Roll No - 160050064
Name - Satti Vamsi Krishna Reddy

==

1. (a) CPU Sockets = 1, CPU cores = 4 each, no of CPUs = 4 (using proc/cpuinfo)
(b) 3.0 GHz capable. Currenty 4 processors running at speeds 1614.023, 1599.960, 1688.906, 1678.593 MHz (using proc/cpuinfo)
(c) 8140356 kB (using proc/meminfo)
(d) free - 5091248 kB, available - 6655988 kB. Available memory is one that can be used in addition to free memory such as cached memory of any process and such memory that can be freed if required. (using proc/meminfo)
(e) (ps -u labuser | wc-l) gives 96 as the number of processes associated with my user session. Similar counts exist for all other users too.
(f) 11736775 (using proc/stat file)
(g) Its 0 bytes. This is because these are not files being stored anywhere in the physical disk, but are created dynamically when user tries to access them (based on current PC's status)

==

2. Memory1 -

VmSize:	8136 kB
VmRSS:	648 kB

Memory2 -

VmSize:	12044 kB
VmRSS:	624 kB

Memory3 -

VmSize:	8140 kB
VmRSS:	3136 kB

Memory4 -

VmSize:	8136 kB
VmRSS:	4972 kB

The 2nd program is demanding relatively lot more (#define ARRAY_SZIE 200000) at-a-time hence the VmSize is large. Whereas, in the VmRSS takes into account the cumulative memory being utilized by the program, which is according to the code larger for 3, 4th program which contain additional for loops. (more loops imply more VMRSS memory as is the case between 3 adn 4th memory .c files)

==

3.

```
15950 pts/2 subprocesses
15951 pts/2 subprocesses
15952 pts/2 subprocesses
15953 pts/2 subprocesses
15954 pts/2 subprocesses
```

15955 pts/2 subprocesses
15956 pts/2 subprocesses

(obtained using ps -all | grep subprocesses command)

Total there are - 7 child processes.

==

4.

(a)

[EMPTY] -

```
execve("./empty", ["./empty"], [/* 65 vars */]) = 0
brk(NULL)                               =
0x1083000=====================================================================================================================

access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)
open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=125322, ...}) = 0
mmap(NULL, 125322, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f8bd1c35000
close(3)                                = 0
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
open("/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\t\2\0\0\0\0\0"..., 832) = 832
fstat(3, {st_mode=S_IFREG|0755, st_size=1868984, ...}) = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7f8bd1c34000
mmap(NULL, 3971488, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3, 0)
= 0x7f8bd1665000
mprotect(0x7f8bd1825000, 2097152, PROT_NONE) = 0
mmap(0x7f8bd1a25000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x1c0000) = 0x7f8bd1a25000
mmap(0x7f8bd1a2b000, 14752, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_ANONYMOUS, -1, 0) = 0x7f8bd1a2b000
close(3)                                = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7f8bd1c33000
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7f8bd1c32000
arch_prctl(ARCH_SET_FS, 0x7f8bd1c33700) = 0
mprotect(0x7f8bd1a25000, 16384, PROT_READ) = 0
mprotect(0x600000, 4096, PROT_READ)     = 0
mprotect(0x7f8bd1c54000, 4096, PROT_READ) = 0
munmap(0x7f8bd1c35000, 125322)          = 0
```

```
exit_group(0)                     = ?
+++ exited with 0 +++
```

[HELLO] -

```
execve("./hello", ["./hello"], [/* 65 vars */]) = 0
brk(NULL)                           = 0x20fc000
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)
open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=125322, ...}) = 0
mmap(NULL, 125322, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f56245ae000
close(3)                            = 0
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
open("/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\t\2\0\0\0\0\0"..., 832) = 832
fstat(3, {st_mode=S_IFREG|0755, st_size=1868984, ...}) = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7f56245ad000
mmap(NULL, 3971488, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_DENYWRITE, 3, 0)
= 0x7f5623fde000
mprotect(0x7f562419e000, 2097152, PROT_NONE) = 0
mmap(0x7f562439e000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x1c0000) = 0x7f562439e000
mmap(0x7f56243a4000, 14752, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_ANONYMOUS, -1, 0) = 0x7f56243a4000
close(3)                            = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7f56245ac000
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7f56245ab000
arch_prctl(ARCH_SET_FS, 0x7f56245ac700) = 0
mprotect(0x7f562439e000, 16384, PROT_READ) = 0
mprotect(0x600000, 4096, PROT_READ)     = 0
mprotect(0x7f56245cd000, 4096, PROT_READ) = 0
munmap(0x7f56245ae000, 125322)          = 0
```

==

```
-- [EXTRA Relative to EMPTY] --         getpid()                        = 16624
-- [EXTRA Relative to EMPTY] --         fstat(1, {st_mode=S_IFCHR|0620,
st_rdev=makedev(136, 2), ...}) = 0
-- [EXTRA Relative to EMPTY] --         brk(NULL)                        = 0x20fc000
-- [EXTRA Relative to EMPTY] --         brk(0x211d000)                   = 0x211d000
-- [EXTRA Relative to EMPTY] --         write(1, "\n", 1)                = 1
-- [EXTRA Relative to EMPTY] --         write(1, "Process ID : 16624 \n", 20)   = 20
-- [EXTRA Relative to EMPTY] --         write(1, "\n", 1)                = 1
-- [EXTRA Relative to EMPTY] --         fstat(0, {st_mode=S_IFCHR|0620,
st_rdev=makedev(136, 2), ...}) = 0
-- [EXTRA Relative to EMPTY] --         write(1, "Enter your name : ", 18)      = 18
-- [EXTRA Relative to EMPTY] --           read(0, "Vamsi\n", 1024)              = 6
-- [EXTRA Relative to EMPTY] --           write(1, "\n", 1)                     = 1
-- [EXTRA Relative to EMPTY] --           write(1, "Welcome Vamsi\n", 14)        = 14
-- [EXTRA Relative to EMPTY] --           lseek(0, -1, SEEK_CUR)                 = -1 ESPIPE
(Illegal seek)
```

==
==========================

```
exit_group(0)                     = ?
+++ exited with 0 +++
```

(b) the unique system calls (for 2nd file) were, getpid() to get process id, write() to write to user's console, read() to get input. An exaustive list of all is too long to describe, but they deal with importing the lib files and initializing the memory to run the program. (common to both 1st and 2nd program)

==
==

5.

[Command used is :--- lsof -p $(ps -u labuser | grep openfiles | awk '{print $1}') ---:]

```
COMMAND     PID    USER   FD   TYPE DEVICE SIZE/OFF   NODE NAME
openfiles 16759 labuser  cwd    DIR    8,1     4096 397869 /home/labuser/Downloads/lab1/files
openfiles 16759 labuser  rtd    DIR    8,1     4096      2 /
openfiles 16759 labuser  txt    REG    8,1     8760 397870
/home/labuser/Downloads/lab1/files/openfiles
openfiles 16759 labuser  mem    REG    8,1  1868984 262318 /lib/x86_64-linux-gnu/libc-2.23.so
openfiles 16759 labuser  mem    REG    8,1   162632 262314 /lib/x86_64-linux-gnu/ld-2.23.so
openfiles 16759 labuser    0u   CHR  136,2      0t0      5 /dev/pts/2
openfiles 16759 labuser    1u   CHR  136,2      0t0      5 /dev/pts/2
openfiles 16759 labuser    2u   CHR  136,2      0t0      5 /dev/pts/2
openfiles 16759 labuser    3w   REG    8,1        0 668285 /tmp/welocme to OS
openfiles 16759 labuser    4w   REG    8,1        0 668286 /tmp/CS333
openfiles 16759 labuser    5w   REG    8,1        0 668287 /tmp/CS347
```

The files opened by the process are in the NAME column above.

==
==

6.

(command used is --- lsblk -o +FSTYPE)

```
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT FSTYPE
sda      8:0    0 59.6G  0 disk
```

```
|-sda1   8:1    0 58.7G  0 part /          ext4
|-sda2   8:2    0    1K  0 part
`-sda5   8:5    0  975M  0 part [SWAP]     swap
```

The filesystems and mountpoints are as above.

===
==

7.

In boot_sector1.asm, the magic number (last two bytes) did not match to 0xaa55, hence it found the hard-disk is not bootable and said 'not a bootable disk'.

But in boot_sector2.asm, the magic number matched since the assembly code explicitly wrote it. So, the boot from hard-disk was successful and later went on into a loop as written in the assembly code.