PRACTICAL NO.1

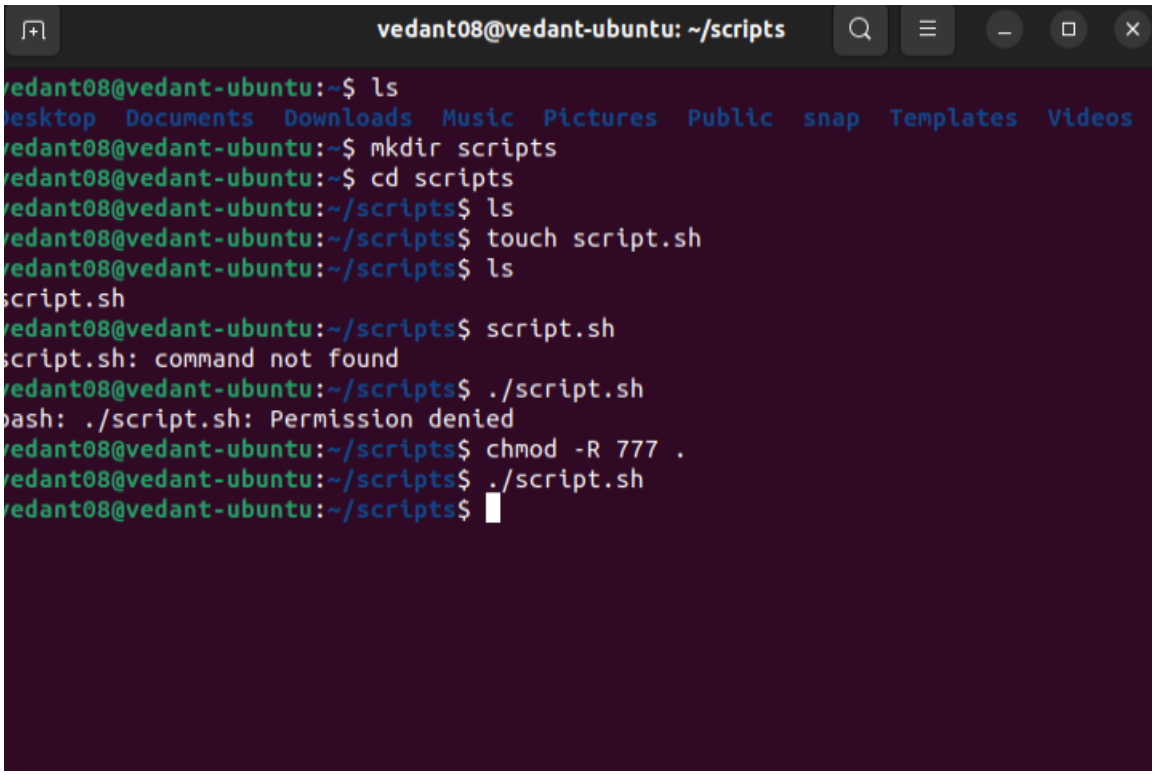
NAME: VEDANT BHUTADA

ROLL: 69

Aim: To study and execute basic Linux System Commands and write shell scripts to display the system particulars (processor, processes and memory).

Task1: Execution of all the steps given in below snippet. (touch, ls, cd, mkdir , chmod –R 777, how to create and handle and execute script.

Output:



2) Create a shell script to perform the following tasks.

(a) Find the number of processors your machine has.

(b) How many cores does your machine have?

(c) What is the frequency of each processor?

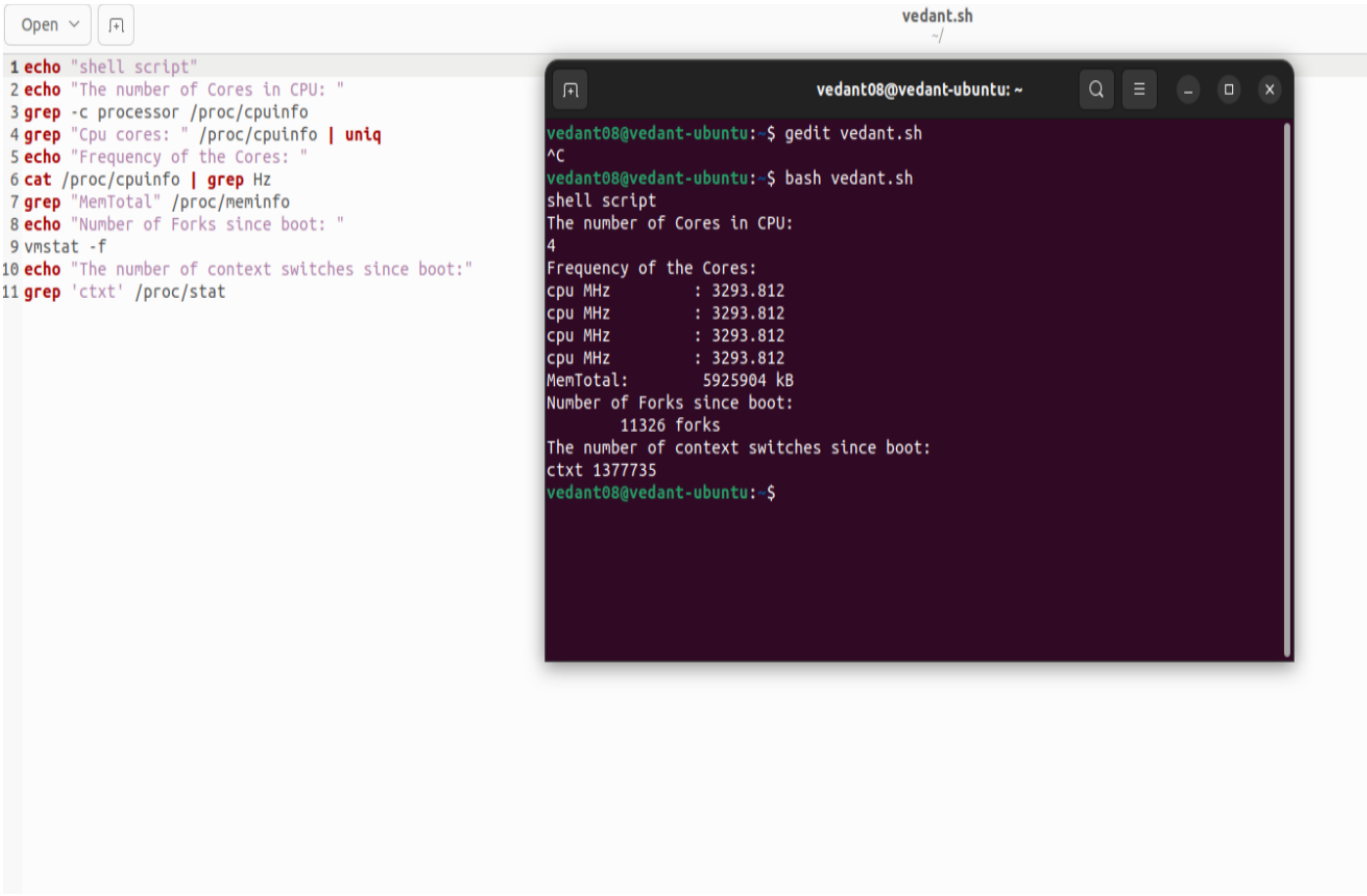
(d) How much physical memory does your system have?

(e) How much of this memory is free?

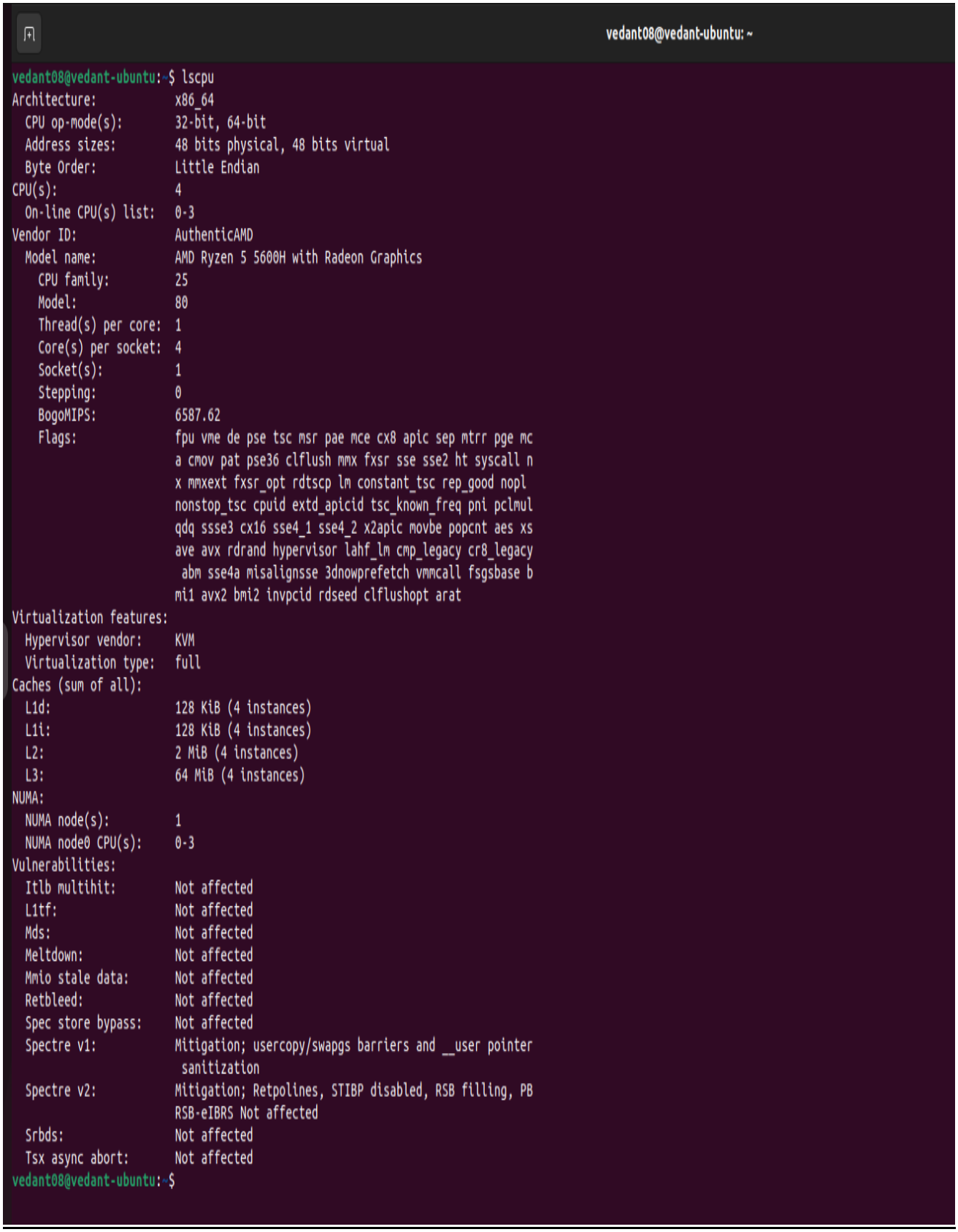
(f) What is total number of number of forks since the boot in the system?

(g) How many context switches has the system performed since bootup?

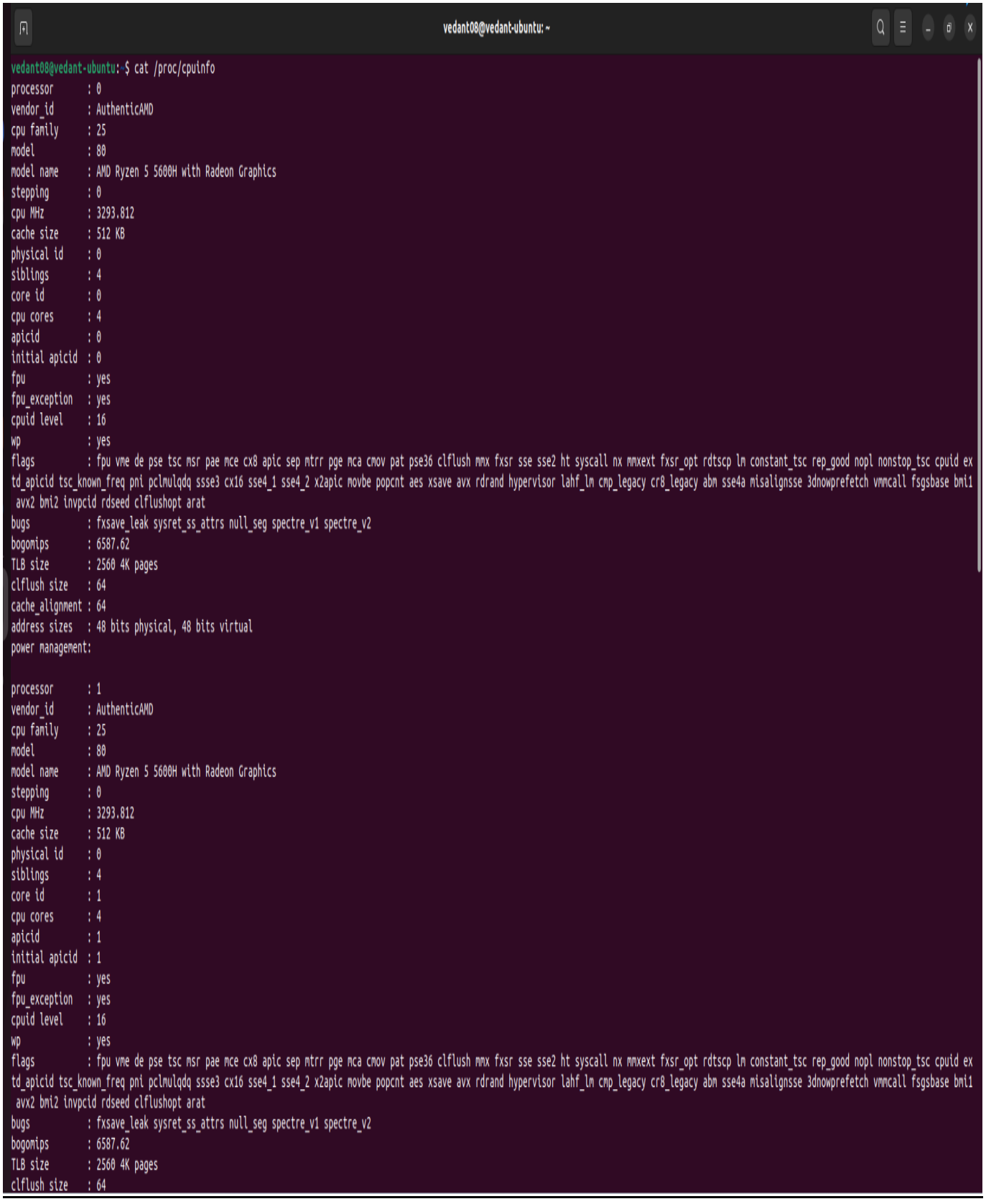
Output:

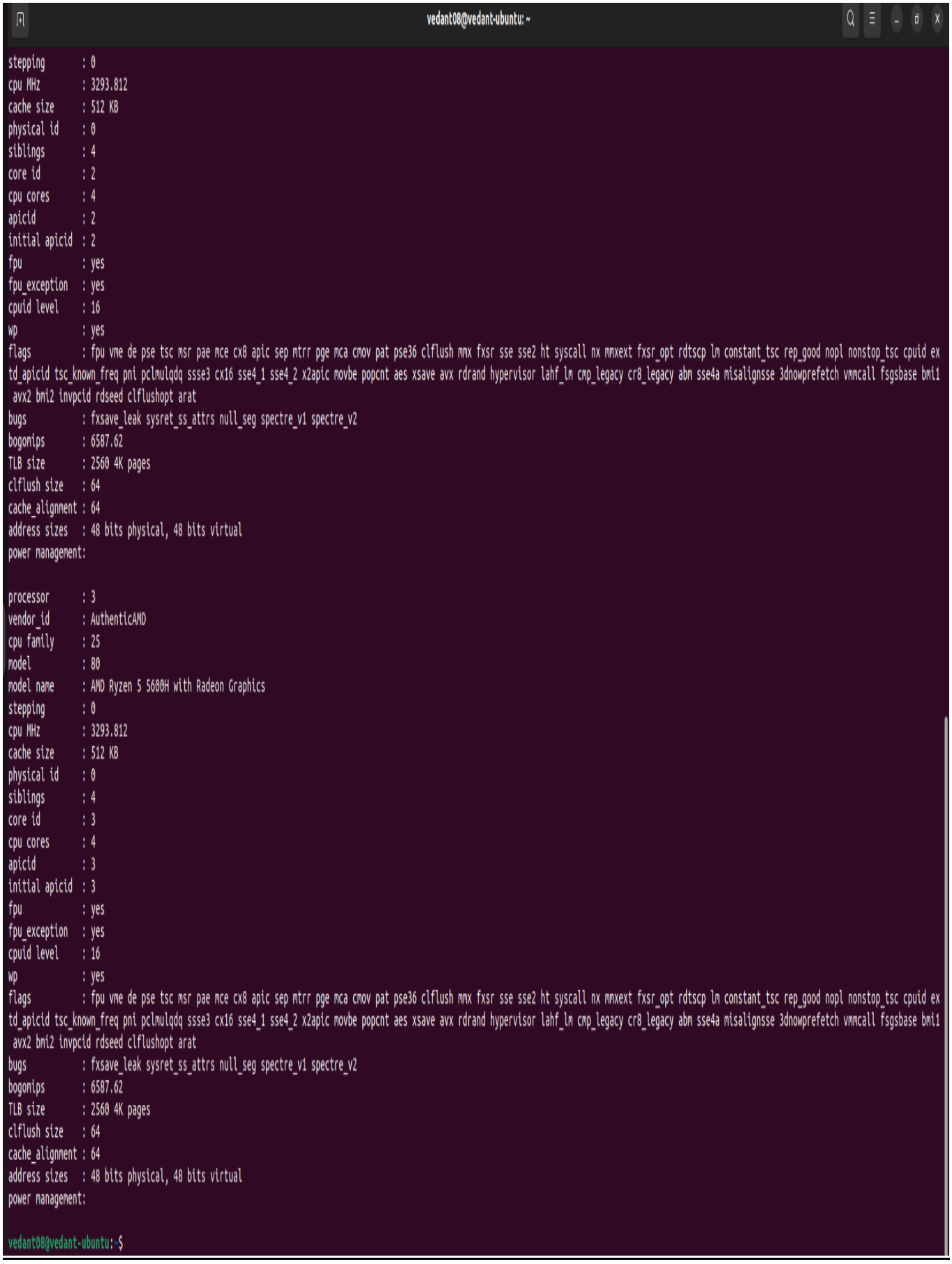


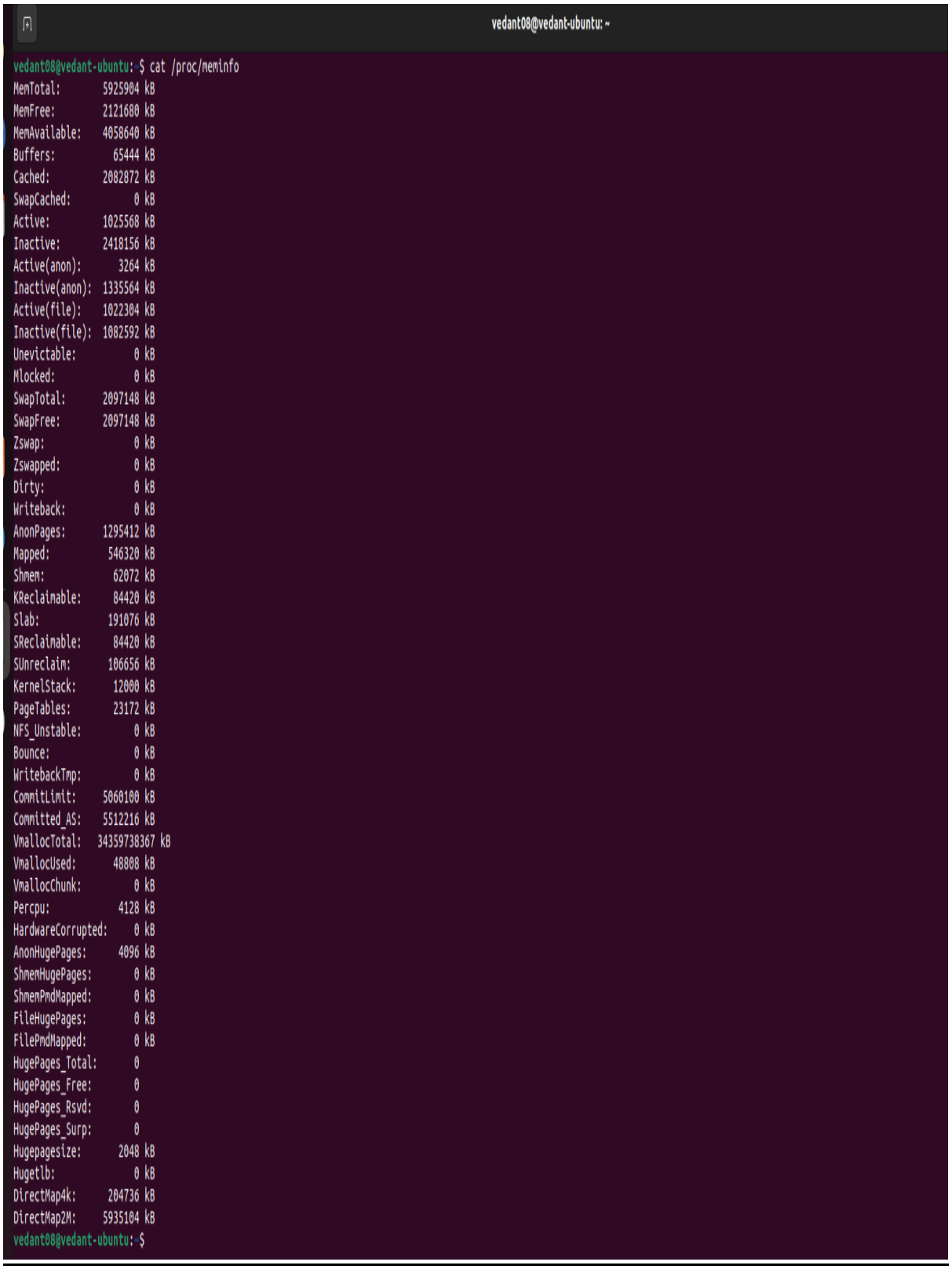
Output: lscpu


vedant08@vedant-ubuntu: ~
vedant08@vedant-ubuntu:~$ lscpu
Architecture: x86_64
CPU op-mode(s): 32-bit, 64-bit
Address sizes: 48 bits physical, 48 bits virtual
Byte Order: Little Endian
CPU(s): 4
On-line CPU(s) list: 0-3
Vendor ID: AuthenticAMD
Model name: AMD Ryzen 5 5600H with Radeon Graphics
CPU family: 25
Model: 80
Thread(s) per core: 1
Core(s) per socket: 4
Socket(s): 1
Stepping: 0
BogoMIPS: 6587.62
Flags: fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx mmxext fxsr_opt rdtscp lm constant_tsc rep_good nopl nonstop_tsc cpuid extd_apicid tsc_known_freq pni pclmulqdq ssse3 cx16 sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx rdrand hypervisor lahf_lm cmp_legacy cr8_legacy abm sse4a misalignsse 3dnowprefetch vmmcall fsgsbase bmi1 avx2 bmi2 invpcid rdseed clflushopt arat
Virtualization features:
Hypervisor vendor: KVM
Virtualization type: full
Caches (sum of all):
L1d: 128 KiB (4 instances)
L1i: 128 KiB (4 instances)
L2: 2 MiB (4 instances)
L3: 64 MiB (4 instances)
NUMA:
NUMA node(s): 1
NUMA node0 CPU(s): 0-3
Vulnerabilities:
Itlb multihit: Not affected
L1tf: Not affected
Mds: Not affected
Meltdown: Not affected
Mmio stale data: Not affected
Retbleed: Not affected
Spec store bypass: Not affected
Spectre v1: Mitigation; usercopy/swapgs barriers and __user pointer sanitization
Spectre v2: Mitigation; Retpolines, STIBP disabled, RSB filling, PBRSB-eIBRS Not affected
Srbds: Not affected
Tsx async abort: Not affected
vedant08@vedant-ubuntu:~$

Output:


vedant08@vedant-ubuntu: ~
vedant08@vedant-ubuntu:~$ cat /proc/cpuinfo
processor : 0
vendor_id : AuthenticAMD
cpu family : 25
model : 80
model name : AMD Ryzen 5 5600H with Radeon Graphics
stepping : 0
cpu MHz : 3293.812
cache size : 512 KB
physical id : 0
siblings : 4
core id : 0
cpu cores : 4
apicid : 0
initial apicid : 0
fpu : yes
fpu_exception : yes
cpuid level : 16
wp : yes
flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx mmxext fxsr_opt rdtscp lm constant_tsc rep_good nopl nonstop_tsc cpuid extd_apicid tsc_known_freq pni pclmulqdq ssse3 cx16 sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx rdrand hypervisor lahf_lm cmp_legacy cr8_legacy abm sse4a misalignsse 3dnowprefetch vmmcall fsgsbase bmi1 avx2 bmi2 invpcid rdseed clflushopt arat
bugs : fxsave_leak sysret_ss_attrs null_seg spectre_v1 spectre_v2
bogomips : 6587.62
TLB size : 2560 4K pages
clflush size : 64
cache_alignment : 64
address sizes : 48 bits physical, 48 bits virtual
power management:
processor : 1
vendor_id : AuthenticAMD
cpu family : 25
model : 80
model name : AMD Ryzen 5 5600H with Radeon Graphics
stepping : 0
cpu MHz : 3293.812
cache size : 512 KB
physical id : 0
siblings : 4
core id : 1
cpu cores : 4
apicid : 1
initial apicid : 1
fpu : yes
fpu_exception : yes
cpuid level : 16
wp : yes
flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx mmxext fxsr_opt rdtscp lm constant_tsc rep_good nopl nonstop_tsc cpuid extd_apicid tsc_known_freq pni pclmulqdq ssse3 cx16 sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx rdrand hypervisor lahf_lm cmp_legacy cr8_legacy abm sse4a misalignsse 3dnowprefetch vmmcall fsgsbase bmi1 avx2 bmi2 invpcid rdseed clflushopt arat
bugs : fxsave_leak sysret_ss_attrs null_seg spectre_v1 spectre_v2
bogomips : 6587.62
TLB size : 2560 4K pages
clflush size : 64


vedant08@vedant-ubuntu: ~
stepping : 0
cpu MHz : 3293.812
cache size : 512 KB
physical id : 0
siblings : 4
core id : 2
cpu cores : 4
apicid : 2
initial apicid : 2
fpu : yes
fpu_exception : yes
cpuid level : 16
wp : yes
flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx mmxext fxsr_opt rdtscp lm constant_tsc rep_good nopl nonstop_tsc cpuid extd_apicid tsc_known_freq pni pclmulqdq ssse3 cx16 sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx rdrand hypervisor lahf_lm cmp_legacy cr8_legacy abm sse4a misalignsse 3dnowprefetch vmmcall fsgsbase bmi1 avx2 bmi2 invpcid rdseed clflushopt arat
bugs : fxsave_leak sysret_ss_attrs null_seg spectre_v1 spectre_v2
bogomips : 6587.62
TLB size : 2560 4K pages
clflush size : 64
cache_alignment : 64
address sizes : 48 bits physical, 48 bits virtual
power management:
processor : 3
vendor_id : AuthenticAMD
cpu family : 25
model : 80
model name : AMD Ryzen 5 5600H with Radeon Graphics
stepping : 0
cpu MHz : 3293.812
cache size : 512 KB
physical id : 0
siblings : 4
core id : 3
cpu cores : 4
apicid : 3
initial apicid : 3
fpu : yes
fpu_exception : yes
cpuid level : 16
wp : yes
flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht syscall nx mmxext fxsr_opt rdtscp lm constant_tsc rep_good nopl nonstop_tsc cpuid extd_apicid tsc_known_freq pni pclmulqdq ssse3 cx16 sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx rdrand hypervisor lahf_lm cmp_legacy cr8_legacy abm sse4a misalignsse 3dnowprefetch vmmcall fsgsbase bmi1 avx2 bmi2 invpcid rdseed clflushopt arat
bugs : fxsave_leak sysret_ss_attrs null_seg spectre_v1 spectre_v2
bogomips : 6587.62
TLB size : 2560 4K pages
clflush size : 64
cache_alignment : 64
address sizes : 48 bits physical, 48 bits virtual
power management:
vedant08@vedant-ubuntu:~$

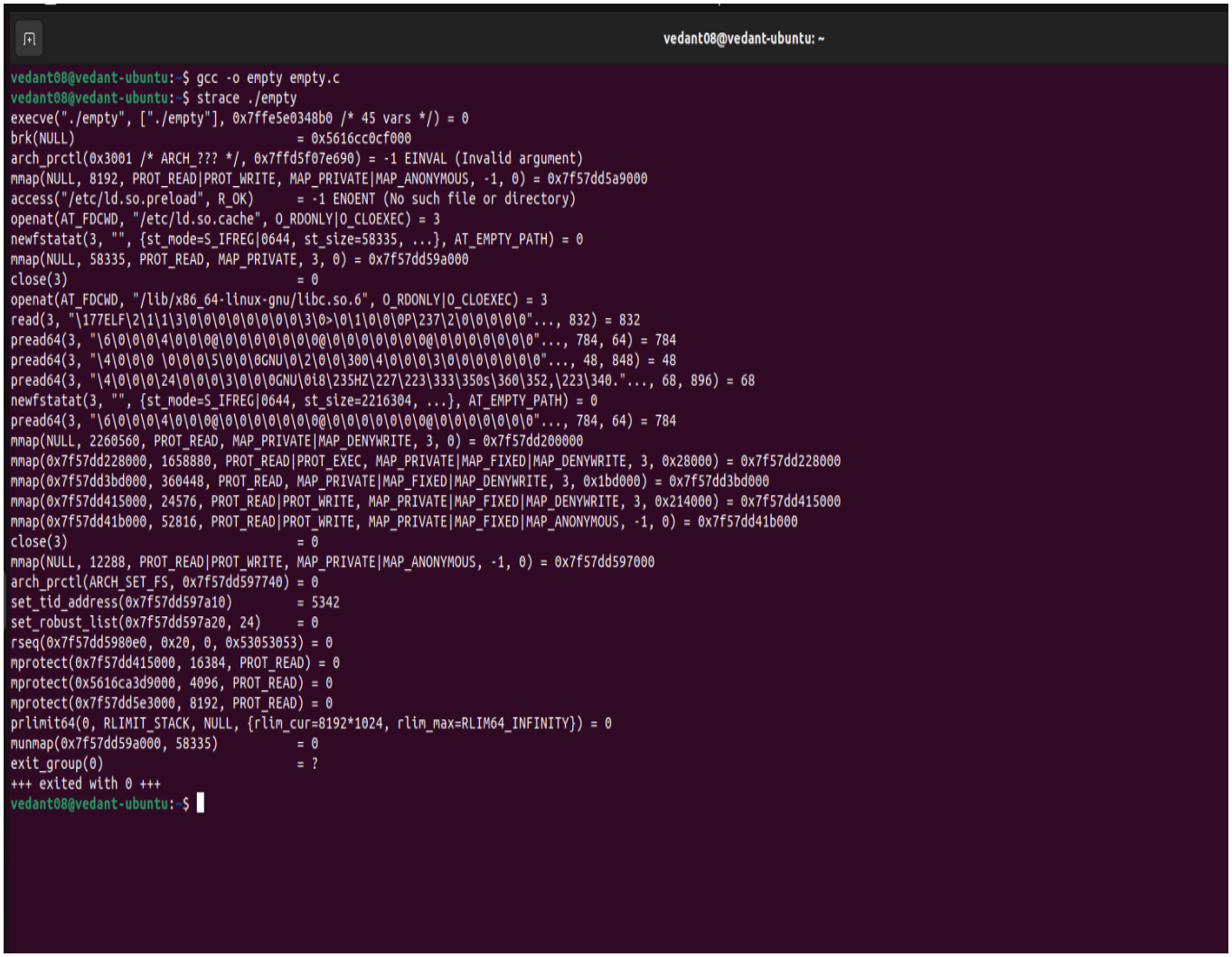

vedant08@vedant-ubuntu: ~
vedant08@vedant-ubuntu:~$ cat /proc/meminfo
MemTotal: 5925904 kB
MemFree: 2121680 kB
MemAvailable: 4058640 kB
Buffers: 65444 kB
Cached: 2082872 kB
SwapCached: 0 kB
Active: 1025568 kB
Inactive: 2418156 kB
Active(anon): 3264 kB
Inactive(anon): 1335564 kB
Active(file): 1022304 kB
Inactive(file): 1082592 kB
Unevictable: 0 kB
Mlocked: 0 kB
SwapTotal: 2097148 kB
SwapFree: 2097148 kB
Zswap: 0 kB
Zswapped: 0 kB
Dirty: 0 kB
Writeback: 0 kB
AnonPages: 1295412 kB
Mapped: 546320 kB
Shmem: 62072 kB
KReclaimable: 84420 kB
Slab: 191076 kB
SReclaimable: 84420 kB
SUnreclaim: 106656 kB
KernelStack: 12000 kB
PageTables: 23172 kB
NFS_Unstable: 0 kB
Bounce: 0 kB
WritebackTmp: 0 kB
CommitLimit: 5060100 kB
Committed_AS: 5512216 kB
VmallocTotal: 34359738367 kB
VmallocUsed: 48808 kB
VmallocChunk: 0 kB
Percpu: 4128 kB
HardwareCorrupted: 0 kB
AnonHugePages: 4096 kB
ShmemHugePages: 0 kB
ShmemPmdMapped: 0 kB
FileHugePages: 0 kB
FilePmdMapped: 0 kB
HugePages_Total: 0
HugePages_Free: 0
HugePages_Rsvd: 0
HugePages_Surp: 0
Hugepagesize: 2048 kB
Hugetlb: 0 kB
DirectMap4k: 204736 kB
DirectMap2M: 5935104 kB
vedant08@vedant-ubuntu:~$

3) Run strace along with the binary program of empty.c given in subdirectory strace .

What do you think the output of strace indicates in this case? How many different system

call functions do you see? Next, use strace along with another binary program of hello.c (which is in the same directory).Compare that is observe the two strace outputs a. Which part of the strace output is common, and which part has to do with the specific program? b. Observe the list of all unique system calls along with input and output parameters and study those system call ?

Output: empty.c file



Output: hello.c file

```
vedant08@vedant-ubuntu:~$ gcc -o hello hello.c
vedant08@vedant-ubuntu:~$ strace ./empty
execve("./empty", ["./empty"], 0x7ffcc13291a0 /* 45 vars */) = 0
brk(NULL)                               = 0x559d18028000
arch_prctl(0x3001 /* ARCH_??? */, 0x7ffeb57b3390) = -1 EINVAL (Invalid argument)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f4cad1f7000
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=58335, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 58335, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f4cad1e8000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\237\2\0\0\0\0\0"..., 832) = 832
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0\0@\0\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784
pread64(3, "\4\0\0\0 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) = 48
pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0i8\235HZ\227\223\333\350s\360\352,\223\340."..., 68, 896) = 68
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=2216304, ...}, AT_EMPTY_PATH) = 0
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0\0@\0\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784
mmap(NULL, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f4cace00000
mmap(0x7f4cace28000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7f4cace28000
mmap(0x7f4cacfbd000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1bd000) = 0x7f4cacfbd000
mmap(0x7f4cad015000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x214000) = 0x7f4cad015000
mmap(0x7f4cad01b000, 52816, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f4cad01b000
close(3)                                = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f4cad1e5000
arch_prctl(ARCH_SET_FS, 0x7f4cad1e5740) = 0
set_tid_address(0x7f4cad1e5a10)         = 5567
set_robust_list(0x7f4cad1e5a20, 24)     = 0
rseq(0x7f4cad1e60e0, 0x20, 0, 0x53053053) = 0
mprotect(0x7f4cad015000, 16384, PROT_READ) = 0
mprotect(0x559d179b4000, 4096, PROT_READ) = 0
mprotect(0x7f4cad231000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
munmap(0x7f4cad1e8000, 58335)           = 0
exit_group(0)                           = ?
+++ exited with 0 +++
```

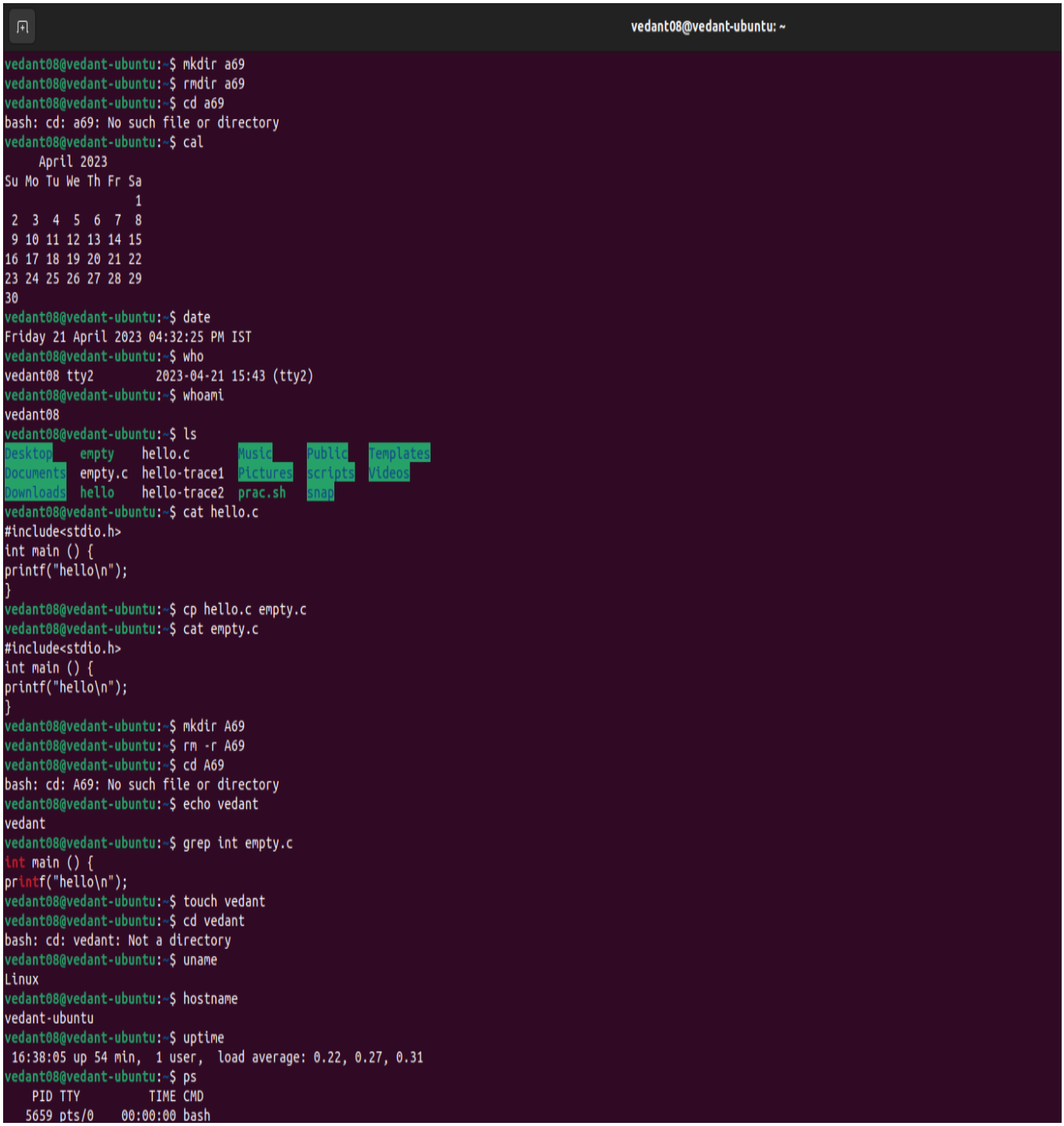
```
vedant08@vedant-ubuntu: ~
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f4cad1e5000
arch_prctl(ARCH_SET_FS, 0x7f4cad1e5740) = 0
set_tid_address(0x7f4cad1e5a10)         = 5567
set_robust_list(0x7f4cad1e5a20, 24)     = 0
rseq(0x7f4cad1e60e0, 0x20, 0, 0x53053053) = 0
mprotect(0x7f4cad015000, 16384, PROT_READ) = 0
mprotect(0x559d179b4000, 4096, PROT_READ) = 0
mprotect(0x7f4cad231000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
munmap(0x7f4cad1e8000, 58335)           = 0
exit_group(0)                           = ?
+++ exited with 0 +++
vedant08@vedant-ubuntu:~$ strace ./hello
execve("./hello", ["./hello"], 0x7ffcd262aab0 /* 45 vars */) = 0
brk(NULL)                               = 0x55a5d65eb000
arch_prctl(0x3001 /* ARCH_??? */, 0x7ffcb2195ac0) = -1 EINVAL (Invalid argument)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f17f40da000
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=58335, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 58335, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f17f40cb000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\237\2\0\0\0\0\0"..., 832) = 832
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0\0@\0\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784
pread64(3, "\4\0\0\0 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) = 48
pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\018\235HZ\227\223\333\350s\360\352,\223\340."..., 68, 896) = 68
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=2216304, ...}, AT_EMPTY_PATH) = 0
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0\0@\0\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784
mmap(NULL, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f17f3e00000
mmap(0x7f17f3e28000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7f17f3e28000
mmap(0x7f17f3fbd000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1bd000) = 0x7f17f3fbd000
mmap(0x7f17f4015000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x214000) = 0x7f17f4015000
mmap(0x7f17f401b000, 52816, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f17f401b000
close(3)                                = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f17f40c8000
arch_prctl(ARCH_SET_FS, 0x7f17f40c8740) = 0
set_tid_address(0x7f17f40c8a10)         = 5571
set_robust_list(0x7f17f40c8a20, 24)     = 0
rseq(0x7f17f40c90e0, 0x20, 0, 0x53053053) = 0
mprotect(0x7f17f4015000, 16384, PROT_READ) = 0
mprotect(0x55a5d5d3a000, 4096, PROT_READ) = 0
mprotect(0x7f17f4114000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
munmap(0x7f17f40cb000, 58335)           = 0
newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}, AT_EMPTY_PATH) = 0
getrandom("\x51\xe0\xc5\xe4\x59\xff\x8e\x9e", 8, GRND_NONBLOCK) = 8
brk(NULL)                               = 0x55a5d65eb000
brk(0x55a5d660c000)                     = 0x55a5d660c000
write(1, "hello\n", 6hello
)                  = 6
exit_group(0)                           = ?
+++ exited with 0 +++
vedant08@vedant-ubuntu:~$
```

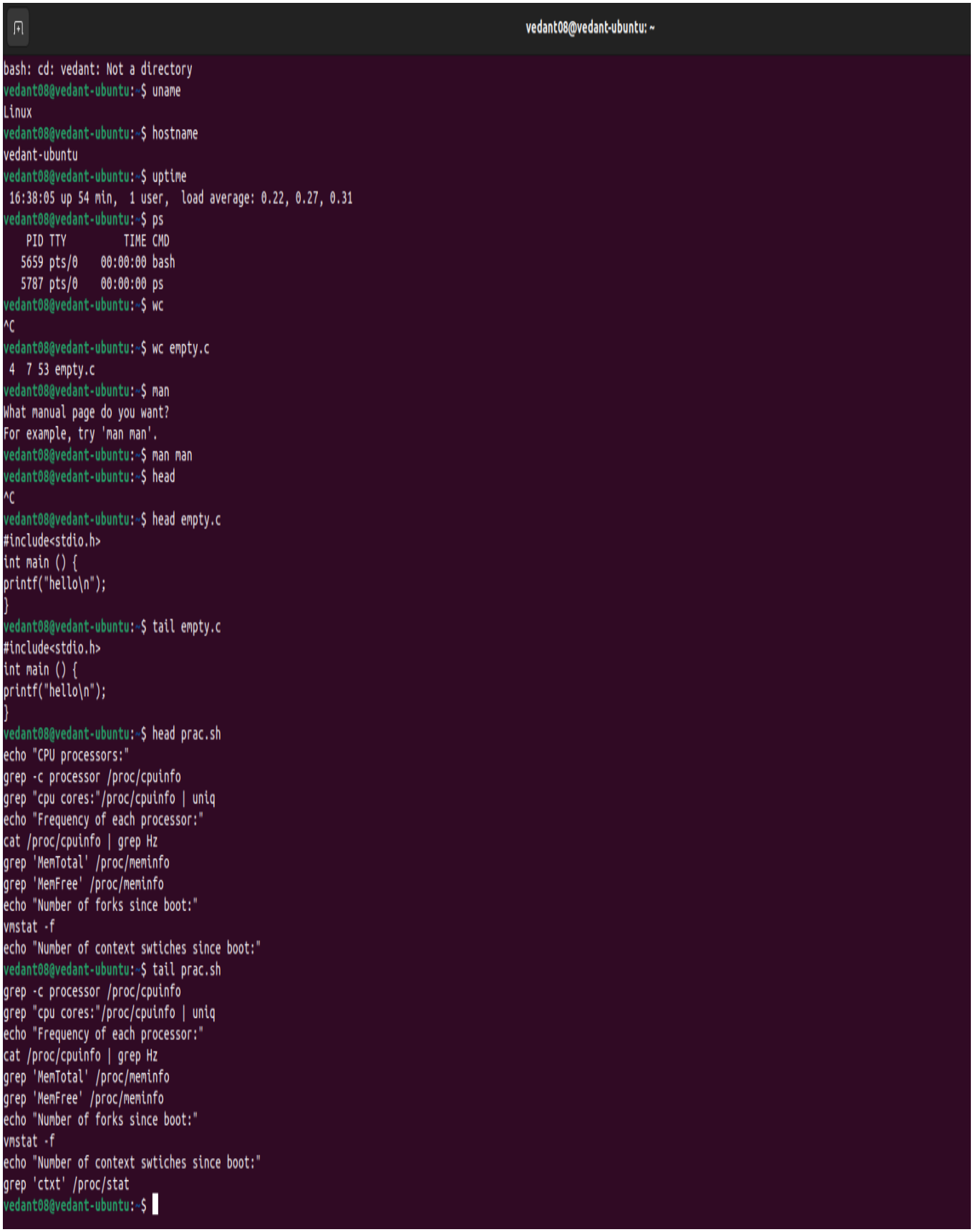
Output: Difference

```
vedant08@vedant-ubuntu:~$ strace -o hello-trace1 ./hello
hello
vedant08@vedant-ubuntu:~$ strace -o hello-trace2 ./hello
hello
vedant08@vedant-ubuntu:~$ diff hello-trace1 hello-trace2
1,4c1,4
< execve("./hello", ["./hello"], 0x7ffe7029bad0 /* 45 vars */) = 0
< brk(NULL)                               = 0x5583eb378000
< arch_prctl(0x3001 /* ARCH_??? */, 0x7ffcd392b770) = -1 EINVAL (Invalid argument)
< mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7efc56e45000
---
> execve("./hello", ["./hello"], 0x7ffffc4a0ce0 /* 45 vars */) = 0
> brk(NULL)                               = 0x5618e5a4c000
> arch_prctl(0x3001 /* ARCH_??? */, 0x7ffc950f3850) = -1 EINVAL (Invalid argument)
> mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fbb4ee09000
8c8
< mmap(NULL, 58335, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7efc56e36000
---
> mmap(NULL, 58335, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7fbb4edfa000
17,21c17,21
< mmap(NULL, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7efc56c00000
< mmap(0x7efc56c28000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7efc56c28000
< mmap(0x7efc56dbd000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1bd000) = 0x7efc56dbd000
< mmap(0x7efc56e15000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x214000) = 0x7efc56e15000
< mmap(0x7efc56e1b000, 52816, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7efc56e1b000
---
> mmap(NULL, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fbb4ea00000
> mmap(0x7fbb4ea28000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7fbb4ea28000
> mmap(0x7fbb4ebbd000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1bd000) = 0x7fbb4ebbd000
> mmap(0x7fbb4ec15000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x214000) = 0x7fbb4ec15000
> mmap(0x7fbb4ec1b000, 52816, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7fbb4ec1b000
23,30c23,30
< mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7efc56e33000
< arch_prctl(ARCH_SET_FS, 0x7efc56e33740) = 0
< set_tid_address(0x7efc56e33a10)         = 5578
< set_robust_list(0x7efc56e33a20, 24)     = 0
< rseq(0x7efc56e340e0, 0x20, 0, 0x53053053) = 0
< mprotect(0x7efc56e15000, 16384, PROT_READ) = 0
< mprotect(0x5583eaf04000, 4096, PROT_READ) = 0
< mprotect(0x7efc56e7f000, 8192, PROT_READ) = 0
---
> mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fbb4edf7000
> arch_prctl(ARCH_SET_FS, 0x7fbb4edf7740) = 0
> set_tid_address(0x7fbb4edf7a10)         = 5595
> rseq(0x7fbb4edf80e0, 0x20, 0, 0x53053053) = 0
> mprotect(0x7fbb4ec15000, 16384, PROT_READ) = 0
> mprotect(0x5618e5381000, 4096, PROT_READ) = 0
> mprotect(0x7fbb4ee43000, 8192, PROT_READ) = 0
32c32
```

```
23,30c23,30
< mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7efc56e33000
< arch_prctl(ARCH_SET_FS, 0x7efc56e33740) = 0
< set_tid_address(0x7efc56e33a10)         = 5578
< set_robust_list(0x7efc56e33a20, 24)     = 0
< rseq(0x7efc56e340e0, 0x20, 0, 0x53053053) = 0
< mprotect(0x7efc56e15000, 16384, PROT_READ) = 0
< mprotect(0x5583eaf04000, 4096, PROT_READ) = 0
< mprotect(0x7efc56e7f000, 8192, PROT_READ) = 0
---
> mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fbb4edf7000
> arch_prctl(ARCH_SET_FS, 0x7fbb4edf7740) = 0
> set_tid_address(0x7fbb4edf7a10)         = 5595
> set_robust_list(0x7fbb4edf7a20, 24)     = 0
> rseq(0x7fbb4edf80e0, 0x20, 0, 0x53053053) = 0
> mprotect(0x7fbb4ec15000, 16384, PROT_READ) = 0
> mprotect(0x5618e5381000, 4096, PROT_READ) = 0
> mprotect(0x7fbb4ee43000, 8192, PROT_READ) = 0
32c32
< munmap(0x7efc56e36000, 58335)           = 0
---
> munmap(0x7fbb4edfa000, 58335)           = 0
34,36c34,36
< getrandom("\xa1\xb0\xbb\x80\xf7\xb4\xd7\x8e", 8, GRND_NONBLOCK) = 8
< brk(NULL)                               = 0x5583eb378000
< brk(0x5583eb399000)                     = 0x5583eb399000
---
> getrandom("\x83\xe9\x62\xad\x7f\xf8\xf1\xb8", 8, GRND_NONBLOCK) = 8
> brk(NULL)                               = 0x5618e5a4c000
> brk(0x5618e5a6d000)                     = 0x5618e5a6d000
vedant08@vedant-ubuntu:~$ ^C
vedant08@vedant-ubuntu:~$
```

4) Execute following commands: 1. clear 2. Cal 3. who&whoami 4. Date 5. Mkdir 6. Rm 7. Cat 8. Cd 9. Cp 10. Grep 11. ls (execute all options) 12. Mv 13. Rm 14. Rmdir 15. Echo 16. Uptime 17. uname, hostname 18. Touch 19. cut 20. Head, tail 21. Ps 22. chmod (execute for all permissions, (reference: OS lab manual pdf , pg no. 34) 23. Wc 24. Chown 25. Man





Result: Basic Linux System Commands and shell scripts to display the system particulars (processor, processes and memory) has been implemented.