Canterbury Christ Church University

School of Engineering, Technology and Design

Undergraduate Computing Suite

2019-2020

Operating Systems (MCOMD3PST)

Date Set: 28-Oct-2019

Assignment 1

General guidelines for submission

- This is an individual submission and must be your own work.
- The required date of submission is **25-Nov-2019 before 14:00.** Scripts should be submitted via normal Blackboard, and the written part via Blackboard (Turnitin) – see below for details. **If you experience any problems with these systems, then please contact Andy Hook (**andy.hook@canterbury.ac.uk) **.**
- This assignment has been set by **Andy Hook**
- This assignment has been moderated by Dr Hannan Azhar.
- This assignment is worth **50%** of the individual course mark

General advice

- You are required to back up your work regularly, both on the campus network and on removable storage devices, and in any other suitable way. Always check the date stamp on your files before submission.
- You *must* submit your work using the versions of software currently in use on the University's network. This means that your scripts should run correctly on the student UNIX system, and that your report should be either a PDF file, or prepared using a compatible version of Microsoft Word.
- Please note that a high standard of presentation is expected. Reports should have an appropriate title page and contents page, and pages should be numbered.
- The title page should have the following information:
    - Module code
    - Module name
    - Assignment number and title
    - Lecturer's name
    - Student's name
    - Submission date
- You are required to use the originality checking feature available in Turnitin to help improve your academic report writing skill.
- You are advised to perform the originality check at least 2 days before the hand-in date to give you sufficient time to benefit from the feedback you receive from Turnitin. You can submit your work to Turnitin as many times as you wish, but only the most recent submission will be marked.

Before you start

You will need to do your research in order to complete the assignment tasks to the correct level. This will involve looking at common shell commands and some more advanced aspects. It is suggested that you look at directory creation, how to accept arguments in a script, the manipulation of data from text or CSV files, writing to files and process management.

Learning outcomes tested by this assignment

- Demonstrate an advanced understanding of process/thread creation, process/thread management and scheduling and process/thread communication.
- Write effective scripts and/or relevant API calls to perform basic administrative tasks and/or construct simulations.

The task

You are working as a system administrator in a start-up company that is experiencing a rapid period of growth. As such, they are recruiting lots of new staff, and your manager has tasked you with speeding up the process of user account creation.

The HR department has provided you with a CSV file of new employees (on Blackboard) who need accounts creating on the system.

Your task is to write some shell scripts to provide the following:

1. An entry point to your program (command wrapper) that will enable the user to perform the following tasks:
   1.1. Provision of simple help
   1.2. Data **extraction** and **transformation**
   1.3. User account creation
   1.4. Process management
2. Logging of the date and time (and the username) whenever each action is invoked

This should all be as "bullet proof" as possible, so you should test your scripts. Document what you have done in your report.

Note: seeing as you will not be able to create **real** user accounts on the system, the extent of user account creation will be the provision of a directory based on the users' name. It is suggested that these user folders are created in a "users" directory. Consider placing your scripts and the "users" directory inside a parent folder to keep your work separate from anything else you have done on the student-unix system.

In more detail

The program entry point (wrapper) [14 marks]

As you are required to provide multiple features in your program, the user must have an entry point into the program where they can specify the action they wish to use. You will need to figure out an appropriate way to do this. For example, the user should be able to run a command such as **userman import <filename>** or **userman list**.

Remember that you will need to use appropriate encapsulation and separation of your program logic in order to facilitate all parts of this task. You must make sure that you have appropriate error handling and exception reporting along with the use of exit codes.

No matter the action the user takes, program execution should not be blocked. For example, when you simulate the long-running user account creation process, the main entry point to your program should exit and those account creation processes must run in the background.

Help [2 marks]

If the wrapper script is invoked with no arguments, it should output a short help message to standard error, then exit with an exit code indicating failure.

Data extraction and transformation [18 marks]

The input file will be a CSV file with the following fields (one record per line):

Name,Email,Department,Manager

You will need to use appropriate extraction and transformation techniques ready for the next part of the task. This will include basic data validation and you should handle and log any exceptions that occur.

A file of test data can be found on Blackboard.

User account creation [22 marks]

Using the data from the previous task, you must create user account directories as per the indicative mark scheme (below). You will need to scaffold a user directory for each user in the CSV file and they should all have at least the following directories:

- Documents
- Pictures
- Videos

You should also generate a "Welcome.txt" file for each user in their Documents directory that has the following content:

> Welcome to the business, <NAME>.
>
> We are pleased to have you working in the <DEPARTMENT> department.
>
> If you have any questions, please speak to your manager, <MANAGER>
>
> Your email address is <EMAIL>

It is also necessary for you to simulate this process as a long-running task in order for you to be able to complete the next task.

Process management [10 marks]

For this part of the task you will need to provide three process management options to the user that enables them to display a list of current account creation processes (as you will have simulated them to be long-running). They should also be able to kill all of these processes at one and export a list of currently running processes to a file name "export.txt".

Logging [14 marks]

You will need to provide a log file to report the results of at least the following:

- Successful user account creation
- User account already exists
- When the option to ‘Kill’ all account creation processes is used

The log files should be in their own directory and the file name should be the current date in the format specified in the mark scheme.

Each entry in the log file should be timestamped and provide the users username (which you will have formed), where applicable. You should not overwrite any log files, so whenever they are written to, ensure the entries are added to the file.

Write Up [20 marks]

In addition to the scripts, you are required to provide a short informal write up (800 words) describing how key aspects of code meet the requirements for this task.

Assessment

You will be assessed on your ability to present clear explanations and precise descriptions. The written work must show your understanding of the subject matter and be Harvard referenced throughout.

This report will be marked in accordance with the Level 6 Generic Assessment Criteria for Undergraduates available at:

http://www.canterbury.ac.uk/quality-and-standards-office/assessment-criteria.aspx

Deliverables and submission

There are two separate deliverables for this assessment. You *must* submit both.

1. Upload a **zipped** version of your scripts to Blackboard (*not* the Turnitin part for Assignment 1). Include copies of your generated directory structure and any output files, as well as a sample log file. Details will be provided in the same location where you downloaded this assignment.
2. Upload your write up onto Blackboard (Turnitin); a link will be provided in the same location where you downloaded this assignment.

A separate document will be provided in the same place as this one to provide assistance and details of file transfer between UNIX and Windows.

Failure to provide an adequate write up will result in SCALING of code marks.

Indicative mark scheme

Task	Mark	Subtotal
Program entry point (wrapper)		
Decode and interpret arguments and trigger associated action	4	
Encapsulation and separation of program logic	5	
Error handling	2	
Case independence of action keywords	2	
Exit correctly, with valid exit code	1	
		14
Help		
Help message printed to standard error	2	
		2
Data extraction and transformation		
Any file name as an argument for the input file	4	
Check for existence of input file	2	
Correct extraction of fields	6	
Basic data validation, handling and logging of exceptions	6	
		18
User account creation		
User directories created with skeleton structure	4	
User directories named consistently	2	
Log exception if user account already exists	2	
n users with the same name can have an account	6	
Welcome file with user’s details inside their Documents directory	2	
Simulation of account creation taking a long time (e.g. 30 secs)	2	
Account creation jobs should run in background	2	
Write to log when account has been created successfully	2	
		22
Process management		
Display a list of current account creation processes	4	
Export a list of current account creation processes to a text file	2	
Kill all current account creation processes	4	
		10
Logging		
New log file to be used each day (format: YYYY-MM-DD)	4	
Event timestamp in the format: [12/10/2019 14:20:00]	4	
Name of user affected (where applicable)	2	
Log file should be appended to, not overwritten	4	
		14

Write up		
800 word write up	20	
		20
TOTAL		100