Datenstrukturen & Algorithmen Gruppeübung Gruppe 10

Ciheng Zhang (3472321) zch3183505@gmail.com
Yao He (3487882) st168323@stud.uni-stuttgart.de
Yuchan Bian (3496226) st170182@stud.uni-stuttgart.de

May 8, 2020

Ciheng Zhang(3472321)
Yao He(3487882)
YuchanBian(3496226)

1 Aufgabe 1

Sequenzielle Suche für a:

13	23	47	48	68	12	69	73	81	83	84	85	87	90	94

Das suchen Program sucht von die erste Element in die Array, Wenn diese Element it gleich wie die gesuchtes Element, die Program return die element und die Index von diese Element.Wie die Bild die rote Element ist gesucht, Gray Element ist nicht gesucht. D.h. die Program hat 6 mal Loop und dann bekommt mann die result.
Sequenzielle Suche für b:

12	13	23	47	48	68	69	73	81	83	84	85	87	90	94

Änhlich Wie die letzen Methode. Durch Sequenzielle Suche braucht mann 6 mal Loop, um die Zahl 68 zu suchen

Binäre Suche für a:
Mann kann Binäre suchen für Aufgabe a nicht benutzen. Denn die Folgen ist nicht sortiert. Wenn mann möchte Binäre Suche für ein Folgen benutzen, muss die Folgen gesortirt wird.
Binäre Suche für b:

12	13	23	47	48	68	69	73	81	83	84	85	87	90	94
u							m							o

zuerst man bekommt die Middle Zahl in die ganze Folgen und zeichen es m und bekommt mann die wert. dann 68<73. Man macht ein neue Folgen von u bis m

12	13	23	47	48	68	69	73
u			m				o

Wieder Machen die letzen Methode. Zahl m ist 47. gesucht Zahl 68 ist großer als 48. dann Wieder ein neue Folgen von m bis o Machen.

47	48	68	69	73
u		m		o

Dann bekommt man die m Wert ist 68 gleich wie gesucht Element.Und man bekommt die gesucht Wert mit 3 mal Loop. Es ist schneller als die Sequenzielle Suche.Aus diese Grund benutzen mann die Binäre Suche für Aufgabe b

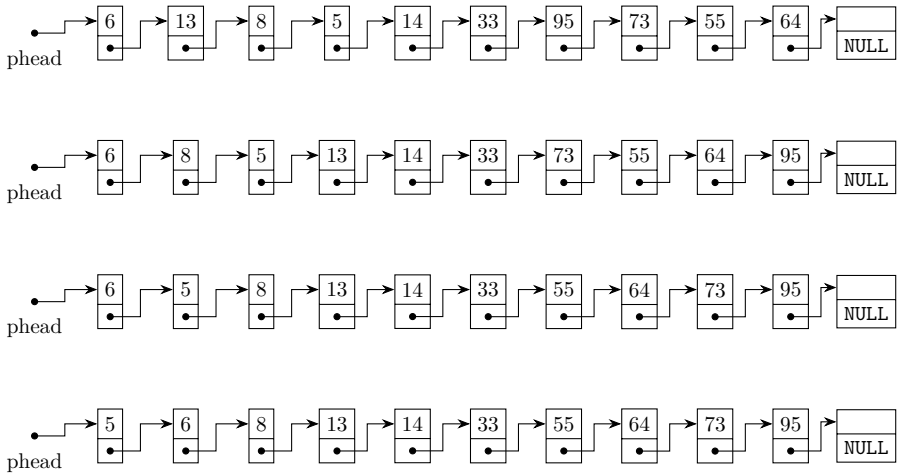
2 Aufgabe 2

Lösung im beigefügten Eclipse-Project

Ciheng Zhang(3472321)
Yao He(3487882)
YuchanBian(3496226)

3 Aufgabe 3

(a)



(b)

Wenn Sie von vorne nach hinten fortfahren, muss das Programm jedes Mal nur seinen eigenen Wert mit seinem nächsten Wert vergleichen. Wenn er größer als sein nächste Wert ist, tauschen Sie den Next Zeiger direkt aus. Aber Wenn Sie von hinten nach vorne ausführen, müssen Sie die Liste von head itaration, um bei jedem Vergleich und Austausch den vor Ihnen liegenden Wert zu ermitteln. Dies erhöht den Rechenaufwand.

4 Aufgabe 4

Fall A

MergeSort
Denn die Durchschnittlicher Rechenaufwand von MergeSort ist $nlog_2n$.D.h. Wenn die Folgen ist größer als 2. die MergeSort ist schnellest Methode.

Fall B

InsertionSort BubbleSort
Beiden diese 2 Methoden brauchen nur ein mal die Folgen iteration. Und keine Vertauch. Dann fertig. D.h. nur linearerAufwand.
die Komplexität ist n

Ciheng Zhang(3472321)
Yao He(3487882)
YuchanBian(3496226)

4.1 Fall C

QuickSort und MergeSort
denn die Komplexität von InsertionSort,BubbleSort und SelectSort in diese Situation ist fast n^2. Aber die MergeSort und QuickSort benutzen die Teil-und-Herrsche-Grundsatzes.D.h. diese1 Beiden Methoden brauchen nur $nlog_2n$ Komplexität.
die Komplexität ist $nlog_2n$