Question 1:

(Decimal to hex) Write a program that prompts the user to enter an integer and displays its corresponding hex number. Here are some sample runs:

Enter a decimal value: 11
The hex value is B

Enter a decimal value: 16
The hex value is 10

For converting decimal to hex, you CANNOT use Python's built-in methods such as hex() function, string's format(), or format specifier, so you must use a while loop to resolve the problem.

Question 2:

Write a menu-driven program that allows a user to enter a list of scores and then choose between finding the smallest, largest, sum, average or mode. The mode is the score that occurs the most often. It can be determined as a by-product of building the frequency array. After building the frequency array, use it to determine which score occurred the most.

Use a if-elif statement to determine what action to take. Provide an error message if a valid choice is entered. Run the program five times, once with each option and once with an invalid option.

Note: You cannot use Python built-in functions for this question. For example, collections.Counter, count(), sum() etc cannot be used. Additionally, you cannot use a Python dictionary to find the mode.

Question 3:

Write a program that merges two sorted lists of numbers into a new sorted list. You cannot use any Python built-in functions, such as sort() or sorted(). Other lists or data structures cannot be used to store temporary results. Most importantly, the complexity of your algorithm must be of the order of n, i.e. O(n).

Example,
List1: [1, 3, 5, 7]
List2: [2, 4, 6, 8]
Result: [1, 2, 3, 4, 5, 6, 7, 8]

Question 4:

A school cafeteria is giving an electronic survey to its students to improve their lunch menu. Create a program that uses a list to store votes for the survey. The program should display four food items in the display. Then it prompts students to enter whether they like or dislike a particular food. Display a report that's updated as each

new set of responses is entered. Use a 2-D Integer array named votes, with four rows and two columns to summarize the results. Each row corresponds to a food item. The columns store the number of “like” and “dislike” votes, respectively.

The output is expected to be printed in a nice tabular format that is easy for humans to read, as shown in the sample below. Your marks will be reduced if you simply code it as "print(votes)".

	Like	Dislike
Pizza	2	0
Hot Dog	1	1
Ham	0	2
Cheese	1	1

Question 5:

Create a program that displays a 4 option menu. The options to be included are:

1 - Enter a new employee's information
2 - Display all employees’ information
3 - Find employees’ information by name
4 - Display all employees' information in chronological order by age.
5 - Remove the employee by ID
6 - Quit

- Option 1:
  - Prompt a user to provide the name, ID, department number, and age of an employee. Store the information as a list of employees. Each employee is represented by a list of items that include their name, ID, department number, and age. As a result, the data structure in which you will store a list of employees is a list of lists.
- Option 2:
  - Display information entered for all employees.
- Option 3:
  - If option 3 is chosen, request the user for the name of the employee. Look through the list. Show all employees with that name. If no employee can be found, provide the user the option to enter them.
- Option 4:
  - Display all employee information in chronological order by age. The action should not modify the list's original order.
- Option 5:
  - If option 5 is selected, ask the user for the employee's ID. Remove the employee with the given ID from the list. The relevant message should be printed to inform the user whether the employee was successfully removed or not.
- Option 6:

  If option 6 is selected, ask the user if they want to exit the program. If this is the case, a relevant message will be displayed before the program gracefully exits.
