

Protokoll der Vordiplomprüfung im Fach Praktische Informatik (Info1+2)

Prüfling: Markus Treinen (für Fragen: Treinen@castor.uni-trier.de)

Prüfer: Stefan Näher

Beisitzer: Oliver Zlotowski

Datum: 11.06.2004

Block 1

Was ist denn ein allgemeines Sortierverfahren?

Erklärung, daß nur mit Vergleichen und Vertauschen sortiert wird.

Dann sollte ich mir eins aussuchen und erklären: **Heapsort**

Verfahren erklärt, zwei Phasen, Darstellung des Heaps im Feld, Referenzierung ohne Pointer möglich

Beide Phasen haben die Funktion *sink* gemeinsam. Funktionsweise erklären (ohne Algorithmus). Dann sollte ich die Schleifen der Phasen im Pseudocode aufschreiben und erklären, was passiert.

Anschließend wollte er die Laufzeit von *sink* wissen, mit Erklärung, wie diese zustande kommt. *Laufzeit ist $2i$, wobei i die Höhe des Levels ist, von dem *sink* startet.*

Zu guter letzt wollte er noch die Laufzeit vom kompletten Heapsort wissen, natürlich mit Beweis. *Setzt sich aus den Laufzeiten der beiden Phasen zusammen, im Endeffekt $O(n \log n)$.*

Block 2

Ich sollte mir ein spezielles Sortierverfahren aussuchen und zuerst die Unterschiede zum allgemeinen erklären: **Countingsort**

Spezielle Sortierverfahren sortieren nicht durch Vergleichen, sondern z.B. wie hier durch Zählen.

Zuerst wollte er die Funktionsweise von mir wissen.

Es werden alle Werte durchgezählt, dann ein Positionsarray erzeugt und aufgrund dessen das Eingabefeld sortiert.

Dann sollte ich den Algorithmus im Pseudocode aufschreiben.

Aufgeschrieben und die Funktionsweise nochmal genauer erläutert anhand des Codes.

Warum baut man das Aufgabefeld von rechts nach links auf?

Wegen der Stabilität.

Was ist Stabilität?

Daß gleiche Werte in der Ausgabe die gleiche Reihenfolge wie in der Eingabe haben.

Wo braucht man das (mit Beispiel)?

Z.B. beim Sortieren eines Telefonbuchs, wenn also von einem Buchstaben noch andere Dinge abhängen.

Block 3

Hier kam dann mein Schwachpunkt Graphen an die Reihe.

Zuerst sollte ich erklären, was eine topologische Sortierung ist und was dieser für Eigenschaften hat.

Ein topologisch sortierter Graph ist azyklisch und umgekehrt.

Von links nach rechts kann man erklären, indem man den Graph "in einer Reihe" aufschreibt. Von rechts nach links wirds kniffliger, das muß mit Induktion bewiesen werden. Hier habe ich mich sehr

verhaspelt und das Ganze mit DFS durcheinandergeworfen.

Als nächstes sollte ich die rekursive Funktion von DFS im Pseudocode aufschreiben. Er wollte nur das Hauptprogramm sehen.

Im Endeffekt muß man hier nur aufschreiben, daß von jedem Knoten aus jede Kante rekursiv erreicht wird.

Dann sollte noch die Laufzeit analysiert werden.

Die Laufzeit ist $n+m$, da für jeden Knoten n mal geprüft werden muß, ob eine Kante vorhanden ist (n) und dann jede Kante genau 1 mal durchlaufen wird (m).

Wie kann man denn mit der rekursiven DFS-Funktion eine topologische Sortierung bewerkstelligen?

Man schreibt sich ja die Zähler $dfsnum$ und $compnum$ auf. $n-compnum$ ergibt hier die topologische Sortierung.

Block 4

Was ist eine Objektorientierte Programmiersprache (OOP)?

Gegensatz zu funktionsorientiert, Klassen, Vererbung, Hierarchie.

Wie schreibt man in Java eine Vererbung einer Klasse?

Mit dem Schlüsselwort "extends".

Kann man nur mit "extends" ableiten?

Nein, auch mit "implements" im Fall eines Interfaces.

Dann sollte ich ein Beispiel für die Ableitung einer Klasse UND Interface beschreiben.

Z.B. enthält das Interface eine Sammlung von Konstanten, die ich in der abgeleiteten Unterklasse benutzen will.

Wie funktioniert denn in Java die Fehlerbehandlung?

Mit Exceptions, die in einem try-catch-Block geschrieben werden. Diesen sollte ich dann aufschreiben und erklären, wie der Ablauf im Fehlerfall und im Normalfall ist. Wichtig ist, daß "finally" IMMER ausgeführt wird, in beiden Fällen.

Kann man vom catch-block zurück in try?

Zlotowski schüttelte den Kopf, was mir dann zum Glück die Antwort verriet ;-). Es geht also nicht.

Fazit

Dauer: ca. 35 Minuten

Note: 2.0

Es war eine recht entspannte Atmosphäre. Herr Näher ist ein etwas verpeilter Prüfer (er wollte mir weismachen, daß eine Laufzeit nicht $\log i$ ist, dann hat er's aber doch eingesehen ;-).

Die Prüfung war prinzipiell genau nach Schema der anderen Protokolle, allerdings ist DFS bei mir nicht so gut gelaufen. Es kommt aber auch vor, daß er etwas detaillierter fragt, ich hatte den Vorteil, daß ich der erste am Tag war.

Viel Erfolg und Spaß beim Lernen!