

## Lista zadań nr 6: *Sortowanie (algorytmy zaawansowane)*

### **Zadanie 1.**

Zaimplementuj każdy w omawianych na wykładzie *zaawansowany algorytm sortowania* (np. rozbudowując pakiet realizujący zadanie 1. z listy nr 5) i zbadaj (na stosownych, samodzielnie zdefiniowanych zbiorach danych testowych) liczbę porównań i przesunięć (przepisań) wykonanych w każdym z algorytmów. Sformułuj stosowne wnioski.

### **Zadanie 2.**

Rozbuduj zadanie 2. z listy nr 5, implementując:

1. *bezpośrednio*,
2. z użyciem ogólnych mechanizmów

każdy z omawianych na wykładzie, *zaawansowany algorytm sortowania*, na zbiorze obiektów (zapropionowanych przez siebie) tak, by możliwe było uzyskiwanie kilku alternatywnych porządków, wyznaczanych przez wartości różnych atrybutów (np. w programie wielofunkcyjnym, z określonym *menu* funkcji). Traktując to zadanie jako przykład *rozbudowy* istniejącego już oprogramowania sformułuj stosowne wnioski dotyczące sposobu projektowania tego typu oprogramowania.

### **Zadanie 3. – dla hobbystów**

Zaimplementuj algorytm ShellSort z automatycznym doбором ciągu przyrostów (uzależnionym od rozmiaru danych) według dwóch różnych strategii (jedną z nich może być strategia omawiana na wykładzie, druga powinna być własną propozycją) i przeprowadź (w stosownej klasie testowej) badanie liczby operacji porównań i przesunięć (przepisań) dla każdej z tych strategii. Porównaj otrzymane wyniki.