

Politechnika Śląska w Gliwicach

Wydział Automatyki, Elektroniki i Informatyki



# Algorytmy i Struktury Danych

## Algorytmy sortowań

---

autor	Paweł Rother
prowadzący	mgr inż. Agnieszka Danek
rok akademicki	2015/2016
kierunek	teleinformatyka
rodzaj studiów	SSI
semestr	2
termin laboratorium	środa nieparzysta 10:15 – 11:45
grupa	1
data oddania sprawozdania	13.04.16.

---

Ćwiczenie miało na celu porównanie różnych algorytmów sortowania danych w tablicy: sortowania bąbelkowego, sortowania przez proste wstawianie, sortowania przez proste wybieranie, sortowania Shell'a, sortowania szybkiego i zmodyfikowanych algorytmów sortowania szybkiego - sortowanie szybkie z innym kluczem osiowym, który wybrany został jako mediana 3 elementów tablicy: pierwszego, ostatniego i środkowego oraz sortowanie szybkie z przerwaniem rekurencji dla przedziału elementów tablicy mniejszego od 20 i dalszym sortowaniem przez proste wstawianie.

Ćwiczenie polegało na pomiarze średniego (z 5 pośrednich pomiarów) czasu sortowania przez każdy algorytm dla coraz większego rozmiaru tablicy.

W tym celu w języku C++ zaimplementowano każdy algorytm na podstawie jego pseudokodu oraz środowisko testowe.

Program ustala rozmiar tablicy, wypełnia ją kolejnymi liczbami naturalnymi, później miesza całą tablicę, rozpoczyna sortowanie i wykonuje 5 pomiarów czasu dla danego sortowania. Sprawdza czy tablica została posortowana (jeżeli nie - zwraca na konsoli błąd), a następnie zwraca średnią tych 5 pomiarów i jeżeli średnia ta jest mniejsza niż 30 sekund, program zwiększa rozmiar tablicy dwukrotnie i wykonuje pomiary ponownie.

Zestawienie wyników pomiarowych wygląda następująco:

N	średni czas sortowania, s						
	A	B	C	D	E	F	G
1000	0,0080	0,0002	0,0002	0,0000	0,0000	0,0000	0,0000
2000	0,0030	0,0006	0,0010	0,0000	0,0000	0,0000	0,0000
4000	0,0130	0,0020	0,0040	0,0002	0,0002	0,0002	0,0002
8000	0,0664	0,0094	0,0150	0,0004	0,0002	0,0002	0,0002
16000	0,2982	0,0282	0,0624	0,0006	0,0008	0,0006	0,0008
32000	1,2450	0,1196	0,2366	0,0020	0,0014	0,0014	0,0012
64000	5,3004	0,4732	0,9476	0,0046	0,0046	0,0032	0,0026
128000	21,7332	1,8320	3,7814	0,0100	0,0062	0,0064	0,0054
256000	87,1388	7,5996	15,1922	0,0228	0,0132	0,0136	0,0114
512000		30,5686	60,7592	0,0506	0,0306	0,0286	0,0242
1024000				0,1090	0,0590	0,0604	0,0514
2048000				0,2386	0,1252	0,1256	0,1086
4096000				0,5248	0,2716	0,2640	0,2298
8192000				1,1628	0,5706	0,5492	0,4876
16384000				2,5396	1,1826	1,1384	0,9988
32768000				5,6896	2,4758	2,3796	2,0896
65536000				12,8802	5,1108	4,9352	4,3512
131072000				29,0064	10,6520	10,2652	9,0702
262144000				67,2526	22,0122	21,1804	18,7986
524288000					45,5672	43,8440	39,0498

gdzie:

**N** - Ilość elementów tablicy.

**A** - Sortowanie bąbelkowe.

**B** - Sortowanie przez proste wstawianie.

**C** - Sortowanie przez proste wybieranie.

**D** - Sortowanie Shell'a.

**E** - Sortowanie szybkie podstawowe.

**F** - Sortowanie szybkie z kluczem osiowym jako medianą 3 elementów tablicy.

**G** - Sortowanie szybkie z przerwaniem rekurencji dla mniej niż 20 elementów i dalszym sortowaniem przez proste wstawianie.

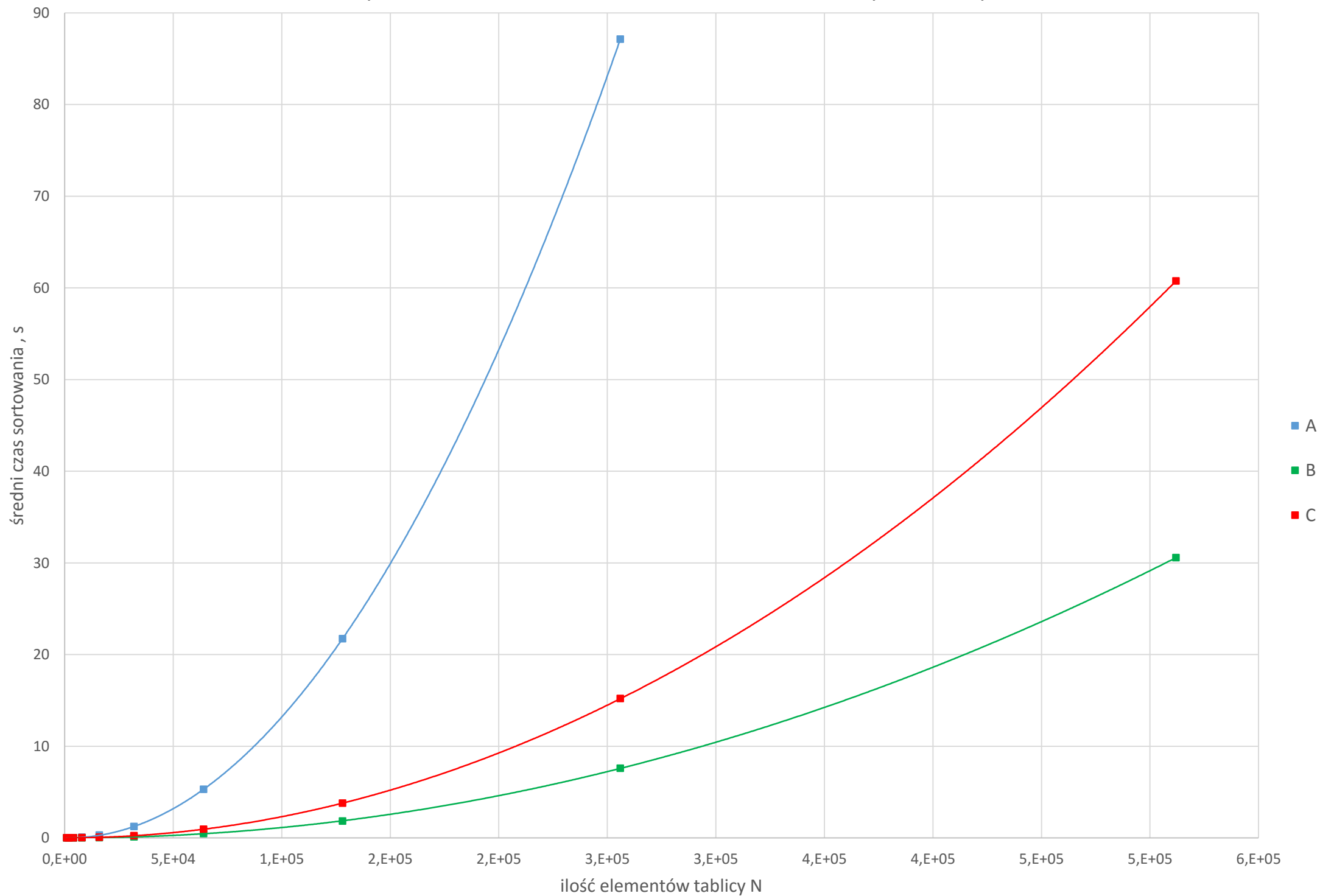
Platforma testowa:

- Procesor intel Core i5 2500k 3.3GHz (OC 4.6 GHz)
- Płyta główna Asrock z77 Pro3
- RAM Corsair 2 x 4 GB DDR3-1600 CL8 (CMZ8GX3M2A1600C8)
- Pamięć masowa SSD Crucial CT250MX200SSD1
- system operacyjny MS Windows 10 Pro

Obciążenie procesora podczas trwania programu oscylowało między 35% a 40%.

Na następnej stronie przedstawione są wykresy zależności czasu trwania sortowania od ilości danych w tablicy:

Wykres zależności czasu trwania sortowania od ilości danych w tablicy



Wykres zależności czasu trwania sortowania od ilości danych w tablicy

