Assignment 1

Advanced Algorithms (UE16CS311)

# Problem statement :

1. Implement the following string matching algorithms
   1. Naive String Matching
   2. Horspool
   3. Boyer-Moore
   4. Rabin-karp
   5. Finite automaton
   6. KMP
2. Search for the pattern in the *patterns\_file* in the entire text in the *text\_file*
   1. Perform the search for each algorithm and note down the index of match and time taken for the search. The pattern can occur more than once in the text file.

## Input Format:

1. Text\_file : Contains the entire text on which search has to be performed
2. Patterns\_file : Each line contains a pattern (maximum length of a pattern 105 )

## Output Format:

For each pattern, print the number of pattern matches ‘n’ in a new line. The next n lines to have the shift value of each pattern occurrence in the text.

## Sample input:

Text\_file: GCCGCGCGTAGCGTA

Patterns:

GCG

TAGG

## Sample output:

3

3

5

10

0

## Template provided :

* main.c : Main driver program. The text and pattern files are read and calls to the string matching functions are made.
* Helper.c : Functions to read entire file into a string, get the next pattern and validate results.
* AA\_A1\_UE16CS000\_string\_matching.c : This is the file which **should contain the implementation of each string matching algorithm** (and additional helper functions if needed)

## Deliverables (due by Sunday, 16-Sep-2018 11:59 PM):

1. Soft copy : Source Code - Just the implementation file from the template provided. Name your implementation file as AA\_A1\_<YourSRN>\_string\_matching.c or cpp.