

Zadania z programowania w języku Java dla II roku Informatyki

dr Agnieszka Zbrzezny

Klasa **ArrayList**

Metody przy których opisie jest powiedziane, że tworzą nową listę, nie powinny modyfikować tablic zadanych jako argumenty.

1. Napisz publiczną i statyczną metodę

```
ArrayList<Integer> append(ArrayList<Integer> a, ArrayList<Integer> b),
```

która tworzy nową listę tablicową poprzez dołączenie drugiej listy tablicowej do pierwszej. Przykładowo, jeżeli lista **a** składa się z elementów 1 4 9 16, natomiast lista **b** z elementów 9 7 4 9 11, to wywołanie `append(a, b)` powinno zwrócić listę tablicową złożoną z elementów 1 4 9 16 9 7 4 9 11. Napisz funkcję `main`, która sprawdzi czy metoda `append` działa poprawnie.

2. Napisz publiczną i statyczną metodę

```
ArrayList<Integer> merge(ArrayList<Integer> a, ArrayList<Integer> b),
```

która tworzy nową listę tablicową poprzez pobieranie elementów na przemian z obu list tablicowych. Jeżeli jedna lista tablicowa jest krótsza od drugiej, to po wyczerpaniu elementów z krótszej listy należy dołączyć pozostałe elementy z dłuższej listy. Przykładowo, jeżeli lista **a** składa się z elementów 1 4 9 16, natomiast lista **b** z elementów 9 7 4 9 11, to wywołanie `merge(a, b)` powinno zwrócić listę tablicową złożoną z elementów 1 9 4 7 9 4 16 9 11. Napisz funkcję `main`, która sprawdzi czy metoda `merge` działa poprawnie.

3. Napisz publiczną i statyczną metodę

```
ArrayList<Integer> mergeSorted(ArrayList<Integer> a, ArrayList<Integer> b),
```

która scala dwie posortowane listy tablicowe tworząc nową posortowaną listę tablicową. Dla każdej z list należy używać zmiennej indeksowej wskazującej kolejny element do porównania. Metoda pobiera do nowo tworzonej listy mniejszy ze wskazywanych elementów. Po wyczerpaniu elementów z którejś z list należy dołączyć pozostałe elementy z niepustej listy. Przykładowo, jeżeli lista **a** składa się z elementów 1 4 9 16, natomiast lista **b** z elementów 4 7 9 9 11, to wywołanie `merge(a, b)` powinno zwrócić listę tablicową złożoną z elementów 1 4 4 7 9 9 9 11 16. Napisz funkcję `main`, która sprawdzi czy metoda `mergeSorted` działa poprawnie.

4. Napisz publiczną i statyczną metodę

```
ArrayList<Integer> reversed(ArrayList<Integer> a),
```

która tworzy nową listę tablicową poprzez odwrócenie kolejności elementów listy **a**. Nie korzystaj z metod klasy `Collections`. Napisz funkcję `main`, która sprawdzi czy metoda `reversed` działa poprawnie.

5. Napisz publiczną i statyczną metodę

```
void reverse(ArrayList<Integer> a),
```

która odwraca kolejność elementów na liście tablicowej **a**. Nie korzystaj z metod klasy `Collections`. Napisz funkcję `main`, która sprawdzi czy metoda `reverse` działa poprawnie.