

CIS 108 - Scripting for Systems Automation and Data Analysis

---

**COLLEGE:**

Merritt College

**ORIGINATOR:** Brown, Courtney**DIVISION/DEPARTMENT:**

Merritt - Division II/M - Technology

**STATE CONTROL NUMBER:** CCC000621581**DATES:****BOARD OF TRUSTEES APPROVAL DATE:** 12/14/2020**STATE APPROVAL DATE:** 01/13/2021**CURRICULUM COMMITTEE APPROVAL DATE:** 05/14/2020**REQUISITE VALIDATION:** 02/03/2018**CURRENT EFFECTIVE DATE:** 01/01/2021**1. REQUESTED CREDIT CLASSIFICATION:**

---

D - Credit - Degree Applicable

N - Not Basic Skills

1 - Program Applicable

**2. DEPT/COURSE NO:**

---

CIS 108

**3. COURSE TITLE:**

---

Scripting for Systems Automation and Data Analysis

**4. COURSE:**

---

**Course**

MC New Course w/DE Addendum

**TOP NO.** 0707.10 - Computer Programming\***5. UNITS:**

---

**Variable No****Units (Min)** 3.000**Min Total**

Hours

**Lecture Hours (Min)** 2.000

35

**Lab/Studio/Activity Hours (Min)** 3.000

52.5

## 6. SELECTED TOPIC:

---

**NO. OF TIMES OFFERED AS SELECTED TOPIC:**

**AVERAGE ENROLLMENT:**

## 7. JUSTIFICATION FOR COURSE:

---

Systems administration and analysis of data such as those found in system logs and data from scientific experiments require scripting skills. These skills integrate use of UNIX/Linux shells and its abilities to launch programs with languages such as Python which interfaces to many distinct libraries in different disciplines such as Biology, Chemistry, and Physics. These scripts solve problems in analysis of scientific experiments and orchestration of DevOps infrastructure through the control domain specific libraries such as Basic Linear Algebra Subprograms (BLAS) used across many mathematical and science domains, Biopython an open source library found in Genomic analysis, and DevOps Application Programming Interfaces (API) offered by Platform-as-a-Service providers like Amazon Web Services (AWS), Microsoft Azure, and IBM Blumix.

## 8. COURSE/CATALOG DESCRIPTION

---

Command line tools for automation and analysis: Integration of tools including UNIX/Linux shell, Python, and domain specific libraries for operations automation in DevOps and analysis in Data Science & Machine Learning; emphasis on acquisition of broadly applicable skills including files and correct modes of access use for distinct categories of files, file attributes, passing of variables, error handling, and exchange of result codes between application components.

## 9. OTHER CATALOG INFORMATION

---

**a. Modular:** No

**If yes, how many modules:**

**b. Open entry/open exit:** No

**c. Grading Policy:** Both Letter Grade or Pass/No Pass

**d. Eligible for credit by Exam:** No

**e. Repeatable according to state guidelines:** No

**f. Required for degree/certificate (specify):**

**g. Meets GE/Transfer requirements (specify):**

**h. C-ID Number:**

**Expiration Date:**

**i. Are there prerequisites/corequisites/recommended preparation for this course?** Yes

## 10. LIST STUDENT PERFORMANCE OBJECTIVES (EXIT SKILLS):

---

If an Objective cannot be deleted, make sure a Content-Review found in the Content Validation Page is not using that objective.

### Objectives

1. Use command redirection or substitution to capture program output
2. Use conditional statements to control the execution of shell scripts.
3. Write shell scripts to perform repetitive tasks using while and for loops.

## 11. COURSE CONTENT:

---

### LECTURE CONTENT:

- A. Review of Basic Shell Concepts (6.25%)
- B. Shell Programming (6.25%)
- C. Quoting (6.25%)
- D. Flow Control (6.25%)
- E. Loops (15%)
- F. Documentation (6.25%)
- G. Parameters (10%)
- H. Input/Output file descriptors and redirection in scripts (10%)
- I. Functions (15%)
- J. Debugging (6.25%)
- K. Function Libraries (6.25%)
- L. Techniques for increasing portability (6.25%)

### LAB CONTENT:

#### A) Basic Shell Concepts (20%)

1. Shell differences
2. Script basics
3. Working with files and directories
4. Using find to locate files and directories by name pattern and type
5. Manipulating file attributes: owners, groups and permissions.
6. Using text filters such as head, tail, cut, tr, sort, uniq, grep, sed and awk.
7. Basic regular expressions

#### B) Shell Programming (20%)

1. Script Basics
2. Variables
  - a. local
  - b. environment
3. Substitution
  - a. shell wildcards
  - b. variable
  - c. command, including backquote and \$(...) forms

d. arithmetic, including expr and \$((...)) forms

C) Parameters and Return Codes (10%)

- a. backslashes
- b. single quotes
- c. double quotes
- d. quoting rules

D) Flow Control (20%)

E) Quoting (10%)

- a. backslashes
- b. single quotes
- c. double quotes
- d. quoting rules

F) Input/Output (20%)

- a. standard input, standard output, and standard error
- b. output redirection, including: > 2> >> and 2>>
- c. input redirection, including < and <<
- d. combining output streams using: 2>&1 >&2
- e. using the read command to read data from files and interactively from the user

## 12. METHODS OF INSTRUCTION (List methods used to present course content):

---

- Lecture
- Lab
- Observation and Demonstration
- Discussion
- Critique
- Projects
- Work Experience
- Multimedia Content
- Threaded Discussions

**Other Methods:**

Interaction with cloud based resources.

## 13. ASSIGNMENTS

---

**Out-of-class Assignments (List all assignments, including library assignments. Requires two (2) hours of independent work outside of class for each unit/weekly lecture hour. Outside assignments are not required for lab-only courses, although they can be given.)**

**Override Outside Class Hours:** No

**Outside-of-Class Hours (Min)** 4.000

**Outside-of-Class Hours (Max)** 0.000

**Override Outside-of-Class Hours (Min)** 0.000

**Override Outside-of-Class Hours (Max)** 0.000

### **Out of class Assignment**

Creating shell scripts that use text filters to filter the output of Unix commands, analyze the results, and report findings to the user

Combining redirection, conditionals, and looping in a script.

Writing several functions and including them in a script

Writing scripts that use and analyze arguments to perform their tasks as do standard Unix/Linux commands.

## 14. STUDENT ASSESSMENT: (Grades are based on):

---

- ESSAY (Includes "blue book" exams and any written assignment of sufficient length and complexity to require students to select and organize ideas, to explain and support the ideas, and to demonstrate critical thinking skills.)
- COMPUTATION SKILLS
- SKILL DEMONSTRATION
- MULTIPLE CHOICE
- OTHER (Describe)

### **OTHER (Describe):**

Projects, presentations, demonstration.

## 15. TEXTS, READINGS, AND MATERIALS

---

A. Textbooks:

### **YesNo37**

Kochan, Stephen G. and Wood, Patrick. *Unix Shell Programming*. 3 edition SAMS, 2003.

Vanderplas, Jake. *Python Data Science Handbook: Essential Tools for Working with Data*. 1 edition O'Reilly, 2016.

Kavis, Michael J.. *Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS)*. 1 edition Wiley, 2014.

\*Date is required: Transfer institutions require current publication date(s) within 5 years of outline addition/update.

B. Additional Resources:

Library/LRC Materials and Services:

The instructor, in consultation with a librarian, has reviewed the materials and services of the College Library/LRC in the subject areas related to the proposed new/updated course

**Print Materials were reviewed?** Yes

**Non-Print Materials were reviewed?** No

**Online Materials were reviewed? Yes**

**Services were reviewed? Yes**

**Specific materials and/or services needed have been identified and discussed. Librarian comments:**

The library provides sufficient information resources in both print and electronic format to support this course. A librarian is available in person at the reference desk or online via chat to assist students whenever the library is open.

C. Readings listed in A and B above are: (See definition of college level):

**YesNo39**

Primarily college level

**16. DESIGNATE OCCUPATIONAL CODE:**

---

C - Clearly Occupational

**17. LEVEL BELOW TRANSFER:**

---

Y - Not applicable

**18. CALIFORNIA CLASSIFICATION CODE:**

---

Y - Credit Course

**19. NON CREDIT COURSE CATEGORY:**

---

Y - Not Applicable, Credit course

**20. FUNDING AGENCY CATEGORY:**

---

Y - Not Applicable (funding not used to develop course)

**REQUISITES AND ADVISORIES**

---

**RECOMMENDED PREPARATION:**

CIS 005 Introduction to Computer Science or CIS 006 Introduction to Computer Programming or CIS 007 Control Structures and Objects

**STUDENT LEARNING OUTCOMES**

---

1. **Design a data analysis or systems automation solution that uses functions.**

Assignment or project that requires a modular approach for successful implementation.

2. **Identify, select, and process command-line arguments.**

Assignment or project where key resources are provided as command line parameters.

