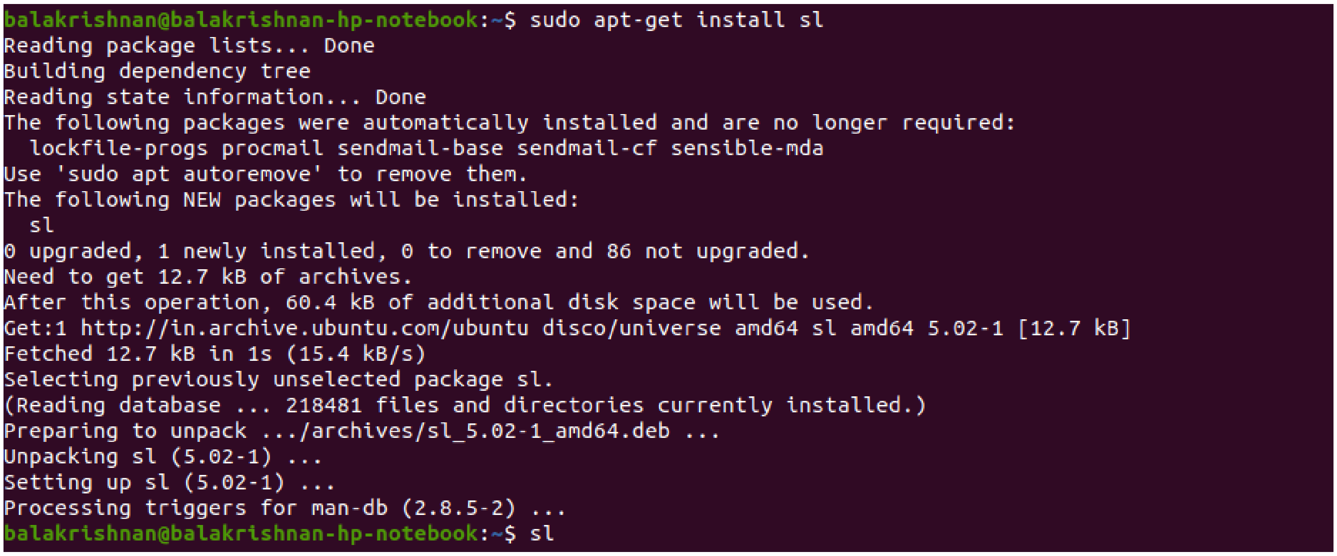
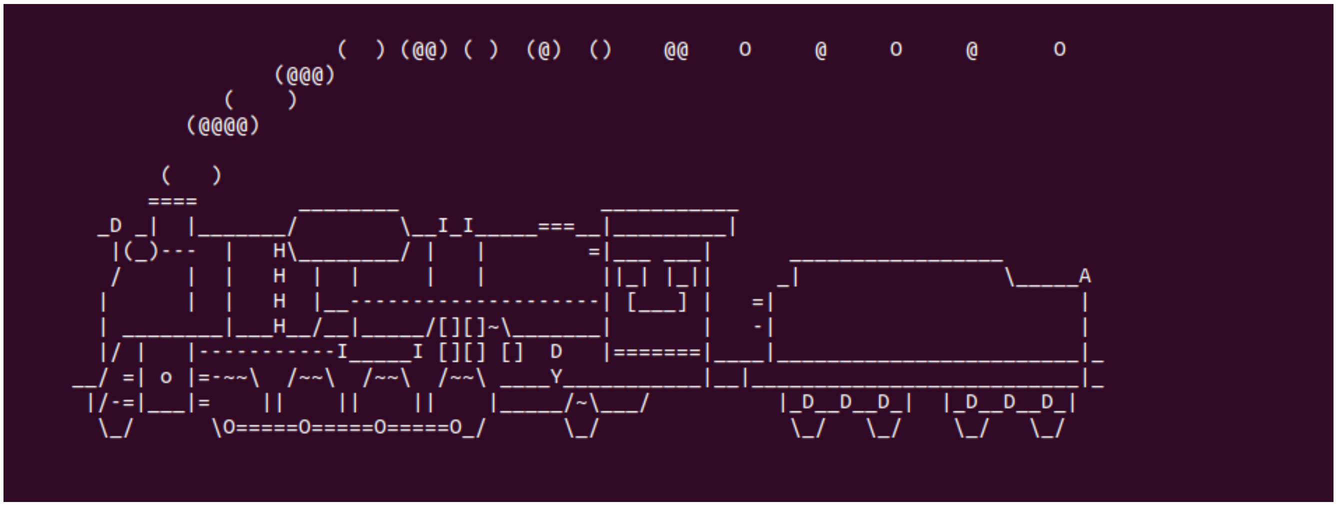
Linux Programming Lab-9
Name: Balakrishnan.P
Register number: 16MIS1159

Script 1

apt-get install sl
sl



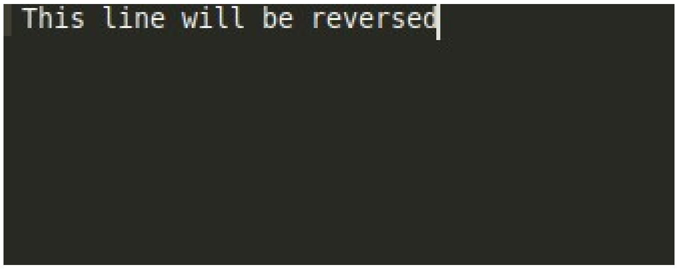
```
balakrishnan@balakrishnan-hp-notebook:~$ sudo apt-get install sl
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  lockfile-progs procmail sendmail-base sendmail-cf sensible-mda
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  sl
0 upgraded, 1 newly installed, 0 to remove and 86 not upgraded.
Need to get 12.7 kB of archives.
After this operation, 60.4 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu disco/universe amd64 sl amd64 5.02-1 [12.7 kB]
Fetched 12.7 kB in 1s (15.4 kB/s)
Selecting previously unselected package sl.
(Reading database ... 218481 files and directories currently installed.)
Preparing to unpack .../archives/sl_5.02-1_amd64.deb ...
Unpacking sl (5.02-1) ...
Setting up sl (5.02-1) ...
Processing triggers for man-db (2.8.5-2) ...
balakrishnan@balakrishnan-hp-notebook:~$ sl
```



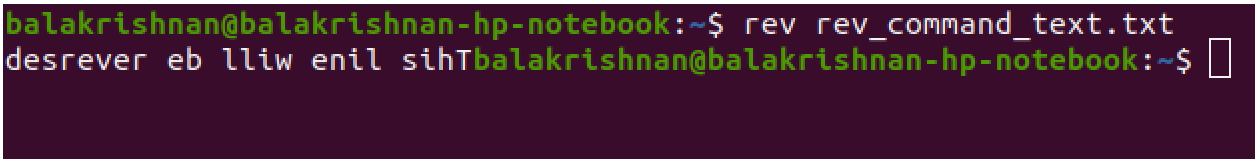
```
                        (  ) (@@) ( )  (@)  ()    @@    O     @     O     @      O
                   (@@@)
               (    )
            (@@@@)

          (   )
        ====        ________                ___________
    _D _|  |_______/        \__I_I_____===__|_________|
     |(_)---  |   H\________/ |   |        =|___ ___|      _________________
     /     |  |   H  |  |     |   |         ||_| |_||     _|                \_____A
    |      |  |   H  |__--------------------| [___] |   =|                        |
    | ________|___H__/__|_____/[][]~\_______|       |   -|                        |
    |/ |   |-----------I_____I [][] []  D   |=======|____|________________________|_
  __/ =| o |=-~~\  /~~\  /~~\  /~~\ ____Y___________|__|__________________________|_
   |/-=|___|=    ||    ||    ||    |_____/~\___/          |_D__D__D_|  |_D__D__D_|
    \_/      \O=====O=====O=====O_/      \_/               \_/   \_/    \_/   \_/
```

Script 2

rev

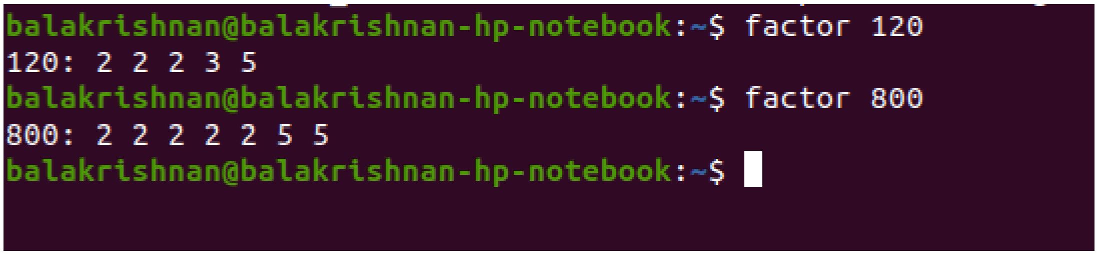


```
This line will be reversed
```



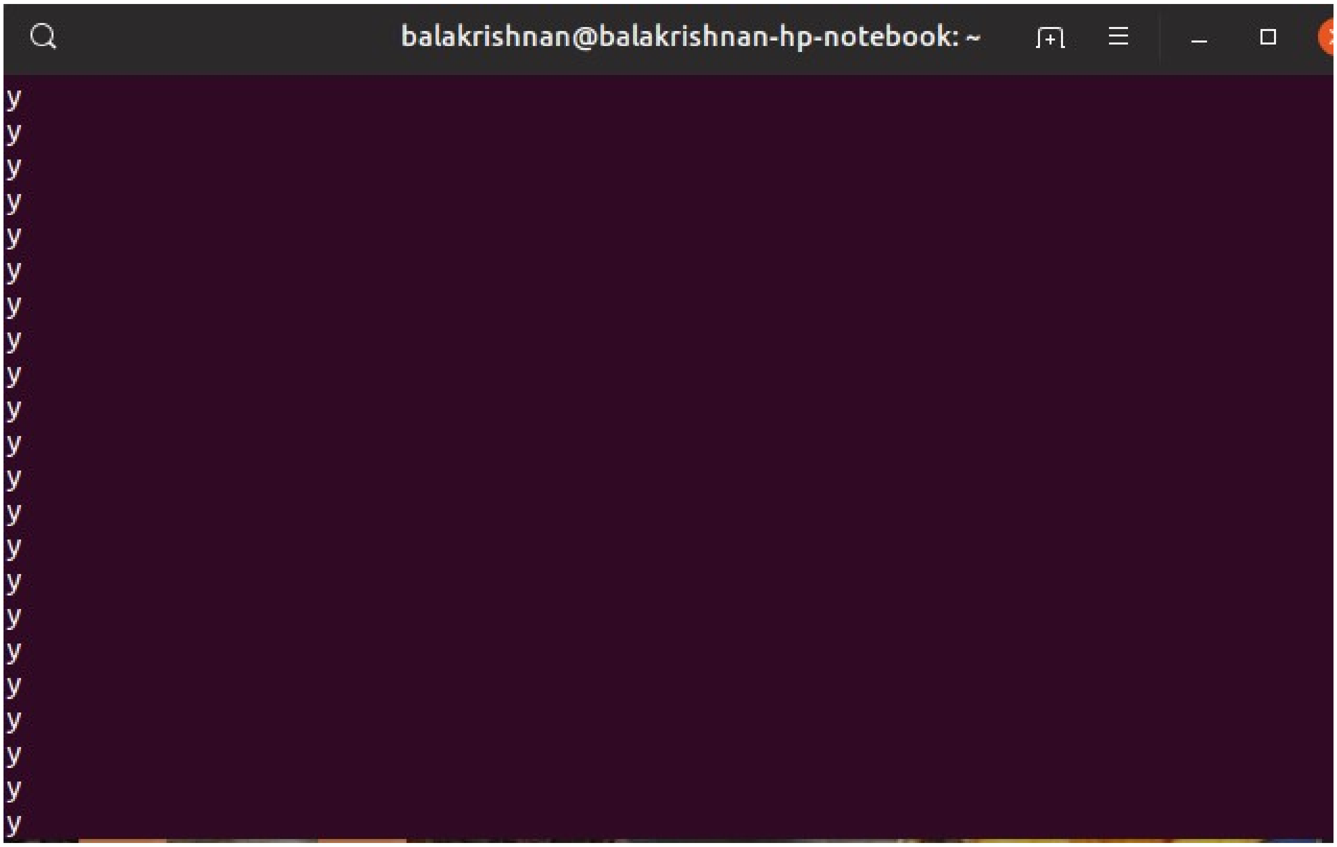
```
balakrishnan@balakrishnan-hp-notebook:~$ rev rev_command_text.txt
desrever eb lliw enil sihTbalakrishnan@balakrishnan-hp-notebook:~$ 
```

factor



```
balakrishnan@balakrishnan-hp-notebook:~$ factor 120
120: 2 2 2 3 5
balakrishnan@balakrishnan-hp-notebook:~$ factor 800
800: 2 2 2 2 2 5 5
balakrishnan@balakrishnan-hp-notebook:~$ 
```

Yes



```
y
y
y
y
y
y
y
y
y
y
y
y
y
y
y
y
y
y
y
y
y
y
```

Question 2:

Write a bash shell script to monitor the health of your system. Let the details be stored and archived in any folder of your choice.

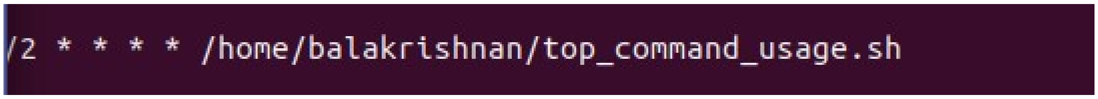
Code

#!/usr/bin/env bash

top -b -n1>top_output.txt

tar -cvf 'lab 9 online.tar' 'top_output.txt'

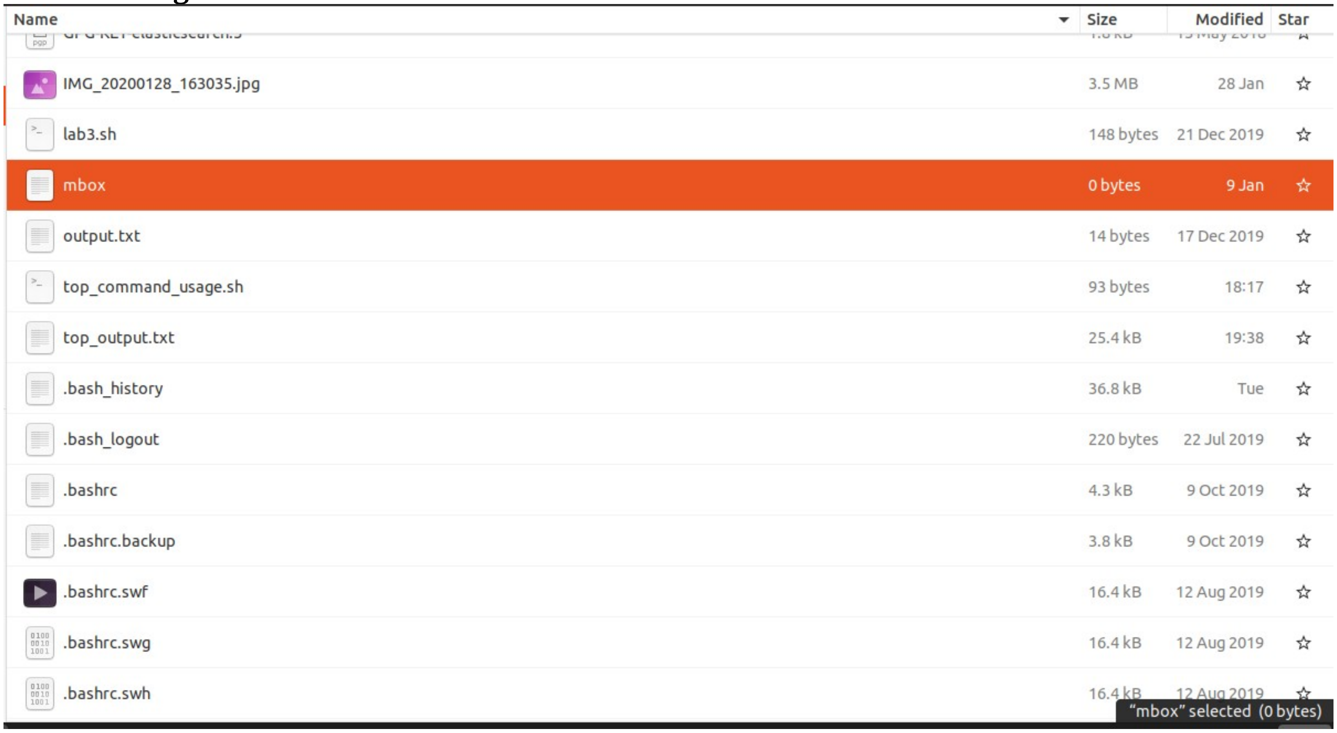
Crontab editor code



```
/2 * * * * /home/balakrishnan/top_command_usage.sh
```

Note: Here the Archive folder creation will happen after every two minutes just two verify the output.

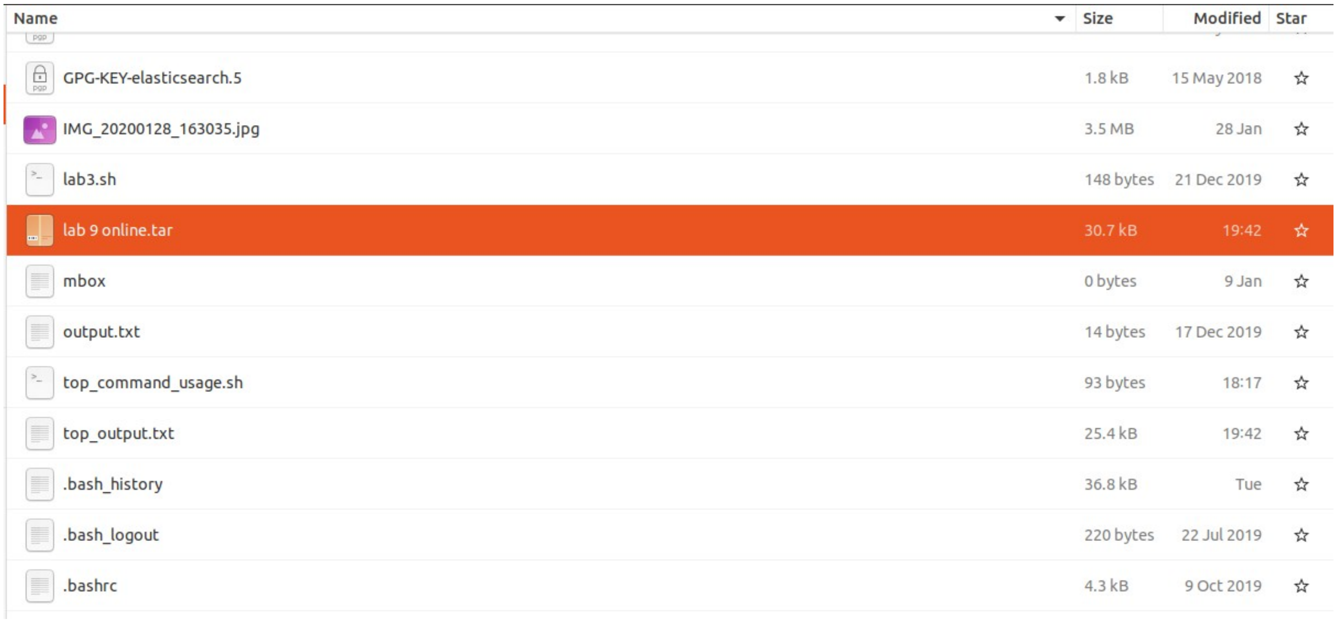
Before saving crontab editor



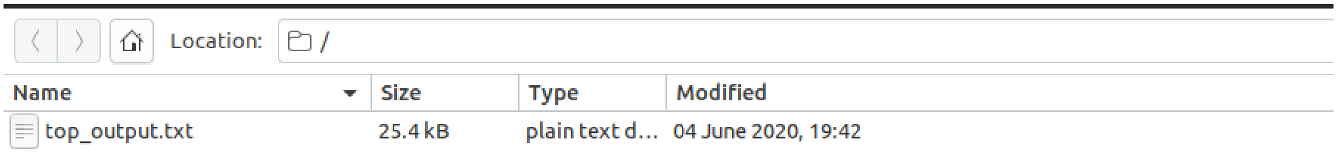
Name	Size	Modified	Star
GPG-KEY-elasticsearch.s	1.6 kB	15 May 2018	☆
IMG_20200128_163035.jpg	3.5 MB	28 Jan	☆
lab3.sh	148 bytes	21 Dec 2019	☆
mbox	0 bytes	9 Jan	☆
output.txt	14 bytes	17 Dec 2019	☆
top_command_usage.sh	93 bytes	18:17	☆
top_output.txt	25.4 kB	19:38	☆
.bash_history	36.8 kB	Tue	☆
.bash_logout	220 bytes	22 Jul 2019	☆
.bashrc	4.3 kB	9 Oct 2019	☆
.bashrc.backup	3.8 kB	9 Oct 2019	☆
.bashrc.swf	16.4 kB	12 Aug 2019	☆
.bashrc.swg	16.4 kB	12 Aug 2019	☆
.bashrc.swh	16.4 kB	12 Aug 2019	☆

“mbox” selected (0 bytes)

After saving the crontab editor

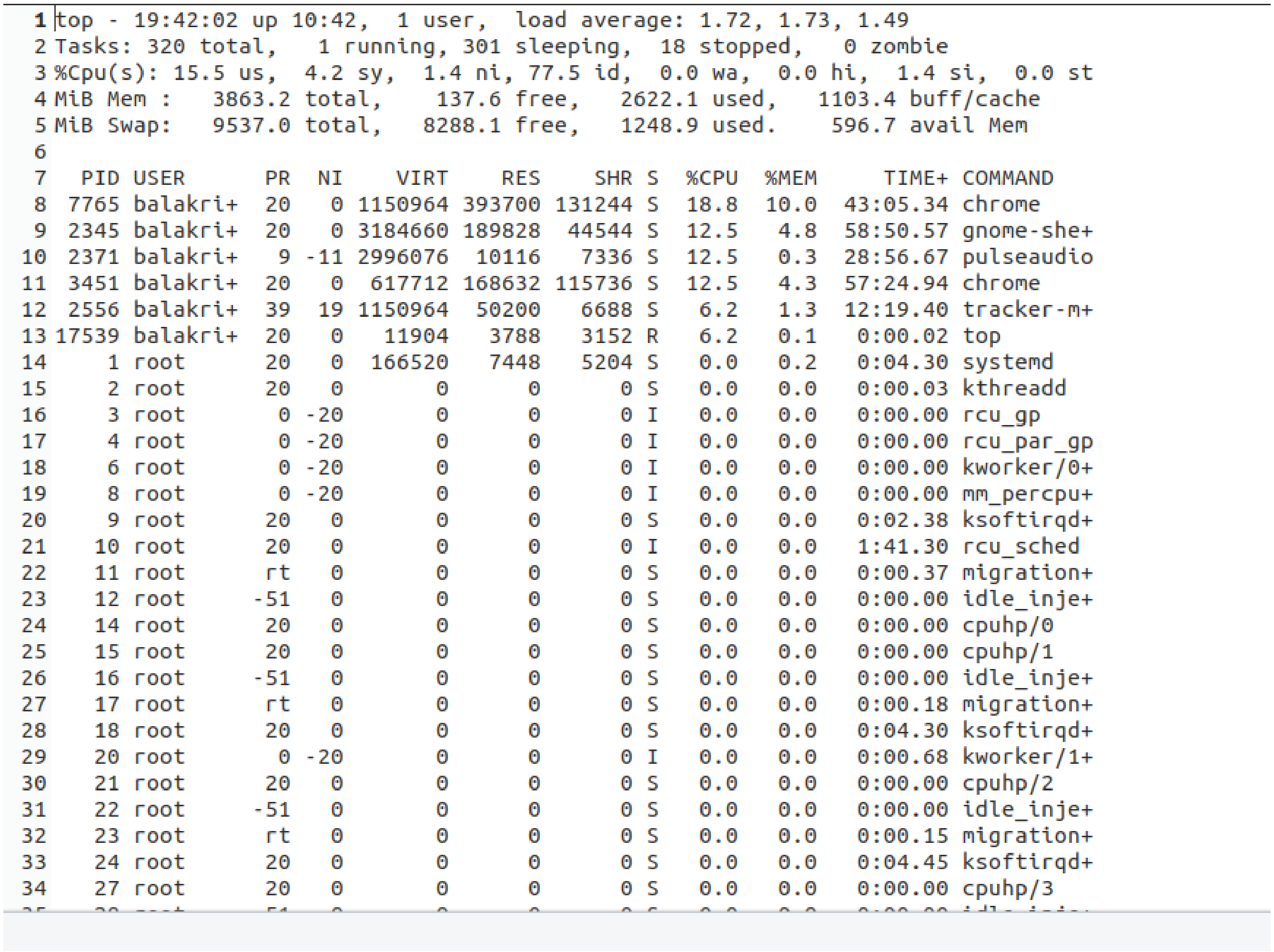


Name	Size	Modified	Star
GPG-KEY-elasticsearch.5	1.8 kB	15 May 2018	☆
IMG_20200128_163035.jpg	3.5 MB	28 Jan	☆
lab3.sh	148 bytes	21 Dec 2019	☆
lab 9 online.tar	30.7 kB	19:42	☆
mbox	0 bytes	9 Jan	☆
output.txt	14 bytes	17 Dec 2019	☆
top_command_usage.sh	93 bytes	18:17	☆
top_output.txt	25.4 kB	19:42	☆
.bash_history	36.8 kB	Tue	☆
.bash_logout	220 bytes	22 Jul 2019	☆
.bashrc	4.3 kB	9 Oct 2019	☆



Location: /

Name	Size	Type	Modified
top_output.txt	25.4 kB	plain text d…	04 June 2020, 19:42



```
top - 19:42:02 up 10:42,  1 user,  load average: 1.72, 1.73, 1.49
Tasks: 320 total,   1 running, 301 sleeping,  18 stopped,   0 zombie
%Cpu(s): 15.5 us,  4.2 sy,  1.4 ni, 77.5 id,  0.0 wa,  0.0 hi,  1.4 si,  0.0 st
MiB Mem :   3863.2 total,    137.6 free,   2622.1 used,   1103.4 buff/cache
MiB Swap:   9537.0 total,   8288.1 free,   1248.9 used.    596.7 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
 7765 balakri+  20   0 1150964 393700 131244 S  18.8  10.0  43:05.34 chrome
 2345 balakri+  20   0 3184660 189828  44544 S  12.5   4.8  58:50.57 gnome-she+
 2371 balakri+   9 -11 2996076  10116   7336 S  12.5   0.3  28:56.67 pulseaudio
 3451 balakri+  20   0  617712 168632 115736 S  12.5   4.3  57:24.94 chrome
 2556 balakri+  39  19 1150964  50200   6688 S   6.2   1.3  12:19.40 tracker-m+
17539 balakri+  20   0   11904   3788   3152 R   6.2   0.1   0:00.02 top
    1 root      20   0  166520   7448   5204 S   0.0   0.2   0:04.30 systemd
    2 root      20   0       0      0      0 S   0.0   0.0   0:00.03 kthreadd
    3 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 rcu_gp
    4 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 rcu_par_gp
    6 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/0+
    8 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 mm_percpu+
    9 root      20   0       0      0      0 S   0.0   0.0   0:02.38 ksoftirqd+
   10 root      20   0       0      0      0 I   0.0   0.0   1:41.30 rcu_sched
   11 root      rt   0       0      0      0 S   0.0   0.0   0:00.37 migration+
   12 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inje+
   14 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
   15 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/1
   16 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inje+
   17 root      rt   0       0      0      0 S   0.0   0.0   0:00.18 migration+
   18 root      20   0       0      0      0 S   0.0   0.0   0:04.30 ksoftirqd+
   20 root       0 -20       0      0      0 I   0.0   0.0   0:00.68 kworker/1+
   21 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/2
   22 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inje+
   23 root      rt   0       0      0      0 S   0.0   0.0   0:00.15 migration+
   24 root      20   0       0      0      0 S   0.0   0.0   0:04.45 ksoftirqd+
   27 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/3
```

Similarly the output from the top command will save the details regarding the health of the system into the file and will be archived for every two hours if the crontab is adjusted as per the requirement.

Additional question

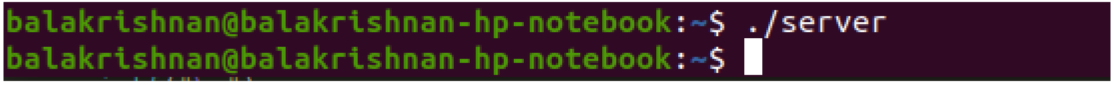
Write a C Program to implement Simple reader-writer algorithm using shared memory segment with semaphore

Code

server.c(this file acts as a writer and this file will writes the text “Balakrishnan” in the shared memory).

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<sys/types.h>
#include<sys/ipc.h>
#include<sys/shm.h>
```

```
#define SHSIZE 100

int main(int argc,char *argv[])
{
        int shmid;
        key_t key;
        char *shm;
        char *s;
        //Let key=some ram=ndom integer
        key=9876;

        shmid=shmget(key,SHSIZE,IPC_CREAT | 0666);

        if(shmid<0)
        {
                perror("Shmget");
                exit(1);
        }

        shm=shmat(shmid,NULL,0);

        if(shm==(char *)-1)
        {
                perror("shmat");
                exit(1);
        }

        memcpy(shm,"Balakrishnan",12);

        s=shm;
        s+=12;

        *s=0;

        while(*shm!='*')
        {
                sleep(1);
        }


}
```

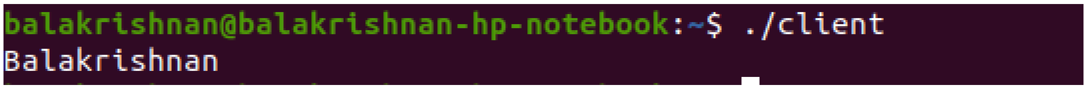


```
balakrishnan@balakrishnan-hp-notebook:~$ ./server
balakrishnan@balakrishnan-hp-notebook:~$ 
```

Client.c(this file acts as a reader and will read the text and prints the text in terminal)

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
```

```
#include<sys/types.h>
#include<sys/ipc.h>
#include<sys/shm.h>

#define SHSIZE 100

int main(int argc,char *argv[])
{
        int shmid;
        key_t key;
        char *shm;
        char *s;
        //Let key=some ram=ndom integer
        key=9876;

        shmid=shmget(key,SHSIZE,0666);

        if(shmid<0)
        {
                perror("Shmget");
                exit(1);
        }

        shm=shmat(shmid,NULL,0);

        if(shm==(char *)-1)
        {
                perror("shmat");
                exit(1);
        }

        for(s=shm;*s!=0;s++)
        {
                printf("%c",*s);
        }
        printf("\n");
        *shm='*';


}
```



```
balakrishnan@balakrishnan-hp-notebook:~$ ./client
Balakrishnan
```

Note: Both the server.c and client.c must be executed at same time using two different terminals.