



Bachelor-Studiengang

Übungen zur Vorlesung "Grundlagen der Informatik" (GDI), WS 2007/2008

Übungsblatt 5: Arrays

Ausgabe am: 30.11.2007

Abgabe verschoben: 15.11.2007
... 3.11.2007

Aufgabe 1

16+15=31 Punkte

In der Programmiersprache C werden Strings als Array of Character verwaltet. Das Array hat eine feste Länge, das Ende des gerade darin gespeicherten Strings ist durch ein Zeichen '\0' gekennzeichnet.

Beispiel

Der String "Hello World!" wird in einem Array of Char der Länge len wie folgt gespeichert

	H	e	l	l	o		W	o	r	l	d	!	\0			...	
Index:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14		len-1

Das Null-Byte an der Index-Position 12 zeigt an, dass der String keine weiteren (sinnvollen) Zeichen mehr enthält, sondern 12 Zeichen lang ist; unabhängig von der tatsächlichen Länge des Arrays (das viel länger sein kann).

Zur Sprache C gehört eine Bibliothek, die eine Reihe von String-Operationen, sehr ähnlich zu den hier beschriebenen, zur Verfügung stellt:

- `int strlen(char[] s)`
liefert die Länge des Strings
- `char[] strcat(char[] s1, char[] s2)`
fügt die Zeichen des 2. Strings an das Ende des 1. Strings an und liefert eine Referenz darauf als Ergebnis; der erste String muss eine Variable sein
- `boolean strcmp(char[] s1, char[] s2)`
vergleicht die beiden Argumente und liefert genau dann *true* als Ergebnis, wenn die repräsentierten Strings gleich sind
- `boolean stricmp(char[] s1, char[] s2)`
vergleicht die beiden Argumente und liefert genau dann *true* als Ergebnis, wenn die repräsentierten Strings gleich sind; Groß- und Kleinbuchstaben werden nicht unterschieden
- `int printf(char[] s)`
druckt den übergebenen String auf den Bildschirm und liefert die Anzahl ausgegebener Zeichen als Ergebnis (Es sollen nicht die vielen verschiedenen Formatangaben berücksichtigt werden, die printf bietet)

Aufgabe:

Implementieren Sie die beschriebenen Operationen in Java, indem Sie Character-Arrays anstelle der in Java verfügbaren Strings benutzen. Die Strings sollen wie in C auch durch ein Null-Zeichen abgeschlossen werden

- a) Implementieren Sie die beschriebenen Operationen **iterativ** (Rekursion ist (auch in Hilfsmethoden) **nicht** erlaubt) in einer Klasse "Aufgabe1" im Package "uebung05".
- b) Implementieren Sie eine geeignete Umgebung, in der Sie die Operationen testen können und testen Sie sie. Achtung: Es soll möglich sein, verschiedene "Strings" zur Laufzeit Ihres Programms einzugeben, auf denen dann Ihre String-Funktionen arbeiten. Denken Sie beim Testen an Sonderfälle!

Bemerkungen

- Es sind keine Methoden der Java-Bibliotheken erlaubt außer den in der Vorlesung behandelten.
- Die Methoden dürfen intern keine Strings und keine zusätzlichen Character-Arrays (nur die als Parameter übergebenen Parameter) verwenden; die Länge der als Parameter übergebenen Arrays darf nicht verändert werden.
- Sie dürfen eigene Hilfsmethoden je nach Bedarf vereinbaren und verwenden.
- Dokumentieren Sie explizit in *jedem* (Sonder-)Fall, was bei "fehlerhaften" Eingaben passiert (z.B. Abschluss-0-Byte fehlt, Quelle einer Kopieroperation länger als das Ziel etc.).
- Denken Sie daran, Ihre Programme mit vernünftigen Kommentaren zu versehen! Tipp: Fragen Sie Kommilitonen, ob diese Ihre Programme mit Ihren Kommentaren *ohne weitere mündliche Erklärung* verstehen oder was sie vermissen. Eventuell finden sie einige Kommentare auch überflüssig?
- Halten Sie sich genau an die oben angegebenen Vereinbarungen:
Es wird ein automatisches Testprogramm für die Implementierung Ihrer Methoden geben.