

Zadania z programowania w języku Java dla II roku Informatyki.

dr Agnieszka Zbrzezny

Typy uogólnione

1. Dodaj do klasy **Pair** z przykładowych programów z wykładu niestaticzną generyczną metodę **swap**, która zamienia ze sobą pierwszy i drugi element pary. Przetestuj tę metodę w funkcji **main** klasy **PairDemo**.
2. Korzystając z klasy **Pair** z poprzedniego zadania zdefiniuj klasę **PairUtil** z jedną staticzną generyczną metodą **PairUtil.swap**, której jedynym argumentem jest obiekt klasy **Pair** i która zwraca nowo utworzoną parę z zamienionymi elementami. Przetestuj tę metodę w funkcji **main** klasy **PairUtilDemo**.
3. Napisz staticzną generyczną metodę **ArrayUtil.isSorted**, która sprawdza czy podana jako argument tablica obiektów klasy **T** jest uporządkowana niemalejąco. W definicji metody zadeklaruj, że klasa **T** musi implementować generyczny interfejs **Comparable**. W funkcji **main** przetestuj działanie metody **isSorted** dla posortowanych i nieposortowanych tablic obiektów klas **Integer** oraz **localDate**.
4. Zaimplementuj metodę uogólnioną, która dopisuje wszystkie elementy z jednej tablicy typu **ArrayList** do innej. Użyj symboli wieloznacznych dla jednego z typów. Utwórz dwa równoważne rozwiązania, jedno z symbolem wieloznacznym **<? extends E>**, a drugie z **<? super E>**.
5. Napisz generyczną wersję algorytmu wyszukiwania binarnego w postaci staticznej generycznej metody **ArrayUtil.binSearch**, której argumentem są tablica elementów klasy **T** oraz szukany obiekt klasy **T** i która zwraca liczbę typu **int** (indeks elementu, gdy znaleziono szukany obiekt, natomiast **-1** w przeciwnym przypadku). W definicji metody zadeklaruj, że klasa **T** musi implementować generyczny interfejs **Comparable**. W funkcji **main** przetestuj działanie metody dla tablic obiektów klas **Integer** oraz **localDate**.
6. Napisz generyczną wersję algorytmu sortowania przez wybieranie w postaci staticznej generycznej metody **ArrayUtil.selectionSort**, której jedynym argumentem jest tablica elementów klasy **T**. W definicji metody zadeklaruj, że klasa **T** musi implementować generyczny interfejs **Comparable**. W funkcji **main** przetestuj działanie metody dla tablic obiektów klas **Integer** oraz **localDate**.
7. Napisz generyczną wersję algorytmu sortowania przez scalanie w postaci staticznej generycznej metody **ArrayUtil.mergeSort**, której jedynym argumentem jest tablica elementów klasy **T**. W definicji metody zadeklaruj, że klasa **T** musi implementować generyczny interfejs **Comparable**. W funkcji **main** przetestuj działanie metody dla tablic obiektów klas **Integer** oraz **localDate**.