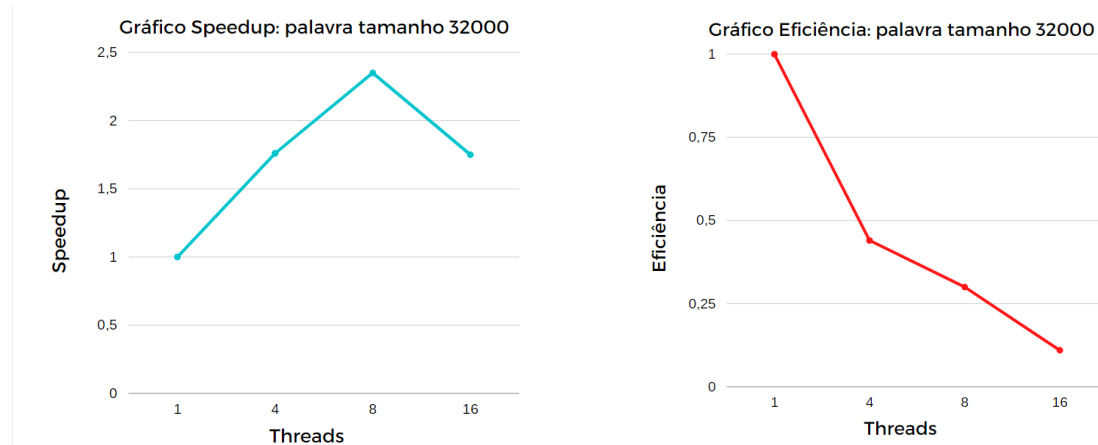


Nome: Arthur Machado Capaverde

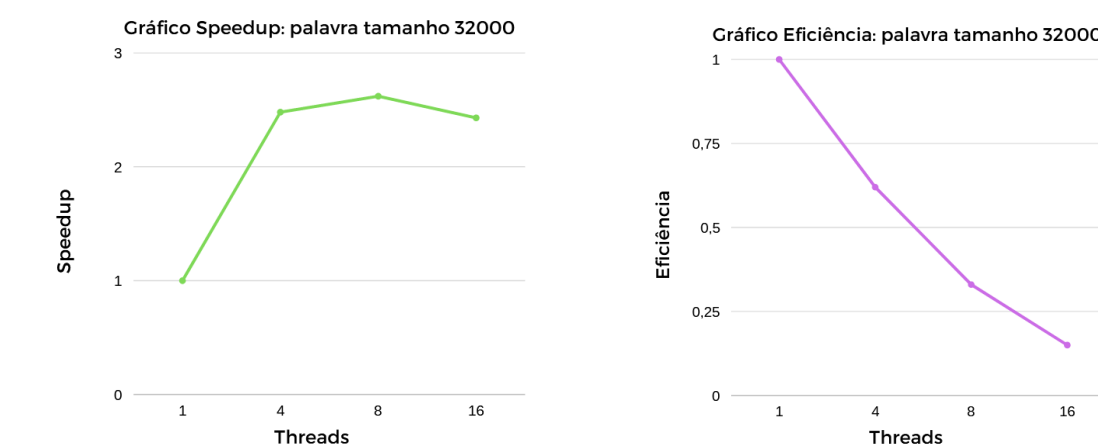
Matrícula: 17207215

Testes realizados em uma máquina com 4 cores.

POSIX THREADS:

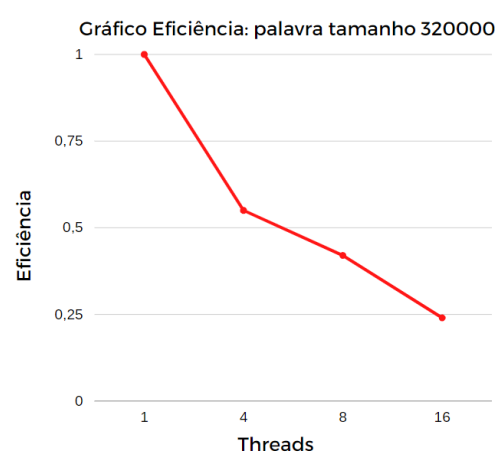
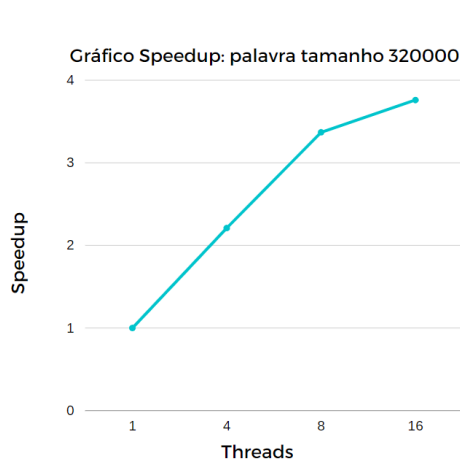


OPEN_MP:

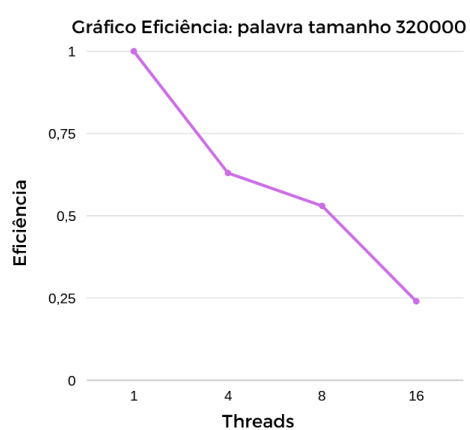
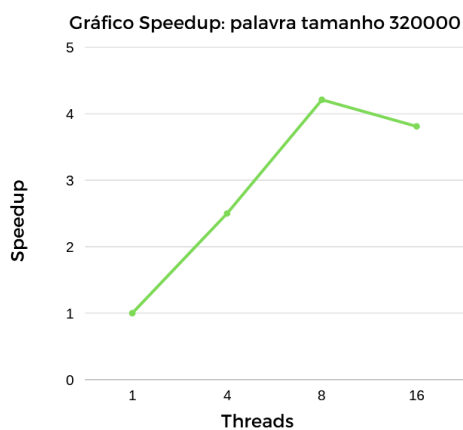


Quando a palavra tem um tamanho relativamente pequeno, observa-se que com um número excessivo de threads o Overhead se torna alto e há pouco ganho de velocidade ao aumentar o número de threads além de 4. Assim, no OPEN_MP, conclui-se que quando a região paralelizável é pequena o aumento no número de threads não causa grandes mudanças, em contraste com as threads POSIX, em que o aumento de threads causou um overhead alto que piorou o desempenho com um número alto de threads. Com o OPEN_MP o speedup foi muito maior para 4 e 16 threads e continuou o mesmo para 8, mostrando que o OPEN_MP causa um overhead menor.

POSIX THREADS:

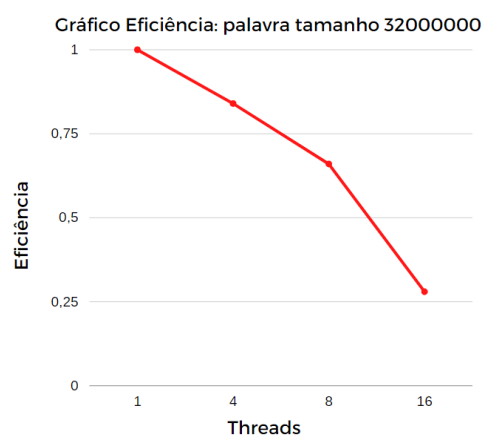
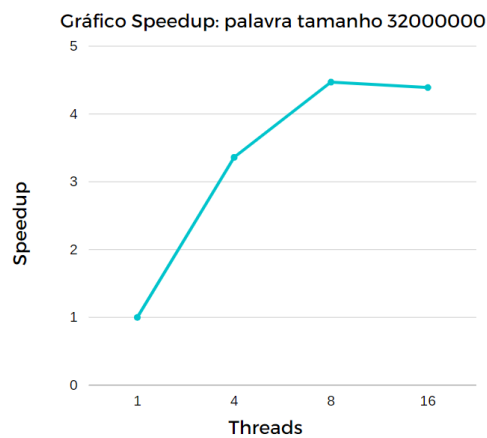


OPEN_MP:

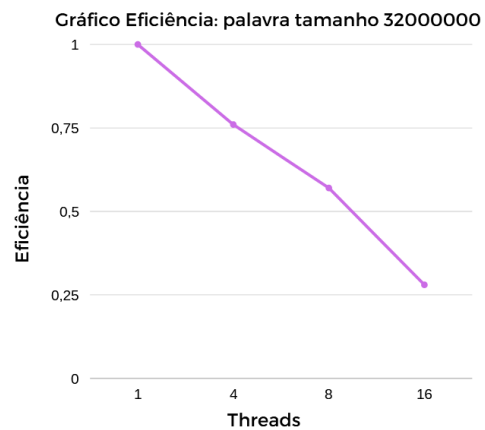
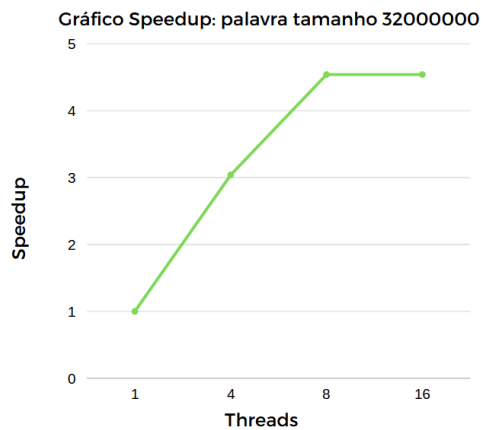


Quando a palavra tem um tamanho mediano o OPEN_MP teve um desempenho muito melhor do que as threads POSIX com 8 threads, enquanto os outros casos se mantiveram relativamente iguais.

POSIX THREADS:



OPEN_MP:



Quando a palavra é relativamente grande percebemos que as threads POSIX começam a ter um desempenho um pouco melhor do que o OPEN_MP quando utilizando apenas 4 threads. Mas nos outros casos os resultados foram extremamente parecidos, levando a (possivelmente errada) conclusão de que com palavras suficientemente grandes as duas ferramentas de paralelização tem desempenho praticamente igual. Vale destacar também que por ser usada uma palavra muito maior, a perda de eficiência se deu de maneira extremamente mais tênue do que quando comparada com os outros tamanhos de palavras.