

Der bestehende Iterator Ihres **ADS_set** ist um einen zusätzlichen „Modus“ zu erweitern. In der bisherigen Implementierung liefert der Iterator alle Elemente in einer beliebigen Reihenfolge, wobei die Reihenfolge immer dieselbe sein muss, solange das **ADS_set** nicht geändert wird (Modus „normal“). Im neuen Modus „speziell“ sollen die Elemente in derselben Reihenfolge geliefert werden, aber es werden nach dem ersten Element alle Elemente mit Ausnahme des letzten Elementes übersprungen. In beiden Modi erreicht der Iterator nach dem letzten Element **end()**. Anders ausgedrückt: Der von **z()** erzeugte Iterator liefert für ein nicht-leeres **ADS_set** immer genau zwei Elemente, nämlich das erste und das letzte (bezogen auf die normale Iteratorreihenfolge), und dann **end()**.

Details: Erweitern Sie Ihre Implementierung **ADS_set** um die Methode

```
const_iterator z() const;
```

Diese soll einen Iterator im Modus „speziell“ liefern. Wenn kein Element im **ADS_set** vorhanden ist, dann gilt **z() == end()**. Andernfalls liefert **z()** einen Iterator, der auf das 1. Element zeigt. Der von **z()** gelieferte Iterator soll nach dem ersten Inkrementieren auf das letzte Element (bezogen auf die Iteratorreihenfolge der Elemente) zeigen, nach dem zweiten Inkrementieren wird **end()** geliefert. Achtung: wenn das **ADS_set** genau ein Element enthält, dann ist dieses Element das erste Element und zugleich das letzte Element und wird daher vom Iterator zwei Mal geliefert.

Die Zeitkomplexität und Speicherkomplexität der Operatorfunktionen müssen unverändert bleiben. So sind zB zusätzliche Felder mit nicht konstanter Größe unzulässig.

Beispiele:

Angenommen der von begin() retournierte Iterator liefert alle gespeicherten Elemente in der Reihenfolge	dann zeigt der von z() retournierte Iterator auf das Element	und wenn dieser Iterator bis end() inkrementiert wird, liefert er die folgenden Elemente in der Reihenfolge
(4,7,1,5,3,6,0,8,10,2,9)	4	(4, 9)
(7,8,9,4,1,5,10)	7	(7, 10)
(7,8,9,4,1,5)	7	(7, 5)
(7,8,9)	7	(7, 9)
(7,9)	7	(7, 9)
(9)	9	(9, 9)
()	end()	()

Anleitung: Schreiben Sie **keine** neue Iteratorklasse!

Erweitern Sie die bestehende Iterator-Klasse wie folgt (dies ist nur einer der möglichen Lösungsansätze, abweichende korrekte Lösungen sind natürlich zulässig):

- Es muss ein Iterator im Modus „speziell“ erzeugt werden können. Dazu ist ein neuer Konstruktor zu schreiben und/oder bestehende zu erweitern, um die Instanzvariablen entsprechend zu initialisieren.
- Passen Sie die Inkrement-Operationen so an, dass beim ersten Inkrementieren entsprechend viele Werte übersprungen werden, wenn der Iterator im Modus „speziell“ ist.

Die Methode **ADS_set::z()** erzeugt einen Iterator im Modus „speziell“ und retourniert diesen.

Die Methode **ADS_set::begin()** liefert wie bisher einen Iterator im Modus „normal“.