Day 22

Lab Assignments

1. WAP to compute the sum of all elements in an array using dynamic memory allocation.
   Input: Enter the number of elements: 5
   Enter 5 elements: 10 20 30 40 50
   Output: Sum of the elements: 150
2. WAP to find the largest element stored in an array of n elements by using dynamic memory allocation.
   Input: Enter the number of elements: 5
   Enter 5 elements: 14 35 24 14 5
   Output: Largest element: 35
3. Dynamically allocate an array using calloc function, store elements into the array, then print the elements. Change the size of the array, get the new elements and print the array. Finally free the memory.
   Input: Enter the number of elements: 5
   Enter 5 elements: 14 35 24 14 5
   Elements are: 14 35 24 14 5
   Enter the number of more elements you want to store: 3
   Enter the 3 elements: 10 20 23
   Output: Elements are: 14 35 24 14 5 10 20 23
4. WAP to allocate a 2D array depending on the number of rows and columns read from the user. Store the elements in the array and display them.
   Input: Enter the number of rows and columns of the 2D array: 4 3
   Enter the elements of the 2D array:
   1 2 3
   4 5 6
   2 3 4
   7 6 8
   Output: Enter the elements of the 2D array:
   1 2 3
   4 5 6
   2 3 4
   7 6 8
5. WAP to read an integer number as a string and find out the sum of the digits of a number. Extract each of the characters of the string, store them in a dynamically allocated integer array and then find out the sum of all the elements of the array.
   Input: Enter a number: 12345
   Output: Sum of the digits of the number: 15
6. WAP to change the value of constant integer by using pointers.

Home Assignments

1. WAP to read the elements of an integer array using dynamic memory allocation. Multiply each element of that array by a constant (read from the user) using an user-defined function. Display the elements of the array before and after the multiplication.
   Input: Enter the number of elements: 5
   Enter 5 elements: 10 20 30 40 50

Enter a constant: 5

Output: Elements of the array are: 50 100 150 200 250

2. WAP to dynamically allocate memory to store n number of names and sort them in alphabetical order.

   Input: Enter the value of n: 5
   Enter 5 names:
   Raj
   Ajay
   Bikram
   Prakash
   John

   Output: After sorting the names are:
   Ajay
   Bikram
   John
   Prakash
   Raj

3. WAP to read the elements of an integer array using dynamic memory allocation. Create two more arrays dynamically to store the even and odd elements separately. Print both arrays separately.

   Input: Enter the number of elements: 10
   Enter 10 elements: 11 20 34 41 52 26 39 90 30 21

   Output: Even Array: 20 34 52 26 90 30
   Odd Array: 11 41 39 21

4. WAP to read two sorted integer arrays and combine them to produce another array dynamically which contains one occurrence of each of the elements of the original arrays.

   Input : Enter the first array size: 4
   Enter the element of the array: 10 23 41 50
   Enter the second array size: 5
   Enter the element of the array: 12 23 46 55

   Output : After merging the resultant array: 10 12 23 41 46 50 55

5. WAP to allocate a 2D float array dynamically depending the number of rows and columns. Note that the number of columns in each row may not be same. So read the number of columns for each row. Read the elements, store them in the array and display them.

   Input: Enter the number of rows of the 2D array: 3
   Enter the column size of row 1: 3
   Enter 3 elements: 1 2 3
   Enter the column size of row 2: 4
   Enter 3 elements: 4 2 3 5
   Enter the column size of row 3: 5
   Enter 3 elements: 6 2 3 9 8

   Output: Elements of the 2D array:
   1 2 3
   4 2 3 5
   6 2 3 9 8

6. WAP to allocate a 2D integer array dynamically to store and display the elements of pascal's triangle depending on the number of rows inputted by the user. Note that the number of columns in each row are not same.

Input: Enter the number of rows of the Pascal's Triangle: 5
Output: Pascal's Triangle of 5 rows:

```
1
1   1
1   2   1
1   3   3   1
1   4   6   4   1
```