

Lista dwukierunkowa cykliczna – losowo

Lista - losowo	Ilość wst.	Wstawianie	Ilość wysz.	Wyszukiwanie	Ilość us.	Usuwanie
N = 10000	9202	732 ms	1560	1448 ms	1361	1242 ms
N = 15000	13258	1742 ms	3247	3174 ms	2724	2658 ms
N = 20000	17050	2855 ms	5481	5370 ms	4241	4594 ms
N = 25000	20559	4391 ms	8087	8266 ms	6021	6935 ms
N = 30000	23821	6141 ms	10978	11605 ms	7904	9555 ms

BST - losowo

BST - losowo	Ilość wst.	Wstawianie	Ilość wysz.	Wyszukiwanie	Ilość us.	Usuwanie
N = 10000	9202	19 ms	1559	14 ms	1361	15 ms
N = 15000	13258	26 ms	3246	23 ms	2724	22 ms
N = 20000	17050	31 ms	5479	26 ms	4241	32 ms
N = 25000	20559	44 ms	8087	37 ms	6021	45 ms
N = 30000	23820	52 ms	10978	43 ms	7903	47 ms

Lista dwukierunkowa cykliczna – przemian

Lista - przemian	Ilość wst.	Wstawianie	Ilość wysz.	Wyszukiwanie	Ilość us.	Usuwanie
N = 10000	9836	755 ms	5630	1198 ms	5361	511 ms
N = 15000	14018	1466 ms	8948	2261 ms	8244	1184 ms
N = 20000	18279	2664 ms	12412	3906 ms	11242	1991 ms
N = 25000	22342	4203 ms	16043	6123 ms	14299	2919 ms
N = 30000	26258	5659 ms	19975	8399 ms	17317	3989 ms

BST - przemian

BST - przemian	Ilość wst.	Wstawianie	Ilość wysz.	Wyszukiwanie	Ilość us.	Usuwanie
N = 10000	9536	20 ms	5630	19 ms	5361	13 ms
N = 15000	14018	30 ms	8948	23 ms	8244	18 ms
N = 20000	18279	37 ms	12411	27 ms	11242	23 ms
N = 25000	22342	46 ms	16042	36 ms	14299	30 ms
N = 30000	26258	52 ms	19974	43 ms	17317	39 ms

Wykonywane dla: srand(20)

Wnioski:

W liście w przypadku pesymistycznym przy wstawianiu elementu musimy przejść przez wszystkie N elementów, co znacznie wpływa na czas wykonania przy dużej ilości elementów. W przypadku optymistycznym stanie się to za pierwszym razem lecz przeważnie będziemy przechodzić przez połowę drogi.

Przy drzewie BST przypadkiem pesymistycznym jest dodanie elementu na maksymalnej wysokości drzewa, co i tak jest kilkaset razy szybsze w porównaniu z listą. Im więcej elementów, tym większą przewagę w czasie wykonania zyskuje drzewo BST.