AACS2284 Operating Systems

**Practical 5: Working with the BASH Shell and Shell Scripts**

1. Which of the following files is always executed immediately after a user logs in to a Linux system and receives a BASH shell?

a. /etc/profile b. ~/.bash\_profile

c. ~/.bash\_login d. ~/.profile

Answer: a. /etc/profile



2. Which of the following will display the message **welcome home** if the **cd /home/user1** command is successfully executed?

a. **cd /home/user1 && echo “welcome home”**

b. **cat “welcome home” || cd /home/user1**

c. **cd /home/user1 || cat “welcome home”**

d. **echo “welcome home” && cd /home/user1**

Answer: a. **cd /home/user1 && echo “welcome home”**

3. What would be the effect of using the **alias** command to make an alias for the **date** command named **cat** in honor of your favorite pet?

a. It cannot be done as there already is an environment variable **cat** associated with the **cat** command.

b. It cannot be done as there already is a command **cat** on the system.

c. When you use the **cat** command at the command prompt with the intention of viewing a text file, the date appears instead.

d. There is no effect until the alias is imported as it is a user-declared variable.

Answer: c. When you use the **cat** command at the command prompt with the intention of viewing a text file, the date appears instead.

4. Consider the following shell script:

**echo -e “What is your favorite color?--> \c”**

**read REPLY**

**if [ “$REPLY” = “red” –o “$REPLY” = “blue” ]**

**then**

**echo “The answer is red or blue.”**

**else**

**echo “The answer is not red nor blue.”**

**fi**

What would be displayed if a user executes this program and answered **Blue** when prompted? a. The answer is red or blue.

b. The answer is not red nor blue.

c. The code would cause an error.

d. The answer is red or blue. The answer is not red nor blue.

Answer: b. The answer is not red nor blue

5. Using sort as a filter, rewrite the following sequence of commands:

**sort list > temp**

**lpr temp**

**rm temp**

|  |
| --- |
| cat list | sort>temp | lpr && rm temp |

AACS2284 Operating Systems

6. What would happen if the user executed the following commands? Explain the output. (i) **file /usr/bin/\* | grep "Again shell script" | sort -r**

|  |
| --- |
| **file /usr/bin/\*** = to display a list of files within the /usr/bin directory.    **file /usr/bin/\* | grep “Again shell script”** = to display a list of files within the /urs/bin directory, then filter those files with keyword “Again shell script”.    **file /usr/bin/\* | grep”Again shell script” | sort -r** = to display a list files within the /urs/bin directory, then filter those file with keyword “Again shell script”, then sort the output in reverse order |

(ii) **tr a A < /etc/hosts | sort -r | pr -d > /etc/hosts**

|  |
| --- |
| **tr a A < /etc/hosts** = replace all “a” with “A” in /etc/hosts file.    **tr a A < /etc/hosts | sort -r** = replace all “a” with “A” in /etc/hosts file, then sort the output in reverse order.    **tr a A < /etc/hosts | sort -r | pr -d > /etc/hosts =** replace all “a” with “A” in /etc/hosts file, then sort the output in reverse order, then print the output in double-spacing into /etc/hosts. |

7. (i) Create and export a variable called **newhome** that is equivalent to the value contained in the **HOME** variable.

|  |
| --- |
| Export newhome = “$HOME” |

(ii) Find all files that start with the word “host” starting from /etc directory and save the stdout to a file called **file1** and the stderr to the same file.

|  |
| --- |
| Find /etc-name “host”>file2>&1 |

(iii) Display only the lines from the output of the set command that have the word “bash” in them. This output on the terminal screen should be sorted alphabetically.

|  |
| --- |
| Set | grep “bash” | sort |

8. Write the expression that can be used to test whether:

a. the user has read and execute permission to the /etc directory

a)

|  |
| --- |
| [-r/etc/hosts -a -x/etc/hosts] |

b. the contents of the variable **$TEST** are equal to the string “success” and the file /etc/hosts exists

|  |
| --- |
| [$TEST= “success” -a -f /etc/hosts] |

c. the contents of the variable **$TEST** are equal to the string “success, or the number 5 or the contents of the variable **$RESULT**

|  |
| --- |
| [“TEST” = “success” -o “$TEST” = “5” -o &“TEST” = “$RESULT”] |

2

AACS2284 Operating Systems

9. Write a shell script that performs the following:

● Displays a list of currently logged-in users

● Display the system’s host name

● Display the time and date

● Displays the disk usage

● Display the pathname of the BASH shell

|  |
| --- |
| #!/bin/bash  echo “Currently logged-in users:”  who  echo “The system’s hostname is:”  echo $HOSTNAME  echo “The current date and time is:”  date  echo “Current disk usage is:”  df  echo “The pathname to the bash shell:”  echo $BASH |

10. Write a script to display the time every 15 seconds. Read the date man page and display the time, using the **%r** field descriptor. Clear the window (using the clear command) each time before you display the time.

|  |
| --- |
| #!/bin/bash  while (true)  do  sleep 15  clear  date +%r  done |

11. Write a shell script to accept basic pay of an employee. Display the following details in a pay list Pay slip details, *House rent allowance (HRA)*, *Dearness allowance (DA)*, *Provident fund (PF)*. HRA is to be calculated at the rate of 15% of basic, DA at the rate of 30% of basic and PF at the rate of 10% of basic.

.

|  |
| --- |
| #!/bin/bash  proceed="y";  while [ $continue="y" ]  do  clear  echo "Please enter your basic pay (RM): ";  read basic  echo -e "\n\n";  echo -e "PAY SLIP DETAILS\n";  echo -e "1. House Rent Allowance (HRA)\n";  echo -e "2. Dearness Allowance (DA)\n";  echo -e "3. Provident Fund (PF)\n";  read choice  echo -e "\n\n";  case $choice in  1)hra=`expr $basic \\* 15 / 100`  echo "House Rent Allowance: RM $hra";;  2)da=`expr $basic \\* 30 / 100`  echo "Dearness Allowance: RM $da";;  3)pf=`expr $basic \\* 10 / 100`  echo "Provident Fund: RM $pf";;  esac  echo -e "\n\n";  echo -e "Do you want to continue?";  read continue  if [ $continue != "y" ]; then  exit  fi  done  #!/bin/bash  echo -n “Enter basic pay of an employee: “  read pay  hra=`bc <<< $pay\*0.15`  da=`bc <<< $pay\*0.3`  pf=`bc <<< $pay\*0.1`  echo “House rent allowance: $hra”  echo “Dearness allowance: $da”  echo “Provident fund: $pf” |

3

AACS2284 Operating Systems

**Extra exercise (Optional)**

Create a shell script named “myscript” to check whether the directory name inserted by user exists beforehand under the user’s home directory.

If the directory does not exist, the system will prompt the message “Folder does not exist”. (Refer to Figure 1)

If the directory exists, the system will prompt the message “Folder *<directory\_name>* exists” and the system will list/display all the contents inside such directory in a single column format. Then, the system will remove such directory together with all its content. (Refer to Figure 2)

**Sample Output:**

|  |
| --- |
| If the user types myscript in the terminal ***with parameter*** e.g. ***myscript dir1*** and the ***dir1 directory does not exist beforehand inside the user’s home directory***, the following output should be shown: |

**Figure 1**

|  |
| --- |
| If the user types myscript in the command prompt ***with parameter*** e.g. ***myscript dir1*** and the ***dir1 directory already exists beforehand inside the user’s home directory*** , the following output should be shown: |

**Figure 2**

4

AACS2284 Operating Systems

**Answer:**

|  |
| --- |
| #!/bin/bash  if [ ! -d $1 ]  then  echo "Folder does not exist"  else  echo "Folder $1 exists"  echo "The content in $1 folder are listed below"  echo "========================================================================="  ls -1 $1/\*  echo "========================================================================="  echo "START REMOVING $1 FOLDER AND ALL THE CONTENT INSIDE IT"  rm -r $1  fi |

5