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 Werte in derselben Reihenfolge geliefert werden, aber es wird nach jedem gelieferten Wert eine bestimmte Anzahl von Werten übersprungen. Die Anzahle der übersprungenen Werte beträgt 1 nach dem 1. Wert, 2 nach dem 2. Wert und ganz allgemein *n* nach dem *n*. Wert. In beiden Modi erreicht der Iterator **end()**, sobald keine Elemente mehr vorhanden sind.

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 beim Inkrementieren in weiterer Folge jeweils entsprechend viele Werte überspringen und auf den darauffolgenden Wert zeigen, falls ein solcher vorhanden ist, andernfalls wird **end()** 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 Schlüsselwerte in der Reihenfolge	dann zeigt der von **z()** retournierte Iterator auf den Wert	und wenn dieser Iterator bis **end()** inkrementiert wird, liefert er die folgenden Schlüsselwerte in der Reihenfolge
(**4**,7,**1**,5,3,**6**,0,8,10,**2**,9)	4	(4, 1, 6, 2)
(**7**,8,**9**,4,1,**5**,10)	7	(7, 9, 5)
(**7**,8,**9**,4,1,**5**)	7	(7, 9, 5)
(**7**,8,**9**)	7	(7, 9)
(**7**,9)	7	(7)
(**9**)	9	(9)
()	end()	()

Anleitung: Schreiben Sie **keine** neue Iteratorklasse! Erweitern Sie die bestehende Iterator-Klasse wie folgt:

- Es ist zu speichern, ob der Iterator im Modus „normal“ oder „speziell“ ist und wie viele Werte im nächsten Schritt zu überspringen sind.
- 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 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“.