

Zadanie: lambda-wyrażenia - kompozycja funkcji

Zbudować klasę InputConverter, która pozwala przekształcić dane wejściowe (ustalane w konstruktorze klasy) za pomocą funkcji, podanych jako argumenty metody convertBy.

Np. jeśli mamy zdefiniowane operacje (funkcje):

- flines - zwraca listę wierszy z pliku tekstowego
- join - łączy napisy z listy (zwraca napis połączonych ze sobą elementów listy napisów)
- collectInts - zwraca listę liczb całkowitych zawartych w napisie
- sum - zwraca sumę elementów listy liczb całkowitych

to sumę liczb całkowitych, występujących w pliku o nazwie *fname* możemy uzyskać poprzez:

```
Integer s = new InputConverter<String>(fname).convertBy(flines, join, collectInts, sum);
```

a listę liczb całkowitych z napisu txt tak:

```
List<Integer> n = new InputConverter<String>(txt).convertBy(collectInts);
```

Istotą metody convertBy jest to, że pozwala ona w różny sposób kombinować różne operacje na różnych danych i łatwo uzyskiwać wyniki.

Zapewnić prawidłowe działanie następującego fragmentu kodu (dostarczonego w szablonie projektu, pliku Main.java nie wolno modyfikować):

```
public static void main(String[] args) {
    /*<--
    *   definicja operacji w postaci lambda-wyrażeń:
    *   - flines - zwraca listę wierszy z pliku tekstowego
    *   - join - łączy napisy z listy (zwraca napis połączonych ze sobą elementów listy napisów)
    *   - collectInts - zwraca listę liczb całkowitych zawartych w napisie
    *   - sum - zwraca sumę elementów listy liczb całkowitych
    */

    String fname = System.getProperty("user.home") + "/LamComFile.txt";
    InputConverter<String> fileConv = new InputConverter<>(fname);
    List<String> lines = fileConv.convertBy(flines);
    String text = fileConv.convertBy(flines, join);
    List<Integer> ints = fileConv.convertBy(flines, join, collectInts);
    Integer sumints = fileConv.convertBy(flines, join, collectInts, sum);

    System.out.println(lines);
    System.out.println(text);
    System.out.println(ints);
    System.out.println(sumints);

    List<String> arglist = Arrays.asList(args);
    InputConverter<List<String>> slistConv = new InputConverter<>(arglist);
    sumints = slistConv.convertBy(join, collectInts, sum);
    System.out.println(sumints);

}
```

Dla następujących danych z pliku:

Cars:
- Fiat: 15, Ford: 20
- Opel: 8, Mitsubishi: 10

oraz następujących argumentów wywołania metody main:

Warszawa 100 Kielce 200 Szczecin 300

program powinien wyprowadzić na konsolę:

```
[Cars:, - Fiat: 15, Ford: 20, - Opel: 8, Mitsubishi: 10]
Cars:- Fiat: 15, Ford: 20- Opel: 8, Mitsubishi: 10
[15, 20, 8, 10]
53
600
```

Uwaga: w klasie InputConverter musi występować tylko jedna metoda o nazwie convertBy (czyli w tym zadaniu nie dopuszczamy przeciążania metody convertBy).