**Name : Nteile Israel**

**Number: ALT/SOE/023/2691**

**Assignment 02**

Question 1: Write a script that check disk usage in a given directory. The script can take two optional argument and one compulsory argument.

-d: which means that all the files and directory within the specified directories should be listed.

-n : which means that the top N entries should be returned.

List of directories: this will be the directories you want to check its disk usage.

E.g. yourscript.sh -n 5 /var should return the top 5 directories wrt to the /var directory.

**Code - code of diskuse.sh script**

**vagrant@ubuntu-focal:~$ cat diskuse.sh**

**#!/usr/bin/bash**

**#function to display usage information**

**option(){**

**echo "option: $0 [-d] [-n N] directory"**

**echo "options:"**

**echo " -d: List all files and directories within the specified directory"**

**echo " -n N: Return the top N entries (default is 8)"**

**exit 1**

**}**

**#Default value for top N entries**

**N=8**

**#parse options**

**while getopts ":dn:" opt; do**

**case ${opt} in**

**d)**

**listfiles=true**

**;;**

**n)**

**N=$OPTARG**

**;;**

**\?)**

**echo "Invalid option: -$OPTARG" 1>&2**

**option**

**;;**

**esac**

**done**

**shift $((OPTIND -1))**

**#check if directory argument is provided**

**if [ -z "$1" ]; then**

**echo "Error: Directory path is missing."**

**option**

**fi**

**#check if directory exist**

**if [ ! -d "$1" ]; then**

**echo "Error: $1 does not exist."**

**exit 1**

**fi**

**#list directories and files based on options**

**if [ "$listfiles" = true ]; then**

**du -ah "$1" | sort -rh | head -n "$N"**

**else**

**du -h "$1" | sort -rh | head -n "$N"**

**fi**

**Implementation of Code...**

**vagrant@ubuntu-focal:~$ pwd**

**/home/vagrant**

**vagrant@ubuntu-focal:~$ cp ./diskusage.sh ./home**

**cp: cannot stat './diskusage.sh': No such file or directory**

**vagrant@ubuntu-focal:~$ nano diskuse.sh**

**vagrant@ubuntu-focal:~$ vagrant@ubuntu-focal:~$**

**vagrant@ubuntu-focal:~$**

**vagrant@ubuntu-focal:~$ chmod +x ./diskuse.sh**

**vagrant@ubuntu-focal:~$**

**vagrant@ubuntu-focal:~$ ./diskuse.sh -n 5 Doc1**

**212K Doc1**

**196K Doc1/.git**

**68K Doc1/.git/objects**

**56K Doc1/.git/hooks**

**20K Doc1/.git/logs**

**vagrant@ubuntu-focal:~$ ./diskuse.sh -n 5 Doc2**

**28K Doc2**

**8.0K Doc2/Doc2A**

**4.0K Doc2/Doc2B**

**vagrant@ubuntu-focal:~$ ./diskuse.sh -d Doc2**

**28K Doc2**

**8.0K Doc2/Doc2A**

**4.0K Doc2/backup.sh**

**4.0K Doc2/Doc2B**

**4.0K Doc2/Doc2A/D1.txt**

**4.0K Doc2/B2.txt**

**4.0K Doc2/B1.txt**

**0 Doc2/B3.txt**

**vagrant@ubuntu-focal:~$ ./diskuse.sh -d Doc1**

**212K Doc1**

**196K Doc1/.git**

**68K Doc1/.git/objects**

**56K Doc1/.git/hooks**

**20K Doc1/.git/logs**

**16K Doc1/.git/refs**

**12K Doc1/.git/logs/refs**

**8.0K Doc1/.git/refs/heads**

**vagrant@ubuntu-focal:~$**

**Question 2**

Create a backup script. This script creates a backup of a given directory and saves it in another directory with a timestamp. It takes two arguments: the source directory and the destination directory. Note the backup should be a tar archive.

**The Script**

**vagrant@ubuntu-focal:~$ cat backup.sh**

**#!/usr/bin/bash**

**#check if both source and destination directories are provided**

**if [ $# -ne 2 ]; then**

**echo "usage: $0 <source\_directory> <destination\_directory>"**

**exit 1**

**fi**

**#Assigning arguments to variables**

**source\_dir=$1**

**destination\_dir=$2**

**#Check if source directory exist**

**if [ ! -d "$source\_dir" ]; then**

**echo "source directory does not exist"**

**exit 1**

**fi**

**#check if destination directory exist, if not create it**

**if [ ! -d "$destination\_dir" ]; then**

**mkdir -p "$destination\_dir"**

**fi**

**#Creating tar archive with timestamp**

**timestamp=$(date +%Y%m%d\_%H%M%S)**

**backup\_file="$destination\_dir/backup\_$timestamp.tar.gz"**

**#Creating a backup**

**tar -czf "backup\_file" "$source\_dir"**

**#check if backup was successful**

**if [ $? -eq 0 ]; then**

**echo "Backup created successfully: $backup\_file"**

**else**

**echo "Backup failed"**

**fi**

**vagrant@ubuntu-focal:~$**

**Implementation of the script**

**Implementation 1**

vagrant@ubuntu-focal:~$ cd Doc3

vagrant@ubuntu-focal:~/Doc3$ ls

C1.txt

vagrant@ubuntu-focal:~/Doc3$ ls -a

. .. C1.txt

vagrant@ubuntu-focal:~/Doc3$ cd

vagrant@ubuntu-focal:~$ cd Doc2

vagrant@ubuntu-focal:~/Doc2$ ls

B1.txt B2.txt B3.txt Doc2A Doc2B Doc3B backup.sh

vagrant@ubuntu-focal:~/Doc2$ mv backup.sh

mv: missing destination file operand after 'backup.sh'

Try 'mv --help' for more information.

vagrant@ubuntu-focal:~/Doc2$ rm backup.sh

vagrant@ubuntu-focal:~/Doc2$ ls

B1.txt B2.txt B3.txt Doc2A Doc2B Doc3B

vagrant@ubuntu-focal:~/Doc2$ cd

vagrant@ubuntu-focal:~$ ./backup.sh ./Doc2/Doc2A ./Doc3/Doc3A

**Backup created successfully: ./Doc3/Doc3A/backup\_20240309\_114730.tar.gz**

vagrant@ubuntu-focal:~$ cd Doc3

vagrant@ubuntu-focal:~/Doc3$ ls

C1.txt Doc3A

vagrant@ubuntu-focal:~/Doc3$ cd Doc3A

vagrant@ubuntu-focal:~/Doc3/Doc3A$ ls

vagrant@ubuntu-focal:~/Doc3/Doc3A$

vagrant@ubuntu-focal:~/Doc3/Doc3A$ cd

vagrant@ubuntu-focal:~$ cat backup.sh

**Implemetation 2**

vagrant@ubuntu-focal:~$ ls

Doc1 Doc2 Doc3 backup.sh backup\_file code diskuse.sh misc personal script.sh tests

vagrant@ubuntu-focal:~$ cd Doc2

vagrant@ubuntu-focal:~/Doc2$ ls

B1.txt B2.txt B3.txt Doc2A Doc2B Doc3B

vagrant@ubuntu-focal:~/Doc2$ cd Doc2A

vagrant@ubuntu-focal:~/Doc2/Doc2A$ ls

D1.txt

vagrant@ubuntu-focal:~/Doc2/Doc2A$ cd

vagrant@ubuntu-focal:~$ chmod +x ./backup.sh

vagrant@ubuntu-focal:~$

vagrant@ubuntu-focal:~$ ./backup.sh ./Doc2/Doc2A ./Doc4

**Backup created successfully: ./Doc4/backup\_20240309\_121419.tar.gz**

vagrant@ubuntu-focal:~$

vagrant@ubuntu-focal:~$ ls

Doc1 Doc2 Doc3 Doc4 backup.sh backup\_file code diskuse.sh misc personal script.sh tests

vagrant@ubuntu-focal:~$ cd Doc4

vagrant@ubuntu-focal:~/Doc4$ ls

vagrant@ubuntu-focal:~/Doc4$ ls

vagrant@ubuntu-focal:~/Doc4$ cd

vagrant@ubuntu-focal:~$ ls

Doc1 Doc2 Doc3 Doc4 backup.sh backup\_file code diskuse.sh misc personal script.sh tests

vagrant@ubuntu-focal:~$