CS140U – Fundamentals Of Unix/Linux
Spring 2021, CRN: 40616, Credits: 4
Instructor: Joseph Jess
Web: http://cf.linnbenton.edu/bcs/cs/jessj/web.cfm?pgID=5404
email: jessj@linnbenton.edu

Initial class note: This is an introductory course to Linux operating systems and systems administration, though that does not mean every topic will be easy to understand or practice. I assume that you have some operating system usage experience. Expect to do a good bit of reading on the web, much practice in labs, and much discussion as a group of the topics we cover.

1. **LBCC catalog course description, including pre-requisites/co-requisites**:
    A laboratory-intensive course that provides new users with an introduction to the Linux operating system. Students will install and administer their own Linux systems, primarily using professional command-line tools. Topics will include file system navigation and permissions, text editors, shell scripting and network-oriented utilities. Provides partial preparation for the Linux+ exam.

    Prerequisite: MTH 075 Variables and Linear Equations and CIS 151 Networking Essentials, both with a grade of C or better.

2. **Class Time-space**:
    1.1 **Note: spring 2020 will be delivered completely remotely due to local and state rules.**
    2.1 Lecture + demo + lab: TR   1400 – 1620,   remote

3. **Measurable student learning outcomes:**
    (incorrect outcomes from the catalog) At the completion of the course, students will be able to:
    3.1 Write correct source code for creating and using data structures and abstract data types such as stacks, ques, lists, trees, hash tables and heaps.
    3.2 Analyze, evaluate, and write various searching and sorting algorithms using pseudo-code and source code.
    3.3 Describe programs and algorithms in terms of time complexity and space complexity.

     (Closer to what they should be...)
    3.4 Install, maintain, upgrade and administer from the command line a basic Linux server with user accounts and basic applications.
    3.5 Write command-line shell scripts to automate basic tasks in Linux system administration.
    3.6 Research a relatively advanced topic in Linux system administration and make a presentation to the class.

     (or perhaps)
    3.7 Identify and use Linux utilities to create, locate, and manage file processing in a security minded way.
    3.8 Use a command line interface to perform typical personal, office, software development, and systems administration tasks.
    3.9 Develop shell scripts or other programs to perform complex tasks.

4. **Learning resources:**
    4.1 **Note:** All class materials and storage will be freely available in a digital format
    4.2 (**recommended**) Access to a computer outside of class to practice and work on assignments (there are a few labs with suitable setups around on and off campus that I can point you to, just ask).
    4.3 A text editor. We could use a smart IDE to do this work as well!
    4.4 (**strongly recommended**) A desire to learn, experiment, design, test, and problem solve with code (both on and off of a computer).

5. **Grading:**
    5.1 Scores for coursework items will be initially available when the instructor gets to grading them... We will discuss this in class, including how to access it and keep yourself organized (which may affect your grade).

    5.2 Students will be required to turn in all coursework items before 23:59 (Pacific Time Zone) on the date that they are due (generally the first day of class each week in my courses).
        5.2.1   Students must be sure to give themselves plenty of time to submit coursework, as late work will not be accepted without prior consent or special circumstances.

5.3 To earn a passing grade in this course you must pass each of the following coursework categories:

    5.3.1   **Demonstration**: Discussion and weekly assignments – 50%

        5.3.1.1 There are a number of weekly assignments to be completed for this class, designed to introduce and reinforce skills related to our course topics.

            5.3.1.1.1   **Note**: careful design, systematic testing, consistent style, and readability of code and documentation are important software quality factors (all of which are subject to interpretation but graded by the instructor based on the spirit and letter of the requirements, so be sure to explain your decisions).

            5.3.1.1.2   **Note well**: Your submission needs to be understandable and able to function from just the files in your final submission. This means that you need to include any and all files necessary for your project to function, even if the instructor provided some of them.

            5.3.1.1.3   **Note very well**: Graded work must be designed and implemented by the student submitting the work and must function on one of the instructor's machines in order to get a final grade. (to create a working project quickly: get it working simply, then add to it; if at some point it stops functioning, then you will better know where an error was introduced)

    5.3.2   **Final**: Final project and presentation – 50%

        5.3.2.1 There will be a final project allowing students to demonstrate topics related to the class.

        5.3.2.2 Each person (or small group) will select a topic related to the class topics, a presentation of that topic will be made in class, each presentation will receive critique from both the instructor and other students, and students will be able to take these critiques into account to produce a final artifact supporting the topic and presentation (it may be a short paper, a poster, a video, or other supporting media related to the topic in question).

    5.3.3   Final grades will be given out based on the following based on score in the class:

    90-100%: A
    80-89%: B
    70-79%: C
    60-69%: D
    00-59%: F

5.4 Reminder: A passing grade in order to count for course requirements for CS classes is generally a C or above.

6. **Other Administrative Information:**

6.1 For a list of general administration information (note that this list is not intended to be exhaustive), such as:

    6.1.1   contacting me,
    6.1.2   accessibility resources,
    6.1.3   expectations of student conduct,
    6.1.4   communications,
    6.1.5   student assistance,
    6.1.6   miscellany,
    6.1.7   nondiscrimination & nonharrasment,
    (each section contains a number of sub-sections and is not meant to be exhaustive of all situations)

    see my administrative information document: *administrative_information* document.