Strings, Lists, and a little File I/O
C++

- C++ programs will still work properly in C, so older C programs can also be used in C++, this is mostly to use C-strings.
- C++ strings are more flexible and easier to use than C-strings.
- The C++ string is more flexible and easier to use.
- C++ has a new class called "string" ("#include <string>") that ends with the '\0' character.
- In C, a string is simply an array of characters that ends with the '\0' character.

C++ Strings
other functions

- The string class has many convenient member functions. For example:
  - The operations <, >, <=, >= at position i of string s (position starts at 0, like an array)
  - s[i] or s.at(i) returns the character at position i of string s
  - s.length() returns the length of string s
  - With cout and cin

The operations and >> so you can use a string naturally

The operations + to add (concatenate) two strings together:
  - z > ... > b > a > Z > ... > A

Compare two strings lexicographically:
  - The operations >, >=, ==, <=, < can all be used to

Example:

The string class has many convenient member functions.
```cpp
#include <iostream>
#include <string>
using namespace std;

int main()
{
  string word1, word2;

  cout << "Enter two words:" << endl;
  cin >> word1 >> word2;

  if (word1 == word2)
    cout << (word2 + "", + word1) << endl;
  else
    cout << (word1 > word2) << endl;

  return 0;
}
```

---OUTPUT---
Enter two words: texas iowa
iowa, texas
When a string is used to store input from cin, a blank space denotes the end of the string.

When you want to read an entire line, including blank spaces, use the getline function.

The syntax for getline with a string named "s" is:

```cpp
generate(chin, s);
```
```c++
int main()
{
    string line1, line2;
    char temp;
    cout << "Enter two lines of text:" << endl;
    getline(cin, line1);
    getline(cin, line2);
    cout << "Line #1 contains " << line1.length() << " characters." << endl;
    cout << "Line #2 contains " << line2.length() << " characters." << endl;
    cout << line2 << endl;
    return 0;
}
```
If it snows all night?

Line #2 contains 22 characters.

Will classes be cancelled if it snows all night?

Enter two lines of text:

---OUTPUT---
mmcmendozathemert@string2

---QUIT---
How to use lists without having to write a list class yourself.
The STL List Template

To access the STL List template, we must use `#include <list>` to be able to access the STL list template.

- `list<int>::iterator intItemPtr = intList.begin();`
- `list<string>::iterator strItemPtr = strList.end();`

Generalized pointer.

To access list elements, we use an iterator of strings.

- `list<int>::iterator intItemPtr;`
- `list<string> strList;`
STL list example code

use the functions that come with list<int>

– allows us to write our own functions and also

creates a new class, derived from list<int>

mylist.h, mylist.cpp, useMyList.cpp

– read from cin, sorts, output

– uses a list of integer and a list of strings

list.h, list.cpp

STL list example code
The Poor Man’s File I/O

- Using a file for input data
  - Use the Unix "read" operator to direct the contents of the input file into the program
  - Use Emacs to create an input file that contains your input data
  - Use the Unix "<" operator to redirect the contents of the input file into the program

- Output will be printed on the screen.
  - ./Newton < input.txt

- Works under Linux/unix, and Mac OS X, I'm not sure about Windows
Sending output to a file

- Use the Unix redirect command `>` to direct output to a file.

- To take input values from input.txt and save output to a file, type `./Newton > input.txt < output.txt`

- I don't advise redirecting output without redirecting input. Just input is okay, or both, but not just output.

- Both input and output are redirected by `./Newton < input.txt > output.txt`.
The cerr stream

- Even though input and output are being redirected to a file, error messages using "cerr" will still be printed to the screen.

- If you print error messages using "cerr", they will be printed to the screen even when output is directed to a file.
```cpp
#include <iostream>

using namespace std;

int main(){
  int n;
  cin >> n;
  cout << "n is " << n << endl;
  cerr << "This message is not redirected." << endl;
  return 0;
}
```
Exercise #10

Write a program that will use an STL list of strings.

- How large must the list of strings be before there is noticeable difference in the time it takes to sort after the list has been read vs sorting the list as it is being input?

- Test this program with some very large input files. (1) II

- Generate some and put them on the website.

- Now try to organize your input process so the strings are sorted as they are read from input.

- Read in strings, sort the strings, print the strings back out.

- Write a program that will use an STL list of strings.