

**Daffodil International University**  
**Department of Computer Science and Engineering**  
**Semester Final, Summer 2021**  
**Course: Compiler Design Lab, Course No: CSE332, Section- PC-C**  
**Full Marks: 40, Time: 3 hrs**

1. Consider the following strings:

**String 1:** #ACTTTTTTAAA>TAAGAGGATA>TTAA#AAACCTTTAAAA

**String 2:** TTTTT<AAAACCCCTTTTAAAAACCC##TTTAAAAA

**String 3:** >>GGGTTTAAAAACCCCTACTACTGGTACAAAAATTTTTTA

**String 4:** AAAAAAAA###TTTTTAAAAA>>AAAAACCCCTTTTAAAAA

Compute the length of each string and concatenate them into a single string and then compute the length of the final concatenated string. Reverse the final concatenated string (**Without using built-in library function**). Perform tokenization of the final concatenated string using the symbol #, >, <.

2. Suppose you have designed a new language. In your language you indicate a line comment with '~'. And for indicating a block comment you surround the comment starting with the string '#@' and end the block comment with the string '@#'. Now perform comment removal.

**Example of line comment:**

*~ This is a line comment*

**Example of block comment:**

*#@ This is a*

*block comment @#*

3. Write a **C-program** that accepts inputs starting with three **c**'s, followed by odd number of **a**'s or even number of **b**'s. The strings will always end by two **c**'s or three **d**'s. Write down the regular expression, draw the DFA and then give the C-code.
4. Write a code that can identify a valid identifier from an invalid identifier. And if the identifier is valid than it can also identify all the keywords of C- language.

<b>Sample Input:</b>	<b>Sample Output:</b>
while	valid identifier keyword
var l	valid identifier variable
lab	invalid identifier