Lab 14 (Template)

Task 1: Write a template class Set (have distinct elements) with flexible size that is Set size can grow by adding elements. Write following member functions: (Avoid duplication to secure full marks)

- Non-parameterized constructor, initialize data members for 10 elements
- Parameterized constructor, initialize data members according to size passed
- Parameterized constructor with two parameters array and size, initialize data members according to size and store distinct elements of array in set
- Add Element, if element does not already exist
- Print all elements of Set
- Destructor
- Copy constructor
- Assignment operator

Write main function and create integer and character type Set objects and check all member functions for both type of Set

Task 2: Write a class **Point** to add in Set class. Overload **== operator** in Point class, so that only unique Point class objects will be added in Set class. Also overload **ostream operator** in Point class required to show elements (Point class objects) in Set class. Next write the main function and add multiple Point class objects in Set class objects. Try to add some duplicate Point class objects as well. Display elements of set to check whether duplicate elements exist or not in Set class.

Task 3: Write a **SetList** (template) class with aggregation of **Set** (template) class. List may have zero or maximum ten objects of Set class. Keep current count of objects in SetList class. Create following member functions and main program to test your class:

- Non-parameterized constructor, initialize data members for 10 elements
- Add Set object in list
- Print Set List