# Projektowanie Algorytmów i Metody Sztucznej Inteligencji Projekt 1

Mgr inż. Marcin Ochman  
Grupa: Wtorek 1515-1655

1. **Wstęp**Celem projektu jest porównanie różnych sposobów sortowania oraz sprawdzenie jaki wpływ na czas sortowania ma ich złożoność obliczeniowa. Zaimplementowane algorytmy: sortowanie szybkie, sortowanie przez scalanie, sortowanie introspektywne. Złożoność obliczeniowa tych algorytmów to O gdzie dla sortowania szybkiego trzeba uwzględnić pesymistyczny przypadek O. Sortowanymi obiektami były dynamicznie zaalokowane tablice wypełnione pseudolosowo przez funkcję *rand()*.
2. **Sortowanie szybkie**Sortowanie szybkie jest algorytmem rekurencyjnym. Dla tablicy elementów wybiera jeden element zwany osią i tworzy dwie tablice. Jedna z elementami mniejszymi od elementu osiowego i drugą z elementami większymi. Algorytm następnie wywołuje się rekurencyjnie dla utworzonych tablic doprowadzając do ich uporządkowania. Jego złożoność obliczeniowa dla przypadku średniego to O. W sytuacji kiedy jako oś wybrany zostanie element największy lub najmniejszy złożoność obliczeniowa wynosi O.
3. **Sortowanie przez scalanie**Sortowanie przez scalanie jest algorytmem rekurencyjnym. Dzieli tablice na dwie mniejsze tablice do momentu uzyskania tablic jednoelementowych. Kiedy cały zestaw danych jest podzielony na pojedyncze elementy zaczyna scalać i sortować elementy aż do uzyskania całego zestawu danych. Algorytm ma złożoność obliczeniową O.
4. **Sortowanie introspektywne**Sortowanie introspektywne jest hybrydowym algorytmem sortowania. Znaczy to, że składa się on z kilku innych algorytmów sortowania. Główną zaletą tego typu algorytmu jest wyeliminowanie pesymistycznego przypadku sortowania szybkiego. Zaczyna on sortowanie za pomocą sortowania szybkiego. W trakcie działania sortowania badana jest maksymalna głębokość wywołań rekurencyjnych. Jeśli wywołania te osiągną 0 wywołania rekurencyjne kończą się a dla danego podproblemu jest wykorzystywane sortowanie przez kopcowanie. Jeśli uzyskano w trakcie sortowania tablice o rozmiarze mniejszym od 10 to sortowane są one przy pomocy sortowania przez wstawianie który pomimo swojej złożoności obliczeniowej radzi sobie lepiej z małymi zestawami danych.
5. **Przebieg eksperymentów**