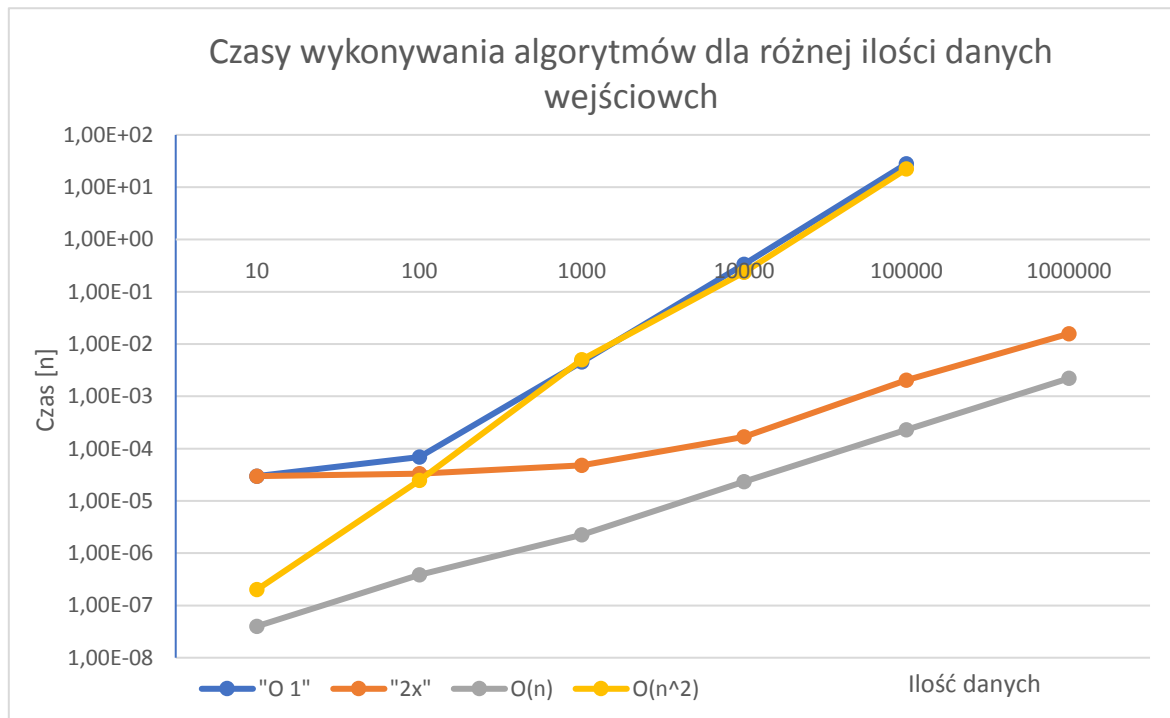


Powiększanie Tablicy Dynamicznej

Badaniu podlegają dwie metody powiększania tablicy dynamicznej: PowiększO1 (powiększa tablicę o 1 element) oraz Powiększ2x (powiększa rozmiar tablicy dwukrotnie).

W celu ułatwienia prowadzenia badań napisano program test.cpp, który liczy średnią czasów wykonywania z podanej ilości wywołań. Do celów analitycznych wykonano pomiary algorytmów o złożoności obliczeniowej $O(n)$ i (n^2) . Dla metody powiększającej tablicę o 1 i dla algorytmu o złożoności $O(n^2)$ nie można było uzyskać wyników przy 1 000 000 danych wejściowych.

Ilość danych wejściowych	Średni czas z 20 wywołań [s]			
	"O1"	"2x"	$O(n)$	$O(n^2)$
n				
10	0,0000297646	0,0000297646	0,000000004	0,00000002
100	0,0000689437	0,0000331726	0,000000383	0,00002447
1000	0,00455578	0,0000476804	0,00000224	0,005
10000	0,335063	0,000168176	0,0000233	0,236
100000	28,021	0,00202628	0,00023	22,27
1000000	-	0,015717	0,0022	-



Na wykresie widać, że powiększanie tablicy dynamicznej „O1” przy rozmiarach większych niż 100 jest znacznie wolniejsze od powiększania „2x”. Powiększanie tablicy dwukrotnie prowadzi jednak do alokowania większej ilości pamięci niż jest potrzebna (np. mają tablice o rozmiarze 10 chcąc wpisać 11. element tablica zostanie powiększona do rozmiaru 20). W zależności od potrzeb metody te mogą być stosowane zamiennie. Ograniczeniem dla korzystania z metody „O1” jest rozmiar tablicy (nie można było otrzymać wyniku dla 1 000 000). Widać, że czas wykonywania metody „O1” pokrywa się z czasem wykonywania innego algorytmu o złożoności obliczeniowej $O(n^2)$, różna dla $n = 10$ wynika z tego, że dla 10 wykonuje się tylko inicjalizacja tablicy, a nie sam algorytm. Można sądzić, że powiększanie o 1 ma złożoność obliczeniową $O(n^2)$. Złożoność algorytmu powiększania o 2 razy znajduje się między złożonością liniową i kwadratową.