**Exercise Objective(s):** *Multi-dimensional arrays*

**Exercise:** *Write a program to construct two matrices and display the sum of those.*

**Recommended duration:** *30 Mins*

**Solution Guidance (if applicable):** *NA*

**Lab Exercise No:** 13

**Exercise Objective(s):** *String class, String immutability*

**Exercise:** *Write class that declares the following String.*

***“The quick brown fox jumps over the lazy dog”.***

*Perform the following modifications to the above string using appropriate methods.*

1. *Print the character at the 12th index.*
2. *Check whether the String contains the word “is”.*
3. *Add the string “and killed it” to the existing string.*
4. *Check whether the String ends with the word “dogs”.*
5. *Check whether the String is equal to “The quick brown Fox jumps over the lazy Dog”.*
6. *Check whether the String is equal to “THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG”.*
7. *Find the length of the String.*
8. *Check whether the String matches to “The quick brown Fox jumps over the lazy Dog”.*
9. *Replace the word “The” with the word “A”.*
10. *Split the above string into two such that two animal names do not come together.*
11. *Print the animal names alone separately from the above string.*
12. *Print the above string in completely lower case.*
13. *Print the above string in completely upper case.*
14. *Find the index position of the character “a”.*
15. *Find the last index position of the character “e”.*

**Recommended duration:** *30 Mins*

**Solution Guidance (if applicable):** *NA*

**Lab Exercise No:** 14

**Exercise Objective(s):** *StringBuilder class*

**Exercise:** *Given an array of 10 students’ names. Convert the array as a single string and print it.*

**Recommended duration:** *15 Mins*

**Solution Guidance (if applicable):** *Hint: Efficient use of memory is the focus here*

**Lab Exercise No:** 15

**Exercise Objective(s):** *String class*

**Exercise:** *Write a program to check whether the given strings are an anagram or not. An anagram is a word or a phrase made by transposing the letters of another word or phrase; for example, “Ate" is an anagram of “Eat". The program should ignore white space and punctuation.*

**Recommended duration:** *30 Mins*

**Solution Guidance (if applicable):** *NA*