

## Data Structures and Algorithms Assignment Set – 2

1. Define an ADT for Polynomials.  
Write C data representation and functions for the operations on the Polynomials in a Header file.  
Write a menu-driven main program in a separate file for testing the different operations and include the above header file.
2. Define an ADT for Sparse Matrix.  
Write C data representation and functions for the operations on the Sparse Matrix in a Header file.  
Write a menu-driven main program in a separate file for testing the different operations and include the above header file.
3. Define an ADT for List.  
Write C data representation and functions for the operations on the List in a Header file.  
Write a menu-driven main program in a separate file for testing the different operations and include the above header file. Two data structures with and without using sentinels in arrays are to be implemented.
4. Define an ADT for Set.  
Write C data representation and functions for the operations on the Set in a Header file, with array as the base data structure.  
Write a menu-driven main program in a separate file for testing the different operations and include the above header file.
5. Define an ADT for String.  
Write C data representation and functions for the operations on the String in a Header file, with array as the base data structure, without using any inbuilt function in C.  
Write a menu-driven main program in a separate file for testing the different operations and include the above header file.
6. Given a large single dimensional array of integers, write functions for sliding window filter with maximum, minimum, median, and average to generate an output array. The window size should be an odd integer like 3, 5 or 7. Explain what you will do with the boundary values.
7. Take an arbitrary Matrix of positive integers, say, 128 X 128. Also take integer matrices of size 3 X 3 and 5 X 5. Find out an output matrix of size 128 X 128 by multiplying the small matrix with the corresponding submatrix of the large matrix with the centre of the small matrix placed at the individual positions within the large matrix. Explain how you will handle the boundary values.
8. Find whether an array is sorted or not, and the sorting order.
9. Given two sorted arrays, write a function to merge the array in the sorting order.