

Lab 07 Specification – Exploring Scripting Languages
Due (via your git repo) no later than 8 a.m., Tuesday, 4th December 2018.
50 points

Lab Goals

The goal for this Lab, is to explore Scripting languages such as Bash and Javascript and to take a look at some fundamental concepts in Scripting.

Assignment

- Part 1: Create a bash script file, that would generate a statistics report of an user defined directory. Let us suppose that the user provides a directory [along with the directory path, for example: /home/a/amohan/tmp], then your bash code should first fetch all the items [files and folders] in the folder. You may store this information into a list, so that it is useful to perform additional processing.

The directory path is provided as an argument during bash script execution. In linux, the following command is used to list all the items in a directory:

```
ls -l // list all items [files and folders] within a given directory.
```

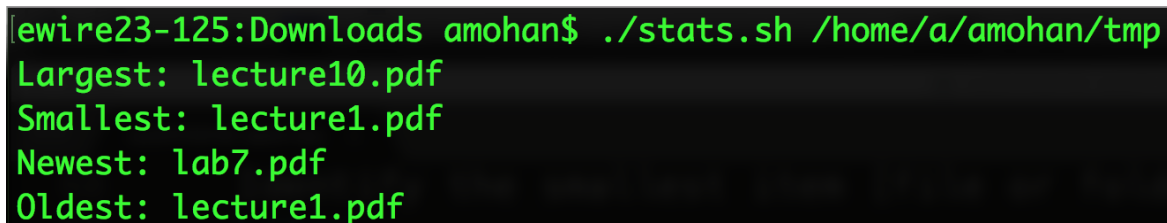
After storing the item list, you may use an iterative approach such as [Bash loops], to further process the data. Implement the following tasks within your Bash script:

1. Identify the largest item [file or folder] in the user defined directory.
2. Identify the smallest item [file or folder] in the user defined directory.
3. Identify the newest [most recent] item [file or folder] in the user defined directory.
4. Identify the oldest item [file or folder] in the user defined directory.

I had run the script file on my machine, on a directory that contains 128 items as follows:

```
./stats.sh /home/a/amohan/tmp
```

A sample statistics report is shown below:



```
ewire23-125:Downloads amohan$ ./stats.sh /home/a/amohan/tmp
Largest: lecture10.pdf
Smallest: lecture1.pdf
Newest: lab7.pdf
Oldest: lecture1.pdf
```

- Part 2: Create a HTML page with one textbox, and ask the user to type in a random word (based on user choice) using the textbox. In the HTML page, create a button called Validate, that would validate if the given word is a palindrome or not. In your implementation, the button ("Validate") click event should be binded to a Javascript function, that would validate if the given string is a palindrome. The binding may be created similar to the [11/26/2018] class activity, where the class had implemented a Password Validation functionality. It is highly recommended to use "table", "tr", and "td" HTML elements in your page, so that you are able to match the sample web page design shown below.

In your Javascript function, you may implement the palindrom check using the algorithm below:

- 1) Store the word (text box 01) into a variable "firstStr"
- 2) Reverse the value in "firstStr" and store it into a variable "secondStr"
- 3) Compare "firstStr" and "secondStr" for equality and alert if the string is a palindrome or not based on If/Else conditional check.

A sample webpage design is shown below:

Enter your word:

- Part 3: Create a HTML page to calculate the BMI value and indicate the BMI based risk level to your user. A sample webpage design is shown below for your reference:

Enter your height:

Enter your weight:

Height can be entered using any one of the following formats: inches, centimeters, and meters. In the sample design, "select" and "option" HTML elements are used to implement.

Weight can be entered using any one of the following formats: pounds, kilograms. In the sample design, "select" and "option" HTML elements are used to implement.

The formula for the BMI calculation may be found online. One such reliable web source to get the formula is provided below:

https://www.cdc.gov/healthyweight/assessing/bmi/childrens_bmi/childrens_bmi_formula.html

After computing the BMI score, use the following rules to display the corresponding message related to the BMI Risk Level to the user. The message can either be displayed through a hidden field's innerHTML and/or through an alert message. It is highly recommended to use a hidden field in HTML and through Javascript function activate the content of the hidden field through its inner HTML. By doing this approach, the message displayed is nicely integrated inside the webpage.

```
if (output < 18.5)
    Display "Underweight";
else if (output >= 18.5 && output <= 25)
    Display "Normal";
else if (output >= 25 && output <= 30)
    Display "Obese";
else if (output > 30)
    Display "Overweight"
```

Required Deliverables

Please submit electronic versions of the following deliverables to your GitHub repository by the due date:

1. Properly completed and commented source programs for Part 1, 2, and 3.
2. A Read me file, that states clearly how to run the program and a self reflection section that includes either the list of challenges faced and/or list of things liked and disliked in this lab.

Grading Rubric

1. If you complete [Part 1] completely as per the requirement outlined above, you will receive 20 points. There is 5 points awarded for each part.
2. If you complete [Part 2] completely as per the requirement outlined above, you will receive 15 points. There is 10 points awarded for each part.
3. If you complete [Part 3] completely as per the requirement outlined above, you will receive 15 points. There is 10 points awarded for each part.
4. If you fail to upload the lab solution file to your git repo by the due date, there will be no points awarded for your submission towards this lab assignment. Late submissions will be accepted based on the late submission policy described in the course syllabus. It is the student's responsibility to communicate to the professor if it is a late submission. If the student had not communicated in advance about the late submission, the lab work shall not be graded as such.
5. Partial credit will be awarded, based on the work demonstrated in the lab submission file.
6. If you needed any clarification on your lab grade, talk to the Professor. The lab grade may be changed if deemed appropriate.

