

**Programming Fundamentals**

|  |  |
| --- | --- |
| Lab 06 | |
| **Topic** | Functions with arrays |
| **Objective** | * Hands on practice to use arrys in combination with arrays. |

**SECTION-A**

1. Write a C++ program and declare an integer array of size 100. Populate this arrays with random function. Value at each index should in between -250 and 2395. Now, ask user either to sort it in Ascending or Descending order. Write two user defined functions SortAscending() and SortDescending() respectively to sort array and call appropriate method as per user requirement. Print array in main function after sorting performed.

**Note**: *You should pass array to relevant function and this function should return array after performing sorting operation.*

## Sample output:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Initial array:**  0 1 2 . . . 98 99   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 55 | -78 | -45 |  | 245 | 1247 |   Press 1 to sort in Ascending order:  Press 2 to sort in descending order:  Choice: 2  **Output:**  0 1 2 . . . 98 99   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 1247 | 245 | 55 |  | -45 | -78 | |

1. Write a C++ program and declare a character type array with capacity to store an address upto 50 characters. Ask user to enter his/her address and declare a function which takes this address as parameter and removes all vowels in this address and returns arrays back to main function. Print address in main function before and after calling user defined function.

## Sample output:

|  |
| --- |
| Enter your address:  H no 21, block b, model town, lahore.  Your address is: H no 21, block b, model town, lahore.  **New address is: H n 21, blck b, mdl twn, lhr.** |

1. Write a C++ program which ask user to enter his favourite sentence in capital letters or lower case letters and store this in an array of character type. Write down a function which takes this array as parameter and convert the case (to lower if it is in capital letters or vice versa.) and return array to main function. Print this sentence in main function.

**Note**: *Please don’t use any built-in method for case conversion.*

## Sample output:

|  |
| --- |
| Enter your favourite sentence: ALL IS WELL  **OUTPUT**:  all is well |

1. Write a C++ program having a user defined function( with name: printCommonElements) which takes two integers arrays (take two arrays of size 20 and fill them using random function with values in range (0-20)) as parameter and store common elements of two arrays in 3rd array (declare this in user defined function). First index of this 3rd array will contain total no. of common elements found. Return this array to main function and print all common elements.

## Sample output:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Array\_01:**  0 1 2 3 4 5 6 7 8 9   |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 0 | 5 | 7 | 3 | 5 | 7 | 12 | 11 | 8 | 14 |   **Array\_02:**  0 1 2 3 4 5 6 7 8 9   |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 10 | 15 | 5 | 19 | 14 | 11 | 7 | 14 | 18 | 8 |   **Output**:  5, 5, 7, 11, 8, 14 |

1. Write a C++ program having a user defined function( with name: printUniqueElements) which takes two integers arrays (take two arrays of size 20 and fill them using random function with values in range (0-10)) as parameter and store those elements of array\_01 which are not in array\_02 in 3rd array(declare this in user defined function). First index of this 3rd array will contain total no. of unique elements found. Return this array to main function and print all common elements.

## Sample output:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Array\_01:  0 1 2 3 4 5 6 7 8 9   |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 0 | 5 | 7 | 3 | 5 | 7 | 12 | 11 | 8 | 14 |   Array\_02:  0 1 2 3 4 5 6 7 8 9   |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 10 | 15 | 5 | 19 | 14 | 11 | 7 | 14 | 18 | 8 |   **Output**:  3, 0, 3, 12 |