

Prof. Markus Gross, Bruno Heidelberger, Richard Keiser, Nicky Kern, Edouard Lamboray, Christoph Niederberger, Tim Weyrich, Felix Eberhard, Manuel Graber, Nathalie Kellenberger, Marcel Kessler, Lior Wehrli

Übung 13 – Abgeleitete Klassen

Ausgabe: 9. Februar 2004
Abgabe: keine
Autor: Nicky Kern

1. Palindrom Checker

10 Punkte

Ein Palindrom ist ein Satz, der von vorne wie von hinten gelesen das gleiche (oder auch etwas anderes - aber es muss einen Sinn haben) ergibt. Auf <http://www.ulf-hinze.de/kram/palindrom.htm> findet ihr einige Beispiele für Palindrome.

Einleitung

Wir haben in der Vorlesung den abstrakten Datentyp "Stack" kennen gelernt. Dieser Datentyp ist sehr gut für einen Palindrom-Test geeignet. Da ein abstrakter Datentyp ja nicht durch eine *Datenstruktur* sondern durch *Zugriffsfunktionen* definiert ist, wollen wir verschiedene Implementationen derselben Datenstruktur ausprobieren.

Array-Stack

Implementiere den Stack aus den Vorlesungsfolien mit einem Array. Leite dafür die Klasse Stack aus `stack.h` ab und implementiere die vier Methoden. Das allgemeine Item wurde durch einen einfachen `char` ersetzt. Beachte, dass Du eine fixe Arraygrösse allozieren musst.

Palindrom-Checker

Schreibe eine Prozedur, welche testet, ob ein gegebener String ein Palindrom ist. Sie bekommt einen String und ein Objekt der Stack-Klasse übergeben und liefert einen `bool` zurück, je nachdem ob der übergebene String ein Palindrom ist oder nicht.

Gehe wie folgt vor:

- Füge die Zeichen des Strings nach und nach in den Stack ein (mit `push()`). Füge keine Leer- oder Sonderzeichen, sondern nur Buchstaben ein. Mit `isalpha()` kannst Du testen, ob es sich bei einem gegebenen `char` um einen Buchstaben handelt.
- Durchlaufe den String ein zweites Mal und vergleiche das aktuelle Zeichen mit dem obersten vom Stack. Ueberspringe auch hier Leer- und sonstige Sonderzeichen. Für den Vergleich empfiehlt es sich, beide Zeichen mittels `tolower()` in Kleinbuchstaben zu konvertieren.

Testprogramm

Schreibe ein Testprogramm. Es soll einen String von der Kommandozeile auf sein Palindrom-Sein testen. Benutze dafür die vorhin entwickelte Funktion.

Tip: Um einen String, der Leerzeichen enthält, auf der Kommandozeile als ein Argument zu übergeben, muss man ihn in doppelte Anführungszeichen einschliessen. Beispielsweise:

```
> ./palindrom "Vitaler Nebel mit Sinn ist im Leben relativ"
```

Listen-Stack

Die Implementation des Stacks mit einem Array funktioniert zwar wunderbar, ist aber nicht unter allen Umständen ideal, weil die Länge des Strings begrenzt ist. Beispielsweise ist ein 20er-Array für das obige Palindrom bereits zu klein! Aus diesem Grund wollen wir eine zweite Implementation entwerfen, die eine Liste verwendet.

Implementiere einen zweiten Stack, der auf einer Liste basiert. Erweitere das Hauptprogramm dahingehend, dass die Palindrom-Checker-Funktion je einmal mit einem Array- und einem Listen-Stack aufgerufen wird. Was sind die Vor- und Nachteile von Array- respektive Listen-Stack?