

## Zadanie 3

1. Dla układu losowego QS(skrajny) był trochę szybszy niż QS(średni) z tego powodu, że dla QS(skrajny) tablica liczb w całym losowym porządku jest przypadkiem najlepszym. Dla układu rosnącego to miało dużo znaczenie. Z tego powodu, że QS(skrajny) bierze ostatni element, który, w przypadku układu rosnącego jest największym elementem w całej tablicy, to dlatego algorytm podzieli tablicę na dwie, gdzie z lewej strony będzie ta cała tablica, ale rozmiaru  $n-1$ . Tak będzie dla każdego następnego podziału tablicy, co sprawia, że to jest przypadek najgorszy danego algorytmu -  $O(n^2)$ .
2. Dla QS(Średni) przypadek najlepszy to taki przypadek, gdy pivot jest medianą (np. Układ posortowany), a przypadek najgorszy - gdy pivot element najmniejszy lub największy. Dla QS(Skrajny) przypadek najlepszy to tablica w całym losowym porządku, a przypadek najgorszy to tablica posortowana(rosnąco lub malejąco).
3. Podział przez medianę. Zaleta: bardzo przyspieszy algorytm, Wada: najgorszy przypadek poszukiwania mediany -  $O(n^2)$ . Podział przez medianę trzech. (Bierze pierwszy, średni i ostatni element, sortuje ich i pivot w tym przypadku - to element średni). Zaleta: jest szansa przyspieszyć algorytm i nie wybrać element skrajni, Wada: jest szansa, że każdy z trzech elementów będzie elementem skrajnym.
4.  $O(\log n)$
5. QS w ogóle jest lepszy niż IS, dlatego że mają taki sam przypadek najgorszy, ale QS ma lepszy przypadek średni. Ale to jeszcze bardzo zależy od sposobu wybrania pivotu. Bo jak możemy zobaczyć, QS(średni) jest szybszy dla dwóch układów, ale QS(skrajny) bardzo przegrywa w przypadku, gdy zachodzi o "zachowaniu naturalnym".
6. **IS** - Zalety: działa w miejscu, dobry przypadek najlepszy, ma zachowanie naturalne, jest stabilny, prosty w implementacji. Wady: przypadek najgorszy -  $O(n^2)$ , jest wrażliwy na dane.

**SS** - Zalety: działa w miejscu, nie jest wrażliwy na układ danych, prosty w implementacji. Wady: przypadek najlepszy = przypadek najgorszy = przypadek średni =  $O(n^2)$ , bardzo słabe zachowanie naturalne, nie jest stabilny, nie ma zachowania naturalnego.

**BS** - Zalety: działa w miejscu, jest stabilny, nie jest wrażliwy na układ danych, bardzo prosty w implementacji. Wady: przypadek najlepszy = przypadek najgorszy = przypadek średni =

$O(n^2)$ , nie ma zachowania naturalnego.

**QS** - Zalety: działa w miejscu, dobry przypadek optymistyczny ( $n \log n$ ), dobry przypadek średni ( $n \log n$ ), ma zachowanie naturalne (zależy). Wady: nie jest stabilny, potrzebna jest pamięć na rekurencję, jest bardzo wrażliwy na dane, przypadek najgorszy ( $O(n^2)$ ), skuteczność zależy od wybrania punktu podziału.

**MS** - Zalety: Przypadek najgorszy = przypadek średni = przypadek najlepszy =  $O(n \log n)$ , jest stabilny, nie jest wrażliwy na dane. Wady: nie działa w miejscu  $O(n)$ , nie ma zachowania naturalnego.

**HS** - Zalety: działa w miejscu, przypadek najgorszy = przypadek średni =  $O(n \log n)$ , przypadek najlepszy  $O(n)$ . Wady: jest wrażliwy na dane, nie jest stabilny, budowa zajmuje czas (przypadek najgorszy -  $O(\log n)$ , przypadek najlepszy  $O(1)$ ), jest nie prosty w implementacji.

**CS** - Zalety: przypadek najlepszy = przypadek średni = przypadek najgorszy =  $O(n+k)$ , jest stabilny. Wady: sortuje liczby z wąskiego zakresu, nie działa w miejscu, jest bardzo wrażliwy na dane.

7. Żeby wybrać metodę sortowania trzeba zwrócić uwagę na czas wykonywania algorytmu sortowania, pamięć, którą algorytm sortowania używa i liczby, które trzeba posortować (liczby dodatnie, liczby ujemne, ilość liczb, maksymalne i minimalne możliwe liczby). Na przykład:

Najszybszymi metodami sortowania są QS i CS. W takim razie najlepiej używać ich.

Ale CS działa tylko na wąskim zakresie liczb i jest lepszy, gdy różnica między największym i najmniejszym elementami jest niezbyt duża.

QS jest uniwersalną metodą sortowania, która jest zawsze szybka i nie używa wielu pamięci, ale jest wrażliwa na punkt podziału tablicy.