

## Übungsblatt 8

Abgabe bis Donnerstag, 07.07.2011, 12:00 Uhr

### Aufgabe 8.1

Betrachten Sie folgende Tabelle. In welche Komplexitätsklassen gehören die Funktionen  $f_i(n)$  jeweils? Beachten Sie, dass eine Funktion auch mehreren Klassen angehören kann.

	$O(n)$	$O(n^4)$	$O(\log(n))$	$O(2^n)$	$O(4^n)$
$f_1(n) = n^2 + \log(n) + n^4$					
$f_2(n) = n + \sqrt{n}$					
$f_3(n) = 2^{n+2}$					
$f_4(n) = 2^{n+n}$					
$f_5(n) = 3^n$					
$f_6(n) = 7777 \cdot n^3 + n^4$					
$f_7(n) = \frac{1}{2^n}$					

### Aufgabe 8.2

Die folgende Methode `findDuplicates` identifiziert Duplikate in einem String-Vektor.

```
static void findDuplicates(Vector<String> v) {  
    int n = v.size();  
    for (int i = 0; i < n-1; i++)  
        for (int j = i+1; j < n; j++)  
            if (v.elementAt(i).equals(v.elementAt(j)))  
                System.out.println("Duplikat");  
}
```

1. Führen Sie für `findDuplicates` eine Aufwandsabschätzung für die Laufzeit in Abhängigkeit von der Länge  $n$  des Vektors  $v$  durch. Gehen Sie davon aus, dass die Methoden `elementAt()`, `equals()` sowie `println()` jeweils konstante Laufzeit haben.
2. Implementieren Sie eine Methode `boolean hasDuplicates(Vector<String> v)`, indem Sie obige Methode so modifizieren, dass genau dann `true` zurückgegeben wird, wenn der Vektor ein Duplikat enthält. Wie verändert sich die Best Case und Worst Case Laufzeit im Vergleich zu der Methode `findDuplicates`?

### Aufgabe 8.3

Erweitern Sie die Klasse `Matrix` aus der Vorlesung um folgende Methoden und führen Sie jeweils eine Aufwandsabschätzung durch:

1. `Matrix submatrix(Vector<Integer> m_vec, Vector<Integer> n_vec)`

Diese Methode soll eine Untermatrix zurückgeben, deren Elemente sich aus den Zeilen  $m \in m\_vec$  sowie den Spalten  $n \in n\_vec$  der ursprünglichen Matrix ergeben. Enthalten die Vektoren  $m\_vec$  und  $n\_vec$  ungültige Zeilen bzw. Spaltennummern, so sollen diese ignoriert werden.

2. `Matrix prod(Matrix matrix)`

Diese Methode soll das Matrixprodukt berechnen. Falls die Form der beiden Matrizen keine Multiplikation erlaubt, soll eine leere Matrix zurückgegeben werden.