HEAPSORT

O algorytmie:

**Sortowanie przez kopcowanie** (ang. *heapsort*) – jeden z algorytmów sortowania, choć niestabilny, to jednak szybki i niepochłaniający wiele pamięci (złożoność czasowa wynosi \scriptstyle \mathrm O(n \log n), a pamięciowa – \scriptstyle \mathrm O(n), przy czym jest to rozmiar sortowanych danych, złożoność pamięciowa dodatkowych struktur wynosi \scriptstyle \mathrm O(1); jest to zatem algorytm sortowania *w miejscu*). Jest on w praktyce z reguły nieco wolniejszy od sortowania szybkiego, lecz ma lepszą pesymistyczną złożoność czasową (przez co jest odporny np. na atak za pomocą celowo spreparowanych danych, które spowodowałyby jego znacznie wolniejsze działanie).

Podstawą algorytmu jest użycie kolejki priorytetowej zaimplementowanej w postaci *binarnego kopca zupełnego*. Zasadniczą zaletą kopców jest stały czas dostępu do elementu maksymalnego (lub minimalnego) oraz logarytmiczny czas wstawiania i usuwania elementów; ponadto łatwo można go implementować w postaci tablicy.

Algorytm sortowania przez kopcowanie składa się z dwóch faz. W pierwszej sortowane elementy reorganizowane są w celu utworzenia kopca. W drugiej zaś dokonywane jest właściwe sortowanie.

Testy:

Załączam przykładowy plik inputowy (wygenerowany programem pomocniczym do jednego z pierwszych zestawów(4000 liczb)) i outputowy.