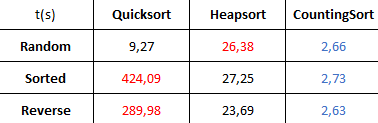
W każdym przypadku zbiór do posortowania składał się z 1000 elementów a czas podany w tabeli jest czasem sumarycznym wykonania sortowania 10000 razy.



Sortowanie przez kopcowanie i zliczanie osiąga podobne rezultaty dla 1000-elementowego zbioru niezależnie od ułożenia elementów w środku podczas gdy efektywność sortowania szybkiego spada w przypadku posortowanej listy 40-krotnie a odwrotnie posortowanej 30-krotnie.

Dla losowo ułożonych list o takim rozmiarze wydaje się być sortowanie przez zliczanie a następnie – kopcowanie. Wydaje mi się, że jest to wina pivota, który jest ustawiony na ostatni element listy w wyniku czego wszystkie pozostałe elementy listy znajdą w „sekcji” większych od pivota lub mniejszych – zależnie od tego czy lista jest posortowana czy też posortowana odwrotnie.

Link do repozytorium: <https://github.com/s20163/ASD/blob/main/asd.py>