

Übungsblatt 8

Abgabe bis Dienstag, den 10. Januar um 16:00 Uhr

Aufgabe 1 (9 Punkte)

In der 5. Vorlesung hatten wir die Heap-Datenstruktur kennengelernt und mit einem einfachen Array realisiert. Wir hatten damit eine Prioritätswarteschlange mit den Operationen *insert*, *getMin* und *deleteMin* realisiert. Die Operation *getMin* hatte dabei eine Laufzeit von $\Theta(1)$, die Operationen *insert* und *deleteMin* im schlechtesten Fall eine Laufzeit von $\Theta(\log n)$.

Bestimmen Sie nun für jede dieser drei Operationen die Anzahl der Block-Operationen im schlechtesten Fall, für eine gegebene Blockgröße B . Selbstverständlich sollen Sie nicht einfach nur einen Ausdruck hinschreiben, sondern diesen auch begründen.

Hinweis: Teilen Sie das Array, in dem die Elemente des Heaps stehen, auf die natürliche Art in Blöcke der Größe B ein. Wenn Sie jetzt die Elemente auf einem Pfad von der Wurzel bis zu einem Blatt betrachten, in vielen verschiedenen Blöcken können diese Elemente liegen?

Aufgabe 2 (9 Punkte)

In der Vorlesung hatten wir eine einfache Methode zur Berechnung einer zufälligen Permutation eines gegebenen Feldes geschrieben.

- Wieviele Block-Operationen benötigt diese Methode im schlechtesten Fall?
- Messen Sie die Laufzeit der Methode für ein möglichst großes Feld. (Wählen Sie das Feld so groß, dass das Programm auf Ihrem Rechner in erträglicher Zeit fertig wird.)
- Versuchen Sie jetzt die Methode cache-effizienter zu implementieren, so dass sie tatsächlich schneller läuft. Nehmen Sie die für b) gemessene Laufzeit als Maßstab. Es ist für diesen Aufgabenteil OK, wenn die berechnete Permutation nicht vollkommen zufällig ist, das heißt, wenn in Ihrer Variante manche Permutationen häufiger vorkommen als andere. Zu einfach dürfen Sie es sich aber nicht machen, insbesondere muss Ihre Methode mindestens so viele Vertauschungen machen wie die einfache aus b) und jedes Element muss potenziell an jede Stelle kommen können. Und Ihr Programm sollte auf jeden Fall schneller sein als das aus b).

Sie brauchen für diese Aufgabe ausnahmsweise keine Tests zu schreiben. Sie sollen aber schon schauen, dass auf Jenkins alles kompiliert.

Aufgabe 3 (2 Punkte)

Schreiben und committen Sie außerdem wie gehabt (in *non-code/uebungsblatt_8*) eine Datei *erfahrungen.txt* zu Ihren Erfahrungen mit dem Übungsblatt und der Vorlesung.