EXPT NO : 5

REPORT

DATE :21/03/2019

AIM : **To write the shell scripts of the given questions and verify the output**

1. Write a shell script to show various system configuration like
   - Currently logged user and his login name
   - Your current shell
   - Your home directory
   - Your operating system type
   - Your current path setting
   - Your current working directory
   - Number of users currently logged in

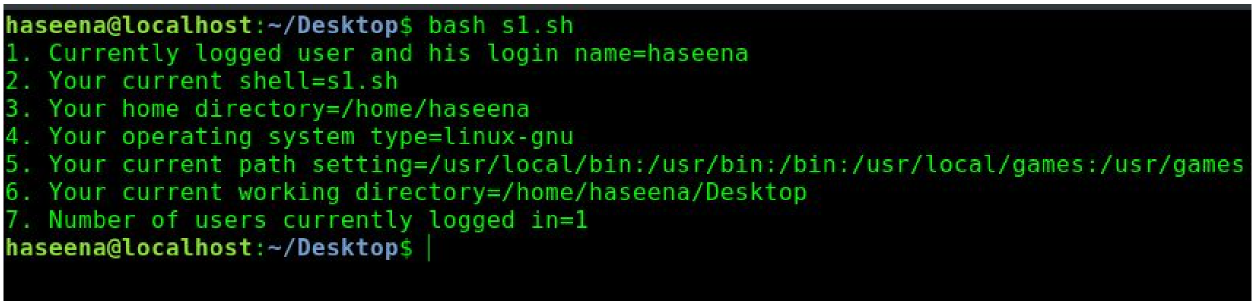
Algorithm:

```
1) Print   Currently logged user and his login name="$USER
2) Print   Your current shell=" $0
3) Print  Your home directory=" $HOME
4) Print  Your operating system type=" $OSTYPE
5) Print   Your current path setting=" $PATH
6) Print  Your current working directory=" $PWD
7) Print   Number of users currently logged in=" `who | wc -l`
```

Program:

```
#! /bin/bash
echo "1. Currently logged user and his login name="$USER
echo "2. Your current shell="$0
echo "3. Your home directory="$HOME
echo "4. Your operating system type="$OSTYPE
echo "5. Your current path setting="$PATH
echo "6. Your current working directory="$PWD
echo "7. Number of users currently logged in="`who | wc -l`
```

Output:



```
haseena@localhost:~/Desktop$ bash s1.sh
1. Currently logged user and his login name=haseena
2. Your current shell=s1.sh
3. Your home directory=/home/haseena
4. Your operating system type=linux-gnu
5. Your current path setting=/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
6. Your current working directory=/home/haseena/Desktop
7. Number of users currently logged in=1
haseena@localhost:~/Desktop$ 
```

2. Write a shell script to show various system configurations like

- your OS and version, release number, kernel version
- all available shells
- computer CPU information like processor type, speed etc
- memory information
- hard disk information like size of hard-disk, cache memory, model etc
- File system (Mounted)

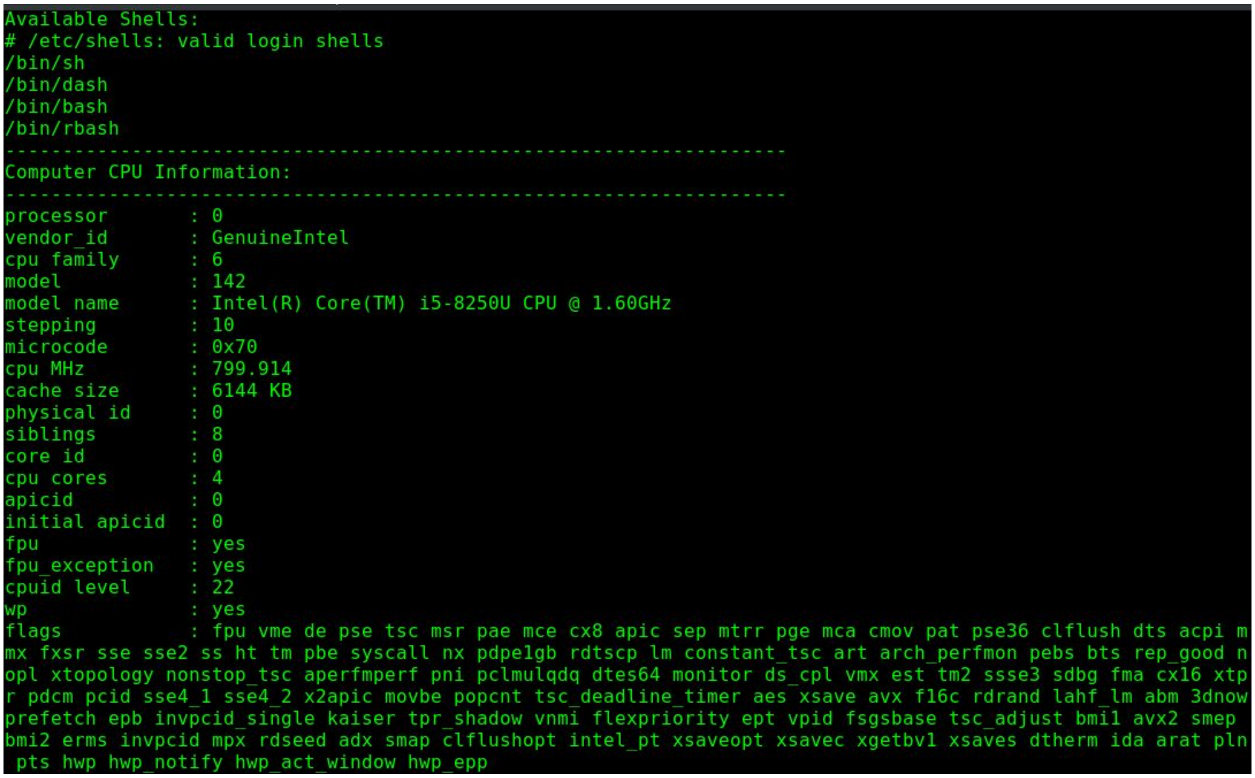
Algorithm:

1) Print OS and version, release number, kernel version : cat /etc/lsb-release
2) Print all available shells : cat /etc/shells
3) Print computer CPU information : cat /proc/cpuinfo
4) Print memory information : cat /proc/meminfo
5) Print hard disk information : cat /proc/ide/hda
6) Print file system (Mounted) : cat /proc/mounts

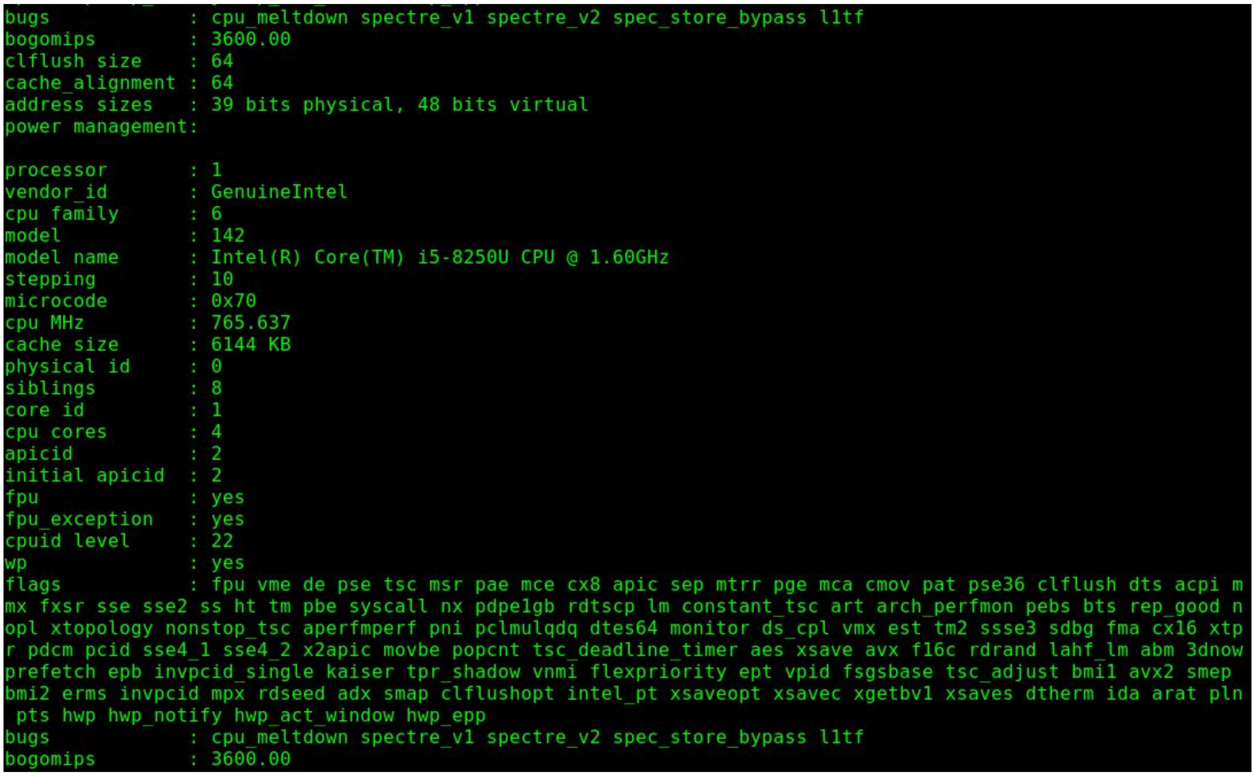
Program:

```
#!/bin/bash
if [ -f /etc/lsb-release ]
then
echo "OS: `cat /etc/lsb-release`" >> myfile
fi
if [ -f /etc/shells ]
then
echo "Available Shells: " >> myfile
echo "`cat /etc/shells`" >> myfile
fi
if [ -f /etc/sysconfig/mouse ]
then
echo "------------------------------------------------------------------" >> myfile
echo "Computer Mouse Information: " >> myfile
echo "--------------------------------------------------------------------" >> myfile
echo "`cat /etc/sysconfig/mouse`" >> myfile
fi
echo "--------------------------------------------------------------------" >> myfile
echo "Computer CPU Information:" >> myfile
echo "--------------------------------------------------------------------" >> myfile
cat /proc/cpuinfo >> myfile
echo "--------------------------------------------------------------------" >> myfile
echo "Computer Memory Information:" >> myfile
echo "--------------------------------------------------------------------" >> myfile
cat /proc/meminfo >> myfile
if [ -d /proc/ide/hda ]
then
echo "--------------------------------------------------------------------" >> myfile
echo "Hard disk information:" >> myfile
echo "--------------------------------------------------------------------" >> myfile
cat /proc/mounts >> myfile
cat myfile
# end of script
```

Output:

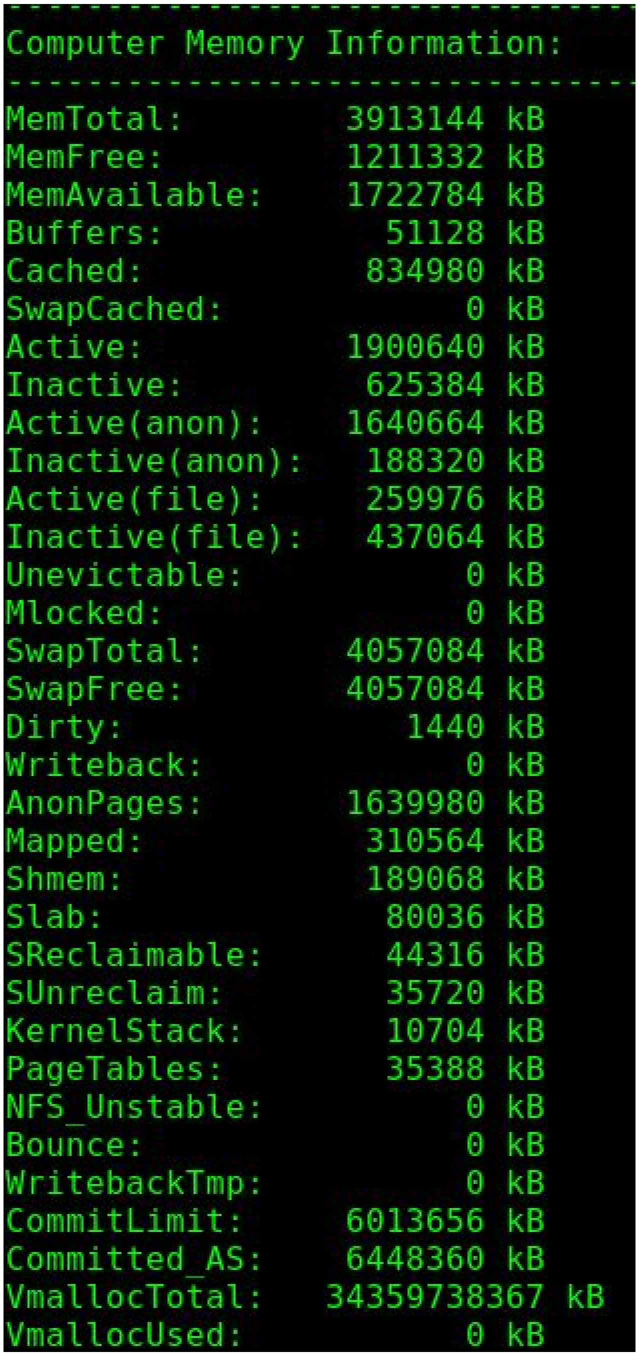


```
Available Shells:
# /etc/shells: valid login shells
/bin/sh
/bin/dash
/bin/bash
/bin/rbash
--------------------------------------------------------------------
Computer CPU Information:
--------------------------------------------------------------------
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 142
model name      : Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz
stepping        : 10
microcode       : 0x70
cpu MHz         : 799.914
cache size      : 6144 KB
physical id     : 0
siblings        : 8
core id         : 0
cpu cores       : 4
apicid          : 0
initial apicid  : 0
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi m
mx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant_tsc art arch_perfmon pebs bts rep_good n
opl xtopology nonstop_tsc aperfmperf pni pclmulqdq dtes64 monitor ds_cpl vmx est tm2 ssse3 sdbg fma cx16 xtp
r pdcm pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdrand lahf_lm abm 3dnow
prefetch epb invpcid_single kaiser tpr_shadow vnmi flexpriority ept vpid fsgsbase tsc_adjust bmi1 avx2 smep
bmi2 erms invpcid mpx rdseed adx smap clflushopt intel_pt xsaveopt xsavec xgetbv1 xsaves dtherm ida arat pln
 pts hwp hwp_notify hwp_act_window hwp_epp
```

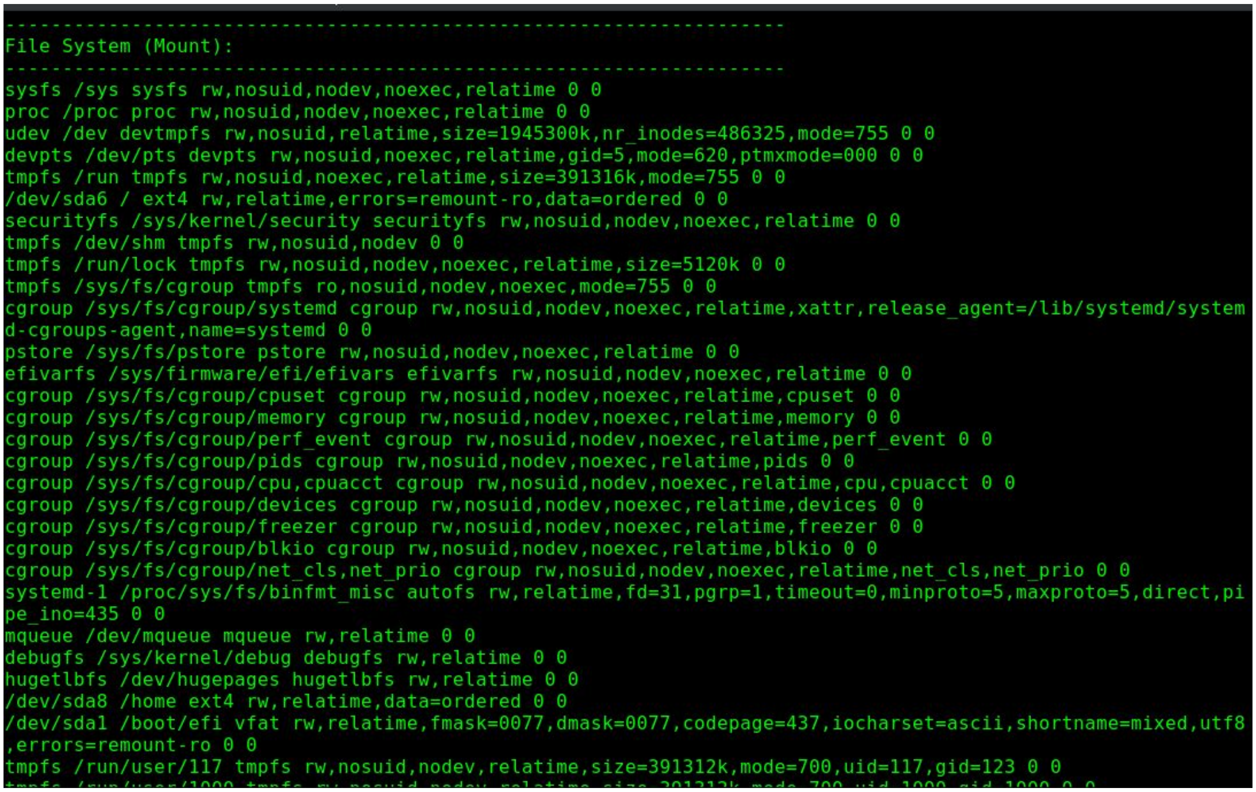


```
bugs            : cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass l1tf
bogomips        : 3600.00
clflush size    : 64
cache_alignment : 64
address sizes   : 39 bits physical, 48 bits virtual
power management:

processor       : 1
vendor_id       : GenuineIntel
cpu family      : 6
model           : 142
model name      : Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz
stepping        : 10
microcode       : 0x70
cpu MHz         : 765.637
cache size      : 6144 KB
physical id     : 0
siblings        : 8
core id         : 1
cpu cores       : 4
apicid          : 2
initial apicid  : 2
fpu             : yes
fpu_exception   : yes
cpuid level     : 22
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi m
mx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant_tsc art arch_perfmon pebs bts rep_good n
opl xtopology nonstop_tsc aperfmperf pni pclmulqdq dtes64 monitor ds_cpl vmx est tm2 ssse3 sdbg fma cx16 xtp
r pdcm pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdrand lahf_lm abm 3dnow
prefetch epb invpcid_single kaiser tpr_shadow vnmi flexpriority ept vpid fsgsbase tsc_adjust bmi1 avx2 smep
bmi2 erms invpcid mpx rdseed adx smap clflushopt intel_pt xsaveopt xsavec xgetbv1 xsaves dtherm ida arat pln
 pts hwp hwp_notify hwp_act_window hwp_epp
bugs            : cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass l1tf
bogomips        : 3600.00
```



```
--------------------------------
Computer Memory Information:
--------------------------------
MemTotal:        3913144 kB
MemFree:         1211332 kB
MemAvailable:    1722784 kB
Buffers:           51128 kB
Cached:           834980 kB
SwapCached:            0 kB
Active:          1900640 kB
Inactive:         625384 kB
Active(anon):    1640664 kB
Inactive(anon):   188320 kB
Active(file):     259976 kB
Inactive(file):   437064 kB
Unevictable:           0 kB
Mlocked:               0 kB
SwapTotal:       4057084 kB
SwapFree:        4057084 kB
Dirty:              1440 kB
Writeback:             0 kB
AnonPages:       1639980 kB
Mapped:           310564 kB
Shmem:            189068 kB
Slab:              80036 kB
SReclaimable:      44316 kB
SUnreclaim:        35720 kB
KernelStack:       10704 kB
PageTables:        35388 kB
NFS_Unstable:          0 kB
Bounce:                0 kB
WritebackTmp:          0 kB
CommitLimit:     6013656 kB
Committed_AS:    6448360 kB
VmallocTotal:   34359738367 kB
VmallocUsed:           0 kB
```



```
--------------------------------------------------------------------
File System (Mount):
--------------------------------------------------------------------
sysfs /sys sysfs rw,nosuid,nodev,noexec,relatime 0 0
proc /proc proc rw,nosuid,nodev,noexec,relatime 0 0
udev /dev devtmpfs rw,nosuid,relatime,size=1945300k,nr_inodes=486325,mode=755 0 0
devpts /dev/pts devpts rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000 0 0
tmpfs /run tmpfs rw,nosuid,noexec,relatime,size=391316k,mode=755 0 0
/dev/sda6 / ext4 rw,relatime,errors=remount-ro,data=ordered 0 0
securityfs /sys/kernel/security securityfs rw,nosuid,nodev,noexec,relatime 0 0
tmpfs /dev/shm tmpfs rw,nosuid,nodev 0 0
tmpfs /run/lock tmpfs rw,nosuid,nodev,noexec,relatime,size=5120k 0 0
tmpfs /sys/fs/cgroup tmpfs ro,nosuid,nodev,noexec,mode=755 0 0
cgroup /sys/fs/cgroup/systemd cgroup rw,nosuid,nodev,noexec,relatime,xattr,release_agent=/lib/systemd/system
d-cgroups-agent,name=systemd 0 0
pstore /sys/fs/pstore pstore rw,nosuid,nodev,noexec,relatime 0 0
efivarfs /sys/firmware/efi/efivars efivarfs rw,nosuid,nodev,noexec,relatime 0 0
cgroup /sys/fs/cgroup/cpuset cgroup rw,nosuid,nodev,noexec,relatime,cpuset 0 0
cgroup /sys/fs/cgroup/memory cgroup rw,nosuid,nodev,noexec,relatime,memory 0 0
cgroup /sys/fs/cgroup/perf_event cgroup rw,nosuid,nodev,noexec,relatime,perf_event 0 0
cgroup /sys/fs/cgroup/pids cgroup rw,nosuid,nodev,noexec,relatime,pids 0 0
cgroup /sys/fs/cgroup/cpu,cpuacct cgroup rw,nosuid,nodev,noexec,relatime,cpu,cpuacct 0 0
cgroup /sys/fs/cgroup/devices cgroup rw,nosuid,nodev,noexec,relatime,devices 0 0
cgroup /sys/fs/cgroup/freezer cgroup rw,nosuid,nodev,noexec,relatime,freezer 0 0
cgroup /sys/fs/cgroup/blkio cgroup rw,nosuid,nodev,noexec,relatime,blkio 0 0
cgroup /sys/fs/cgroup/net_cls,net_prio cgroup rw,nosuid,nodev,noexec,relatime,net_cls,net_prio 0 0
systemd-1 /proc/sys/fs/binfmt_misc autofs rw,relatime,fd=31,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pi
pe_ino=435 0 0
mqueue /dev/mqueue mqueue rw,relatime 0 0
debugfs /sys/kernel/debug debugfs rw,relatime 0 0
hugetlbfs /dev/hugepages hugetlbfs rw,relatime 0 0
/dev/sda8 /home ext4 rw,relatime,data=ordered 0 0
/dev/sda1 /boot/efi vfat rw,relatime,fmask=0077,dmask=0077,codepage=437,iocharset=ascii,shortname=mixed,utf8
,errors=remount-ro 0 0
tmpfs /run/user/117 tmpfs rw,nosuid,nodev,relatime,size=391312k,mode=700,uid=117,gid=123 0 0
```

3. Write a shell script to implement a menu driven calculator with following functions

- Addition
- Subtraction
- Multiplication
- Division
- Modulus

Algorithm:

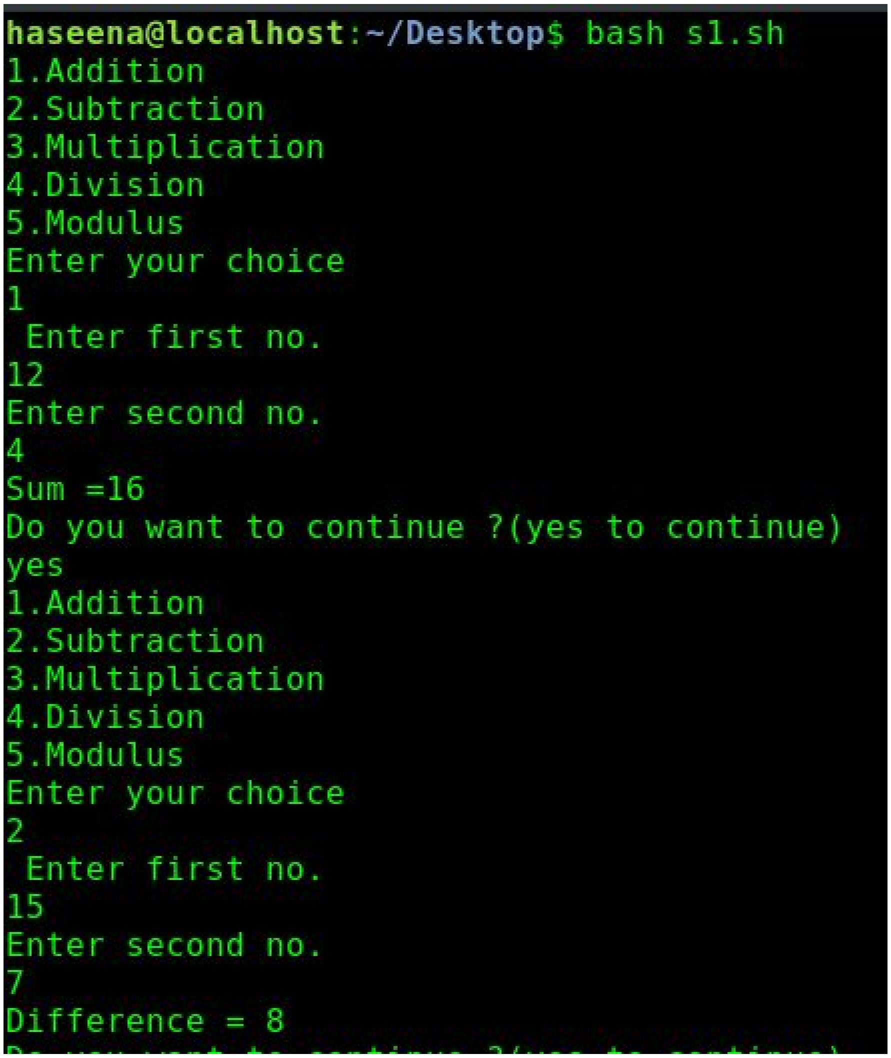
```
1) While true do following steps
2) Print the calculator menu
3) Read choice to c
4) Read the operands , say a and b.
5) Switch c
    a) Case 1 : sum = a+b
       Print sum
    b) Case 1 : diff = a-b
       Print difference
    c) Case 1 : product = a*b
       Print product
```

```
        d) Case 1 : quotient = a/b
           Print quotient
        e) Case 1 : modulus = a%b
           Print modulus
        f) Default : print " invalid choice"
    6) Read user choice to continue
    7) If yes then loop to step 1
```

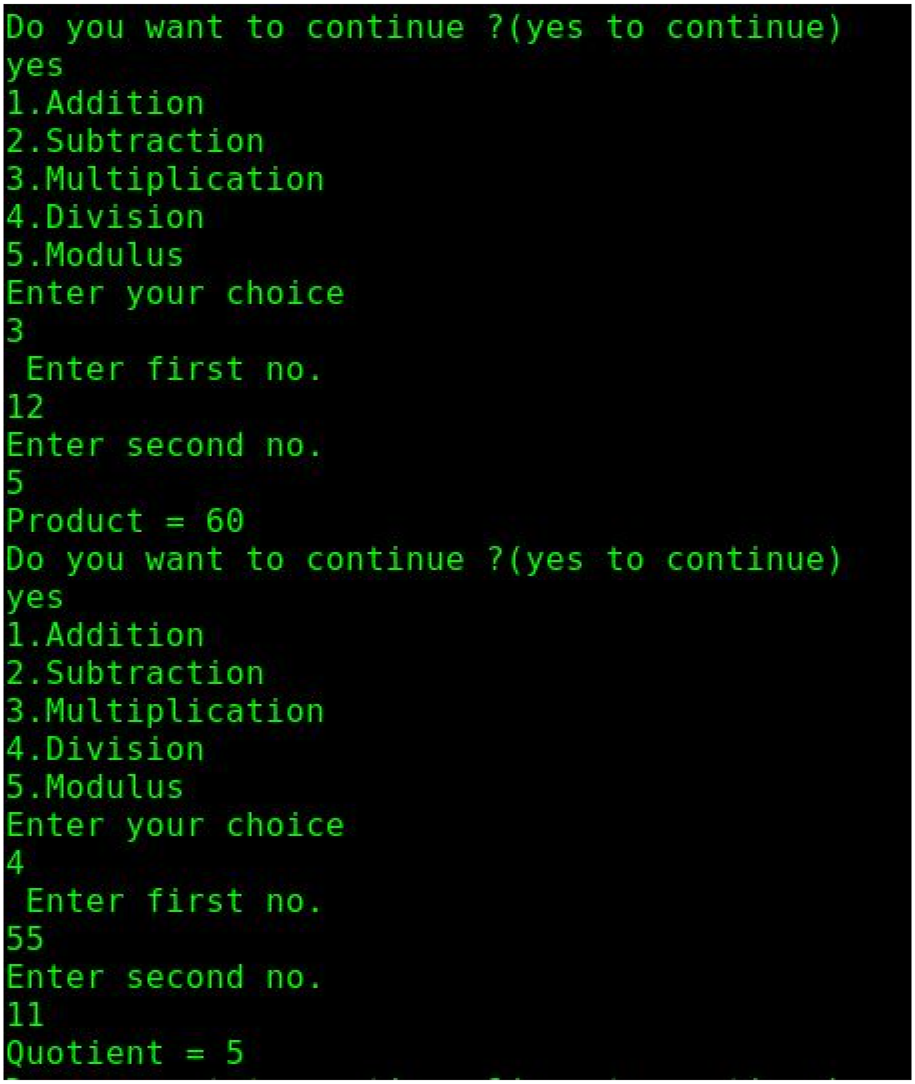
Program:

```
#! /bin/bash
i="yes"
while [ $i = "yes" ]
do
        echo "1.Addition"
        echo "2.Subtraction"
        echo "3.Multiplication"
        echo "4.Division"
        echo "5.Modulus"
        echo "Enter your choice"
        read ch
        echo " Enter first no."
        read a
        echo "Enter second no."
        read b
        case $ch in
                1) sum=`expr $a + $b`
                echo "Sum ="$sum;;
                2) diff=`expr $a - $b`
                echo "Difference = "$diff;;
                3) product=`expr $a \* $b`
                echo "Product = "$product;;
                4) if [ $b = 0 ]
                   then
                        echo Division by zero is not possible
                   else
                        quotient=`expr $a / $b`
                        echo "Quotient = "$quotient
                   fi;;
                5)modulus=`expr $a % $b`
                echo "Modulus = "$modulus;;
                *)  echo "Invalid choice";;
        esac
        echo "Do you want to continue ?(enter yes if you want to continue)"
        read i
        if [ $i != "yes" ]
        then
            exit
        fi
done
```

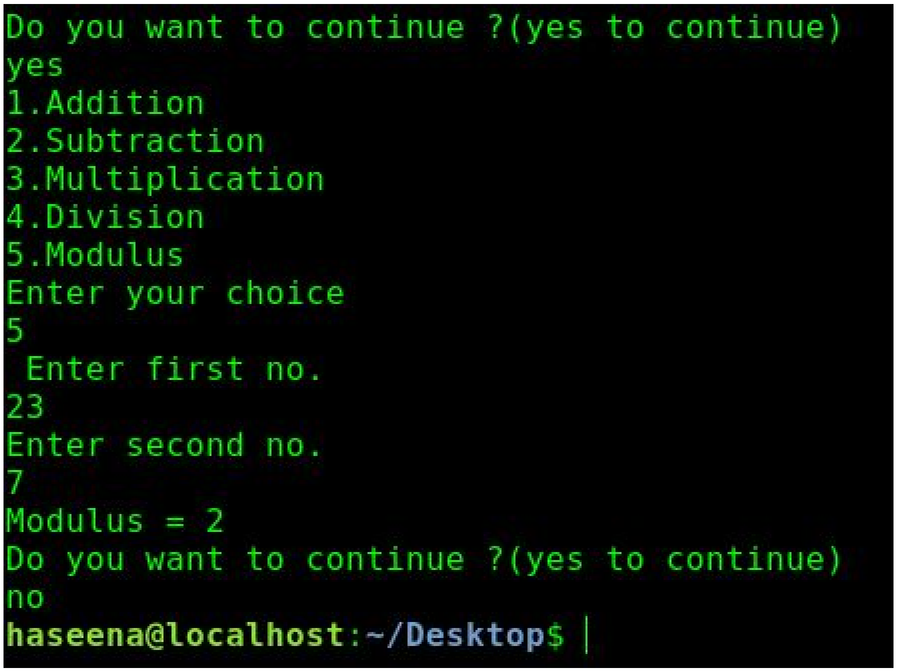
Output:



```
haseena@localhost:~/Desktop$ bash s1.sh
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulus
Enter your choice
1
 Enter first no.
12
Enter second no.
4
Sum =16
Do you want to continue ?(yes to continue)
yes
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulus
Enter your choice
2
 Enter first no.
15
Enter second no.
7
Difference = 8
Do you want to continue ?(yes to continue)
```



```
Do you want to continue ?(yes to continue)
yes
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulus
Enter your choice
3
 Enter first no.
12
Enter second no.
5
Product = 60
Do you want to continue ?(yes to continue)
yes
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulus
Enter your choice
4
 Enter first no.
55
Enter second no.
11
Quotient = 5
```



```
Do you want to continue ?(yes to continue)
yes
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulus
Enter your choice
5
 Enter first no.
23
Enter second no.
7
Modulus = 2
Do you want to continue ?(yes to continue)
no
haseena@localhost:~/Desktop$
```

4. Write a script called addnames that is to be called as follows ./addnames ulist Username . Here ulist is the name of the file that contains list of user names and username is a particular student’s username. The script should

- check that the correct number of arguments was received and print a message, in case the number of arguments is incorrect
- check whether the ulist file exists and print an error message if it does not
- check whether the username already exists in the file. If the username exists, print a message stating that the name already exists. Otherwise, add the username to the end of the list.

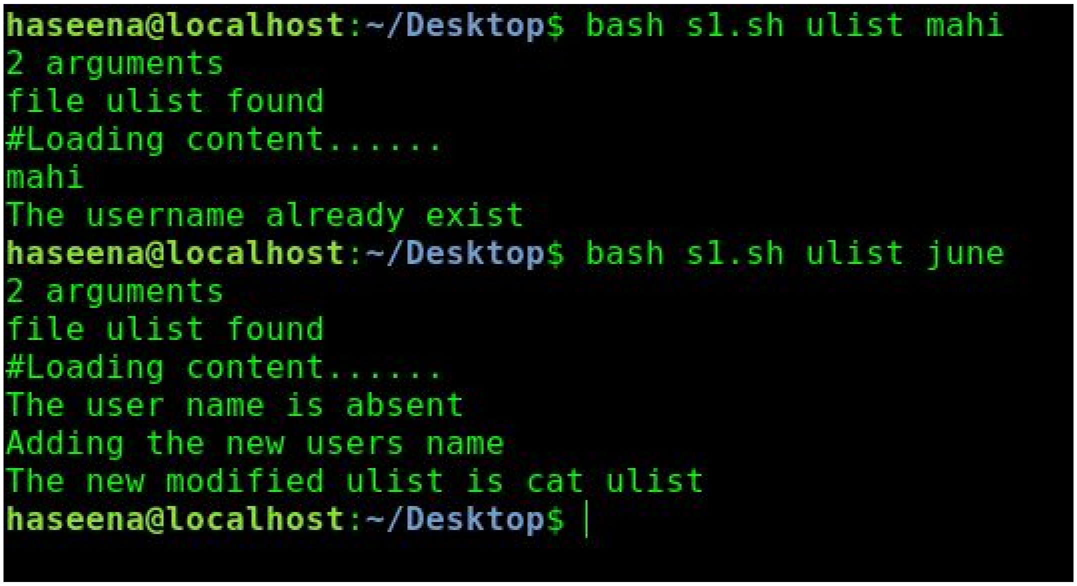
Algorithm:

1) Create a file named ulist and add usernames to it
2) Provide ulist and a username say, name as arguments
3) If $# equals 2 then do
4) Check file ulist exist or not
5) Check if ulist contains the username ‘name’
6) If not found then print “ name not found”
7) Then add ‘name’ to ulist
8) Else print “ name already exist’
9) If $# not equal to 2 then exit

Program:

```
#!  /bin/bash
if [ "$#" -eq 2 ]; then
        echo "2 arguments"
        if [ -f "$1" ]
        then
               echo "file" $1 "found"
                echo "#Loading content......"
                if grep $2 $1
                then
                        echo "The username already exist"
                else
                        echo "The user name is absent"
                        echo "Adding the new users name"
                        echo $2 >> $1
                        echo "The new modified" $1 "is" cat $1
                fi
        else
                echo "error file " $1 "not found"
        fi
else
echo "you must enter 2 arguments"
fi
```

Output:



```
haseena@localhost:~/Desktop$ bash s1.sh ulist mahi
2 arguments
file ulist found
#Loading content......
mahi
The username already exist
haseena@localhost:~/Desktop$ bash s1.sh ulist june
2 arguments
file ulist found
#Loading content......
The user name is absent
Adding the new users name
The new modified ulist is cat ulist
haseena@localhost:~/Desktop$ |
```

5. Write a Shell script which starts on system boot up and kills every process which

uses more than a specified amount of memory or CPU.

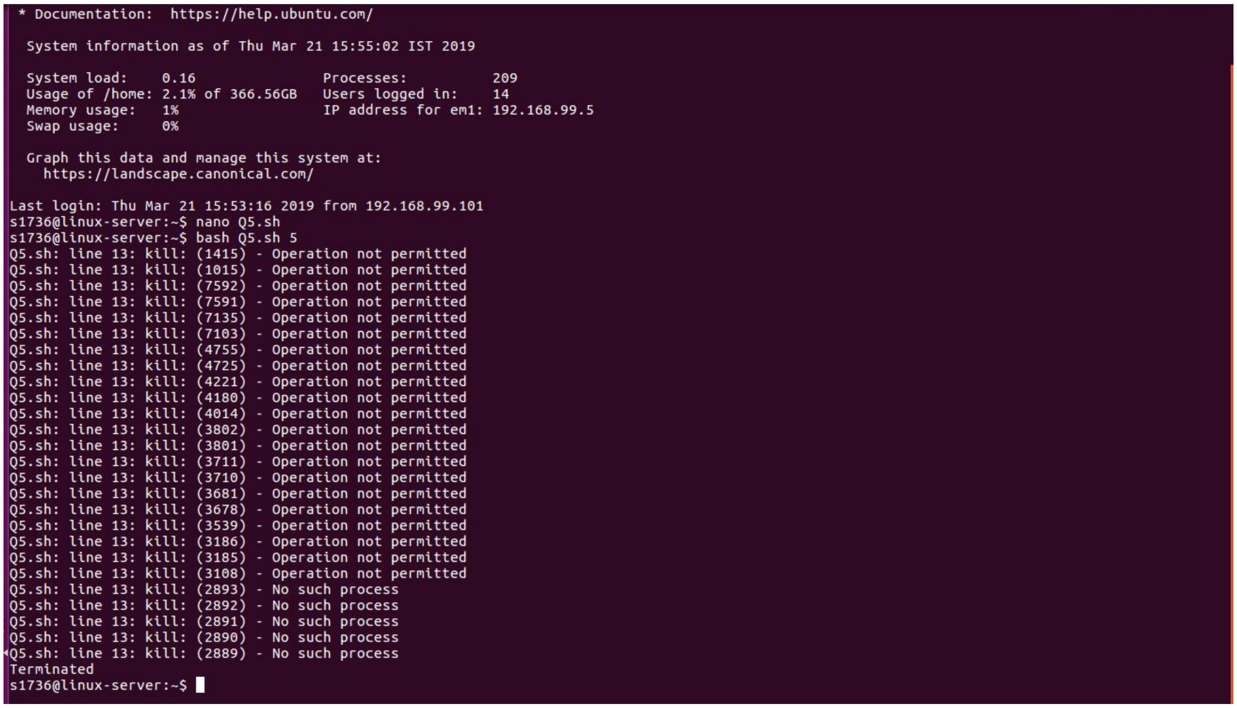
Algorithm:

1) Check CPU_USAGE
2) Set CPU_USAGE_THRESHOLD value
3) List all the running processes into PROCESS
4) If $CPU_USAGE   > $CPU_USAGE_THRESHOLD then
5) Kill $TOPPROCESS
6) Print top-most process killed $TOPPROCESS
7) Print CPU USAGE is at $CPU_LOAD
8) exit

Program:

```
while [ 1 ];
do
echo
echo checking for run-away process …
CPU_USAGE=$(uptime | cut -d"," -f4 | cut -d":" -f2 | cut -d" " -f2 | sed -e "s/\.//g")
CPU_USAGE_THRESHOLD=800
PROCESS=$(ps aux r)
TOPPROCESS=$(ps -eo pid -eo pcpu -eo command | sort -k 2 -r | grep -v PID | head -n
1)
if [ $CPU_USAGE -gt $CPU_USAGE_THRESHOLD] ; then
  kill -9 $(ps -eo pid | sort -k 1 -r | grep -v PID | head -n 1)
  kill -9 $(ps -eo pcpu | sort -k 1 -r | grep -v %CPU | head -n 1)
  kill -9 $TOPPROCESS
  echo system overloading!
  echo Top-most process killed $TOPPROCESS
      echo CPU USAGE is at $CPU_LOAD
else
    fi
    exit 0
    sleep 1;
    done
```

Output:



```
 * Documentation:  https://help.ubuntu.com/

  System information as of Thu Mar 21 15:55:02 IST 2019

  System load:    0.16               Processes:          209
  Usage of /home: 2.1% of 366.56GB   Users logged in:    14
  Memory usage:   1%                 IP address for em1: 192.168.99.5
  Swap usage:     0%

  Graph this data and manage this system at:
    https://landscape.canonical.com/

Last login: Thu Mar 21 15:53:16 2019 from 192.168.99.101
s1736@linux-server:~$ nano Q5.sh
s1736@linux-server:~$ bash Q5.sh 5
Q5.sh: line 13: kill: (1415) - Operation not permitted
Q5.sh: line 13: kill: (1015) - Operation not permitted
Q5.sh: line 13: kill: (7592) - Operation not permitted
Q5.sh: line 13: kill: (7591) - Operation not permitted
Q5.sh: line 13: kill: (7135) - Operation not permitted
Q5.sh: line 13: kill: (7103) - Operation not permitted
Q5.sh: line 13: kill: (4755) - Operation not permitted
Q5.sh: line 13: kill: (4725) - Operation not permitted
Q5.sh: line 13: kill: (4221) - Operation not permitted
Q5.sh: line 13: kill: (4180) - Operation not permitted
Q5.sh: line 13: kill: (4014) - Operation not permitted
Q5.sh: line 13: kill: (3802) - Operation not permitted
Q5.sh: line 13: kill: (3801) - Operation not permitted
Q5.sh: line 13: kill: (3711) - Operation not permitted
Q5.sh: line 13: kill: (3710) - Operation not permitted
Q5.sh: line 13: kill: (3681) - Operation not permitted
Q5.sh: line 13: kill: (3678) - Operation not permitted
Q5.sh: line 13: kill: (3539) - Operation not permitted
Q5.sh: line 13: kill: (3186) - Operation not permitted
Q5.sh: line 13: kill: (3185) - Operation not permitted
Q5.sh: line 13: kill: (3108) - Operation not permitted
Q5.sh: line 13: kill: (2893) - No such process
Q5.sh: line 13: kill: (2892) - No such process
Q5.sh: line 13: kill: (2891) - No such process
Q5.sh: line 13: kill: (2890) - No such process
Q5.sh: line 13: kill: (2889) - No such process
Terminated
s1736@linux-server:~$ 
```

CONCLUSION

Verified the outputs for the above questions and improved the hold over bash shell scripting.