Program 1: **Using Files - Numeric Processing**

- The file named Random.txt contains a long list of random numbers. Copy the file to your hard drive and then write a program that opens the file, reads all the numbers from the file, and calculates the following:

  A. The number of numbers in the file
  B. The sum of all the numbers in the file (a running total)
  C. The average of all the numbers in the file

- The program should display the number of numbers found in the file, the sum of the numbers, and the average of the numbers.

Program 2: **Using Files – Student Line Up**

- The file named LineUp.txt contains a list of student names. Write a program that reads all student names and report two students whose name appears at the front of the phone book and the name appears at the last of the phone book. You may assume that no two students have the same name.

Program 3: **Using Files – Saving Account Balance**

- Write a program that calculates the balance of a savings account at the end of a period of time. It should ask the user for the annual interest rate, the starting balance, and the number of months that have passed since the account was established. A loop should then iterate once for every month, performing the following:
  A. Ask the user for the amount deposited into the account during the month. (Do not accept negative numbers.) This amount should be added to the balance.
  B. Ask the user for the amount withdrawn from the account during the month. (Do not accept negative numbers.) This amount should be subtracted from the balance.
  C. Calculate the monthly interest. The monthly interest rate is the annual interest rate divided by twelve. Multiply the monthly interest rate by the balance, and add the result to the balance.
- After the last iteration, the program should write the following to “Report.txt”.
  - the ending balance, the total amount of deposits, the total amount of withdrawals, and the total interest earned.

Program 4: **Copying file A to file B**

- Write a program that will prompt the user for the names of input and output files. It will then copy the input file to the output file by byte-by-byte.
- After the program completes, compare whether the input and out files are identical in CLion

Program 5: **printing strings from input file A, i.e., HelloWorld.cpp**

- Write a program that will prompt the user for the names of input and print every instance of string in the input file to the console with the line number.