**Operating System Principles Assessment Tasks Report**

**TASK 1**

In this task we are creating a bash function and function to filter through the infile using a single word.

**TaskFilter Function:**

**int TaskFilter(const std::string& input, const std::string& output)**

-> we use a set to contain the readLines we are ensuring they're no duplicate entries

-> If the string is not between 3 and 15 (inclusive), skip

-> If the string is not only lowercase chars ([A-Z][0-9] and special chars, skip)

-> If the string contains more than 2 conseq character, skip

-> Pushing the string to the set handles the uniqueness of each line

**Analysis:**

Shell in Real time takes 2.14 seconds for 708878 with a speed of 1,516,999 words per sec but when optimized it takes 0.432 seconds with a speed of 3511571 words per sec.

**TASK 3**

In this task we were to implement a map and reduce sorting method using two different pthreads.*pThread\_create* creates concurrent threads within the parent process, similarly, to fork 13 mapping threads are created, but also 13 reducing threads are created (as well as the 2 threads to initiate mapping and reducing).

Other than the different concurrency methods, the fork implementation is also using basic .txt file outputs for mapping state, whereas the thread-based implementation is using FIFO files and piping data from one mapping thread to the matching reduce thread before combining and sorting.

**TASK 4**

In this task we were to implement a map and reduce sorting method using two different pthreads . *pThread\_create* creates concurrent threads within the parent process, 13 mapping threads are created, but also 13 reducing threads are created (as well as the 2 threads to initiate mapping and reducing).

Other than the different concurrency methods, the fork implementation is also using basic .txt file outputs for mapping state, whereas the thread-based implementation is using FIFO files and piping data from one mapping thread to the matching reduce thread before combining and sorting.

**Analysis:**

Thread in Real time takes 2.980 seconds for 1,516,999 with a speed of 509401 words per sec but when optimized it takes 2.964 seconds with a speed of 511980 words per sec

**Time Complexity Analysis:**

Big O for Task 3 and 4 goes to O(n^2)