Assignment Submission Due Date: 25 September 2024

**Solve these below Python problems:

If-Else Statements

1. **Grade Evaluation:** Write a program that takes a student's score as input and prints their grade based on the following scale: A (90-100), B (80-89), C (70-79), D (60-69), F (below 60).
2. **Even or Odd:** Create a program that checks if a number provided by the user is even or odd and prints an appropriate message.
3. **Temperature Conversion:** Write a program that converts temperatures from Celsius to Fahrenheit or vice versa based on user choice. Use if-else to determine which conversion to perform.

For Loops

4. **Summing Numbers:** Write a program that takes a list of integers and calculates the sum of all the even numbers in the list using a for loop.
5. **Multiplication Table:** Create a program that generates and prints the multiplication table for a number entered by the user, up to 10.
6. **Counting Vowels:** Write a program that counts the number of vowels in a given string using a for loop and prints the total count.

While Loops

7. **Guess the Number:** Develop a number guessing game where the user has to guess a randomly generated number between 1 and 100. Provide hints (higher/lower) until the correct guess is made.
8. **Countdown Timer:** Create a countdown timer that starts from a user-defined number and decrements to zero, printing each number as it counts down.
9. **Input Validation:** Write a program that continuously prompts the user to enter a positive integer until they do so. If the input is invalid, print an error message.

Lists

10. **List Reversal:** Write a program that takes a list of integers from the user and prints the list in reverse order.
11. **Find Minimum and Maximum:** Create a program that takes a list of numbers and finds the minimum and maximum values, printing both.
12. **Unique Elements:** Write a program that removes duplicate elements from a list of integers and prints the new list with unique values.

Tuples

13. **Tuple Operations:** Define a tuple of cities. Write a program that prints each city using a loop and allows the user to add a new city to a list based on the tuple.
14. **Swapping Tuple Elements:** Create a program that swaps the first and last elements of a given tuple and prints the modified tuple.
15. **Tuple to List Conversion:** Write a program that converts a given tuple of mixed data types into a list, allowing the user to access each item individually.

Combined Concepts

16. **List Sorting:** Write a program that takes a list of numbers and sorts them in ascending order, using a for loop to iterate through the list.
17. **Count Items in a List:** Create a program that counts how many times a specific element appears in a list, using a while loop to traverse the list.
18. **User-Defined List Operations:** Develop a program that allows users to add, remove, and display elements in a list until they choose to exit.
19. **Tuple Index Search:** Write a program that searches for a user-defined value in a tuple and returns its index, or a message indicating that the value is not found.
20. **List Comprehension Challenge:** Create a list of squares of numbers from 1 to 10 and then use a for loop to print each square along with its corresponding number.

Sequence Slicing

21. **Extracting Substrings:** Write a program that takes a string input from the user and allows them to specify the starting and ending indices to extract and print a substring.
22. **Reversing a List with Slicing:** Create a program that takes a list of integers and uses slicing to reverse the list. Print the original and the reversed list.
23. **Middle Elements of a List:** Write a program that takes a list of numbers and slices the list to extract the middle element(s). If the list has an odd length, return the single middle element; if it's even, return the two middle elements.

Dictionaries

24. **Employee Directory:** Create a program that maintains a dictionary of employee names and their corresponding departments. Allow the user to add new employees, remove employees, and look up the department of a specific employee.
25. **Merge Two Dictionaries:** Write a program that merges two dictionaries containing student names and their scores. If a student appears in both dictionaries, sum their scores. Print the resulting dictionary.

26. **Finding Maximum Value in Dictionary:** Create a dictionary that contains names of students as keys and their scores as values. Write a program that finds and prints the name of the student with the highest score.

2D List and Tuples

27. **Transposing a 2D List**:Write a program that takes a 2D list (a list of lists) representing a matrix and transposes it. The transposed matrix should be printed as a new 2D list.
28. **Sum of Each Row in a 2D Tuple**: Create a program that defines a 2D tuple containing integer values. Write a function that calculates and prints the sum of each row in the tuple.
29. **Finding the Maximum Element in a 2D List**: Write a program that takes a 2D list of integers and finds the maximum value in the entire list. Print both the maximum value and its position (row and column indices).
30. **Matrix Multiplication**:Create a program that performs matrix multiplication between two compatible 2D lists (matrices). The program should take user input for both matrices and print the resulting matrix after multiplication.