Introduction & Course Overview

Dr. Robert Lowe

Division of Mathematics and Computer Science
Maryville College

Outline

Outline

Course Description

An introduction to computer science and structured programming with emphasis on program design and implementation, debugging, documentation, and programming projects. Laboratory work supplements and expands lecture topics and offers supervised practice using programming.

Required Materials

- *Big C++. 3/e*. Cay Horstmann. https://tinyurl.com/bigcpp
- An internet connected computer of some sort.

Grading

Category	**Weight**
Exams	30%
Quizzes	15%
Attendance & Hands-On Lab Activities	20%
Programming Assignments	35%

Academic Honesty

- Cheating and Plagiarism are Immoral (you are a bad person if you do this).

Academic Honesty

- Cheating and Plagiarism are Immoral (you are a bad person if you do this).
- Plagiarism is integrating other people's work into your own without attribution.

Academic Honesty

- Cheating and Plagiarism are Immoral (you are a bad person if you do this).
- Plagiarism is integrating other people's work into your own without attribution.
- Cheating is the wholesale copying of another work.

Academic Honesty

- Cheating and Plagiarism are Immoral (you are a bad person if you do this).
- Plagiarism is integrating other people's work into your own without attribution.
- Cheating is the wholesale copying of another work.
- Neither Plagiarism or Cheating will be tolerated in any form!

Academic Honesty

- Cheating and Plagiarism are Immoral (you are a bad person if you do this).
- Plagiarism is integrating other people's work into your own without attribution.
- Cheating is the wholesale copying of another work.
- Neither Plagiarism or Cheating will be tolerated in any form!
- Acceptable Sources of Code:

Academic Honesty

- Cheating and Plagiarism are Immoral (you are a bad person if you do this).
- Plagiarism is integrating other people's work into your own without attribution.
- Cheating is the wholesale copying of another work.
- Neither Plagiarism or Cheating will be tolerated in any form!
- Acceptable Sources of Code:
  - Your own original code.

Academic Honesty

- Cheating and Plagiarism are Immoral (you are a bad person if you do this).
- Plagiarism is integrating other people's work into your own without attribution.
- Cheating is the wholesale copying of another work.
- Neither Plagiarism or Cheating will be tolerated in any form!
- Acceptable Sources of Code:
  - Your own original code.
  - Code from class examples and lab assignments.

Academic Honesty

- Cheating and Plagiarism are Immoral (you are a bad person if you do this).
- Plagiarism is integrating other people's work into your own without attribution.
- Cheating is the wholesale copying of another work.
- Neither Plagiarism or Cheating will be tolerated in any form!
- Acceptable Sources of Code:
  - Your own original code.
  - Code from class examples and lab assignments.
  - Code from your textbook.

Academic Honesty

- Cheating and Plagiarism are Immoral (you are a bad person if you do this).
- Plagiarism is integrating other people's work into your own without attribution.
- Cheating is the wholesale copying of another work.
- Neither Plagiarism or Cheating will be tolerated in any form!
- Acceptable Sources of Code:
  - Your own original code.
  - Code from class examples and lab assignments.
  - Code from your textbook.
- An honest failure is respectable. Cheating is contemptible.

Course Policy Overview

- Do not cheat.

Course Policy Overview

- Do not cheat.
- Deadlines are set, no extensions will be given.

Course Policy Overview

- Do not cheat.
- Deadlines are set, no extensions will be given.
- No late work will be accepted.

Course Policy Overview

- Do not cheat.
- Deadlines are set, no extensions will be given.
- No late work will be accepted.
- Attendance is mandatory.

Course Policy Overview

- Do not cheat.
- Deadlines are set, no extensions will be given.
- No late work will be accepted.
- Attendance is mandatory.
- Excused absences are possible, but do not modify any due dates (except in rare circumstances).

Course Policy Overview

- Do not cheat.
- Deadlines are set, no extensions will be given.
- No late work will be accepted.
- Attendance is mandatory.
- Excused absences are possible, but do not modify any due dates (except in rare circumstances).
- Students with disabilities should contact the academic support center 981-8124 to coordinate accommodations.

Outline

Account Setup & Logging In

1. Signup for a free github account: http://github.com

Account Setup & Logging In

1. Signup for a free github account: http://github.com
2. Visit https://tinyurl.com/cs111-github to share your github username and real name with me.

Account Setup & Logging In

1. Signup for a free github account: http://github.com
2. Visit https://tinyurl.com/cs111-github to share your github username and real name with me.
3. Signup for an MCCS account by visiting https://cs.maryvillecollege.edu/signup

Account Setup & Logging In

1. Signup for a free github account: http://github.com
2. Visit https://tinyurl.com/cs111-github to share your github username and real name with me.
3. Signup for an MCCS account by visiting https://cs.maryvillecollege.edu/signup
4. Click the link in your email to set your password.

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.
3. Click open

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.
3. Click open
4. When prompted, enter your username and then password.

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.
3. Click open
4. When prompted, enter your username and then password.

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.
3. Click open
4. When prompted, enter your username and then password.

ChromeOS

1. Install and launch “Secure Shell”

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.
3. Click open
4. When prompted, enter your username and then password.

ChromeOS

1. Install and launch “Secure Shell”
2. Fill in connection details and click “open”

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.
3. Click open
4. When prompted, enter your username and then password.

ChromeOS

1. Install and launch “Secure Shell”
2. Fill in connection details and click “open”

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.
3. Click open
4. When prompted, enter your username and then password.

ChromeOS

1. Install and launch “Secure Shell”
2. Fill in connection details and click “open”

MacOS X

1. Launch terminal 

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.
3. Click open
4. When prompted, enter your username and then password.

ChromeOS

1. Install and launch “Secure Shell”
2. Fill in connection details and click “open”

MacOS X

1. Launch terminal 

2. Enter the command: `ssh first.last@cs.maryvillecollege.edu`

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.
3. Click open
4. When prompted, enter your username and then password.

ChromeOS

1. Install and launch “Secure Shell”
2. Fill in connection details and click “open”

MacOS X

1. Launch terminal 

2. Enter the command: `ssh first.last@cs.maryvillecollege.edu`
3. Press enter

Maryville COLLEGE

Connecting to MCCS

Username: first.last **Server**: cs.maryvillecollege.edu **Port**: 22

Windows

1. Launch Putty 

2. Enter the servername.
3. Click open
4. When prompted, enter your username and then password.

ChromeOS

1. Install and launch “Secure Shell”
2. Fill in connection details and click “open”

MacOS X

1. Launch terminal 

2. Enter the command: `ssh first.last@cs.maryvillecollege.edu`
3. Press enter
4. Enter password when prompted.

The Linux Command Prompt

- Commands are entered in the following pattern: `command arg1 arg2...`

The Linux Command Prompt

- Commands are entered in the following pattern: `command arg1 arg2...`
- Pressing “enter” executes the commands.

The Linux Command Prompt

- Commands are entered in the following pattern: `command arg1 arg2...`
- Pressing “enter” executes the commands.
- Let’s try it out! Run the following commands:

The Linux Command Prompt

- Commands are entered in the following pattern: `command arg1 arg2...`
- Pressing “enter” executes the commands.
- Let’s try it out! Run the following commands:
  1. `ls`

The Linux Command Prompt

- Commands are entered in the following pattern: `command arg1 arg2...`
- Pressing “enter” executes the commands.
- Let’s try it out! Run the following commands:
  1. `ls`
  2. `ls -a`

The Linux Command Prompt

- Commands are entered in the following pattern: `command arg1 arg2...`
- Pressing “enter” executes the commands.
- Let’s try it out! Run the following commands:
  1. `ls`
  2. `ls -a`
  3. `ls -l -a`

The Linux Command Prompt

- Commands are entered in the following pattern: `command arg1 arg2...`
- Pressing “enter” executes the commands.
- Let’s try it out! Run the following commands:
  1. `ls`
  2. `ls -a`
  3. `ls -l -a`
  4. `man ls`

The Linux Command Prompt

- Commands are entered in the following pattern: `command arg1 arg2...`
- Pressing “enter” executes the commands.
- Let’s try it out! Run the following commands:
  1. `ls`
  2. `ls -a`
  3. `ls -l -a`
  4. `man ls`
  5. `man man`

Working With Directories

`ls` List files in the current directory

Working With Directories

`ls` List files in the current directory

`mkdir` *name* Create a Directory

Working With Directories

`ls` List files in the current directory

`mkdir` *name* Create a Directory

`pwd` Show the present working directory

Working With Directories

`ls` List files in the current directory

`mkdir` *name* Create a Directory

`pwd` Show the present working directory

`cd` *directory* Change Directory

Working With Directories

`ls` List files in the current directory

`mkdir` *name* Create a Directory

`pwd` Show the present working directory

`cd` *directory* Change Directory

Let's try it out!

Working With Directories

`ls` List files in the current directory

`mkdir` *name* Create a Directory

`pwd` Show the present working directory

`cd` *directory* Change Directory

Let's try it out!

1. `mkdir humor`

Working With Directories

`ls` List files in the current directory

`mkdir` *name* Create a Directory

`pwd` Show the present working directory

`cd` *directory* Change Directory

Let's try it out!

1. `mkdir humor`
2. `ls`

Working With Directories

`ls` List files in the current directory

`mkdir` *name* Create a Directory

`pwd` Show the present working directory

`cd` *directory* Change Directory

Let's try it out!

1. `mkdir humor`
2. `ls`
3. `cd humor`

Working With Directories

`ls` List files in the current directory

`mkdir` *name* Create a Directory

`pwd` Show the present working directory

`cd` *directory* Change Directory

Let's try it out!

1. `mkdir humor`
2. `ls`
3. `cd humor`
4. `pwd`

Working With Files

`cp` *src dest* Copy a file

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

`less` *filename* View a file

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

`less` *filename* View a file

Let's try it out!

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

`less` *filename* View a file

Let's try it out!

1. `cp /home/robert.lowe/humor/bees bess`

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

`less` *filename* View a file

Let's try it out!

1. `cp /home/robert.lowe/humor/bees bess`
2. `ls`

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

`less` *filename* View a file

Let's try it out!

1. `cp /home/robert.lowe/humor/bees bess`
2. `ls`
3. `mv bess bees`

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

`less` *filename* View a file

Let's try it out!

1. `cp /home/robert.lowe/humor/bees bess`
2. `ls`
3. `mv bess bees`
4. `ls`

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

`less` *filename* View a file

Let's try it out!

1. `cp /home/robert.lowe/humor/bees bess`
2. `ls`
3. `mv bess bees`
4. `ls`
5. `rm bees`

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

`less` *filename* View a file

Let's try it out!

1. `cp /home/robert.lowe/humor/bees bess`
2. `ls`
3. `mv bess bees`
4. `ls`
5. `rm bees`
6. `ls`

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

`less` *filename* View a file

Let's try it out!

1. `cp /home/robert.lowe/humor/bees bess`
2. `ls`
3. `mv bess bees`
4. `ls`
5. `rm bees`
6. `ls`
7. `cp  robert.lowe/humor/* .`

Working With Files

`cp` *src dest* Copy a file

`mv` *src dest* Move a File

`rm` *filename* Remove a file (be careful!)

`less` *filename* View a file

Let's try it out!

1. `cp /home/robert.lowe/humor/bees bess`
2. `ls`
3. `mv bess bees`
4. `ls`
5. `rm bees`
6. `ls`
7. `cp  robert.lowe/humor/* .`
8. `less threes`

Text Editing

- Programming is about entering text into files.

Text Editing

- Programming is about entering text into files.
- A variety of text editors exist.

Text Editing

- Programming is about entering text into files.
- A variety of text editors exist.
- People tend to choose friends based on common text editor preferences.

Text Editing

- Programming is about entering text into files.
- A variety of text editors exist.
- People tend to choose friends based on common text editor preferences.
- The most common editors on our system are:

Text Editing

- Programming is about entering text into files.
- A variety of text editors exist.
- People tend to choose friends based on common text editor preferences.
- The most common editors on our system are:

  `nano` - A simple editor, original designed for muggles to use with email.

Text Editing

- Programming is about entering text into files.
- A variety of text editors exist.
- People tend to choose friends based on common text editor preferences.
- The most common editors on our system are:

  `nano` - A simple editor, original designed for muggles to use with email.

  `vi` - Visual Editor a modal text editor.

Text Editing

- Programming is about entering text into files.
- A variety of text editors exist.
- People tend to choose friends based on common text editor preferences.
- The most common editors on our system are:

`nano` - A simple editor, original designed for muggles to use with email.

`vi` - Visual Editor a modal text editor.

`emacs` - A very elaborate and powerful editor.

Playing Games

Let's take a look at games installed on the system!

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`
2. `ls /usr/local/games`

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`
2. `ls /usr/local/games`
3. Try typing some of the names that came up. Some of my favorites are:

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`
2. `ls /usr/local/games`
3. Try typing some of the names that came up. Some of my favorites are:
   - `tetris-bsd`

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`
2. `ls /usr/local/games`
3. Try typing some of the names that came up. Some of my favorites are:
   - `tetris-bsd`
   - `moon-buggy`

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`
2. `ls /usr/local/games`
3. Try typing some of the names that came up. Some of my favorites are:
   - `tetris-bsd`
   - `moon-buggy`
   - `lander`

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`
2. `ls /usr/local/games`
3. Try typing some of the names that came up. Some of my favorites are:
   - `tetris-bsd`
   - `moon-buggy`
   - `lander`
   - `robots`

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`
2. `ls /usr/local/games`
3. Try typing some of the names that came up. Some of my favorites are:
   - `tetris-bsd`
   - `moon-buggy`
   - `lander`
   - `robots`
   - `fortune`

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`
2. `ls /usr/local/games`
3. Try typing some of the names that came up. Some of my favorites are:
   - `tetris-bsd`
   - `moon-buggy`
   - `lander`
   - `robots`
   - `fortune`
   - `adventure`

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`
2. `ls /usr/local/games`
3. Try typing some of the names that came up. Some of my favorites are:
   - `tetris-bsd`
   - `moon-buggy`
   - `lander`
   - `robots`
   - `fortune`
   - `adventure`
   - `zork`

Playing Games

Let's take a look at games installed on the system!

1. `ls /usr/games`
2. `ls /usr/local/games`
3. Try typing some of the names that came up. Some of my favorites are:
   - `tetris-bsd`
   - `moon-buggy`
   - `lander`
   - `robots`
   - `fortune`
   - `adventure`
   - `zork`
   - `nethack`

Homework For Next Time

Play around with the three major editors and pick which one you wish to use. (You can change your mind later.)

- **vi** - Run `vimtutor` from the command prompt to get the vim tutorial.
- **emacs** - Run `emacs` from the command prompt to run emacs.
  press “Control+h” followed by “t” to get to the tutorial
- **nano** - Just mess around. This is not an editor for serious programmers, but if you like it then you do you!