**COP 5611 - Advanced Operating Systems**

**Weekly Project Report - 6**

**Team: Diablo**

**Coding Implementation Design:**

There were two optimizations that we decided would assist the performance of the processing our code was outputting.

The first optimization is related to the levels of patterns that are formed. At first we were developing patterns on the basis of levels. That is, a two level pattern would consist of two websites accessed under the pattern criteria, a three level pattern would consist of three websites accessed under the pattern criteria, and so on. However, we could benefit from the fact that if a certain two level pattern of sites are not resulted with a certain chain (say sites A, B), then there was no point in considering a combination of sites starting with the non-resulting chain at further levels. To clarify, there is no need to check for patterns with sites formed by the non-resulting chains at lower levels (There is no need to look for patterns such as A, B, C or A, B, D etc.). This dropping of combination was eventually a performance booster.

The second change that we applied was related to the number of patterns formed with respect to a particular pattern type for the window it currently resides in. If a pattern type A has 50 patterns including the previous window, and the previous window accounts for 5 of such patterns, then these 5 patterns would be dropped and the resulting pattern count would be 45. This creates the liveliness of patterns for each sliding window.

**Team member: Harish Chetty**

I worked on the following part related to the implementation:

I worked on the first optimization scheme involving the removal of search for combination of pattern chains at higher levels for non-existent pattern chains at lower levels.

**Team member: Gaurav Sinha (gs13h)**

I worked on the following part related to the implementation:

I worked on the second chain dealing with recalculating pattern count for a specific pattern with respect to the current sliding window by removing the pattern count for the previous window.