// OOP345 Workshop 9: **Multi-Threading**

// File: reflect.txt

// Name: Jingmin (Jessica) Zhou

// Email: jzhou175@myseneca.ca

// ID: 119766194

// Date: 07/28/2020

In Workshop 9 (Multi-Threading), I have learned how to use multiple threads for processing partitioned data, read and write a set of characters from/into in binary mode and bind a function to its arguments.

Firstly, a binary file is usually very much smaller than a text file (e.g. input.txt) that contains an equivalent amount of data. Because small files save storage space, can be transmitted faster, and are processed faster, binary file is useful for encrypting and decrypting a file in this workshop, e.g. std::ifstream::binary and std::ofstream::binary are used to read and write a binary file.

Secondly, binding a function to its arguments helps to manipulate the position and number of values to be used by the function and modifies the function according to the desired output. In this workshop, converter() was bound to its parameters and return its function object. When the function object is called in thread(), it would return the result for these specific parmaters.

Thirdly, multiple threads allow the execution of multiple parts of a program at the same time and lead to maximum utilization of the CPU by multitasking. These parts are known as threads and are lightweight processes available within the process. In this workshop, two threads are used to encrypt and decrypt the text file.

===========================================================================================================================================

Quiz 7 reflection: 8/8.