

## Labb 6 Iterators

Implementera iteratorer som itererar över den String klass som ni gjorde i Labb 3. Läs gärna på hur iteratorerna för STL containers fungerar. STL står för C++ Standard Template Library.

Syftet med en iterator är att den ska fungera på nästan samma sätt som en pekare. Läs mera i C++ primer page 106 ff. Se även <http://en.cppreference.com/w/cpp/string>.

Observera att göra en helt korrekt implementering av iterators som stämmer helt med STL-systemet är ganska komplicerat. Vi nöjer oss med att göra en ganska minimal implementering som räcker för att visa idéerna med iterators.

En iterator itererar över elementen i ett container objekt. I vårt fall så är det String objekten från den tidigare labben vi skall iterera över. Observera att vi behöver lägga till några saker i String klassen för att det ska fungera.

Uppgift:

Ni ska göra alla fyra iteratorer typerna:

- iterator
- const\_iterator
- reverse\_iterator
- const\_revers\_iterator

I String klassen lägg till:

- typedefs för iterator, const\_iterator, reverse\_iterator och const\_revers\_iterator
- funktionerna begin, end, cbegin, cend, rbegin, rend, crbegin och crend.

Iteratorerna ska kunna göra:

- \*it, ++it, it++, (it+i), it[i], == och !=
- default constructor, copy constructor och tilldelning (=)

Kommentarer:

```
String s("hej");  
for (String:: reverse_iterator it=s.rbegin(); it!= s.rend(); ++it)  
    cout << *it;
```

kommer att skriva ut "jeh".

**OBS!** För att implementera iterator, const\_iterator och funktionerna begin, end, cbegin, cend krävs det i stort sätt ingen kod alls!

### Testprogrammet i Main.cpp

Observera att det testprogram som finns i Main.cpp bara är en hjälp och varken fullständigt eller garanterat helt korrekt. Det är möjligt att testprogrammet kör felfritt fast er lösning är felaktig. Det är även möjligt – men inte troligt – att er lösning är korrekt fast testprogrammet inte kör/kompilerar felfritt.

### Krav för G:

Gör ovanstående.

### Krav för VG:

Det finns inga särskilda VG krav på denna laboration men det som står för VG i labb 3 gäller även för denna labb, dvs. om ni ska ha VG så måste även iteratorerna ha const på rätt ställe och var effektiva etc.