

# Strings

Victor Eijkhout and Charlie Dey

spring 2017

# String declaration

```
#include <string>
using namespace std;

// .. and now you can use 'string'
```

# String creation

A *string* variable contains a string of characters.

```
string txt;
```

You can initialize the string variable, or assign it dynamically:

```
string txt{"this is text"};  
string moretxt("this is also text");  
txt = "and now it is another text";
```

# Concatenation

Strings can be *concatenated*:

```
txt = txt1+txt2;  
txt += txt3;
```

# String is like vector

You can query the *size*:

```
int txtlen = txt.size();
```

or use subscripts:

```
cout << "The second character is <<" <<  
      txt[1] << ">>" << endl;
```

# More vector methods

Other methods for the vector class apply: `insert`, `empty`, `erase`, `push_back`, et cetera.

[http://en.cppreference.com/w/cpp/string/basic\\_string](http://en.cppreference.com/w/cpp/string/basic_string)

# Exercise 1

Write a function to print out the digits of a number: 156 should print one five six. Use a vector of strings.

Hint: it is easiest to generate the digits last-to-first. Then figure out how to print them reversed.