Universität Stuttgart
Fakultät 5, Institut FMI
Einführung in die Informatik II

Sommersemester 2006
Claus, Draskoczy, Schmid
08.07.06

Aufgabenblatt 11 **Abgabe: 15.07.06, 24.00 Uhr**

1. **Zusammenfassung: Suchen** (mittel) (schriftlich, 3 Punkte)
   Fassen Sie das gesamte Kapitel 8 zusammen (einen Teil haben Sie bereits zusammengefasst). Die Zusammenfassung soll handschriftlich sein und auf maximal eine DIN-A4-Seite passen.
   Das Erstellen der Zusammenfassung können Sie gerne gemeinsam in Ihrer Lerngruppe machen, es muss allerdings jeder Übungsteilnehmer eine eigene Zusammenfassung verfassen! Schreiben Sie Ihren Namen und Übungsgruppe auf die Seite, scannen Sie die Seite ein und geben Sie sie als JPEG-Datei im eClaus-System ab.

2. **Zusammenfassung: Hashing** (leicht–mittel) (schriftlich, 2 Punkte)
   Erstellen Sie eine Zusammenfassung über Kapitel 9 (Hashing). Auch diese Zusammenfassung soll handschriftlich sein und auf maximal eine DIN-A4-Seite passen.
   Das Erstellen der Zusammenfassung können Sie gerne gemeinsam in Ihrer Lerngruppe machen, es muss allerdings jeder Übungsteilnehmer eine eigene Zusammenfassung verfassen! Schreiben Sie Ihren Namen und Übungsgruppe auf die Seite, scannen Sie die Seite ein und geben Sie sie als JPEG-Datei im eClaus-System ab.

3. **Mergesort** (leicht–mittel) (schriftlich, 1+2 Punkte)
   Sortieren Sie die Zahlenfolge 8, 7, 99, 16, 1, 19, 23, 8, 2, 42 und 3 mit dem Verfahren Mergesort. Geben Sie im eClaus-System die Zahlenfolge für jeden Zwischenschritt und jeden Aufruf von `Mergesort(x, y)` ab!
   Ist Mergesort ein stabiles Sortierverfahren? Beweisen Sie Ihre Aussage und geben Sie den Beweis ebenfalls im eClaus-System ab.

4. **Bucketsort** (leicht) (schriftlich, 3 Punkte)
   Schreiben Sie ein Ada-Programm, das einzelne Zeichen vom Benutzer einliest (also einen beliebig langen String) und nach dem Verfahren *Streuen und Sammeln* sortiert. Zum Abschluss soll die sortierte Folge ausgegeben werden.
   Selbstverständlich kann ein Zeichen mehrfach im Eingabestring vorkommen!

5. **Paralleles Sortieren** (mittel) (schriftlich, 4 Punkte)
   Schreiben Sie ein Ada-Programm, das Integerwerte vom Benutzer einliest und nach dem Verfahren *lineare Kette* sortiert. Dabei soll jede Eingabe sofort in die lineare Kette geschoben werden und die bisherige Kette ausgegeben werden. Sobald der Benutzer den letzten Wert eingegeben hat, werden nur noch die benötigten Schiebetakte ausgeführt und nach jedem Takt die Kette ausgegeben. Die zuletzt ausgegebene Kette ist die sortierte Folge.
   Natürlich findet hier nicht wirklich paralleles Sortieren statt, sondern wird nur simuliert!

6. Permutationen (mittel) (schriftlich, 5 Punkte)

Sind die einzelnen Datensätze sehr umfangreich, so kann man sich beim Sortieren Zeit sparen, indem man nur die Permutation bestimmt und die einzelnen Datensätze zunächst nicht verschiebt. Ist die Permutation bestimmt, so kann man aufgrund dieser die eigentlichen Datensätze im Array verschieben. Modifizieren Sie einen Sortieralgorithmus Ihrer Wahl, sodass dieser die Permutation bestimmt, nach der die Elemente umsortiert werden müssen, und erst im Anschluss die eigentlichen Datensätze verschiebt. Wie groß ist der Aufwand in O-Notation zur Bestimmung der Permutation, wie groß der zur Umsortierung? Versuchen Sie mit möglichst wenig zusätzlichem Speicherplatz auszukommen.

Zusatzaufgabe: Inwiefern hat diese Aufgabe etwas mit Rehashing zu tun? Demonstrieren Sie diesen Zusammenhang an einem Beispiel.