

## Platforma testowa:

Procesor: Intel Core i3-8100

Płyta główna: AsRock Z370 Pro4

Pamięć RAM: G.Skill 8GB(2x4GB)

Dysk twardy: GoodRam IRDM 240GB

Nazwa metody sortowania	Długość tablicy	Rodzaj tablicy	Liczba porównań	Liczba przestawień	Czas
Cocktail sort	100	tablica uzupełniona losowymi liczbami	3272	2432	~0s
Lomuto Quicksort	100	tablica uzupełniona losowymi liczbami	714	451	~0s
Hoare'a Quicksort	100	tablica uzupełniona losowymi liczbami	456	178	~0s
Heapsort	100	tablica uzupełniona losowymi liczbami	1290	578	~0s
Cocktail sort	100	tablica posortowana rosnąco	99	0	~0s
Lomuto Quicksort	100	tablica posortowana rosnąco	481	287	~0s
Hoare'a Quicksort	100	tablica posortowana rosnąco	482	63	~0s
Heapsort	100	tablica posortowana rosnąco	1376	639	~0s
Cocktail sort	100	tablica posortowana malejąco	4950	4945	~0s
Lomuto Quicksort	100	tablica posortowana malejąco	517	340	~0s
Hoare'a Quicksort	100	tablica posortowana malejąco	384	113	~0s
Heapsort	100	tablica posortowana malejąco	1128	517	~0s
Cocktail sort	100	tablica częściowo nieuporządkowana	674	474	~0s
Lomuto Quicksort	100	tablica częściowo nieuporządkowana	517	324	~0s
Hoare'a Quicksort	100	tablica częściowo nieuporządkowana	474	76	~0s
Heapsort	100	tablica częściowo nieuporządkowana	1364	625	~0s

Dla tablicy o długości 100(powyższej) lub 1000(poniższej) elementów nie widzimy dużych różnic w liczbie porównań i przestawień we wszystkich algorytmach. Czasy wykonywania są zerowe lub zbliżone do zera dla każdego algorytmu i typu danych. Dla tablicy już posortowanej cocktail sort jest najsprawniejszy. Dla tablicy prawie uporządkowanej wyniki cocktail sort'a są zbliżone do quicksort'a. Wniosek jaki z tego płynie dla małej ilości n elementów jak 100,1000 oraz każdego typu danych wejściowych sprawdzają się wszystkie przedstawione tutaj algorytmy.

Nazwa metody sortowania	Długość tablicy	Rodzaj tablicy	Liczba porównań	Liczba przestawień	Czas
Cocktail sort	1000	tablica uzupełniona losowymi liczbami	336 092	252 037	0,005s
Lomuto Quicksort	1000	tablica uzupełniona losowymi liczbami	11538	5704	~0s
Hoare'a Quicksort	1000	tablica uzupełniona losowymi liczbami	7206	2623	~0s
Heapsort	1000	tablica uzupełniona losowymi liczbami	19180	9059	~0s
Cocktail sort	1000	tablica posortowana rosnąco	999	0	0s
Lomuto Quicksort	1000	tablica posortowana rosnąco	8007	4491	~0s
Hoare'a Quicksort	1000	tablica posortowana rosnąco	8015	548	~0s
Heapsort	1000	tablica posortowana rosnąco	20368	9689	~0s
Cocktail sort	1000	tablica posortowana malejąco	499500	499453	0,011s
Lomuto Quicksort	1000	tablica posortowana malejąco	8440	4965	~0s
Hoare'a Quicksort	1000	tablica posortowana malejąco	7028	1048	~0s
Heapsort	1000	tablica posortowana malejąco	17648	8319	~0s
Cocktail sort	1000	tablica częściowo nieuporządkowana	67901	63331	0,002s
Lomuto Quicksort	1000	tablica częściowo nieuporządkowana	10522	6748	~0s
Hoare'a Quicksort	1000	tablica częściowo nieuporządkowana	7590	901	~0s
Heapsort	1000	tablica częściowo nieuporządkowana	20166	9583	~0s

Nazwa metody sortowania	Długość tablicy	Rodzaj tablicy	Liczba porównań	Liczba przestawień	Czas
Cocktail sort	100 000	tablica uzupełniona losowymi liczbami	3 336 399 022	2 498 801 056	58,871s
Lomuto Quicksort	100 000	tablica uzupełniona losowymi liczbami	2 0333 210	1 131 089	0,035s
Hoare'a Quicksort	100 000	tablica uzupełniona losowymi liczbami	1 459 074	411 491	0,022s
Heapsort	100 000	tablica uzupełniona losowymi liczbami	3 249 544	1 574 652	0,074s
Cocktail sort	100 000	tablica posortowana rosnąco	99999	0	0s
Lomuto Quicksort	100 000	tablica posortowana rosnąco	1 469 144	783 382	0,022s
Hoare'a Quicksort	100 000	tablica posortowana rosnąco	1 470 506	66 091	0,007s
Heapsort	100 000	tablica posortowana rosnąco	3 399 646	1 649 937	0,07s
Cocktail sort	100 000	tablica posortowana malejąco	4 999 950 000	4 999 945 064	103,207s
Lomuto Quicksort	100 000	tablica posortowana malejąco	1 514 922	838 331	0,024s
Hoare'a Quicksort	100 000	tablica posortowana malejąco	1 370 511	116 092	0,009s
Heapsort	100 000	tablica posortowana malejąco	3 094 894	1 497 575	0,066s
Cocktail sort	100 000	tablica częściowo nieuporządkowana	608 552 807	580 726 330	12,113
Lomuto Quicksort	100 000	tablica częściowo nieuporządkowana	2 374 730	1 554 519	0,051s
Hoare'a Quicksort	100 000	tablica częściowo nieuporządkowana	1 396 477	123 124	0,021s
Heapsort	100 000	tablica częściowo nieuporządkowana	3 371 292	1 635 777	0,083s

Dla tablicy o długości 100 000 elementów zauważamy ogromne różnice dla poszczególnych algorytmów. Cocktail sort dla losowych, posortowanych malejąco i częściowo nieuporządkowanych wartości jest kompletnie nie efektywny i należy go unikać. Oba quicksort'y sprawują się znakomicie, osiągając świetne wyniki czasowe. Heapsort również osiąga stabilne, zadowalające wyniki, jednakże nieco gorsze od quicksortów. Co ciekawe z tabeli wynika, że algorytm Hoare'a daje jeszcze lepsze wyniki niż Lomuto.

## Wnioski

Wszystkie zaprezentowane algorytmy sortują w miejscu. Cocktail sort ma typową klasę złożoności  $O(n^2)$  co widać wyraźnie po wynikach dla większej ilości elementów. Dla przypadku optymistycznego, czyli gdy tablica jest już posortowana klasa złożoności wynosi  $O(n)$ . Quicksort ma typową klasę złożoności  $O(n \log n)$ , przy czym dla przypadku pesymistycznego, złożoność wynosi  $O(n^2)$ . Natomiast heapsort ma stałą złożoność  $O(n \log n)$ . Jak widzimy po wynikach czasowych quicksort najlepiej się sprawdza praktycznie w każdej sytuacji. Jeśli obawiamy się występowania przypadków pesymistycznym dla quicksorta możemy zastosować heapsort, który daje również przyzwoite wyniki czasowe dla dużej liczby elementów. Jeśli chodzi o implementację, zdecydowanie najprostszy jest cocktail sort, pozostałe trzy stoją na podobnym poziomie zaawansowania.