

CS 180 - More C++

Note Title

9/6/2011

Announcements

- HW is up - due Saturday by midnight
- Lab 2 is posted - prelab before class tomorrow
- Presentation about on campus opportunity
- Math/CS club meets Wed at 4pm

Example from last time

```
double gpa;  
cout << "Enter your gpa: ";  
cin >> gpa;  
if (gpa = 4.0)  
    cout << "Wow!" << endl;
```

`gpa = 4.0` returns `4.0`, not "true".

Why?

$a = (b = 0);$

$y = a = a + 1;$

Arrays

Python has lists, tuples, etc.

In C++, only have arrays.

- Size is fixed at declaration
- type is fixed (& homogeneous)

register address
↓

Ex:

```
int numbers[10];  
numbers[0] = 55;  
numbers[9] = 10;
```

```
numbers[10] = 5; ← (error in Python)  
↑ in C++, seg fault
```

numbers		register address
0	55	2659
1		2660
2		6261
3		
4		
5		
6		
7		
8		
9	10	

Creating Arrays:

Allowed:
int

days In Month = {31, 28, 31, 30, 31, 30,
31, 31, 30, 31, 30, 31};

Error: int days In Month [] ;
no size

Allowed:

char greeting[] = "Hello";
must be a char!

reason - strings are char arrays

Input & Output

C++ has several predefined classes.

Class	Purpose	Library
istream	Parent class for all input streams	<iostream>
ostream	Parent class for all output streams	<iostream>
iostream	Parent class for streams that can process input and output	<iostream>
ifstream	Input file stream	<fstream>
ofstream	Output file stream	<fstream>
fstream	Input/output file stream	<fstream>
istringstream	String stream for input	<sstream>
ostringstream	String stream for output	<sstream>
stringstream	String stream for input and output	<sstream>

Using iostream

```
#include <iostream>  
using namespace std;  
std::cin
```

useful for debugging

Notes: - can now use cin (for input)
 + cout (for output)
 (just like print or raw-input)

- separate distinct variables by >> or <<
 cin cout

- use endl for end of a line

Example

'\n'

Python

```
print "Hello"  
print # blank line  
print "Hello,", first  
print first, last # automatic space  
print total  
print str(total) + "." # no space  
print "Wait... ", # space; no newline  
print "Done"
```

C++

```
1 cout << "Hello" << endl;  
2 cout << endl; // blank line  
3 cout << "Hello, " << first << endl;  
4 cout << first << " " << last << endl;  
5 cout << total << endl;  
6 cout << total << "." << endl;  
7 cout << "Wait... "; // no newline  
8 cout << "Done" << endl;
```


Formatting output

```
cout << team << ": ranked " << rank << " of " << total << " teams" << endl;
```

- No '%d' here to easily format

Can set precision:

```
cout << "pi is " << fixed << setprecision(3) << pi << endl;
```

- Note that precision stays set to 3
pi is 3.141

Using cin

```
int number;  
cout << "Enter a number:";  
cin >> number;
```

Note: - inputs are separated by any white space
cin >> a >> b;
10 _ 20 return
10 ?
20 ?

- type of input must match
type of variable
(not all strings)

One possible problem:

```
string person;  
cout << "What is your name? ";  
cin >> person;  
cin >> age;
```

I type "Erin Chambers".

What happens?

person = "Erin"

Getline

- getline is a function which saves the string up to (but not including) the next newline

Ex: string person;
 cout << "What is your name?";
 getline (cin, person);

Erin Chambers

Another tricky example

```
int age;  
string food;  
cout << "How old are you? ";  
cin >> age;  
cout << "What would you like to eat? ";  
getline(cin, food);
```

← cin ;

I type :

age
15

pizza

Problem:

age = 15
food = ""

Using File Streams - ifstream

```
#include <iostream>
```

```
#include <fstream>
```

```
using namespace std;
```

if file is known:

```
ifstream mydata("scores.txt");
```

~~ifstream~~ mydata >> variable;

if not:

```
ifstream mydata;
```

```
string filename;
```

```
cout << "What file? ";
```

```
cin >> filename;
```

```
mydata.open(filename.c_str( )); // parameter to open must be a C-style string
```

converts to c-style string