Question bank for unit-1 and unit-2

1. Write the characteristic of Programming language.
2. What are the various phases of complier
3. Write a short note on software simulated computer and virtual computer with a suitable example..
4. Explain program verification and validation.
5. What is pointer? Write the advantage and disadvantage of pointer while using in program.
6. Explain the terms “dereferencing” and “deallocation “in the context of pointer.
7. Write a program to implement an abstract data type” complex”. This data type implements a complex number a+ib. Also write methods to determine magnitude and conjugate of this complex number.
8. Explains static scope rule and dynamic scope rule for sub program control.
9. Explain the most suitable data structure to implement heap such that garbage collection and removal of dangling references is carried out efficiently.
10. What is BNF grammer notation to define syntax of a grammar? Using this notation write rules for syntax of a “do.. while “statements and “if.. else “statements.
11. Choose two high level programming languages of your own choices and contrast their definition of array and pointer data types.
12. Develop a formula based representation of a stack where, if during a push operation we do not have space to accommodate the new element, a new array of size one bigger than existing is allocated, elements are copied from old array to new one and the old array is deleted. Similarlly, during a pop operation, if the stack size drops to one fourth of the array capacity, a smaller array of half the size is allocated, the elements copied and the old array is deleted.
13. Suppose that you have designed a programming language which permits procedures and coroutines to be intermixed. If procedure P calls procedure Q, then Q either returns to P via return or it may resume P. Also Q, may recursively call P which may then resume Q. Show how the return and resume statements work together.
14. Some languages permit a procedure to have multiple entry points. Discover one such language and write its rules. Why is such a facility useful?
15. Devise a function F that alters the value of a global variable *x* as well as returns a value. Also, write a program which uses F and X sensitive to the order of evaluation of expression. Discuss the different possible values which might printed by your function F.
16. Discuss about Abstract Data Types.
17. Suppose that one wants to write a function which concatenates two strings and it calls the storage allocator to get space for the new string. Should this be considered a side effect? Justify your answer.
18. The dynamic scope rule specifies that nonlocals must be evaluated in the calling environment. Explain why MACRO expansion of procedure calls produces the same result as would be obtained under the dynamic scope rule.
19. Implement a variant of quicksort for lists *Le,* sort a list as follows : Pick an element and call it PIVOT. Partition the list into two sublists of elements smaller than and larger than the PIVOT. Recursively sort the sublists. Combine the sorted sublists and PIVOT together into sorted list.
20. Describe the concept of orthogonality in programming language design.

21. Differentiate between compile time binding and run time binding.
22. What is loader and explain?
23. What is linker and explain?
24. Some languages, such as C, Java distintinguish between uppercase and lowercase in identifiers. What are the pros and cons of this design decision?
25. Give the accessing formula for computing the location of a component A(I)(J) of a matrix A declared as int A[m][n]; where m and n are constants and A is stored in a raw major order.
26. For a elementary data type in a language with which you are familiar, do the following:
    (i) Show a situation during execution where a data object of that types exists that is neither a variable nor a constant
    (ii) Explain the difference between data objects of that type and the values that those data object may contain.
27. For a language that provides a pointer type for programmer constructed data objects and operations such as new and dispose that allocate and free storage for data object, write a program segment that generate garbage(in terms of storage management). Write a program segment that generates a dangling reference. If one or the other program cannot be written explain why?
28. What makes a good programming language?
29. What are external influencing factors of programming language?
30. What do you understand by binding and binding time? What is the difference between binding at compile time and binding at execution time? Take an expression of a language and discuss various types of binding done for this statement.
31. Write a statement in a language of your choice, list the various bindings that are necessary to completely determine the semantics of the statement when it is executed.
32. What are the general syntactic criteria of a programming language? Explain.
33. Explain the terms:
    (i) Translators
    (ii) Software simulation
    (iii) Garbage collection
    (iv) Run time requirements of a programming language.
    (v) Ambiguous grammar
    (vi) Recursive sub program implementation
34. Explain the terms variables, constants and literals for a language.
35. Describe any attributes for an elementary data type that a data object of that type may have other than its data type.
36. When and why dynamic type checking and static type checking operation are performed? Give major advantages using type checking operations.
37. Explain the following terms with three examples for each :
    (i) Programmer - constructed data object
    (ii) System-defined data object.
    (iii) User defined heterogeneous data object
38. Write a program for your choice, give a complete specification of the arguments and results of each sub program. Include implicit arguments in the specification. Are any of them undefined for some arguments in the domain you specified ?
39. What will be syntax and semantic for the following expression ?
    Expression: a*b ^2 + C / d - e * fun(y)
    Discuss about side effects problem for the above expression
40. Suppose that you wished to design a language that used :
    (i) retained local environments.

(ii) no recursive· subprogram calls and
(iii) non-local references based on dynamic scope.

Explain how the referencing environment is represented. Explain the actions taken on subprogram call and return. Explain how nonlocal referencing is implemented.

41. Why are private parts of a super class generally not made accessible to subclasses?
42. Can we have inheritance without polymorphism? Explain.
43. On the basis of execution time, storage requirements compare the complier and interpreter.
44. What are the major factors that affect development evolution of programming language methodologies?
45. What are main criteria for implementation of an operation in hardware, software (subprogram) and in line sequence?
46. What are part array and variant record? How they differ from conventional array and records?
47. Define abstraction, encapsulation, and polymorphism with example.
48. Discuss sequential and index sequential files with their specification and implementation.
49. Implement CIP and CEP in subprogram control and explain?
50. Compare the language like C, Pascal, LISP on the basis of the following:
    (i) Program structure
    (ii) Parameter passing scheme.
    (iii) Scope rules
    (iv) Control structure available.
    (v) Data object/ type available
51. Explain in brief free and fixed field formats of formats of syntax?
52. What is binding and binding time and discuss various classes of binding time?
53. Describe sequence control and its types? What are recursive sub program.
54. Describe implicit and explicit sequence control with example.
55. What do you understand by operating and programming environment? List various attributes of a good programming and operating environment.
56. Write a short note on :
    (i) unification
    (ii) tree walking algorithm
    (iii) Prolog
    (iv) backtracking
    (v) pattern matching
57. Write the specification and implementation of:
    (i) structured data type
    (ii) composite data type
    (iii) Scalar data type
58. Define inheritance and its types with example.
59. Explain polymorphism with example.
60. Write down a short program in Smalltalk in terms of inheritance.