

## Testat 8 – 1

- Achte darauf, dass die vorgegebene Signatur in Deiner Lösung exakt eingehalten wird.
  - Vervollständige die Datei `LambdaMethods.java` ohne eine `package`-Angabe.
  - **Die abgegebene Datei muss die geforderten drei Methoden enthalten und compilierbar sein. Ergänze daher bei den Methoden `makeInt` und `makePairs` auf jeden Fall eine geeignete `return`-Anweisung.**
  - Gib nur die Datei mit dem Namen `LambdaMethods.java` im Moodle-Bereich ab.
  - Die Bearbeitungsdauer beträgt 60 Minuten.
  - Die Bearbeitungszeit beginnt mit der Ausgabe des Testats an Dich.
- 

Bei diesem Testat erhältst Du drei Klassen und zwei Interfaces:

- Die Klasse `IntIntPairs` (Erläuterung s.u.) und die Interfaces `BoolFunction` und `IntFunction` sind bereits fertig implementiert und sollen nicht geändert werden.
- Die Klasse `LambdaMethods` soll passend zur Aufgabenstellung vervollständigt werden.
- Die Klasse `Testumgebung` stellt eine einfache Testumgebung bereit, die von Dir beliebig geändert werden darf.

Vervollständige in der Klasse `LambdaMethods` die drei vorgegebenen Methoden `makeInt`, `makePairs` und `apply` mit folgender Funktionalität:

- Die Methode `makeInt` soll für das als Argument übergebene `IntIntPairs`-Objekt die Anzahl aller gültigen Werte in `keys` bestimmen und zurückgeben, die gleich `0` sind.
- Die Methode `makePairs` soll ein `IntIntPairs`-Objekt zurückgeben, das Kopien der gültigen Paare des als Argument übergebenen `IntIntPairs`-Objekts enthält, deren Wert in `keys` größer als `0` ist.
- Die Methode `apply` soll bei allen gültigen Paaren des als Argument übergebenen `IntIntPairs`-Objekts die Werte in `keys` mit `5` multiplizieren und die Werte in `values` beibehalten.
- Die Lösung muss durch die Nutzung der von der Klasse `IntIntPairs` angebotenen Methoden erfolgen, denen geeignete Lambda-Ausdrücke als Parameter übergeben werden sollen. An keiner Stelle der Lösung dürfen neue Klassen vereinbart werden.

*Beispiel:*

Die Testumgebung sollte folgende Ausgabe erzeugen:

Paare für Test:

(7,3) (4,3) (1,-9) (6,6) (41,12) (9,-11) (8,12) (5,-60) (14,18) (0,12) (-9,-16) (0,0) (17,32) (1,1)

Test für `makeInt` - Ergebniswert ist:

2

Test für `makePairs` - Ergebnispaaire sind:

(7,3) (4,3) (1,-9) (6,6) (41,12) (9,-11) (8,12) (5,-60) (14,18) (17,32) (1,1)

Test für `apply` - geänderte Paare:

(35,3) (20,3) (5,-9) (30,6) (205,12) (45,-11) (40,12) (25,-60) (70,18) (0,12) (-45,-16) (0,0) (85,32) (5,1)

*Hinweis:*

Die Grundlage für dieses Testat bildet die vom Praktikumsblatt 7 bekannte Klasse `IntIntPairs`, die eine vorgegebene maximale Zahl von Paaren von `int`-Werten verwaltet. Zwei zusammengehörende `int`-Werte werden unter jeweils dem gleichen Index in den beiden Feldern `keys` und `values` abgelegt. Die abgelegten Paare sind nur dann gültig, wenn zugleich in einem dritten `boolean`-Feld `valids` der Wert `true` für diesen Index eingetragen ist. Das Löschen von Paaren in der Klasse `IntIntPairs` erfolgt durch Setzen des entsprechenden Wertes von `valids` auf `false`.