



PART – A (SHORT ANSWER QUESTIONS)

S. No	Question	Blooms Taxonomy Level	Course Outcome
UNIT – I			
1	Define Programming Language.	Knowledge	1
2	Difference between sentence and sentential form.	Analyze	2
3	Define Syntax and Semantics.	Knowledge	2
4	Differentiate between Syntax and Semantics.	Analyze	1
5	Write BNF notation for i) For loop ii) if-else condition	Understand	2
6	Give grammars for simple assignment statements.	Understand	2
7	Give unambiguous grammar for if-then-else.	Understand	2
8	Define Parse trees.	Knowledge	2
9	Define Denotational semantics.	Knowledge	1
10	Define Operational semantics.	Knowledge	1
11	Define Axiomatic semantics.	Knowledge	2
12	Differentiate compiler and interpreter.	Analyze	2

13	Describe language recognizers.	Understand	1
14	Describe generators.	Understand	1
15	Distinguish simplicity and orthogonality.	Analyze	2
UNIT – II			
1	Distinguish static and dynamic scoping.	Analyze	2
2	Define associative arrays.	Knowledge	2
3	Define guarded commands?	Knowledge	2
4	Distinguish named type and structure type compatibility.	Analyze	3
5	List the merits of sub range types.	Knowledge	3
6	Differentiate union and enumeration.	Analyze	3
7	Define data type.	Knowledge	2
8	List the merits of type checking.	Knowledge	2
9	Define user defined data type.	Knowledge	2
10	Define widening and narrowing conversions?	Knowledge	3
UNIT – III			
1	Define scope and lifetime of a variable.	Knowledge	3
2	Differentiate shallow and deep binding.	Analyze	3
3	Define subprogram.	Knowledge	3
4	Define procedures.	Knowledge	3
5	Define local referencing environment.	Knowledge	4
6	Differentiate static and dynamic binding.	Analyze	4
7	Define the phrase 'Adhoc-binding'	Knowledge	4
8	Differentiate co-routines and subprograms	Analyze	4
9	Define the phrase 'overloaded subprogram'	Analyze	3
10	Discuss Generic sub programs.	Understand	4
UNIT – IV			
1	Define an exception.	Knowledge	4
2	Define a thread.	Knowledge	4
3	Define concurrency.	Knowledge	5
4	Define binary semaphore.	Knowledge	5
5	Define monitors.	Knowledge	5
6	Define mutual exclusion.	Knowledge	5
7	Define deadlock.	Knowledge	5
8	Define an abstract data type.	Knowledge	5
9	Define logic programming language.	Knowledge	5
10	Define data abstraction.	Knowledge	6
UNIT – V			
1	Give the meaning of lazy evaluation mean.	Understand	6
2	Define procedural abstraction.	Knowledge	6
3	List few characteristics of Python language.	Knowledge	7
4	Define functional language.	Knowledge	7
5	Define imperative language.	Knowledge	7

S. No	Question	Blooms Taxonomy Level	Course Outcome
6	Give the meaning of scripting language.	Understand	6
7	List few examples of scripting languages.	Knowledge	7
8	List keywords of Python language.	Knowledge	7
9	List data types of Python language.	Knowledge	7
10	Define the term separate compilation in Python.	Knowledge	7

PART – B (LONGANSWER QUESTIONS)

S. No	Question	Blooms Taxonomy Level	Course Outcome
UNIT - I			
1	Discuss language evaluation criteria and the characteristics that affect them.	Understand	1
2	List the potential benefits of studying programming language concept.	Knowledge	2
3	Explain syntax of a “for” statement in PASCAL using BNF notation and syntax graphs.	Understand	2
4	Explain syntax of declaration statement in PASCAL using BNF notation and syntax graphs.	Understand	1
5	Compute the weakest precondition for each of the following simple assignment statements and post conditions. $a=2*(b-1)-1 \{a>0\}$ $b=(c+10)/3 \{b>6\}$ $a=a+2*b-1 \{a>1\}$ $X=2*y+x-1 \{x>11\}$	Apply	2
6	Write BNF notation for following: a) For loop b) If-else condition c) Structure definition	Apply	2
7	Define left most derivation. Prove that the following grammar is ambiguous. $\langle \text{program} \rangle \rightarrow \text{begin} \langle \text{stmt_list} \rangle \text{end}$ $\langle \text{stmt_list} \rangle \rightarrow \langle \text{stmt} \rangle$ $ \langle \text{stmt} \rangle ; \langle \text{stmt_list} \rangle$ $\langle \text{stmt} \rangle \rightarrow \langle \text{var} \rangle = \langle \text{expression} \rangle$ $\langle \text{var} \rangle \rightarrow A B C$ $\langle \text{expression} \rangle \rightarrow \langle \text{var} \rangle + \langle \text{var} \rangle$ $ \langle \text{var} \rangle - \langle \text{var} \rangle$ $ \langle \text{var} \rangle$	Knowledge	2
8	State the given grammar is ambiguous. $\langle \text{assign} \rangle \rightarrow \langle \text{id} \rangle = \langle \text{expr} \rangle$ $\langle \text{id} \rangle \rightarrow A B C$ $\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle + \langle \text{expr} \rangle$ $ \langle \text{expr} \rangle * \langle \text{expr} \rangle$ $ (\langle \text{expr} \rangle)$ $ \langle \text{id} \rangle$	Knowledge	2
9	List the three general methods of implementing a programming language.	Knowledge	1

10	Explain different aspects of the costs of a programming language.	Understand	2
UNIT - II			
1	Define heterogeneous array. Discuss the design issues of arrays.	Knowledge	2
2	Explain in detail the design issues of arithmetic expressions.	Understand	2
3	Discuss structural and name equivalence for types. Give an example of a language used for each approach.	Understand	2
4	Define a variable and what the attributes of a variable are. Elaborate on address of a variable.	Knowledge	3
5	Write a note on Boolean and relational expressions.	Apply	3
6	Explain the different types of relational operators used in C, Ada and Fortran 95.	Understand	3
7	List the advantages and disadvantages of mixed mode arithmetic expressions?	Knowledge	2
8	Define array and record. Classify arrays based on storage allocation. What are the advantages and disadvantages of allocation memory during compilation time and run time.	Knowledge	2
9	Define static, fixed stack dynamic, stack dynamic, fixed heap dynamic and dynamic arrays. What are the advantages of each?	Knowledge	2
10	List the design issues of pointers. Explain.	Knowledge	3
UNIT - III			
1	Discuss generic subprograms in C++ and java.	Understand	3
2	Explain about the concept of local referencing environment?	Understand	3
3	Define subprograms. What are the advantages of subprograms?	Knowledge	3
4	Discuss in detail about lifetime of a variable.	Understand	3
5	Discuss the design issues of subprogram and its operations performed on them.	Understand	4
6	Explain different methods of parameter passing mechanisms to subprograms.	Understand	4
7	Explain about Co routines.		4
8	Explain the design considerations of parameter passing.	Understand	4
9	Discuss about procedures and functions in subprograms.	Understand	3
10	Define shallow and deep binding for referencing environment of subprograms that have been passed as parameters.	Knowledge	4
UNIT - IV			
1	Describe how exception is handled in ADA with an example.	Understand	4
2	Describe briefly about Semaphores.	Understand	4
3	Describe briefly about Monitors.	Understand	5
4	Discuss Object Oriented Programming in SMALLTALK.	Understand	5
5	Write about goal statements and simple arithmetic in PROLOG.	Apply	5
6	Define binary semaphore. What is counting semaphore? What are the primary problems with using semaphores to provide synchronization?	Knowledge	5
7	Explain the following terms: i. process synchronization ii. race condition iii. binary semaphores	Understand	5

	iv. MIMD		
8	Explain different types of propositions present in logic programming.	Understand	5
9	Describe the cooperation synchronization and competition synchronization in message passing.	Understand	5
10	Discuss how dining philosopher's problem and producer consumer problem are solved using concurrency in Ada.	Understand	5
UNIT – V			
1	Write about control constraints in Python.	Apply	6
2	Write about data abstraction in Python.	Apply	6
3	Write about data types and structures of LISP and LISP interpreter.	Apply	7
4	List the ways in which ML is significantly different from scheme.	Knowledge	7
5	Describe the scoping rule in common LISP, ML and HASKELL.	Understand	7
6	Explain the characteristics of scripting languages.	Understand	6
7	Discuss in detail about the python primitive types.	Understand	6
8	Explain about LISP functional programming language.	Understand	7
9	Explain the features of scripting languages.	Understand	7
10	Explain the concept of Pragmatics.	Understand	7

PART – C (PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS)

S. No	Question	Blooms Taxonomy Level	Course Outcome
UNIT – I			
1	Give some reasons why computer scientist and professional software developers should study general concepts of language design and evaluation.	Understand	1
2	Write reasons for the statement: “Exception handling is very important, but often neglected by programming languages”.	Apply	2
3	Write reasons for the statement: “A programming language can be compiled or interpreted”. Give relative advantages and disadvantages of compilation and interpretation. Give examples of compiled and interpreted languages.	Apply	2
4	What does it mean for a program to be reliable?	Apply	1
5	What are the three fundamental features of an object oriented programming language?	Apply	2
6	Describe some design trade-offs between efficiency and safety in some language you know.	Understand	2
7	Why is the Von Neumann bottleneck important?	Understand	2
8	Which produces faster program execution, a compiler or a pure interpreter?	Knowledge	2
9	Why is type checking the parameters of a subprogram important?	Knowledge	1
10	Explain the different aspects of the cost of a programming language?	Understand	1

S. No	Question	Blooms Taxonomy Level	Course Outcome
UNIT - II			
1	List what advantages does java's break statement have over C's and C++'s break statement.	Knowledge	2
2	State whether static binding is more reliable or dynamic binding. Explain why.	Knowledge	2
3	List the design issues for Names with suitable examples.	Knowledge	2
4	After language design and implementation [what are the four times bindings can take place in a program?]	Analyze	3
5	Describe a situation where a history-sensitive variable in a subprogram is useful?	Understand	3
6	What are the arguments for and against representing Boolean values as single bits in memory?	Understand	3
7	List the advantages of using control structures in any of the compiled programming languages.	Knowledge	2
8	Discuss the merits of guarded commands.	Understand	2
9	Discuss the merits and demerits of short circuit evaluation.	Understand	2
10	Discuss the importance of Named constants with suitable examples.	Understand	3
UNIT - III			
1	Compare the parameter passing mechanisms of ALGOL and ADA.	Analyze	3
2	State the importance of Local Referencing Environments with suitable examples.	Knowledge	3
3	Discuss about different parameter passing methods of Ada language.	Understand	3
4	In what way is static checking better than dynamic type checking.	Analyze	3
5	What are the three general characteristics of subprogram.	Understand	4
6	Which two languages allow multiple values to be returned from a function?	Analyze	4
7	In what way co-routines are different from conventional subprograms?	Analyze	4
8	What causes a C++ function to be instantiated?	Knowledge	4
9	What are the three semantic models of parameter passing?	Knowledge	3
10	Describe the way that aliases can occur with pass-by-reference?	Understand	4
UNIT - IV			
1	Discuss the reasons for using exception handlers in a programming language. What if there exist programming languages with no exception handlers.	Understand	4
2	Give the sample code to factorial of a number in PROLOG language.	Understand	4
3	Correlate the importance of logic programming languages over functional programming languages.	Analyze	5
4	List the merits and demerits of subprogram level concurrency.	Knowledge	5
5	Discuss C++ parameterized ADT with suitable example.	Understand	5
6	Explain about Exception Propagation.	Understand	4

S. No	Question	Blooms Taxonomy Level	Course Outcome
7	Discuss various applications of logic programming languages.	Understand	4
8	Explain the concept of object oriented programming in Smalltalk.	Understand	4
9	Give an example for exception handlers in Ada language.	Understand	5
10	Explain the design issues related to Data abstraction	Understand	5
UNIT - V			
1	Write features of Haskell that makes very different from schema.	Apply	6
2	List the ways in which ML is significantly different from scheme.	Knowledge	6
3	Record the advantages of Python scripting language over other scripting languages	Knowledge	7
4	Give example of Python code to find the roots of quadratic equation	Understand	7
5	Discuss the concept of pragmatics in scripting languages.	Understand	7
6	Write the python code to print first ten natural numbers.	Apply	6
7	Write the python code to print factorial of a given number.	Apply	6
8	Write the python code to print prime numbers between the given range.	Apply	7
9	Write the python code to find the given number is even or odd.	Apply	7
10	Write the python code to print Pascal triangle.	Apply	7