Fall 23 Final Exam CIS106

Rules:

1. You are allowed to use your notes, labs and the content of the cis106.com website including the presentations.

2. You are not allowed to use Google or ChatGPT

3. If you use any command that was not covered during the lectures you will have to provided a detailed explanation and be available to answer questions, otherwise, you will not be granted points for the given question.

4. You are allowed to ask questions during the exam, the instructor reserves the right to answer the question

5. This is a performance base exam which means that everything you do will be graded. The only way to get 0 points in a question is by leaving it blank.

6. All the files for the exam are located in a github repository and every question is independent from the other so in theory you can complete the exam in any order you choose.

7. If you completed the final assignment, you have already earned 20 points towards this exam.

8. Here is the value of every question:
    a. Final Assignment: 20 Points
    b. Pre Work: 10 Points
    c. Question 1: 10 Points
    d. Question 2: 20 Points
    e. Question 3: 20 Points
    f. Question 4: 20 Points

9. Pre work has to be completed before you start the exam. All your screenshots will be placed in this directory.

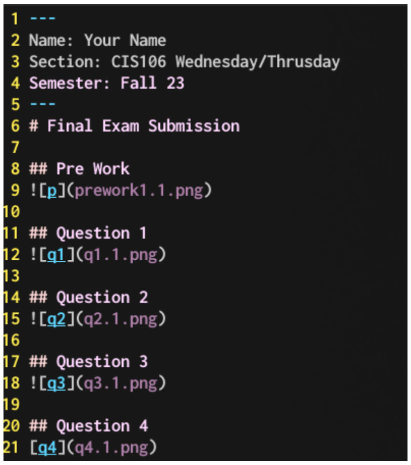
    i. Once you complete this exam, please open the directory in vs code and convert the file to pdf.

    ii. Doublecheck to make sure all the images load correctly. Take the last 10 minutes of the exam to do this.

    iii. **YOU ARE NOT ALLOWED TO UPLOAD THIS EXAM TO GITHUB!** So do not copy the files to your repository. There will be a severe penalty for violating this rule.

Pre-work

1. In your home directory, create a directory called **`finalSubmission`**.
2. Inside the **`finalExamSubmission`** directory, create a file named **`first_name_last_name_final_submission.md`**. For example, my file would be named **`robert_alberto_final_submission.md`**.
3. Open the file **`first_name_last_name_final_submission.md`** in one of the following text editors:
    a. Vim
    b. Nano
    c. Text Editor
    d. Geany
    e. ***Note****: You cannot open or edit this file in VS Code. You will use vs code when you are done with the exam*!
4. Add the following text to the file (see image below). The naming convention that you will use for every screenshot is **q#.1.png**. This will make it easy to type and save you time.



```
---
Name: Your Name
Section: CIS106 Wednesday/Thrusday
Semester: Fall 23
---
# Final Exam Submission

## Pre Work
![p](prework1.1.png)

## Question 1
![q1](q1.1.png)

## Question 2
![q2](q2.1.png)

## Question 3
![q3](q3.1.png)

## Question 4
[q4](q4.1.png)
```

5. Save the file, clear your terminal, and make sure that your present working directory is your home (~).
6. Display a tree of the **`finalExamSubmission`** directory
7. Display the content of the file **`first_name_last_name_final_submission.md`**.
8. on the give question. **The repository must be cloned in your home directory**.All the files for the exam are located in the GitHub repository referenced in question 1, so if you choose to not work the exam in order, then make sure to clone the repository before you start working

Take a screenshot of the commands you use to complete this question. You will receive full credit as long as the following commands are in the screenshot.

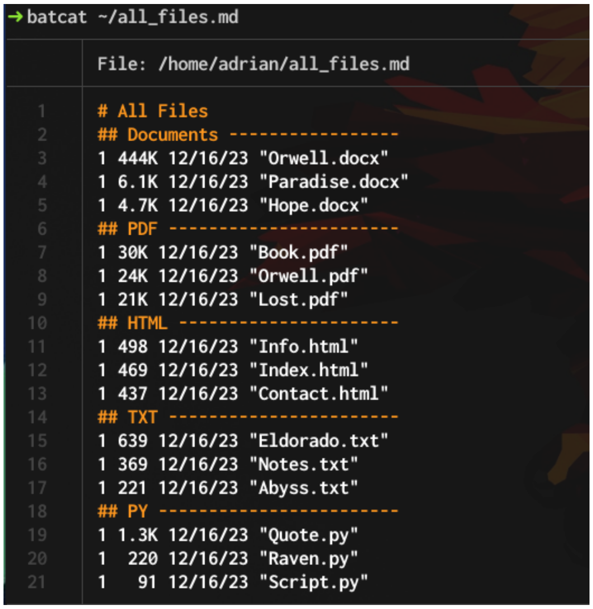
- *The commands you used for creating the directory and file*
- *The commands you used for displaying the content of the directory and file*

Question 1

Tips:

- The formatting string for this is +%D
- There are no options in the list commands for excluding the permissions of a long list
- You will need I/O redirection
- You will need wildcards
- The man page of the ls command describes how to use the time format.
- You don't need to install any packages to complete this question.

For this question, you will create a markdown file that looks like this ->



```
➜ batcat ~/all_files.md

       File: /home/adrian/all_files.md

       # All Files
       ## Documents -----------------
       1 444K 12/16/23 "Orwell.docx"
       1 6.1K 12/16/23 "Paradise.docx"
       1 4.7K 12/16/23 "Hope.docx"
       ## PDF -----------------------
       1 30K 12/16/23 "Book.pdf"
       1 24K 12/16/23 "Orwell.pdf"
       1 21K 12/16/23 "Lost.pdf"
       ## HTML ----------------------
       1 498 12/16/23 "Info.html"
       1 469 12/16/23 "Index.html"
       1 437 12/16/23 "Contact.html"
       ## TXT -----------------------
       1 639 12/16/23 "Eldorado.txt"
       1 369 12/16/23 "Notes.txt"
       1 221 12/16/23 "Abyss.txt"
       ## PY ------------------------
       1 1.3K 12/16/23 "Quote.py"
       1  220 12/16/23 "Raven.py"
       1   91 12/16/23 "Script.py"
```

The command I used in the screenshot is **`batcat`**; an improved version of **`cat`** that can be installed with `` `**sudo apt install bat**` ``. Again, you are not required to install this program so don't waste time.

Complete the following tasks to create the file:

1. Clone the GitHub repository: **`https://github.com/linuxworkshop67/finalFiles`**. *Make sure to clone this repo from your home directory.*

2. In the **`finalFiles`** directory, you will find a subdirectory called **`question1`**. The question1 directory contains all the files for generating the markdown file.

3. Long list all the docx files in the **`question1`** directory, **excluding the owner and group**, **with human-readable file sizes**, **quoted file names**, **and the date formatted as mm/dd/yy** . Also, remove the permissions from the output. Save the output of the long list to a file in your home directory called **`all_files.md`**

4. Repeat the long list command for the other file types and append the output to **`all_files.md`**

5. Open the file **`all_files.md`** in Vim or Nano and add the markdown formatting you see in the screenshot. Save and quit the text editor.

6. Display the content of the file **`all_files.md`**.

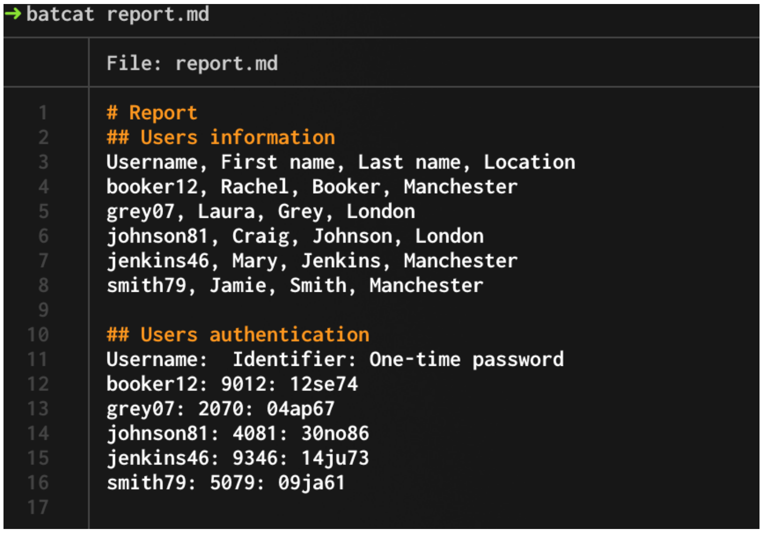
Take a screenshot of the commands you use to complete this question. You will receive full credit as long as the following commands are in the screenshot.

- *Long list commands*
- *Output redirection and append*
- *The content of your file contains all the elements of the screenshot*

Question 2

Scenario-based question

Your boss has provided you with a CSV file containing some user's information. The file can be found in the `question2` directory under the `finalFiles` directory. Your task is to create a markdown file that looks like this:



```
→batcat report.md
        File: report.md

        # Report
        ## Users information
        Username, First name, Last name, Location
        booker12, Rachel, Booker, Manchester
        grey07, Laura, Grey, London
        johnson81, Craig, Johnson, London
        jenkins46, Mary, Jenkins, Manchester
        smith79, Jamie, Smith, Manchester

        ## Users authentication
        Username:  Identifier: One-time password
        booker12: 9012: 12se74
        grey07: 2070: 04ap67
        johnson81: 4081: 30no86
        jenkins46: 9346: 14ju73
        smith79: 5079: 09ja61

```

1. Replace all the semi-colons (;) with commas (,). Save the output to a file called `employes_v1.csv`

2. Using the `employes_v1.csv` file create another file containing the columns **Username** and **One-time password** and **recovery code** delimited by ': ' (including a space after the colon). This file should be called `users_auth.csv`

3. Using the `employes_v1.csv` file create another file containing the columns, **username**, **first name**, **last name**, and **location**. The delimiter for this file should be ', ' (a space after the comma). This file should be called `users_info.csv`

4. Create a markdown file that includes the content of `users_info.csv` and `users_auth.csv` in that order. Add the headings as seen in the image using nano or vim. Save the file as `report.md`

5. Using the command line, find the answers to the following questions:
    a. How many lines are there in the file `report.md`
    b. What is the size of the file `report.md`

Take a screenshot of the commands you use to complete this question. You will receive full credit as long as the following commands are in the screenshot.

- *The command you used to create the file employes_v1.csv, users_auth.csv, users_info.csv and report.md*
- *The commands to answer part 5*

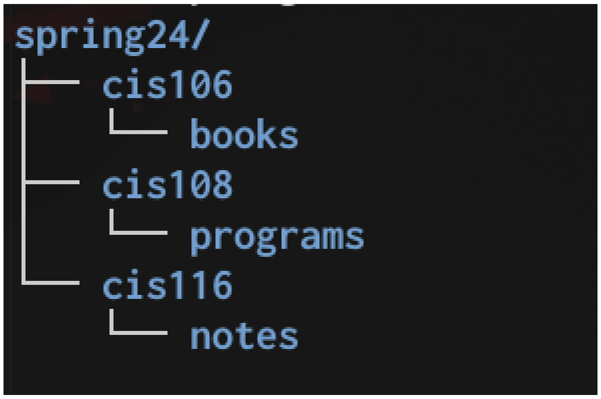
Question 3

Scenario based question

Mario is a network administration student. His friend Ash gave him some files for the next semester and he needs to copy them to a new folder in his **`Documents`** directory. Mario is very organized and for every course he takes every semester creates a directory structure. Your task here is to create the directory structure and copy the files to their respective directory. You will find all the files in the **`question3`** subdirectory of the **`finalFiles`** directory.

Follow the following instructions to complete this question:

1. Change your present working directory to the root of the file system. You will not be allowed to use cd again for the remaining of this question. *Remember the root of the filesystem is not the same as your home directory!*
2. In your **Documents** directory, create the following directory structure **in a single command** and **without using the cd command:**



```
spring24/
├── cis106
│   └── books
├── cis108
│   └── programs
└── cis116
    └── notes
```

3. Copy the files you have in the **`question3`** directory to their respective **`spring24`** subdirectory. You must copy all the files related to each subject in one command using a wildcard.
4. Display a tree of the **`spring24`** directory.

Take a screenshot of the commands you use to complete this question. You will receive full credit as long as the following commands are in the screenshot.

- *Every command should be run from the root of the filesystem*
- *The command used to create the directory structure*
- *The commands used to copy the files with the wildcard*
- *The tree of the spring24 directory*

Question 4

Tips:

- Inspecting a file means looking inside the file using a command that can search for words in every line of a given file.
- Do not get intimidated by the HTML. You are looking for a path that shows you which folders you need to make.

Scenario Description:

Fred is a first-year web development student. He downloaded an example website, but when he opened it in Firefox, he noticed the page looked broken. All the assets (images, scripts, etc.) are not in the right place. Help Fred fix this site. You do not need to know HTML to do this. Also, you don't need to modify the HTML. Inspect the HTML file so that you can place each file in its respective directory.

Instructions to complete

1. Download the broken site by cloning this repository (make sure to clone the repository in your home directory!): `https://github.com/linuxworkshop67/broken_website`
    a. Alternatively the question4 directory contains all the files of the broke_website repository so you can use the question4 directory if you want.
2. Inspect the **`index.html`** file to find the directory structure you need to create to fix the site. Here is what you need to know about HTML and CSS.
    a. The location of files is referenced with the attribute `src="`**`file/location/here`**`"` or with the `href="`**`file/location/here`**`"` or with `url("`**`../location/of/images`**`")`. You can ignore the ../ in the url().
2. Create the necessary directories in a single command.
3. Move all the files to their respective directories using a wildcard when needed
4. Open the website in Firefox. You can use the command **`firefox index.html &`**

Take a screenshot of the commands you use to complete this question. You will receive full credit as long as the following commands are in the screenshot.

- *The command you used to get the repository in your computer. Ignore this if you use the question4 directory*
- *The commands you used to inspect the website files*
- *The command that you used to create the necessary directories*
- *The commands that you used to move the files to their respective locations*

Good Luck