

Übung 8 – Hashing

In dieser Übung soll das Verhalten der Klasse *java.util.HashMap* auf unterschiedliche Hash-Funktionen in den Java Versionen 1.3, 1.4.0 und 1.4.2 untersucht werden.

Gegeben sind die beiden Programme *Test1* und *Test2* (verfügbar auf dem Server) die Elemente in einer Hashtabelle (vom Typ *java.util.HashMap*) ablegen und auf diese wieder zugreifen.

```
public class Test1 {
    private static final int LEN=500;
    public static void main(String args[]) {
        HashMap map = new HashMap();
        Double [] keys = new Double[LEN];

        for (int i=0; i<LEN; i++) {
            keys[i] = new Double(i);
            map.put(keys[i], new Integer(i));
        }

        long t0 = System.currentTimeMillis();
        for (int rep = 0; rep < 1000; rep++) {
            for (int i = 0; i < LEN; i++) { map.get(keys[i]); }
        }
        long t1 = System.currentTimeMillis();
        System.out.println("time = " + (t1-t0));
    }
}

public class Test2 {
    private static final int LEN = 1000;
    public static void main(String[] args){
        HashMap map = new HashMap(203);
        Integer[] keys = new Integer[LEN];

        for (int i=0; i < LEN; i++) {
            keys[i] = new Integer(i * 1024);
            map.put(keys[i], "Value");
        }

        long t0 = System.currentTimeMillis();
        for (int rep = 0; rep < 1000; rep++) {
            for (int i = 0; i < LEN; i++) { map.get(keys[i]); }
        }
        long t1 = System.currentTimeMillis();
        System.out.println("time = " + (t1-t0));
    }
}
```

Lassen Sie beide Programme mit den Java Versionen 1.3, 1.4.0 (1.4.0_00 oder 1.4.0_01) und 1.4.2 laufen (es kann dabei immer derselbe Byte-Code verwendet werden). Die Laufzeiten werden stark variieren, da diese drei Java Versionen unterschiedliche Implementationen der Klasse *java.util.HashMap* enthalten. Alte Java Versionen sind auf der Sun Web-Site im Archiv unter <http://java.sun.com/products/archive/index.html> verfügbar.

Versuchen Sie die Effizienzunterschiede zu erklären. Mögliche Vorgehensweisen um diese Frage zu beantworten sind:

- Studium der Quelltexte der Klasse *HashMap* in den Versionen 1.3, 1.4 und 1.4.2
- Überprüfung mit einem Debugger, wie gut die Elemente in der Hash-Tabelle verteilt sind
- Ausgabe der Hash-Codes modulo verwendete Tabellengröße.

Ziel ist, dass Sie die Problematik der Implementierung effizienter Datenstrukturen sehen und erleben.