Aufgabenblatt 01

1 (a)

Auf einer sortierten Liste ist im Durchschnitt die Binäre Suche am schnellsten. Die Aufwandsanalyse ergibt einen durchschnittlichen Aufwand von etwa log2 n und somit in den meisten Fällen der sequenziellen Suche (durchschnittlicher Aufwand (n + 1) / 2) überlegen.

(b)

Die binäre Suche ist nicht bei unsortierten Listen möglich, da man bei der Teilung nicht weiß, in welcher Hälfte sich der gesuchte Wert befindet. Bei der sequentiellen Suche stellt dies kein Problem da.

©

10 11 12 14 15 16 17

Gesucht: 10

Die sequenzielle Suche findet die 10 im ersten Schritt. Die binäre Suche im dritten Schritt.

10 11 12 13 14 15 16 17 ---- 10 11 12 13 ----- 10

(d)

4. und 9.

1 2 3 4 5 6 7 8 9 --------- 1 2 3 4 ----------- 3 4 ------- 4

1 2 3 4 5 6 7 8 9 ------ 6 7 8 9 --------- 8 9 --------- 9

2(a)

1. F[0]= gelb False
2. F[1]=lila False
3. F[2]=blau False
4. F[3]=rot False
5. F[4]=grün False
6. F[5]=grau False
7. F[6]=weiß True
8. F[0]= A False
9. F[1]= C False
10. F[2]= D False
11. F[3]= E False
12. F[4]= G False
13. F[5]= H True

(b)

Die binäre Suche ist auf die erste Liste nicht anwendbar, da sie nicht sortiert ist.

1. M = 3

F[3] = E False

1. M = 5

F[5] = H True

3 (a)

Auswahlskriterium:

Selection Sort wählt das kleinste (oder größte) Element aus der unsortierten Teilmenge aus und platziert es an die richtige Stelle in der sortierten Teilmenge.

Insertion Sort wählt das aktuelle Element aus der unsortierten Teilmenge aus und fügt es an die richtige Position in der bereits sortierten Teilmenge ein.

Arbeitsweise:

Selection Sort arbeitet durch wiederholtes Finden des minimalen (oder maximalen) Elements in der unsortierten Teilmenge und Austauschen dieses Elements mit dem ersten Element der unsortierten Teilmenge.

Insertion Sort arbeitet durch Einfügen jedes Elements der unsortierten Teilmenge an die richtige Position in der bereits sortierten Teilmenge, indem es schrittweise die Elemente nach oben verschiebt, um Platz für das neue Element zu machen.

(b)

Man sollte eher InsertionSort benutzen da die Aufwandsanalyse zeigt, dass dieser effizienter arbeitet. Der Aufwand von SelectionSort ist konstant etwa (n\*n)/2 wohingegen dies bei InsertationSort nur im worst case so ist und dieser sonst weniger Aufwand erfordert

©

(d)

4

Ein Bild, das Text, Screenshot, Display, Software enthält.

Automatisch generierte Beschreibung

(b)

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung