**See at end of report

Sandmark before optimization is 4 minutes and 1.747 seconds
Sandmark after optimization is 1 minute and 13.174 seconds** (79.77% improvement)

Midmark before optimization is 9.702 seconds
Midmark after optimization is 2.945 seconds** (69.65% improvement)

Advent before optimization is 47.812 seconds
Advent after optimization is 27.572 seconds** (73.4% improvement)

Processes changed:

-Remove function calls in execute.c that do not relate to segment and add in place
(Effectively removes operator.c)

Sandmark: 3 minutes 46.417 seconds
Midmark: 9.089 seconds

These yield a 6.634% improvement in Sandmark
These yield a 6.744% improvement in Midmark

-Remove redundant asserts from all files (specifically redundant asserts in segemnt.c)

Sandmark: 3 minutes 32.646 seconds
Midmark: 8.561 seconds

These yield a 6.476% improvement in Sandmark
These yield a 6.168% improvement in Midmark

-Remove bitpack from execute.c and add in place

Sandmark: 1 minute and 49.917 seconds
Midmark: 4.431 seconds

These yield a 51.690% improvement in Sandmark
These yield a 51.758% improvement in Midmark

-Remove make_word from execute.c and add in place

Sandmark: 1 minute 27.933 seconds
Midmark: 3.540 seconds

These yield a 21.111% improvement in Sandmark
These yield a 21.219% improvement in Midmark

-Add the registers to the heep and make them globally available

Sandmark: 1 minute 22.448 seconds
Midmark: 3.339 seconds

These yield a 6.312% improvement in Sandmark
These yield a 6.213% improvement in Midmark

-Remove all length calculations that are in loops and define beforehand

Sandmark: 1 minute 22.445 seconds
Midmark: 3.331 seconds

These yield a ~.00% improvement in Sandmark
These yield a .034% improvement in Midmark

-Implement inline functions where logical

Sandmark: no difference
Midmark: no difference

These yield a ~0% improvement in Sandmark
These yield a ~0% improvement in Midmark

-Find the 4 pieces in memory that were never freed according to valgrind

Sandmark: no difference
Midmark: no difference

These yield a ~0% improvement in Sandmark
These yield a ~0% improvement in Midmark

-Remove bitpack usage and do in place in read file

Sandmark: 1 minute 13.174 seconds
Midmark: 2.945 seconds

These yield a ~1.082% improvement in Sandmark
These yield a ~1.131% improvement in Midmark

**All of these times were tested against each case using ./um <filepath> and we did not know you could just type “um <filepath>” and surprisingly it made a huge difference.

Times with running just “um <filepath>” are as follows:

Sandmark: 14.75 seconds
Midmark: 0.57 seconds

These yield a ~7.672% improvement in Sandmark
These yield a ~5.166% improvement in Midmark
These were our final times.