SRMINSTITUTE OF SCIENCE AND TECHNOLOGY

Ramapuram Campus, BharathiSalai, Ramapuram, Chennai - 600089

**FACULTY OF ENGINEERING AND TECHNOLOGY**

# DEPARTMENT OFCOMPUTERSCIENCEANDENGINEERING

****

**QUESTIONBANK**

**DEGREE / BRANCH: B.Tech/CSE**

**IV SEMESTER**

**SUB CODE – SUBJECT NAME: 18CSC207J/ADVANCED PROGRAMMING**

**PRACTICE**

**Regulation– 2018**

**AcademicYear: 2021-22**

# SRMINSTITUTE OF SCIENCE AND TECHNOLOGY

**Ramapuram Campus, BharathiSalai, Ramapuram, Chennai-600089**

**DEPARTMENTOFCOMPUTERSCIENCEANDENGINEERING**

**QUESTIONBANK**

**SUBJECT : 18CSC207J -ADVANCED PROGRAMMING PRACTICE**

**SEM/YEAR:IV/II**

**Course Outcomes**

***CO1:***Create Programs using structured, procedural and object oriented programming paradigms

***CO2:***Create Programs using event driven, declarative and imperative programming paradigms

***CO3:***Create Programs using parallel, concurrent and functional programming paradigms

***CO4:***Create Programs using logic, dependent type and network programming paradigms

***CO5:***Create Programs using symbolic, automata based and graphical user interface programming paradigms

***CO6:***Create Programs using different programming paradigms using python language

|  |  |  |  |
| --- | --- | --- | --- |
| **UNITI** | | | |
| Structured Programming Paradigm- Programming Language Theory- Bohm-Jacopini structured program theorem- Sequence, selection, decision, iteration, recursion- Other languages: C, C++, Java, C#, Ruby - Demo: Structured Programing in Python- Procedural Programming Paradigm- Routines, Subroutines, functions- Using Functions in Python- logical view, control flow of procedural programming in various aspects- Other languages: Bliss, ChucK, Matlab- Demo: creating routines and subroutines using functions in Python- Object Oriented Programming Paradigm- Class, Objects, Instances, Methods- Encapsulation, Data Abstraction- Polymorphism, Inheritance- Constructor, Destructor- Example Languages: BETA, Cecil, Lava Demo: OOP in Python | | | |
| **PART-A (Multiple Choice Questions)** | | | |
| **Q.**  **No** | **Questions** | **Course Outcome** | **Competence**  **BT Level** |
| **1** | In Python which parameter passing mechanism is used with function call.   1. Pass by value 2. Pass by Reference 3. Both Pass by value and Pass by reference 4. None | CO1 | L1 |
| **2** | Which one is correct about variable names in Python.   1. All variable names must begin with an underscore. 2. Unlimited length 3. The variable name length is a maximum of 2. 4. All of the above | CO1 | L1 |
| **3** | Which of the following is not the type of function argument?   1. Positional argument 2. Keyword argument 3. Initial argument 4. Default argument | CO1 | L1 |
| **4** | What will be the output of the following Python code?  **x = 50**  **def func(x):**  **print(‘x is’, x)**  **x = 2**  **print(‘Changed local x to’, x)**  **func(x)**  **print(‘x is now’, x)**   1. x is 50   Changed local x to 2  x is now 50   1. x is 50   Changed local x to 2  x is now 2   1. x is 50   Changed local x to 2  x is now 100   1. None | CO1 | L2 |
| **5** | What will be the output of the following Python code?  **values = [[3, 4, 5, 1], [33, 6, 1, 2]]**  **v = values[0][0]**  **for row in range(0, len(values)):**  **for column in range(0, len(values[row])):**  **if v < values[row][column]:**  **v = values[row][column]**  **print(v)**   1. 3 2. 5 3. 6 4. 33 | CO1 | L3 |
| **6** | What will be the output of the following piece of code. [CLO-1,L3]  **def greet(name,msg=’Good Day’):**  **print("Hello",name + ', ' + msg)**  **greet("AAA")**  **greet(“BBB”,”Good Morning”)**   1. Hello AAA Good Morning, Hello BBB Good Morning 2. Hello AAA Good Morning, Hello BBB Good Day 3. Hello AAA Good Day, Hello BBB Good Day 4. Hello AAA Good Day, Hello BBB Good Morning | CO1 | L2 |
| **7** | What is the correct syntax to create a class named Student that will inherit properties and methods from a class named Person in Python?  a) class Student from Person:  b) class Student(Person):  c) Student(Person):  d) class Student : Person | CO1 | L1 |
| **8** | What value will be printed by the print statement given in the following code?  **odd=lambda x: bool(x%2)**  **numbers=[n for n in range(10)]**  **print(numbers)**  **n=list()**  **for i in numbers:**  **if odd(i):**  **continue**  **else:**  **break**   1. [0, 2, 4, 6, 8, 10] 2. [0, 1, 2, 3, 4, 5, 6, 7, 8, 9] 3. [1, 3, 5, 7, 9] 4. Error | CO1 | L3 |
| **9** | The number of arguments taken by lambda function   1. 1 2. 2 3. Any number 4. None | CO1 | L1 |
| **10** | Which of the following is true regarding Generic/meta programming?   1. generates semantic associations 2. Programs about programs 3. generates higher-order programs 4. is used for assembly level manipulations | CO1 | L1 |
| **11** | If a is a dictionary with some key-value pairs, what does a.pop(‘key’) do?   1. Removes an arbitrary element 2. Removes all the key-value pairs 3. Removes the key-value pair for the key given as an argument 4. Invalid method for dictionary | CO1 | L2 |
| **12** | According to Bohm-Jacopini, a function is possible by combining subprograms in which three manners?   1. Jump, Sequence and Loop 2. Sequence, Function Calls and Subroutines 3. Sequence, Iteration and Selection 4. Iteration, Macros and Branching | CO1 | L1 |
| **13** | What are the values printed by the two print statements given below?  **a=10**  **b=20**  **def change():**  **global b**  **a=45**  **b=56**  **change()**  **print(a)**  **print(b)**   1. 10 56 2. 45 56 3. 10 20 4. Syntax Error | CO1 | L3 |
| **14** | Which of the following is the use of id() function in Python?   1. Every object doesn’t have a unique id 2. id returns the identity of the object 3. All of the mentioned 4. None of the mentioned | CO1 | L1 |
| **15** | What will be the value printed by the last print statement in the following Python code?  **d={“id”:101, “name”:”AAA”, “dept”:”QA”}**  **print(d)**  **print(“Emp ID=”,d[‘id’])**  **print(“Emp Name=”,d[‘name’])**  **print(“EmpDept=”,d[‘dept’])**  **d[‘dept’]=”RA”**  **print(d)**  **d.pop(‘dept’)**  **print(d[‘dept’])**   1. QA 2. RA 3. KeyError: ‘dept’ 4. None | CO1 | L3 |
| **16** | Which of the following is correct way to add all classes, methods or other datatypes(list, tuple, dictionary) etc.. of a module in Python?   1. import \* from module\_name 2. from module\_name import \* 3. from module\_name import all 4. import module\_name as m | CO1 | L2 |
| **17** | ---------------- refers to the spaces at the beginning of a code line which is considered as the special important feature of Python.   1. Indentation 2. Input 3. Inherit 4. Identification | CO1 | L1 |
| **18** | \_\_\_\_\_\_\_\_\_ is a graphical representation of structured programming using Top down analysis.   1. Programming Paradigm 2. Structogram 3. Flowchart 4. Proess block | CO1 | L1 |
| **19** | Which of the following statements is incorrect about the following code?  **class People():**  **def \_\_init\_\_(self, name):**  **self.name = name**  **def namePrint(self):**  **print(self.name)**  **person1 = People("John")**  **person2 = People("Sai")**  **person1.namePrint()**   1. person1 and person2 are two different instances of the People class 2. The \_\_init\_\_ method is used to set initial values for attributes 3. 'self' is not needed in def namePrint(self): 4. person2 has a different value for 'name' than person1 | CO1 | L3 |
| **20** | \_\_\_\_\_\_\_\_\_ is not a keyword, but by convention it is used to refer to the current instance (object) of a class.   1. class 2. def 3. self 4. init | CO1 | L2 |
| **21** | Which of the following is the correct way to define an initializer method?   1. def \_\_init\_\_(title, author): 2. def \_\_init\_\_(self, title, author): 3. def \_\_init\_\_(): 4. \_\_init\_\_(self, title, author): | CO1 | L2 |
| **22** | How the constructors and destructors can be differentiated?   1. Destructor have a return type but constructor doesn’t 2. Destructors can’t be defined by the programmer, but constructors can be defined 3. Destructors are preceded with a tilde symbol, and constructor doesn’t 4. Destructors are same as constructors in syntax | CO1 | L2 |
| **23** | What is the output of the function complex()?   1. 0j 2. 0+0j 3. 0 4. Error | CO1 | L2 |
| **24** | What does ~~~5 evaluate to?   1. +5 2. -11 3. +11 4. -5 | CO1 | L2 |
| **25** | Which specifier should be used for member functions of a class to avoid inheritance?   1. Private 2. Default 3. Protected 4. Public | CO1 | L2 |
| **PART B (4 Marks)** | | | |
| **1** | What is Structured programming? How does it minimize the complexity? | CO1 | L1 |
| **2** | Write a python program with an add() function to return the sum of  two integers. | CO1 | L3 |
| **3** | List on Python Variables and its types. | CO1 | L1 |
| **4** | Compare structured programming and Procedural programming. | CO1 | L2 |
| **5** | Write a program to implement recursion. | CO1 | L3 |
| **6** | What is Data abstraction and explain its types. | CO1 | L1 |
| **7** | Define Inheritance. | CO1 | L1 |
| **8** | Write a program to create a list and print the values. | CO1 | L3 |
| **PART C (12 Marks)** | | | |
| **1** | There are 50 computers available in computer programming lab where each computers are used six hours per day. Write a Python program using classes and objects that contain getDetail() for getting input from user,calculatesecondperDay() for calculating the usage of each computer in seconds per day, calculateminutesperWeek() for calculating the usage of each computer in minutes per week ,calculatehourperMonth() for calculating usage of each computer in hour per month and calculatedayperYear() for calculating usage of each computer in day per yearList all the Components of structured programming language | CO1 | L3 |
| **2** | Discuss the features of Procedural programming. | CO1 | L2 |
| **3** | Define Function and recursion and explain them in detail | CO1 | L2 |
| **4** | List out the Features of object oriented programming | CO1 | L2 |
| **5** | Write a python program to get square and cube of a number using  Inheritance concept. | CO1 | L3 |

**Note:**

1. **BT Level –** Blooms Taxonomy Level
2. **CO – Course Outcomes**

BT1 –RememberBT2 – Understand BT3 – Apply BT4 – Analyze BT5 – Evaluate BT6 – Create