U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

December 2023 Supplementary Examinations

Programme: B.E. **Semester: I / II**
Branch: Common to all Branches **Duration: 3 hrs.**
Course Code: 22CS1ESPYP / 22CS2ESPYP **Max Marks: 100**
Course: Introduction to Python Programming

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

			CO	PO	Marks
		UNIT - I			
1	a)	With examples, define an operator and operand in Python. Write a Python program that takes two integers as input from the user, calculates and prints the results of add, subtract, multiply, divide operations. Also calculate remainder when the first integer is divided by the second integer. Division must be with decimal precision.	CO1	PO1	**10**
	b)	Give the general syntax of while loop in Python. Write a Python program that prompts the user to enter a sequence of positive integers (greater than zero) and finds both the maximum and minimum values in the sequence using loops. The program must continue to accept input until the user decides to stop entering numbers (user can end the input sequence by entering a negative integer). Display the count of entered numbers and the average of all the entered numbers. The program must handle invalid input (non-positive integers).	CO3	PO3	**10**
		UNIT - II			
2	a)	List any four string methods in Python along with their purpose and sample code to illustrate the usage.	CO1	PO1	**08**
	b)	Write a Python program that takes two strings as input from the user and determines if they are "anagrams" of each other. Ignore spaces and consider uppercase and lowercase letters as equivalent.	CO1	PO1	**06**
	c)	Write a Python program that takes a list of strings as input. The program should perform the following tasks: i. Sort the list in ascending order of string length. ii. If two strings have the same length, sort them alphabetically in ascending order. iii. Output the sorted list.	CO2	PO2	**06**

			UNIT - III			
	3	a)	Explain the key characteristics and properties of dictionaries. Compare dictionaries and tuples in Python.	*CO1*	*PO1*	**10**
		b)	Write a Python program using function that calculates the area and circumference of a circle. The program should: i. Prompt the user to enter the radius of the circle as a floating-point number. ii. Use the math module to calculate the area and circumference of the circle. iii. Display the calculated area and circumference rounded to two decimal places. iv. Handle cases where the user enters a negative radius by displaying an error message.	*CO3*	*PO3*	**10**
			OR			
	4	a)	What are tuples in Python, and how are they defined? Discuss the immutability of tuples and why this property makes them suitable as dictionary keys. Mention any advantages and limitations or considerations when using tuples as keys in dictionaries.	*CO1*	*PO1*	**10**
		b)	Write a Python program using function that takes a list of tuples as input, where each tuple contains a student's name (a string) and their scores in different subjects (a variable number of integers). The program should do the following: i. Calculate and print the average score for each student. ii. Find and print the student with the highest average score. iii. Find and print the subject with the highest average score across all students. iv. Handle cases where there are multiple students with the same highest average score.	*CO3*	*PO3*	**10**
			UNIT - IV			
	5	a)	Explain the object lifecycle in Python.	*CO1*	*PO1*	**10**
		b)	Create a Python program that models a simple banking system using classes and objects. Define a BankAccount class with the following attributes and methods: **Attributes**: account_number (a unique identifier), account_holder (the name of the account holder), balance (the current balance) **Methods**: deposit(amount) - Deposits the specified amount into the account, withdraw(amount) - Withdraws the specified amount from the account (*Ensure that the withdrawal amount does not exceed the balance*), get_balance() - Returns the current balance, display_account_info() - Displays the account number, account holder, and balance. Create two instances of the BankAccount class, each with unique account numbers, and perform the following operations: i. Deposit money into both accounts. ii. Withdraw money from one of the accounts. iii. Display the account information for both accounts.	*CO2*	*PO2*	**10**

			OR			
	6	a)	Explain how exception handling works in Python, including the use of try, except, else, and finally blocks. Describe the purpose of each block and how they contribute to handling exceptions.	*CO1*	*PO1*	**10**
		b)	Create a Python program with a custom exception class called InvalidAgeException to handle age-related errors in a user registration system. Implement a function register_user(name, age) that takes a user's name (a string) and age (an integer) as input. This function should check if the age is less than 18, raise an InvalidAgeException with the message "User must be 18 years or older." And check if the age is 18 or older, print a message welcoming the user. Illustrate the call to the register_user function multiple times with different user names and ages to test your custom exception handling.	*CO3*	*PO3*	**10**
			UNIT - V			
	7	a)	Explain the concept of the escape character (\) in Python regular expressions. Write a Python program that validates email id's using regular expressions.	*CO1*	*PO1*	**10**
		b)	Appraise Python's support for file handling. Write a Python program to reverse the content of a file and store it in another file	*CO1*	*PO1*	**10**

****