COMP 8567
Summer 2024
Assignment 4
Aug/10/24

Note: This assignment **must be implemented** on the CS Linux sever using your official university login.

Write a bash script **backups24.sh** that runs continuously in the background and performs operations as discussed below.

Synopsis

backup24.sh $1 $2 $3

- backup24.sh can have 0-3 arguments
- $1, $2 and $3 are used to indicate file types (any valid file type)
  - ex: **$ backup24.sh .c .txt .pdf**
  - ex: **$ backup24.sh .zip**
- When no arguments are entered, all file types must be considered

Upon running the bash script with valid arguments as described earlier, the script must continuously run in the background and perform the following steps at specified time intervals

STEP 1 Create a **complete backup** of the **specified file types** found in the entire directory tree rooted at /home/username by tarring all such files into **cbup****.tar stored** at **~/home/backup/cbup24s**

- Update backup.log with the timestamp and the name of the tar file (See Fig.1.0 for file naming)
- **Note:** Naming conventions of the backup files must be followed as per Fig 1.0

(2 minutes interval)

STEP 2 Create an **incremental backup** of the **specified file types** in the entire directory tree rooted at /home/username that were **newly created or modified (only)** after **STEP 1**

- If there are any newly created/modified files **after STEP1**, create a tar file of those files (only): ib****.tar at **~/home/backup/ibup24s** and update backup.log with the timestamp and the name of the tar file (see Fig. 1.0 for logging and the file naming)
- Else update backup.log with the timestamp and a message (See Fig.1.0)

(2 minutes interval)

STEP 3 Create an **incremental backup** of the **specified file types** in the entire directory tree rooted at /home/username that were **newly created or modified (only)** after **STEP 2**

- If there are any newly created/modified files **after STEP 2**, create a tar file of those files (only): ib****.tar at **~/home/backup/ibup24s** and update backup.log with the timestamp and the name of the tar file (see Fig. 1.0 for logging and the file naming)
- Else update backup.log with the timestamp and a message (See Fig.1.0)

(2 minutes interval)

STEP 4 Create a **differential backup** of the **specified file types** in the entire directory tree rooted at /home/username that were **newly created or modified (only)** after **STEP 1**

- If there are any newly created/modified files **after STEP1**, create a tar file of those files (only): db****.tar at **~/home/backup/dbup24s** and update backup.log with the timestamp and the name of the tar file (see Fig. 1.0 for logging and the file naming)
- Else update backup.log with the timestamp and a message (See Fig.1.0)

(2 minutes Interval)

STEP 5 Create an **incremental backup** of the **specified file types** in the entire directory tree rooted at /home/username that were **newly created or modified (only)** after **STEP 4**

- If there are any newly created/modified files **after STEP 4**, create a tar file of those files (only): ib****.tar at **~/home/backup/ibup24s** and update backup.log with the timestamp and the name of the tar file (see Fig. 1.0 for logging and the file naming)
- Else update backup.log with the timestamp and a message (See Fig.1.0)

(PROCEED TO STEP 1) //Continuous loop

(See the next page) …

Fig. 1.0 backup.log (sample)

```
Fri 19 Aug2024 06:16:08 PM EDT   cbup24s-1.tar was created

Fri 19 Aug2024 06:18:08 PM EDT   ibup24s-1.tar was created

Fri 19 Aug2024 06:20:08 PM EDT   No changes-Incremental backup was not created

Fri 19 Aug2024 06:22:08 PM EDT   dbup24s-1.tar was created

Fri 19 Aug2024 06:24:08 PM EDT   ibup24s-2.tar was created

Fri 19 Aug2024 06:26:08 PM EDT   cbup24s-2.tar was created

Fri 19 Aug2024 06:28:08 PM EDT   ibup24s-3.tar was created

Fri 19 Aug2024 06:30:08 PM EDT   ibup24s-4.tar was created

Fri 19 Aug2024 06:32:08 PM EDT   dbup24s-2.tar was created

Fri 19 Aug2024 06:34:08 PM EDT   ibup24s-5.tar was created

-----
-----
-----
```

Note: In practical scenarios, one can expect a weekly complete backup and a daily incremental backup (or could also be differential backup) of files.

Submission Instructions:

You need to submit the following:

1. A4S24.zip (that contains backups24.sh and backup.log)
2. Zoom/Google Drive recording link explaining the following (not more than 15 minutes)
   - Overall working of the code and various modules (around 8-9 minutes)
   - Execution of the code under various inputs/conditions as per the requirements of the assignment (around 6-7 minutes)

Other form of links/MP4 files will NOT be acceptable. Include the link in the COMMENTS section