differs from imperative languages in its handling of arithmetic
27. Justify the following statement, “No single factor determines whether a programming language is good.
28. Justify the following statement, “No single factor determines whether a programming language is good.”
29. Explain concept of currying in haskell with an example

30. Explain what are facts, rules, and queries in logic programming with example.
31. The haskell function head defined in prelude, returns the first element of a list and throws an exception when we try to apply it on an empty list. Define two variants of this function (you can use different names) that work exactly like head function except in the case of an empty list input they will show [] as output instead of throwing an exception. You must use the following constructs in Haskell for defining the functions. a. First implementation should make use of pattern matching. b. Second implementation uses guard equations Note: Students are not expected to write the main function and do uer IO.
32. Describe different parameter passing modes.
33. Compare heap based and stack based principle storage allocation mechanisms.
34. Write a note on Lambda Calculus.
35. What is the difference between normal-order and applicative-order evaluation? What is lazy evaluation?
36. Describe the difference between forward chaining and backward chaining. Which is used in Prolog by default?
37. Define a haskell function named "addUs" that adds 2 input numbers. Using this function as a building block, define a Haskell function "multiplyUs" that multiplies two input numbers. The multiplyUs function should cater to following:
    1. Inputs may be signed numbers e.g. "multiplyUs (-2) * (3)" should result in "-6" and "multiplyUs (-2) * (-6)" should result in "12"
    2. It should use guard expressions and recursion.
    3. No need to write the main function to do user interaction writing definition for "addUs" and "multiplyUs" is sufficient.
38. Discuss Scope with reference to binding in program. Also compare static and dynamic scoping.
39. What is the importance of binding time? What is the difference between lifetime of a name to object binding and visibility.
40. What are the features of Expression evaluation? What do you mean by side effect of an expression?
41. Discuss about Expression evaluation ordering with examples.
42. Explain about the foll: storage allocation mechanisms used.
    a.Stack Based allocation

43. What do you mean by Lazy evaluation?
44. Define the terms names, scope and binding. How is a declaration different from a definition?
45. Examine about the following scope rules with respect to a programming language.
    a. Static scoping
    b. Dynamic scoping
46. Write the pseudo code for the same, and show how output differs in each case.
47. Explain about following concepts of binding of referencing environments with reference to a programming language.
    a.Object closures
48. What are the benefits of side effect freedom in programming?
49. What are Data types? List the different types.
50. What are Non converting Type Casts?