PRACTICE LAB

MALAVIKA P.S

17MIS1080

1. **sl**

If you've ever accidentally typed sl when you meant to list a directory's contents with `ls`, then you may want to install a program "sl".

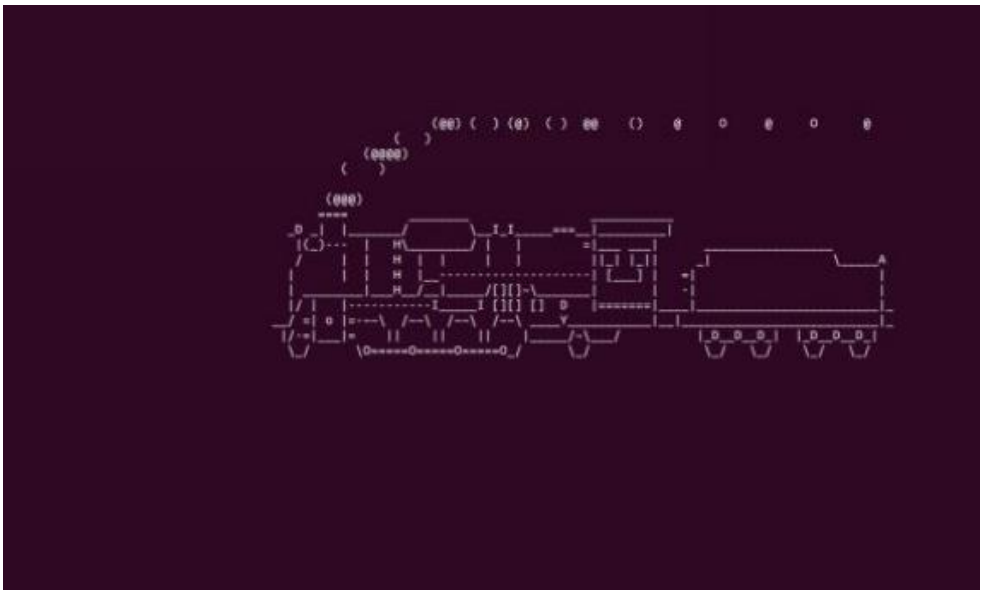
Code:

p1.sh

sudo apt-get install sl

sl

```
malavika@malavika-VirtualBox:~$ gedit p1.sh
malavika@malavika-VirtualBox:~$ chmod +x p1.sh
malavika@malavika-VirtualBox:~$ ./p1.sh
[sudo] password for malavika:
Reading package lists... Done
Building dependency tree
Reading state information... Done
sl is already the newest version (3.03-17build2).
The following package was automatically installed and is no longer required:
  libwayland-egl1-mesa
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 203 not upgraded.
```



1. **rev**

The rev command in Linux is used to reverse the lines character wise. This utility basically reverses the order of the characters in each line by copying the specified files to the standard output. If no files are specified, then the standard input will read.

factor

The **factor** command in Linux is used to print the prime factors of the given numbers, either given from command line or read from standard input.

```
malavika@malavika-VirtualBox:~$ rev
malavika
akivalam
^Z
[2]+  Stopped                 rev
malavika@malavika-VirtualBox:~$ factor 56
56: 2 2 2 7
```

yes

yes command in linux is used to print a continuous output stream of given *STRING*. If *STRING* is not mentioned then it prints ‘y’;

code:

yes malavika

```
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
malavika
```

ADDITIONAL QUESTION

Code for shared memory for writer process:

```
#include <iostream>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <stdio.h>
using namespace std;
int main()
{
// ftok to generate unique key
key_t key = ftok("bhaveshfile",65);
// shmget returns an identifier in shmid
int shmid = shmget(key,1024,0666|IPC_CREAT);
// shmat to attach to shared memory
char *str = (char*) shmat(shmid,(void*)0,0);
cout<<"Write Data : ";
gets(str);
```

```
printf("Data written in memory: %s\n",str);

//detach from shared memory

shmdt(str);


return 0;

}
```

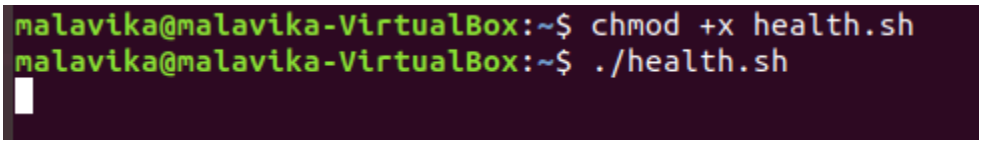
code for shared memory od reader process:

```
#include <iostream>

#include <sys/ipc.h>

#include <sys/shm.h>

#include <stdio.h>

using namespace std;

int main()

{

// ftok to generate unique key

key_t key = ftok("bhaveshfile",65);

// shmget returns an identifier in shmid

int shmid = shmget(key,1024,0666|IPC_CREAT);

// shmat to attach to shared memory

char *str = (char*) shmat(shmid,(void*)0,0);

printf("Data read from memory: %s\n",str);

//detach from shared memory

shmdt(str);

// destroy the shared memory

shmctl(shmid,IPC_RMID,NULL);

return 0;
```

HOT QUESTION

```
vmstat 1200 > vmstat1.data
filename= "/malavika/vmstat1.data"
tail -f $filename |
```

```
while read $line do
if [ (cat vmstat1.data | grep "swpd")>0  ]
then
 echo "Some Rogue Process has consumed massive amounts of Memory" >
swap.txt
fi
if [ (cat vmstat1.data | grep "r")>1  ]
then
   echo "Some processes are waiting to execute" >
runqueue.txt
fi
if [ (cat vmstat1.data | grep "cpu")>1000  ]
then
   echo "CPU usage is more" > cpu.txt
fi
End
```



Therefore my system is in good health ,or else there should be an alert message,popping up.

EXPLANATION:

The vmstat 1200 – monitors every 24 hours and puts the data into the vmstat1.data

grep “swap”- the swap should always be zero if it‘s not then some process has consumed massive memory.

That will be monitored in this line

grep “r”- the running queue is constantly above process 1 it indicates the system is slow and some process is waiting to be executed. That will be monitored here.

Grep “cpu”- it indicates the cpu usage of the system. If the cpu usage is more it will be monitored and will alert in this line.

LINK:

https://github.com/Malavika2609/linux