CS 162
Intro to CS II

STL/Templates
Odds and Ends

• Turn in your extra credit from Expo
• Keep working on Assignment 4
• Assignment 4 questions
Why Function Templates?

//at least C++ has overload
void swap(int &, int &);  
void swap(char &, char &);  
...
void swap(int &a, int &b) {
    int temp = a;  
    a = b;  
    b = temp;
}  
void swap(char &a, char &b) {  
    char temp = a;  
    a = b;  
    b = temp;
}
Function Template...

//Have to have this header
template<class T>
void swap(T &, T &);
...

template<class T>
void swap(T &a, T &b) {
    T temp = a;
    a = b;
    b = temp;
}

OSU Oregon State University
When can you get into trouble?
//Have to have this header

```cpp
template<class T>
void func(T *, T *, int);
...

template<class T>
void func(T a[], T b[], int size) {
    // a is already a reference/static array
    // what if we wanted to swap values in arrays
}
```

OSU Oregon State University
Why make a class templated?

• What functionality have we written so many times, regardless of type?
Class Templates

//Have to have this header

template<class T>

class vector {
    public:
        vector();
        ~vector();
        void push_back(T);
    private:
        T *v;
};
Class Templates

//Have to have this header
template<class T>
vector<T>::vector(){
    v=NULL;
}

template<class T>
vector<T>::~vector(){
    delete [] v;
}

template<class T>
void vector<T>::push_back(T element){
    ...
}

Demo
Using pre-defined vector class...

- What is size() vs. capacity()
- What is resize() vs. reserve()?
- How does push_back() work?
Demo