

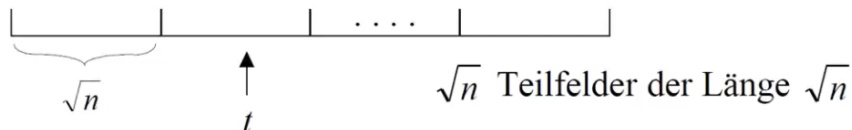
Verfahren

- worst-case der [[Interpolationssuche]] verhindern durch Interpolationssuche
 - auf \sqrt{n} Teilfelder
 - Länge \sqrt{n}

- Berechne Index t durch Interpolationssuche:

$$t = \text{von} + \left\lfloor (bis - \text{von}) \cdot \frac{x - A[\text{von}]}{A[bis] - A[\text{von}]} \right\rfloor$$

- Suche von t aus das korrekte Teilfeld (in Sprüngen von \sqrt{n}) und suche von dort aus weiter:



Identifizieren des korrekten Teilfeldes in $O(1)$ erwarteter Zeit

```

QUADSUCH(von,bis,x)
1: n ← bis - von + 1
2: IF A[von] < A[bis] THEN
3:   t ← von + ⌊ (bis - von) · (x - A[von]) / (A[bis] - A[von]) ⌋
4:   IF x = A[t] THEN RETURN t
5:   ELSE IF x < A[t] THEN
6:     REPEAT t ← t - ⌊√n⌋ UNTIL x ≥ A[t]
7:     RETURN QUADSUCH(t, t + ⌊√n⌋ - 1, x)
8:   ELSE
9:     REPEAT t ← t + ⌊√n⌋ UNTIL x ≤ A[t]
10:    RETURN QUADSUCH(t - ⌊√n⌋ + 1, t, x)
11: ELSE IF x = A[von] THEN RETURN von
12: ELSE RETURN -1
  
```

rekursive Version

Aufruf:
QUADSUCH(1,n,x)

Erwartete Laufzeit:
(Teilfeld in $O(1)$ Zeit)

$T(n) = O(\log \log n)$

Worst-case:
(alle Teilfelder probiert)
 $T(n) = O(\sqrt{n})$