

Gruppe

2/3

ADS

Übungsblatt 3

1131971 René Schmitt

1131998 Kevin Schorpp

Aufgabe 1

1. Finde einen kleineren Fall

wenn zu suchendes Element größer ist als die Mitte der Folge

Untergrenze = Mitte + 1

wenn zu suchendes Element kleiner ist als die Mitte der Folge

Obergrenze = Mitte - 1

2. Drücke die ursprüngliche mit der kleineren Aufgabe aus

binarySearch (F, k, u, m-1)
oder binarySearch (F, k, m+1, o)

3. Finde den trivialen Fall

Mitte ist gleich dem zu suchenden Element

4. Bestimme Fallunterscheidung

wenn Mitte gleich dem zu suchenden Element

return Mitte;

wenn Obergrenze kleiner ist als Untergrenze

return -1

5. Schreibe Funktion

binarySearch (F, k, u, o) → p

Eingabe : Folge F der Länge n, Suchschlüssel k, Untergrenze u, Obergrenze o

Ausgabe : Position p des ersten Elements aus F, das gleich k ist, sonst NO_KEY

$m = (u + o) / 2$

if F [m] = k **then return** m

else if o < u **then return** NO_KEY

else if k < F [m] **then**

return binarySearch (F, k, u, m-1)

else

return binarySearch (F, k, m+1, o)

fi

Aufgabe 2

Durchschnittswerte von insgesamt 100.000 Suchen.

Erfolgreiche Suche

Suchverfahren	Array Länge		
	1024	2048	4096
Linear (letzter Treffer)	1024	2048	4096
Linear (erster Treffer)	421	815	1681
Binär rekursiv	8	9	10
Binär iterativ	8	9	10

Erfolglose Suche

	1024	2048	4096
Linear (letzter Treffer)	1024	2048	4096
Linear (erster Treffer)	1024	2048	4096
Binär rekursiv	10	11	12
Binär iterativ	10	11	12

Lineare Suche

Die lineare Suche (letzter Treffer) muss immer den gesamten Array durchsuchen, da es immer noch einen weiteren Treffer geben könnte. Daher hat sie bei erfolgreicher Suche den gleichen Aufwand wie bei erfolgloser Suche.

Die lineare Suche (erster Treffer) endet bei erfolgreicher Suche mit dem ersten Treffer, muss aber bei erfolgloser Suche ebenfalls den gesamten Array durchsuchen.

Binäre Suche

Die binäre Suche hat den Aufwand $\log_2 n$. In unserem Test wurde die Zahl bei der erfolgreichen Suche allerdings durchschnittlich früher gefunden.

Bei erfolgloser Suche haben diese Algorithmen immer den Aufwand $\log_2 n$

$$\log_2 1024 = 10; \log_2 2048 = 11, \log_2 4096 = 12$$