BUBBLE SORT

Złożoność pesymistyczna

```cpp
template<typename T>
int bubble_sort(T *arr, int size)
{
    int no_operations = 0;  //* licznik operacji
    for(int i = 0; i<size-1; i++) {
        for(int j = 0; j<size-i-1; j++){    //* odejmujemy 'i' poniewaz wiemy ze liczba ktora "wyplynela" na koniec jest najwieksza
            if(arr[j] > arr[j+1]){  //* jesli poprzednik jest wiekszy od nastepnika zamien miejscami i zwieksz liczbe operacji
                std::swap(arr[j], arr[j+1]);
                no_operations++;
            }
            no_operations++;    //* zwiekszam ilosc operacji poniewaz w kazdej iteracji porownuje dwa elementy (arr[j] i arr[j+1])
        }
    }
    return no_operations;
}
```

Najgorszym przypadkiem jest odwrotnie posortowana tablica.

Musimy wykonać wtedy $1+2+3+\ldots+(n-1)$ zamian (ta $\sum = \frac{n(n-1)}{2}$)

Bez względu na przypadek musimy wykonać taką samą ilość porównań $= \frac{n(n-1)}{2}$

Łącznie $n(n-1) = n^2 - n \rightarrow O(n^2)$

Wyliczane poprzez 100 testów:

BUBBLE SORT

Ilosc elementow	Srednia ilosc operacji	Odchylenie standardowe
10	66.61	6
20	283.28	15.7162
50	1831.2	60.1914
100	7449.8	169.346
200	29894.3	465.544
500	187267	1870.47

SELECTION SORT

Złożoność pesymistyczna

```cpp
int selection_sort(T *arr, int size)
{
    int no_operations = 0;
    int min_index = 0;
    T min_value = arr[0];

    for(int i = 0; i<size-1; i++){
        for(int j = i; j<size; j++){
            if(arr[j] <= min_value){    //* jesli aktualna minimalna wartosc jest wieksza, zmien ja na mniejsza i zachowaj index
                min_value = arr[j];
                min_index = j;
            }
            no_operations++;    //* zliczenie porownania
        }
        std::swap(arr[i], arr[min_index]);  //* zamieniam najmniejsza wartosc z elementem o kolejnym indeksie
        no_operations++;    //* zliczenie zamiany
        min_value = arr[i+1];
    }
    return no_operations;
}
```

Algorytm będzie wykonywał zawsze taką samą ilość porównań, równą tak jak w Bubble Sorcie $\frac{n(n-1)}{2}$, szukając elementu najmniejszego

Zamian zawsze będzie $(n-1)$

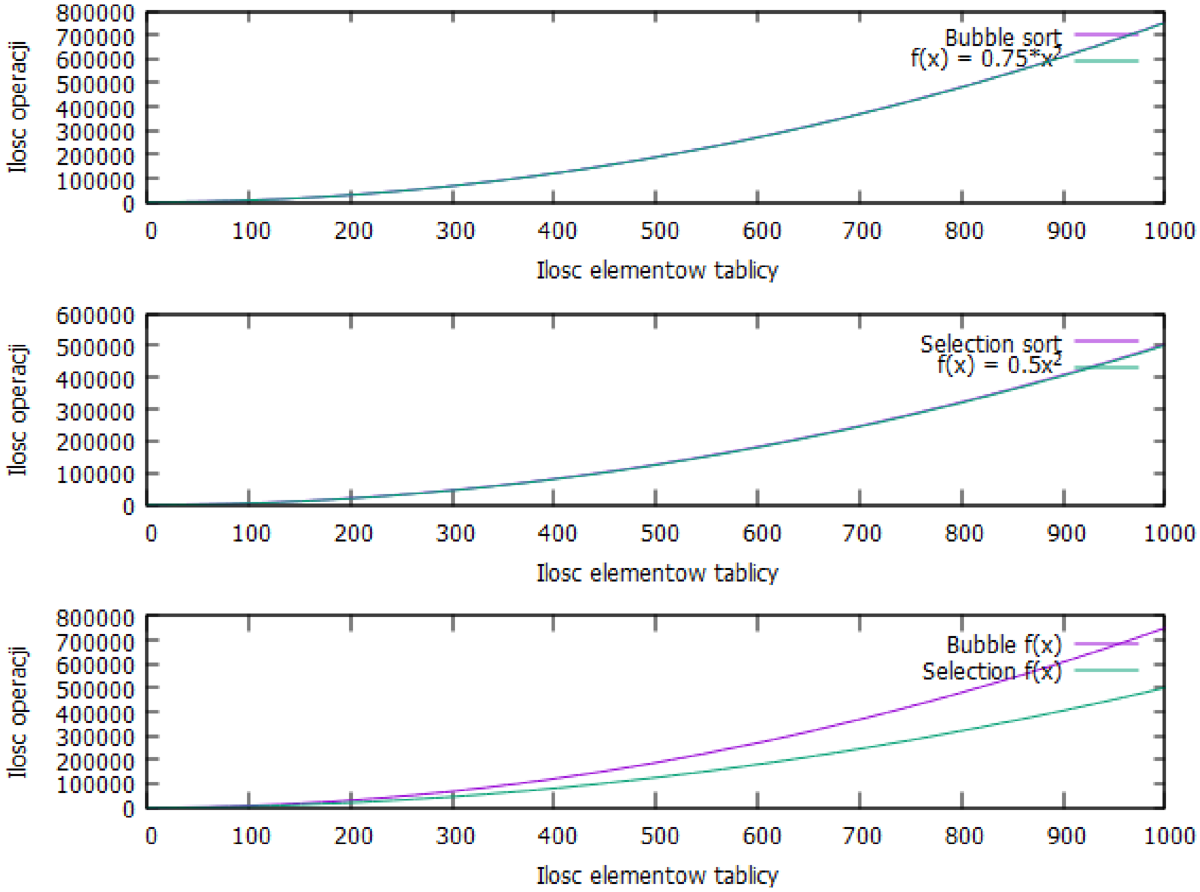
Łącznie: $\frac{n(n-1)}{2} + (n-1) = \frac{n^2}{2} + \frac{n}{2} - 1 \rightarrow O(n^2)$

Wyliczane poprzez 100 testów:

SELECTION SORT

Ilosc elementow	Srednia ilosc operacji	Odchylenie standardowe
10	62	0
20	227	0
50	1322	0
100	5147	0
200	20297	0
500	125747	0

Porównanie sortowań na wykresach



Autor:

Kacper Piątkowski