

Übungsblatt 8

Abgabe bis Mittwoch, den 6. Juli um 16:00 Uhr

Aufgabe 1 (4 Punkte)

Nehmen Sie Ihre *String* Klasse vom 6. Übungsblatt, und templatisieren Sie diese, so dass Ketten von Elementen von einem beliebigen Typ *T* möglich sind. Gehen Sie dabei vor wie bei der Templatisierung der Klasse *Array* in der Vorlesung.

Um die volle Punktzahl für diese Aufgabe zu bekommen, reicht es wenn Sie die Methoden *insert* und *erase*, sowie den *assignment operator* berücksichtigen. Die Methoden *find* und *replace* sowie den *copy constructor* können Sie auch berücksichtigen oder weglassen, wie Sie möchten.

Wenn Ihre Lösung für das 6. Übungsblatt fehlerhaft war, können Sie auch die Musterlösung als Ausgangsbasis für Ihren Code benutzen.

Es reicht, Tests für die Instantiierung *String<char>* zu schreiben. Ein *Main* Programm ist für diese Aufgabe nicht notwendig.

Aufgabe 2 (5 Punkte)

Implementieren Sie eine template Spezialisierung für Bitstrings, d.h. für Ketten von Elementen vom Typ *bool*. Dabei sollten Sie *n* Bits kompakt in $\lceil n/8 \rceil$ Bytes speichern.

Für die volle Punktzahl reicht es wieder, wenn Sie die Methoden *insert* und *erase* und den *assignment operator* implementieren. Ändern Sie aber den *assignment operator* so ab, dass man einen normalen c-string mit Nullen und Einsen zuweisen kann, also etwa schreiben kann:

```
String<bool> bitString;  
bitString = "0011011111011"; // Should be stored in only 2 bytes.
```

Schreiben Sie für Ihre Spezialisierung separate Tests. Achten Sie bei den Tests für *insert* und *erase* auf kritische Grenzfälle, z.B. wenn gerade an einer Byte-Grenze eingefügt wird. Sie müssen nicht alle möglichen solcher Grenzfälle berücksichtigen, aber wenigstens drei.

Aufgabe 3 (1 Punkt)

Committen Sie wie gehabt alles ins SVN, in ein neues Unterverzeichnis *uebungsblatt-8*, stellen Sie sicher, dass alles kompiliert, die Tests ohne Fehler laufen, Lint ohne Fehler durchläuft und auch auf Jenkins alles funktioniert. Und die *erfahrungen.txt* nicht vergessen.