

ITK 279
Algorithms and Data Structures

Fall, 2008

Dr. Califf

Suppose

- You need to put a large file in order alphabetically.
- You need to explain to your boss why you picked a particular way to store a bunch of information.
- You're faced with a problem unlike any you've ever tried to solve before.

Suppose

- You need to write a program that has to be extremely fast.
- Your boss has just asked you to solve a problem that cannot be optimally solved during your lifetime. How will you know that?

Instructor

- Dr. Mary Elaine Califf
- Office: Old Union 106
- Email: mecaliff@ilstu.edu

PowerPoint with Califf

- An organizational tool
- A place to put problems up for you
- Not a primary way to convey information
- It will be posted—usually before class
- It is **NOT** intended to be a source of information for study, although it can be useful in combination with your notes

Class Policies

- Reading
- Homework
- Group work
- Quizzes
- Blackboard
- Website:
<http://www.itk.ilstu.edu/faculty/mecalif/calif.htm>

Course Expectations

- Read before class
- Attend class
- Do your homework
- Start programs promptly
- Follow directions
- Do your own work
- Group work
- Use UNIX
- ASK
- Explain
- Office hours
- Prompt grading
- Answer

What is this course about?

- What makes really good software?

What is this course about?

- Algorithms
- Data structures
- Last “pure” programming course

Why UNIX?

Why C++?

For Wednesday

- Read the syllabus carefully.
- Skim the first 3 chapters of Savitch.

For Friday

- Turn in student information sheet
- Send an email to mecaliff@ilstu.edu with
 - Your name
 - ITK 279

Background

- Language/Data Structures
- Math
- Recursion

C++ (vs. Java)

- Very similar in many ways
 - Same control structures
 - Very similar syntax
- Some important differences
 - C++ is less object-oriented
 - Input/output
 - Arrays
 - Handling of dynamic memory (pointers)
 - Libraries

Hello World in C++

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

int main(int argc, char *argv[])
{
    cout << "Hello world!" << endl;
    return 0;
}
```

Notes

- `#include` is used to access libraries and other code. It actually copies the text of the file into your file, so you must not include things more than once
- Namespaces are used to organize large projects. All identifiers from the various C++ libraries are in the `std` namespace.
- Functions in C++ do not have to be methods (members of a class). The main function is never a member of a class.

Input/Output

- `cout` is the reference to standard output (typically the monitor)
- `cin` is the reference to standard input (typically the keyboard)
- The insertion operator, `<<`, lets you write to an output stream.
- The extraction operator, `>>`, lets you read from an input stream.

Input Example

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

int main()
{
    int age;
    cout << "Please enter your age: ";
    cin >> age;
    cout << "You entered " << age << endl;
    return 0;
}
```

Notes

- The arguments to main are optional.
- You can handle multiple inputs at once (just like outputs). Typically only useful with files.

Primitive Data Types

- Basic integral types:
 - char
 - short
 - int
 - long
- Basic floating point types
 - float
 - double
- Boolean type
 - bool
 - Technically numeric – has values true and false but they are 1 and 0
 - In C++, 0 is false and everything else true

Strings

- Traditional “C” strings
 - Arrays of chars ending in the null char (“\0”)
 - Required for things like file names
- String class
 - Somewhat similar to Java’s
 - Not immutable