

-----  
Polecenie 2.

I.

Złożoność obliczeniowa:

a) Cocktail Sort - W przypadku zwiększenia się rozmiaru tablicy wejściowej czas sortowania może wydłużyć się znacząco.

Dość istotny wpływ na szybkość sortowania algorytmu ma sposób ułożenia danych wejściowych.

Najgorszy przypadek - ciąg malejący - jest o 1/3 czasu dłuższy od przypadku, w którym dane wejściowe są już częściowo posortowane - V-kształny ciąg.

Złożoność obliczeniowa dla ciągów stałych i rosnących jest stała, gdyż ciągi są już posortowane, więc jedynym zadaniem algorytmu jest przeiterowanie całej tablicy, co zajmuje  $O(n)$  czasu, dla pozostałych ciągów jest to złożoność  $O(n^2)$

b) Heap Sort - Zwiększanie się rozmiaru tablicy wejściowej nie ma istotnego wpływu na czas sortowania.

Sposób ułożenia danych wejściowych nie ma istotnego wpływu na czas sortowania.

Złożoność obliczeniowa dla wszystkich typów ciągów jest rzędu  $O(N \log N)$ .

b) Heap Sort - Zwiększanie się rozmiaru tablicy wejściowej nie ma istotnego wpływu na czas sortowania.

Sposób ułożenia danych wejściowych nie ma istotnego wpływu na czas sortowania.

Złożoność obliczeniowa dla wszystkich typów ciągów jest rzędu  $O(N \log N)$ .

c) Insertion Sort - Zwiększanie się rozmiaru tablicy wejściowej ma duży wpływ na czas sortowania.

W tym przypadku, czas sortowania najlepszego - losowego - przypadku jest ponad 2 razy krótszy od czasu sortowania

przypadku najgorszego - malejącego.

Złożoność obliczeniowa dla ciągów rosnących i stałych to  $O(n)$ ,

dla wszystkich innych  $O(n^2)$ .

d) Selection Sort - Zwiększanie się rozmiaru tablicy wejściowej ma duży wpływ na czas sortowania.

W tym przypadku, czas sortowania najgorszego - malejącego - przypadku jest o  $1/4$  dłuższa od

czasu sortowania przypadku najlepszego.

Złożoność obliczeniowa dla dowolnego typu ciągu w przypadku tego algorytmu, wynosi

$O(n^2)$ .

II.

Typ sortowanych danych jest kluczowy dla szybkości sortowania każdego typu algorytmu.

Patrząc na wykresy wywnioskować można, że w przypadku sortowania danych typu Integer

czas tego sortowania jest krótszy - średnio 2 krotnie - od czasu sortowania danych typu Double.

-----  
-----

Polecenie 3.

Złożoność obliczeniowa algorytmu to średnio  $O(N \log N)$ , dla najgorszego przypadku natomiast jest to  $O(n^2)$ .

Wybór klucza dla algorytmu Quick Sort, widocznie wpływa na szybkość sortowania.

W najgorszym przypadku - wybór klucza jest losowy - algorytm sortuje około 2 razy dłużej

niż w przypadku "rozsądniejszego" doboru klucza.

-----