COSC 350 System Software
Lab #3

How to submit

- For each program, you need write detailed comments for each statement.
- You need consider all possible error.
- Submit all codes with output
- You need demonstrate each task in front of instructor during the next lab hours.

Task #1:
Write a c code to copy a content of any readable file (**foo**) to a file (**clone**) without open input file (use standard input and output (file descriptor 0 and 1) with input output redirection).

Task #2:
Write a c code to copy a content of any readable file (**foo**) to a file (**clone1**) with open files. (created output file mode will be rw-rw-rw). read byte by byte.

Task #3:
Write a c code to copy a content of any readable file (**foo**) to a file (**clone2**) with open files using a buffer with size 32 byte (created output file mode will be rwx-rwx---).

Task #4:
Write a simple c code which open two files (**foo, foo1**)and append to a file (**foo12**) by using the lseek system call (created output file mode will be rwxrw---).

Task #5:
Write a simple c code which open a file (**foo**) as a input and write into a file by using the lseek system call (**foorev**)as a reverse order (created output file mode will be rwxrw---).

Task #6:
Rewrite Task#5 by using **pread()** system call instead of using lseek() system call.

Task #7:
Write a C program that receive sequence of integers on the command line and prints their sum to the screen. Define your own function to convert string to integer instead of using atoi function.
Exit the program with an appropriate error message under the following error condition: There is not at least one integer on the command line.

#Task #8:
Write C code which pass input (text file) and output file name as command line arguments. Open the input file as read only and open output file with mode rw-rw-rw. Your program encodes each character to ASCII code number and writes to output file. You need consider argument number error and open file error. You must not use any library function to convert a character to ASCII number.

Task #9:
A **palindrome** is a word, phrase, number or other sequence of units that can be read the same way in either direction. Write a C-function **int palind(int fd1, int fd2)** that takes two independent file descriptors fd1 and fd2 that are already opened to the same file. The function palind() uses lseek to scan the file and returns 1 if the file contains a palindrome (reads the same forward and reverse) and 0 if not. You may assume that the file is well-formed and contains just lower-case alphabetic characters on a single line. You also need write a main function to test your palind() function works properly. The main function accept a file name as a argument. The main() function open the file once and create duplicate file descriptor. And call the function palind() to check whether the file contains palindrome or not.