18/12/24 (FN)

CONTROLLER OF EXAMINATIONS ★ ADDITIONAL AU CHENNAI 600 025

Roll No. | | | | | | | | | | |

B.E. (FT) END SEMESTER EXAMINATIONS – NOV / DEC 2024

Computer Science and Engineering

Fifth Semester

CS6302 – PROGRAMMING PARADIGMS

(Regulation 2018 - RUSA)

Time: 3 Hours | Answer ALL Questions | Max. Marks: 100

CO1	Write programs related to syntax and semantics
CO2	Compare programs between C, Perl and Small Talk
CO3	Write programs using scripting languages
CO4	Demonstrate event-driven and concurrent programming using Prolog
CO5	Apply Prolog for developing distributed systems

PART - A (10 x 2 = 20 Marks)

Sl.No.	Questions	Marks	CO	BL
1.	What is the principle of 'orthogonality' of programming language design?	2	CO1	L1
2.	What is 'type error'? How it can be resolved in language design?	2	CO1	L2
3.	Comment on *copy* vs *reference* semantics of assignment statements.	2	CO1	L1
4.	Give an example of an expression in C, with side effects.	2	CO2	L2
5.	What is a 'partial function'? Give example.	2	CO2	L2
6.	Illustrate the concept of implementation of dynamic arrays, using runtime stack and heap.	2	CO2	L1
7.	Mention the potential problem that occurs in the call to *scanf* function in C.	2	CO2	L2
8.	Why the semantics definition becomes complicated for multiple inheritance?	2	CO1	L2
9.	What are 'eager' and 'lazy' evaluation mechanisms?	2	CO4	L1
10.	Give expansions for the following: ALGOL, FORTRAN, LISP, PHP.	2	CO2	L1

PART – B (8 x 8 = 64 Marks)

(Answer any 8 questions)

Sl.No.	Questions	Marks	CO	BL
11.	How the different programming environments execute the programs using compilation and interpretation techniques? Explain with examples.	8	CO1	L2
12.	What are the design issues for exception handling? Discuss how exceptions are handled in any popular language of your choice.	8	CO2	L1
13.	i) Present the necessary formalism for checking the validity of a typical 'function' in C-lite. ii) Consider the following C-like program:	4 4	CO2	L2

	********** int a; **procedure foo(int x)**{ x = x + 10; a = a + x;} **procedure fie( )**{ a = 5; foo(a); print (a);} ********** What is the output of the above program for each of the following parameter passing mechanisms? a) Argument a is passed by value b) Argument a is passed by Reference c) Argument a is passed by Value-Result			
14.	Present the structure of an 'Activation Record'. Demonstrate the semantics of function call and return using run-time stack, for any simple recursive function of your choice.	8	CO1	L1
15.	Show all steps in the derivation of the meaning of the following assignment statement when executed in the given state, using the denotational sematic rule. M(x=2*x+3/y-4,{<x,6>,<y,-12>,<Z,75>})	8	CO1	L2
16.	Describe the General features of 'Imperative programming paradigm'.	8	CO2	L1
17.	Present the detailed steps of any popular 'Garbage Collection' algorithm of your choice and critically analyze its pros and cons.	8	CO2	L1
18.	**i)** Mention the common characteristics of 'scripting languages'. **ii)** Present the distinctive features of any popular scripting language of your choice and contrast those features in terms of 'numeric types', 'arrays', 'scoping' with that of any popular compiled language.	4 4	CO3	L2
19.	Discuss about the significance of 'abstract' class in object-oriented programming, and how does it differ from an 'interface'.	8	CO2	L1
20.	What are the unique features of 'functional programming'? Demonstrate using any functional programming language of your choice the *list* data structure usage, by defining a function that accepts an unordered list and returns some ordered list.	8	CO4	L2
21.	Write short notes on two popular strategies for supporting 'synchronization' and compare their pros and cons.	8	CO3	L2
22.	What are the differentiating features of 'Event Driven programming' in comparison with 'imperative paradigm'? Present a detailed discussion on how Event handling is implemented in any popular language of your choice choosing any simple example of your choice.	8	CO4	L1

PART – C (2 x 8 = 16 Marks)

Sl.No.	Questions	Marks	CO	BL
23.	What are the distinguishing features of 'logic programming'? Solve the following query using the given facts and rules and provide illustrations wherever necessary: ***cousin (Who, jeff).***	8	CO4	L3

	********** mother(mary, sue). mother(mary, bill). mother(sue, nancy). mother(sue, jeff). mother(jane, ron). father(john, sue). father(john, bill). father(bob, nancy). father(bob, jeff). father(bill, ron). parent(A,B) :- father(A,B). parent(A,B) :- mother(A,B). grandparent(C,D) :- parent(C,E), parent(E,D). sibling(U,V):- parent(P,U), parent(P,V), U\=V. cousin(X,Y):- parent(U,X), parent(V,Y), sibling(U,V), X\=Y. **********			

24.	Consider the following C-like program. What does the program print if the language uses *static* and *dynamic* scoping? Draw symbol table and explain.
	**********
	Program A()
	{ x, y, z: integer;
	procedure B()
	{ y: integer;
	y=0;
	x=z+1;
	z=y+2; }
	procedure C()
	{ z: integer;
	procedure D()
	{ x: integer;
	x = z + 1;
	y = x + 1;
	call B(); }
	z = 5;
	call **D()**; }
	x = 10;
	y = 11;
	z = 12;
	call **C()**;
	print x, y, z;}
	**********
		8	CO1	L3