

Binäre Suche

Gegeben: Ein sortiertes Array **a** mit **n** Komponenten **a[0], . . . , a[n-1]**

Aufgabe: Entscheide, ob ein zur Laufzeit für die Variable **value** eingegebener Wert im Array **a** vorkommt.

Beispiel

value = 13

n = len(a) = 10

Suche **value** in der Liste **a[0], . . . , a[9]**; diese Liste und **value** übergeben wir der Booleschen Funktion **binarysearch**, welche **a[0], . . . , a[9]** als lokale Liste **array[0], . . . , array[9]** verarbeitet.

a[0]	a[1]	a[2]	a[3]	a[4]	a[5]	a[6]	a[7]	a[8]	a[9]
3	4	5	5	7	8	11	13	19	21

array[0]	array [1]	array [2]	array [3]	array [4]	array [5]	array [6]	array [7]	array [8]	array [9]
3	4	5	5	7	8	11	13	19	21

1. Schritt:

Wir bestimmen den mittleren Index des Arrays array: $\text{len(array)}//2 = 5$

2. Schritt:

midvalue = array[len(array)//2] = array[10//2] = array[5] = 8

Wir vergleichen value mit midvalue:

Falls value == midvalue: binarysearch gibt den Wert True zurück; gefunden!

Falls value < midvalue: suche in der Liste a[0], . . . , a[4] links von a[5]

Falls value > midvalue: suche in der Liste a[6], . . . , a[9] rechts von a[5]

hier: wegen $13 > 8$ suchen wir in der Liste a[6], . . . , a[9]

Suche **value** in der Liste **a[6], . . . , a[9]**

a[6]	a[7]	a[8]	a[9]
11	13	19	21

Diese Liste **a[6], . . . , a[9]** und **value** übergeben wir der Booleschen Funktion **binarysearch**, welche **a[6], . . . , a[9]** als lokale Liste **array[0], . . . , array[3]** verarbeitet.

array[0]	array[1]	array[2]	array[3]
11	13	19	21

1. Schritt:

Wir bestimmen den mittleren Index des Arrays array: $\text{len(array)}//2 = 4//2 = 2$

2. Schritt:

$\text{midvalue} = \text{array}[\text{len(array)}//2] = \text{array}[4//2] = \text{array}[2] = 19$

Wir vergleichen value mit midvalue:

Falls value == midvalue: binarysearch gibt den Wert True zurück; gefunden!

Falls value < midvalue: suche in der Liste array[0], . . . , array[1] links von array[2]

Falls value > midvalue: suche in der Liste array[3] rechts von array[2]

hier: wegen $13 < 19$ suchen wir in der Liste array[0], . . . , array[1]

Suche **value** in der Liste **array[0], . . . , array[1]**

array[0]	array[1]
11	13

Diese Liste **array[0], . . . , array[1]** und **value** übergeben wir der Booleschen Funktion **binarysearch**, welche **array[0], . . . , array[1]** als lokale Liste **array[0], . . . , array[1]** verarbeitet.

1. Schritt:

Wir bestimmen den mittleren Index des Arrays array: $\text{len(array)}//2 = 2//2 = 1$

2. Schritt:

midvalue = array[len(array)//2] = array[2//2] = array[1] = 13

Wir vergleichen value mit midvalue:

Falls value == midvalue: binarysearch gibt den Wert True zurück; gefunden!

Falls value < midvalue: suche in der Liste array[0] links von array[1]

Falls value > midvalue: suche in der leeren Liste [] rechts von array[1], dann: binarysearch gibt den Wert False zurück; nicht gefunden!

hier: wegen **13 = value = midvalue = 13**: gefunden!