Digital Integrated Circuit Design

Module CppTmpl
C++ Review:
Templates, STL, & Containers

W. Rhett Davis

Templates

- Templates are functions, structs, classes, etc. with generic types
- The template sum below will work for any type that is defined for the operator +

```
template <class T>
T sum (T a, T b) {
  return (a+b);
}
...
int i = sum<int>(1,2);
double f = sum<double>(3.14,2.71);
string s = sum<string>("Hello ","World!");
```

- See http://www.cplusplus.com/doc/tutorial/templates/

STL Containers

- STL - Standard Template Library
  - » Standard in any C++ compiler
- STL Containers
  - » A set of objects that collect other objects
  - » Most common:
    - – vector
    - – map
- See http://www.cplusplus.com/reference/stl/

Vector

- push_back() to add elements to list, pop_back() to remove
- Use [] to access elements
- Use iterator to traverse all elements
  - acts as a pointer to current element, but also has begin() and end() methods

```
#include <vector>
#include <string>
#include <iostream>

vector<string> days;
days.push_back("Monday");
days.push_back("Tuesday");

cout << days[1] << endl;
vector<string>::iterator it;
for (it= days.begin(); it < days.end(); it++)
  cout << *it << endl;
```

Map

- Map uses first element to index the second element

```
#include <map>
#include <string>
#include <iostream>

map<string,int> months;
months["March"]=3;
months["June"]=6;

map<string,int>::iterator it;
for (it= months.begin(); it != months.end(); it++)
  cout << it->first << ' ' << it->second << endl;
```

- Use "first" field of iterator to get the index
- Use "second" field of iterator to get the value

Digital Integrated Circuit Design

Module CppTmpl
C++ Review:
Templates, STL, & Containers

W. Rhett Davis

Thanks for watching