

	Solve any Four out of Six	5 marks each
Q2.		
A	Explain how Prolog differs from imperative languages in its handling of arithmetic.	
B	Justify the following statement, "No single factor determines whether a programming language is good."	
C	Explain concept of currying in haskell with an example.	
D	Explain what are facts, rules, and queries in logic programming with example.	

226 Q.3

Solve any Four out of Six		5 marks each
Q2	Describe the different Types in Haskell.	
A	With an example explain how constructors are different from other member functions	
B	How scripting languages differ from other programming languages	
C	Mention features of Functional Programming languages.	
D	When and why do we use "is" instead of "=" in Prolog?	
E	Explain lifecycle of a thread.	
F		
Solve any Two Questions out of Three		10 marks each
Q3	Write a Prolog code to find if a list is sorted or not.	
A	Explain the Exception handling mechanism with example	
B	Explain Type System and Type checking.	
C		

Solve any Two Questions out of Three

Explain the different mechanisms in which storage is allocated to a program and data. 10 marks each

i) Explain the concept of higher order functions in Functional Programming. Name and specify input output characteristics of any 2 Haskell higher order functions. (4 marks)

ii) Write your own Haskell implementation for any one of the functions you stated in the previous question. (13 marks)

iii) Write a corresponding imperative algorithm to achieve the same input output characteristics. (15 marks)

Note: you may assume and state a suitable data structure while writing imperative algorithm.

Discuss how to implement Polymorphism in C++ with example program.

Q2.	Solve any Four out of Six	5 marks each
A	What are Scripting Languages? List common characteristics of scripting languages.	
B	Explain with example the difference between declarative and imperative programming paradigm.	
C	Briefly describe the process of resolution and unification in logic programming with example.	
D	What is Data Hiding in Object Oriented Programming Paradigm? Describe how data hiding is implemented in C++ or Java.	
E	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. Note: Students are not expected to write the main function and do user IO.	
F	Explain the different communication and synchronization techniques in Concurrent Programming model.	
		5 marks each
3.	Solve any Four out of Six	
A	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.	
B	Explain following terms: Concurrent system, Parallel system, Distributed system, Race condition, Context switching.	

C	What mathematical formalism underlies functional programming?
D	Write a note on naming and scoping rules for scripting languages.
E	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.
F	What is the role of an Exception Handler in a programming language? Briefly explain important tasks it performs.

E	<p>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.</p> <p>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.</p> <p>You must use the following constructs in Haskell for defining the functions.</p> <ol style="list-style-type: none"> First implementation should make use of pattern matching. Second implementation uses guard equations <p>Note: Students are not expected to write the main function and do user IO.</p>
F	Describe different parameter passing modes.
<p>Solve any Four out of Six</p> <p>5 marks each</p>	
A	Compare heap based and stack based principle storage allocation mechanisms.
B	Write a note on Lambda Calculus.
C	What is the difference between normal-order and applicative-order evaluation? What is lazy evaluation?
D	Describe the difference between forward chaining and backward chaining. Which is used in Prolog by default?
E	<p>Define a haskell function named "addUs" that adds 2 input numbers.</p> <p>Using this function as a building block, define a Haskell function "multiplyUs" that multiplies two input numbers.</p> <p>The multiplyUs function should cater to following:</p> <ol style="list-style-type: none"> Inputs may be signed numbers e.g. "multiplyUs (-2) * (3)" should result in "-6" and "multiplyUs (-2) * (-6)" should result in "12" It should use guard expressions and recursion. No need to write the main function to do user interaction writing definition "addUs" and "multiplyUs" is sufficient.
F	Discuss Scope with reference to binding in program. Also compare static and dynamic scoping.