Assignment 2 – Arrays and Strings

Academic Honesty

The assignment is individual work. Students are allowed to consult books and online resources but must acknowledge the resources used in the report.

We use MOSS to detect software plagiarism.

Sign the Academic Honesty Pledge in the report.

Program Specifications

Assuming that all input strings are non-empty (having length of at least 1) and shorter than MAX_LENGTH, implement the following string functions:

- **strgLen**(s): return the length of string s.

- **strgCopy**(s, d): copy the content of string s (source) to d (destination).

- **strgChangeCase**(s): for each character in the string, if it is an alphabet, reverse the case of the character (upper to lower, and lower to upper). Keep the non-alphabet characters as is.

- **strgDiff**(s1, s2): compare strings s1 and s2. Return the index where the first difference occurs. Return -1 if the two strings are equal.

- **strgInterleave**(s1, s2, s3): copy s1 and s2 to s3, interleaving the characters of s1 and s2. If one string is longer than the other, after interleaving, copy the rest of the longer string to s3. For example, given s1 = "abc" and s2 = "123", then s3 = "a1b2c3". If s1 = "abcdef" and s2 = "123", then s3 = "a1b2c3def".

Notes:

- Do not use any C library function at all in your code. Do not add any header file to the code.
- The functions (algorithms) must be the most efficient in terms of running time and memory usage.
- Submit file strg.c. Complete the header with your student and contact information. Include sufficient comments in your code to facilitate code inspection.
- Submit a report in PDF with following information: references (sources); error conditions and actions taken; brief algorithm; running time of the function (algorithm) and a brief explanation. See the template a2report.docx for an example.
- See file a2output.txt for sample input and output.
- Your code will be graded automatically by a script, so make sure to strictly follow the specifications.
- Do not use any output statements (for example, printf) in your code, or they will mess up the grading scripts.
- Use the main( ) function provided to test your code. Understanding the code in the main( ) function is part of the assignment. Do not change the code in the main( ) function in the final submission, or your program will mess up the grading script.