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

```
key_type x() const;
```

Diese soll das kleinste lokale Maximum in der durch einen Iterator gelieferten Reihenfolge der Schlüsselwerte liefern. Wenn es kein lokales Maximum gibt, soll eine Exception vom Typ **`std::runtime_error`** ausgelöst werden.

Zur Vereinfachung wird für diese Aufgabe festgelegt: Ein lokales Maximum ist ein Wert, der größer ist als sein Vorgänger und größer als sein Nachfolger. Ein lokales Maximum muss daher einen Vorgänger und einen Nachfolger haben.

Für den Vergleich von Werten ist **`std::less`** (bzw. der alias **`key_compare`**, sofern vorhanden) zu verwenden. Der Aufruf **`std::less<T>{}(a,b)`** für die beiden Werte **a** und **b** vom Typ **T** liefert **`true`**, falls **b** größer als **a** ist, und **`false`** sonst. Aufruf von anderen Methoden oder Funktionen, insbesondere die Verwendung von Iteratoren (und damit z. B. auch die Verwendung einer range based for loop), ist nicht erlaubt.

Die Zeitkomplexität der Funktion **x** muss O(n) sein (n ist dabei die Anzahl der Elemente im Set), die Speicherkomplexität O(1). Es ist beispielsweise nicht erlaubt, zusätzliche Felder mit einer nicht konstanten Größe zu verwenden.

Beispiele:

| Angenommen der Iterator liefert alle gespeicherten Schlüsselwerte in der Reihenfolge | dann liefert der Aufruf von x() | denn die lokalen Maxima sind die Werte |
|---|---|---|
| (4,7,1,5,3,6,0,8,10,2,9) | 5 | 7,5,6,10 |
| (9,1,3,5,6,7) | std::runtime_error | |
| (7,8,9,4,1,5,10) | 9 | 9 |
| (7,8,9) | std::runtime_error | |
| (7,9,8) | 9 | 9 |
| (9,1) | std::runtime_error | |
| () | std::runtime_error | |

Tipp: Eine einfache Methode ist, die Werte in Iteratorreihenfolge zu durchlaufen (allerdings, ohne Iteratoren zu verwenden) und dabei eine Minimumsuche durchzuführen, wobei allerdings nur die lokalen Maxima berücksichtigt und alle anderen Werte ignoriert werden.

Für die Erzeugung von Objekten vom Typ **`std::runtime_error`** ist gegebenenfalls zusätzlich **`<stdexcept>`** zu inkludieren.