# File IO Basics and Text File Handling assignments

**Mandatory**

1. Read 2 lines of text as single command line argument, validate the arguments, extract the lines, write to file “fout.txt”. Now open file read the content and display. Implement the functions

int write(FILE \*fptr, char \*line);

int read(FILE \*fptr, char \*linebuf);i

[Assume maximum line length as 80]

1. Accept 3 file names as command line arguments. The first 2 are input files in which first file has to be created as an integer file and the second file has to created as a string file. Merge the contents of these 2 files into the 3rd file. It should be one integer from the first file followed by one line from the second file.
   1. Display the merged file.
   2. Add appropriate error handling.
   3. Modularize the program and do it as multi file program.
   4. Remove all memory leaks
   5. Read "Integer file" using fscanf (Formatted I/O)
   6. Read "Strings file" using fgets (Line I/O)
   7. Write "Output file" using fprintf (Formatted I/O)

Example:

f1.txt

10

20

f2.txt:

hello

hi

fout.txt:

10hello

20hi

1. Copy the file “string\_process\_prg.c“ to your local directory. Consider a line length of 80 characters. Create “input.txt” file with appropriate data.
2. Fix the issues (warnings and errors in file).
3. Implement display()
4. Test the program for the expected output i.e to display file contents.
5. Free the allocated memory

**Optional**:

1. Read a line of text with atleast 3 words or more, remove all the duplicate words and finally display the unique words.

//removes duplicates from input double pointer and returns the double pointer array with only one unique words

char \*\* remove\_duplicate(char \*\*arr, int count);

1. WAP to remove duplicate lines in an input file “inp.txt” and write unique lines to an output file “out.txt”.

[Hint: You may reuse the function used to remove duplicates used in Q1 above]