



Laborübung

Klasse: 1DHIF

Datum:14.03.23

Lernstoff:

Iterieren durch Arrays und manipulieren von Arraydaten
Zusatzattribut anz für die Anzahl der relevanten Werte im Array
Sortieren von Array Daten

Erweitere die im Unterricht entwickelte **CharArray** Klasse mit den im folgenden beschriebenen Methoden. :

CharArray
- satz:char[] - anz:int
+ CharArray (satz:char[]) + entferneZeichen(c:char) + ausgeben() + reverseZeile() + istIdentMit(satz:char[]):boolean + istTeilVon(satzTeil:char[]):String + ersterIndexVon(c:char):int + ersterIndexVon(c:char, abIndex:int):int + letzterIndexVon(c:char):int + sort()

Die Methode **reverseZeile()** dreht die Anordnung der Buchstaben in der Zeile um. Achte dabei darauf, dass du nur die relevanten Buchstaben in Betracht ziehst - >anz!!!

Beispiel „Hallo Welt!“ wird zu „!tleW ollaH!

Die Methode **istIdentMit(char[] satz)** gibt zurück ob der übergebene Satz mit dem Attribut *satz* identisch ist. Achte auf die Größe des beiden Arrays und auf *anz* – der Vergleich darf nicht zu einer *IndexOutOfBoundsException* führen!

Die Methode **istTeilVon(char[] satzTeil)** stellt fest ob der übergebene *satzTeil* ein Teil des Attributs *satz* ist. Zurückgegeben wird ein String, der das Ergebnis des Vergleichs in Worten in einem String zurückgibt.

Beispiele für den Rückgabewert:

„Hallo“ ist in „Hallo Welt!“ enthalten

„Hello“ ist nicht in „Hallo Welt“ enthalten

Die Methode **ersterIndexVon(char c)** bekommt ein Zeichen und gibt zurück an welcher Stelle im *satz* das Zeichen zuerst auftritt. Falls das Zeichen nicht vorkommt wird -1 zurückgegeben

Beispiel:

Inhalt von <i>satz</i>	char	Rückgabe
Hallo Welt!	l	2
Hallo Welt!	x	-1

Die Methode ***ersterIndexVon(char c, int abIndex)*** bekommt ein Zeichen und eine Position im Array und gibt zurück an welcher Stelle im Satz das Zeichen ab der Position zuerst auftritt. Falls das Zeichen nicht vorkommt wird -1 zurückgegeben

Beispiel:

Inhalt von <i>satz</i>	char	abIndex	Rückgabe
Hallo Welt!	l	4	8
Hallo Welt!	l	12	-1

Die Methode ***letzterIndexVon(char c)*** bekommt ein Zeichen und gibt zurück an welcher Stelle im Satz das Zeichen zuletzt auftritt. Falls das Zeichen nicht vorkommt wird -1 zurückgegeben

Beispiel:

Inhalt von <i>satz</i>	char	Rückgabe
Hallo Welt!	l	8
Hallo Welt!	x	-1

Die Methode ***sort()*** sortiert die relevanten Elemente des Arrays (beachte das Attribut *anz!!!*) absteigend nach den Alphabet (ASCII Code) – nachdem Großbuchstaben in der Tabelle vor Kleinbuchstaben stehen wird sich dies auch auf die Sortierung auswirken! Verwende folgenden beschriebenen Algorithmus, der eine Abwandlung des SelectionSort Algorithmus ist:

- Finde die Position des größten Elements im relevanten Array (*anz!!*)
- Tausche dieses Element mit dem Element, das an der letzten Stelle (\Rightarrow *anz-1*) steht
- Betrachte im nächsten Schritt den um 1s verkürzten Arrayabschnitt (\Rightarrow also Position 0 bis Position *anz-2*)
- Finde die Position des größten Elements in diesem Arrayabschnitt
- Tausche das größte Element des Arrayabschnitts, mit dem Element, das an der vorletzten Stelle (\Rightarrow *anz-2*) steht
- Betrachte im nächsten Schritt wieder den um 1s verkürzten Array (\Rightarrow also Position 0 bis Position *anz-3*)
- Finde wieder die Position des größten Elements in diesem Arrayabschnitt
- Tausche das größte Element des Arrayabschnitts, mit dem Element, das an der vorletzten Stelle (\Rightarrow *anz-3*) steht
- ... wiederhole diesen Algorithmus solange bis der betrachtete Arrayabschnitt nur mehr aus einem Element besteht.

Erweitere dein main um die Funktionalität der “neuen“ Methoden zu testen.