

# C# Generics, Collections, Iterators, and Regular Expressions

# Rasmus Lystrøm

## External Associate Professor

### ITU

nie@itu.dk

[illegible][illegible]

# Agenda

Generics

Iterators

Collections

Regular Expressions

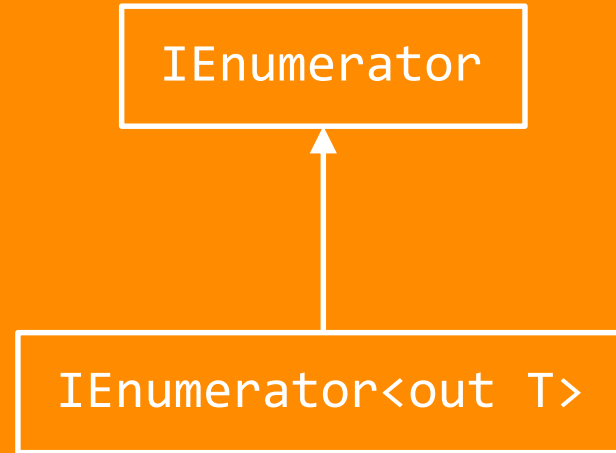
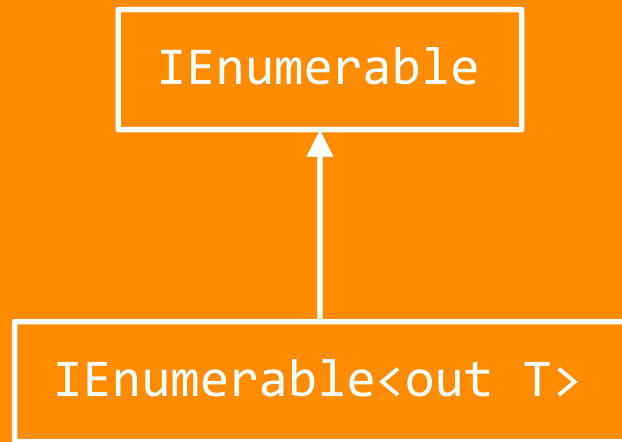
# Generics

Parametric Polymorphism

Type Constraints

Co- and contravariance

# Iterators



Producer

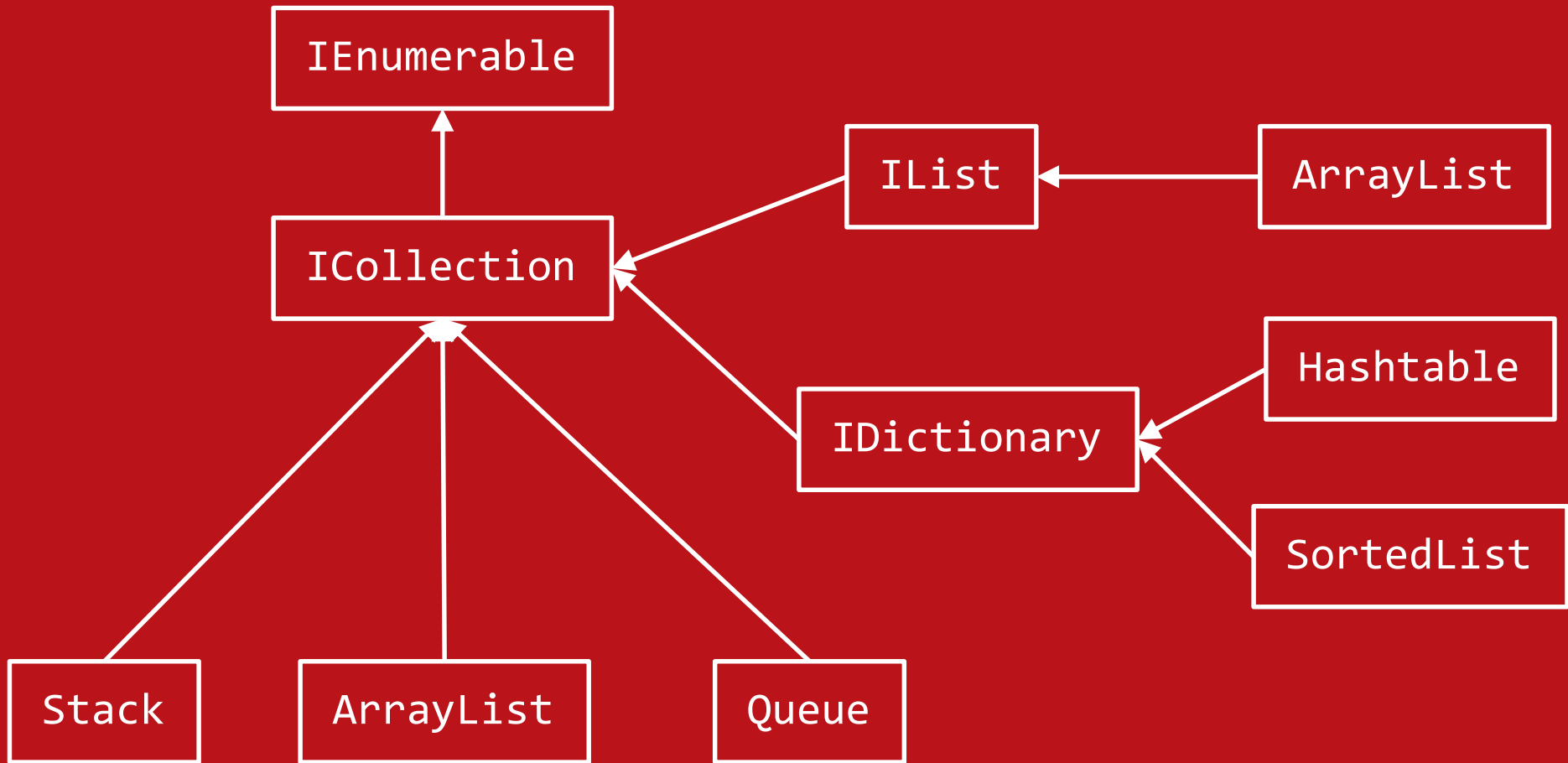
`yield return T;`

`yield break;`

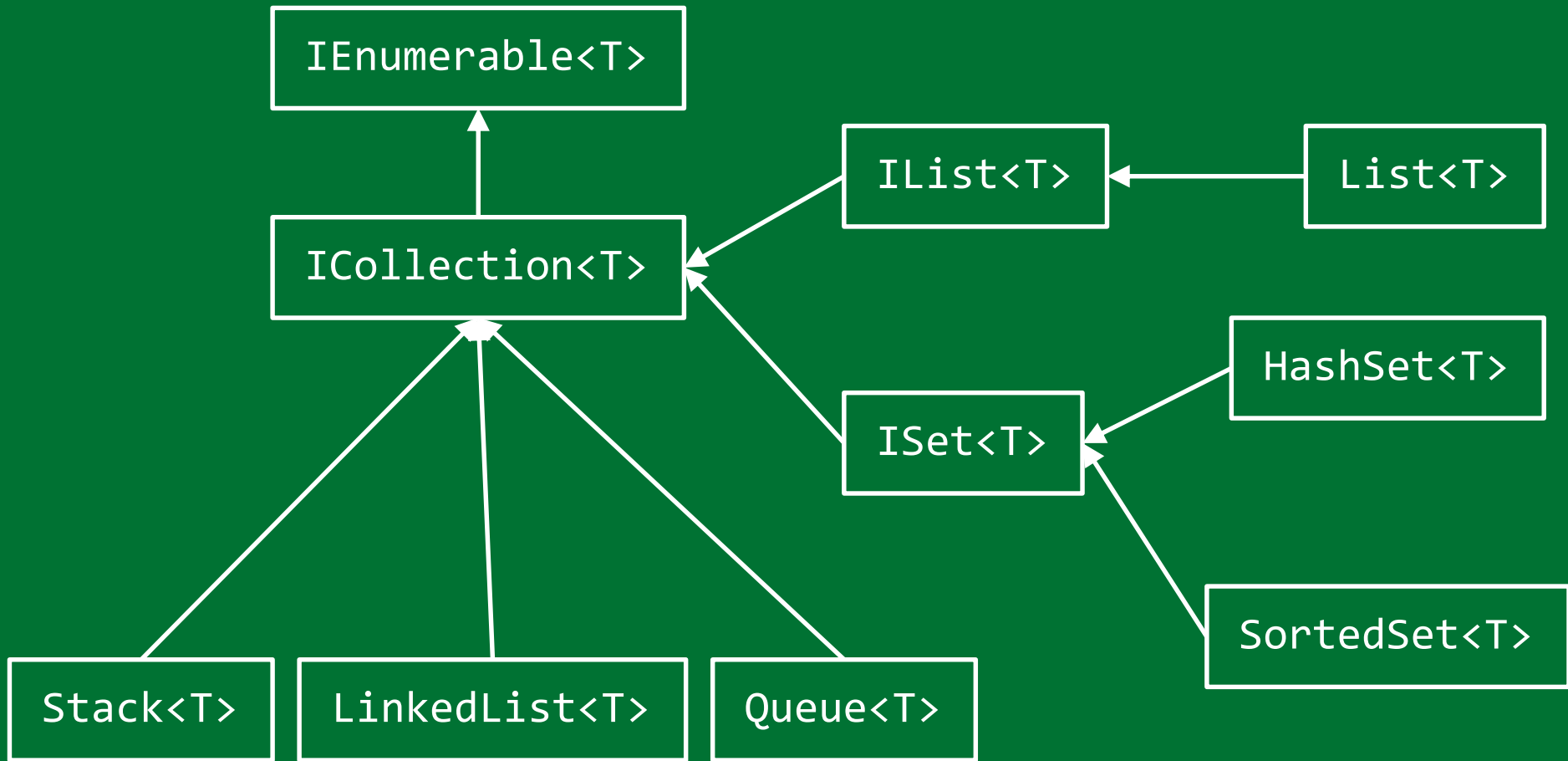
Building block for Linq

```
foreach (var item in items)
{
}
```

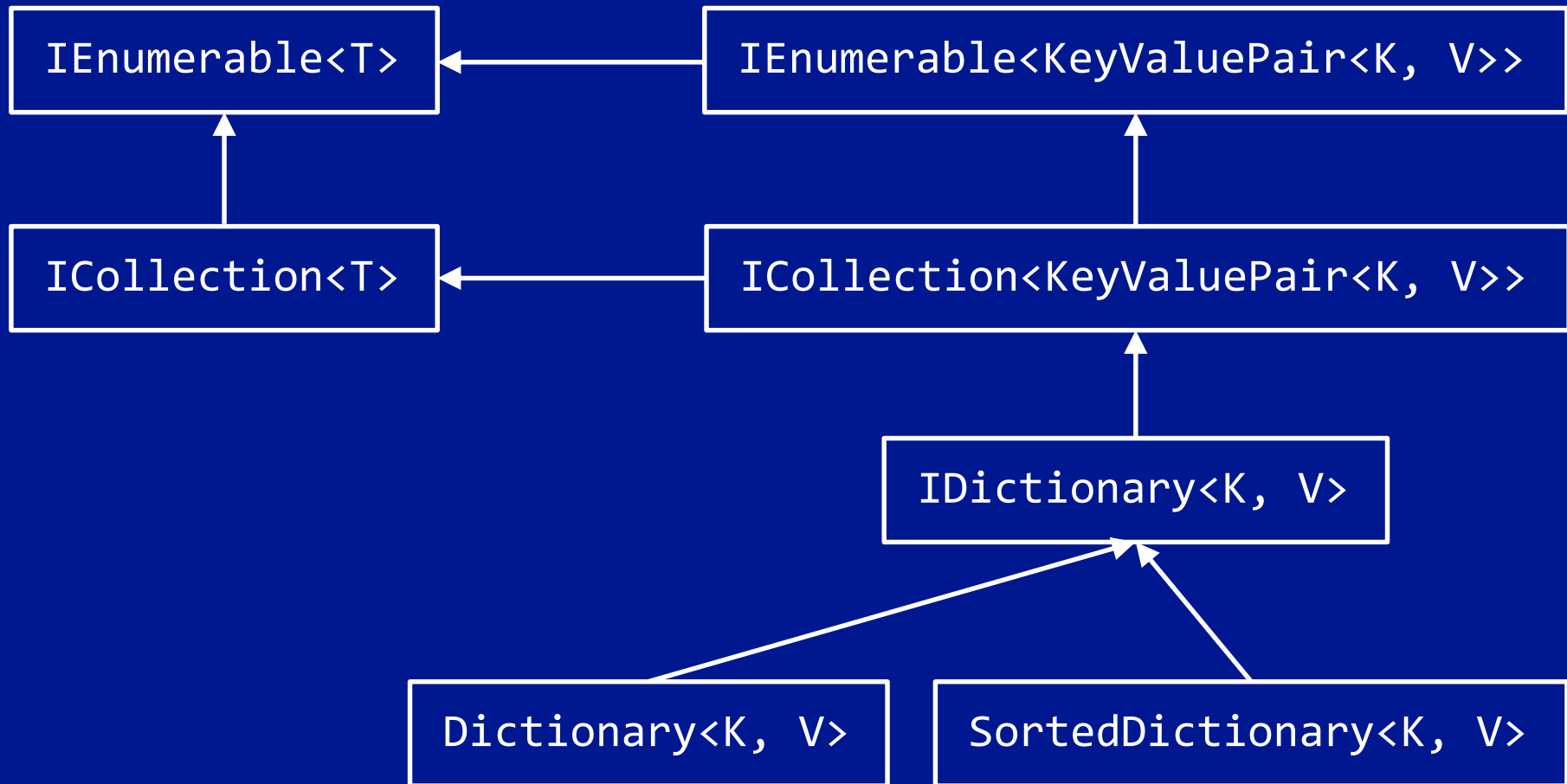
# System.Collections



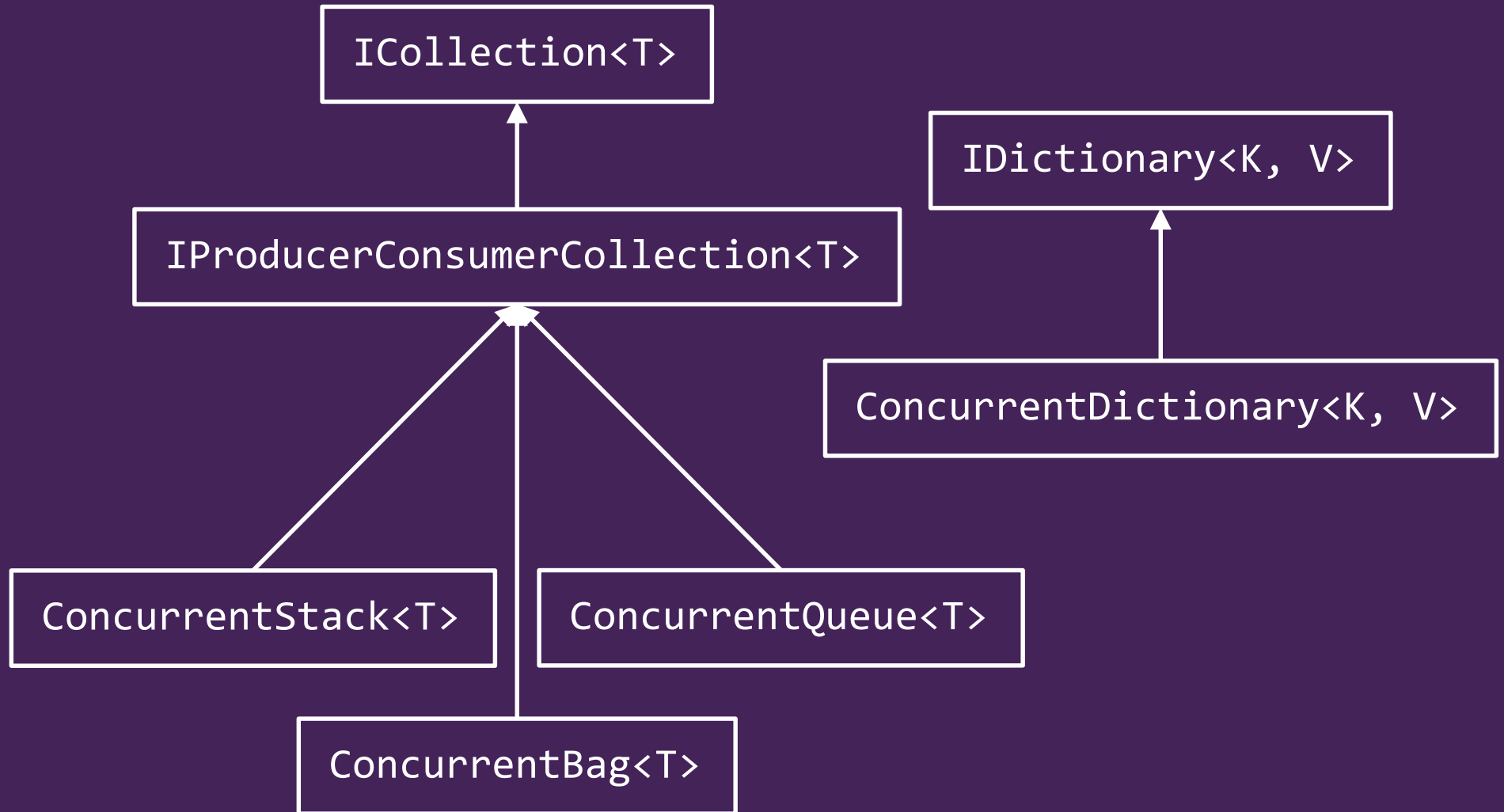
# System.Collections.Generic



# System.Collections.Generic 2



# System.Collections.Concurrent





# Regular Expressions

*	Zero or more times the previous character
+	Once or more times the previous character
?	Zero or one time the previous character
.	Any single character (not \n)
\s	Any whitespace character (e.g. tab)
\S	Any non-whitespace character
\b	Word boundary
\B	Any non-word boundary position
\w	Any word character (a-z, A-Z, 0-9)
\W	Any non-word character
^	Start of the input text
\$	End of the input text

# Regular Expressions

[1c]	matches character '1' or 'c'
[a-z]	matches all lower-case letters
[a-zA-Z]	matches all letters
[0-9]+	matches integer numbers
[0-9]+\.[0-9]+	matches a floating point
[0-2][0-9]:[0-5][0-9]	matches a time e.g. 12:34