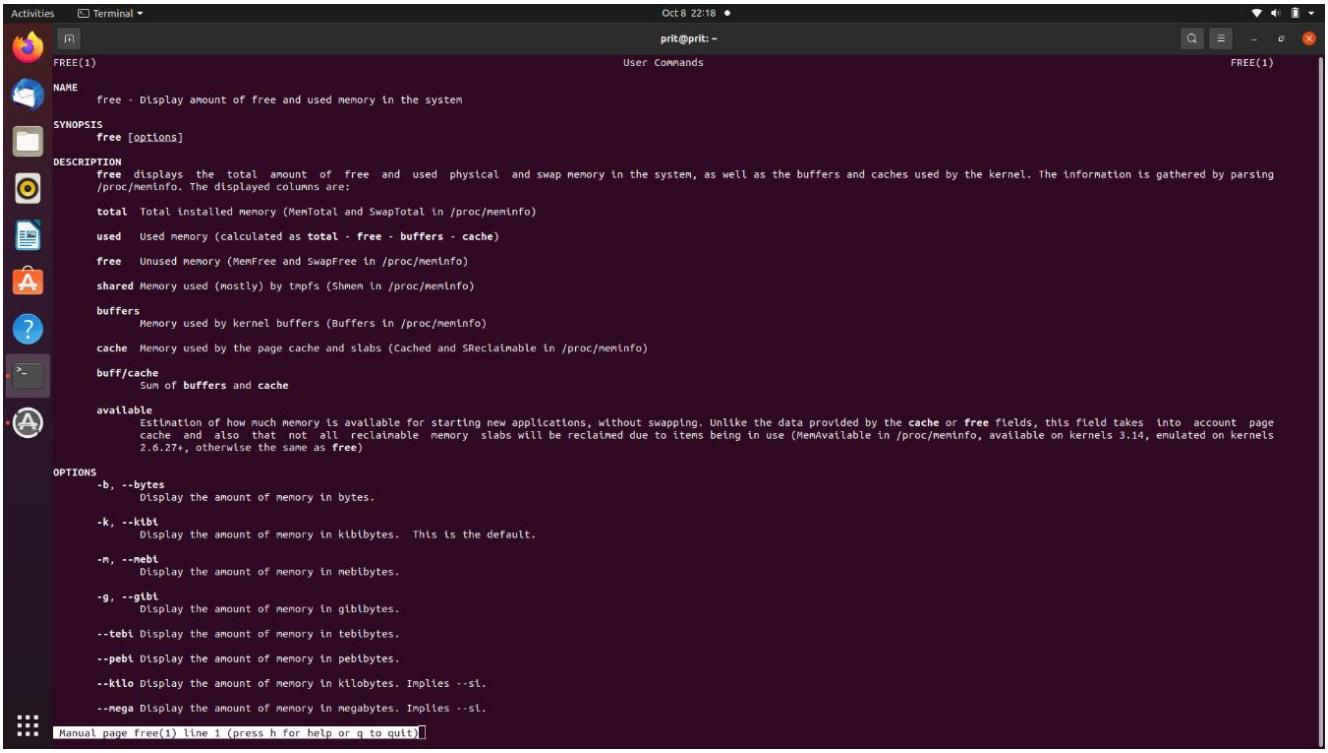
OS HW 2: Prithul Sarker

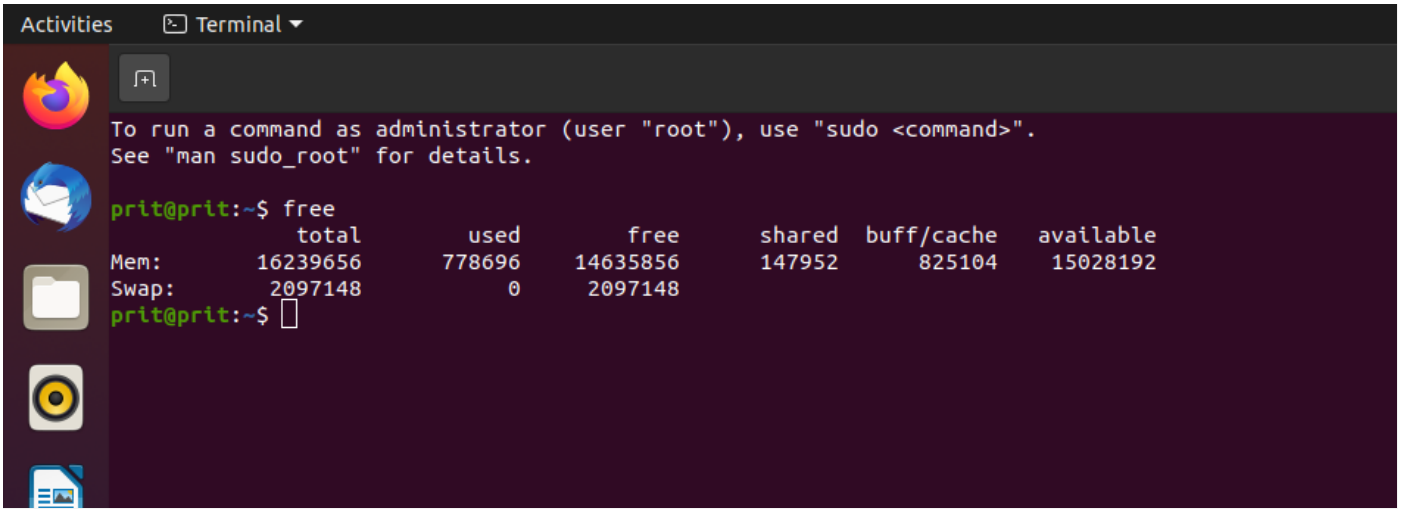
Problem 1

Man page of free command:

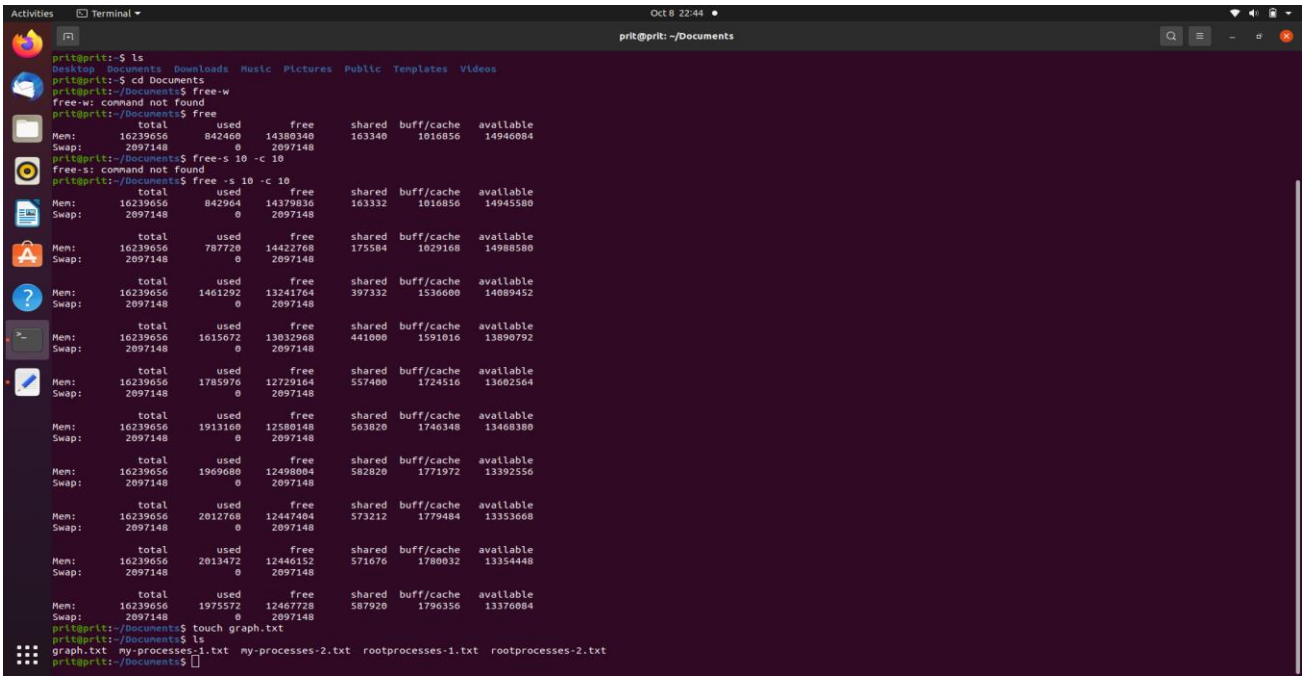


In my computer free-o command does not work so I have done the same task using only free command.

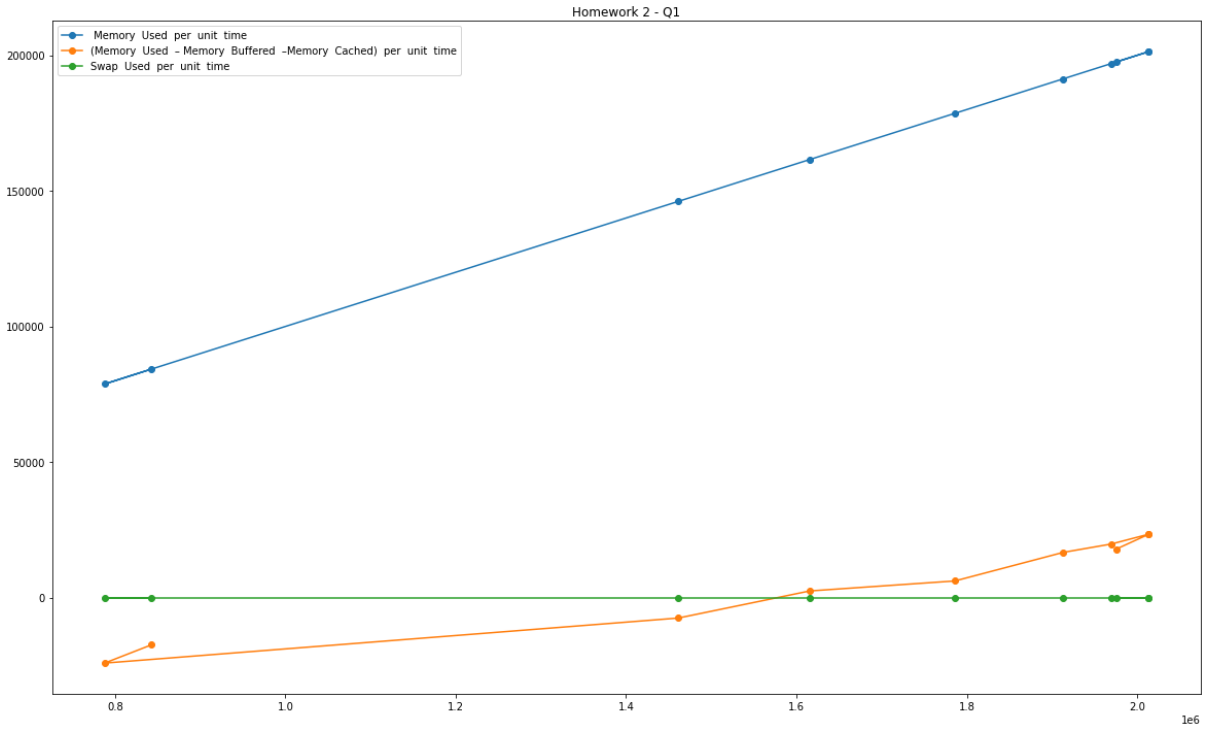
Execution of free command:



Executing free command 10 times at a constant time interval (10 seconds) running other programs in between:




```
total        used        free      shared  buff/cache   available
Mem:       16239656      842964    14379836      163332     1016856    14945580
Swap:       2097148           0     2097148

               total        used        free      shared  buff/cache   available
Mem:       16239656      787720    14422768      175584     1029168    14988580
Swap:       2097148           0     2097148

               total        used        free      shared  buff/cache   available
Mem:       16239656     1461292    13241764      397332     1536600    14089452
Swap:       2097148           0     2097148

               total        used        free      shared  buff/cache   available
Mem:       16239656     1615672    13032968      441000     1591016    13890792
Swap:       2097148           0     2097148

               total        used        free      shared  buff/cache   available
Mem:       16239656     1785976    12729164      557400     1724516    13602564
Swap:       2097148           0     2097148

               total        used        free      shared  buff/cache   available
Mem:       16239656     1913160    12580148      563820     1746348    13468380
Swap:       2097148           0     2097148

               total        used        free      shared  buff/cache   available
Mem:       16239656     1969680    12498004      582820     1771972    13392556
Swap:       2097148           0     2097148

               total        used        free      shared  buff/cache   available
Mem:       16239656     2012768    12447404      573212     1779484    13353668
Swap:       2097148           0     2097148

               total        used        free      shared  buff/cache   available
Mem:       16239656     2013472    12446152      571676     1780032    13354448
Swap:       2097148           0     2097148

               total        used        free      shared  buff/cache   available
Mem:       16239656     1975572    12467728      587920     1796356    13376084
Swap:       2097148           0     2097148
```



I used free command on an interval of 10 sec. So for the per unit time measurement, the memory used, buffered/cached and swap used are divided by 10 sec. In the above diagram, we can see that the memory used per unit time increases with the memory being used. However, swap memory could only be used when the main memory is full. So, the swap memory is zero for the figure. Again, at first, buffer is being used more than the memory used. For this reason, the (memory used -memory buffered – memory cached) per unit time values are in negative region. However, after few 10 second intervals, the buffer memory does not increase in the similar way compared to the memory used. This is why, (memory used -memory buffered – memory cached) per unit time is in the positive region.

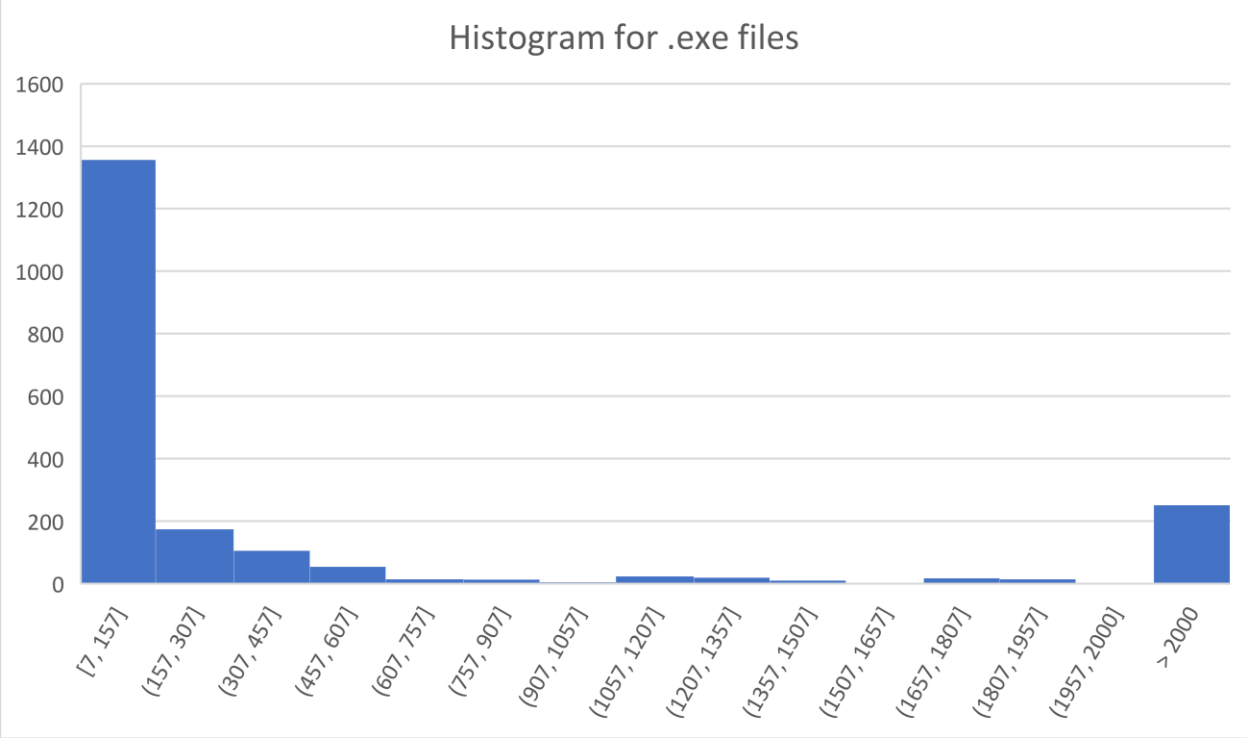
Problem 2

For .exe files, the mean and median of the files are 4617.24 and 73 Kb respectively.

And optimal page size = $(2* s * e)^{1/2}$

Here, e = 8 bytes, and s = pages per process * page number = $2^8 * 4617.24 * 1024$

Therefore, optimal page size = $(2 * 2^8 * 4617.24 * 1024 * 8)^{1/2}$ = 139162 bytes

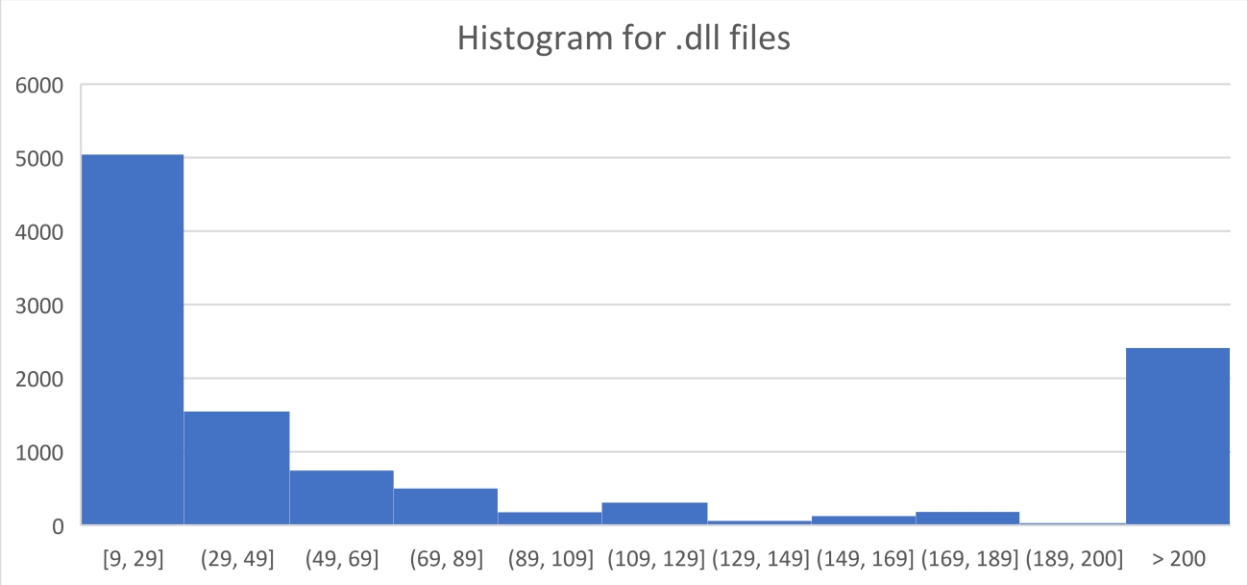


For .dll files, the mean and median of the files are 31121.72 and 35.375 Kb respectively.

And optimal page size = $(2* s * e)^{1/2}$

Here, e = 8 bytes, and s = pages per process * page number = $2^8 * 31121.72 * 1024$

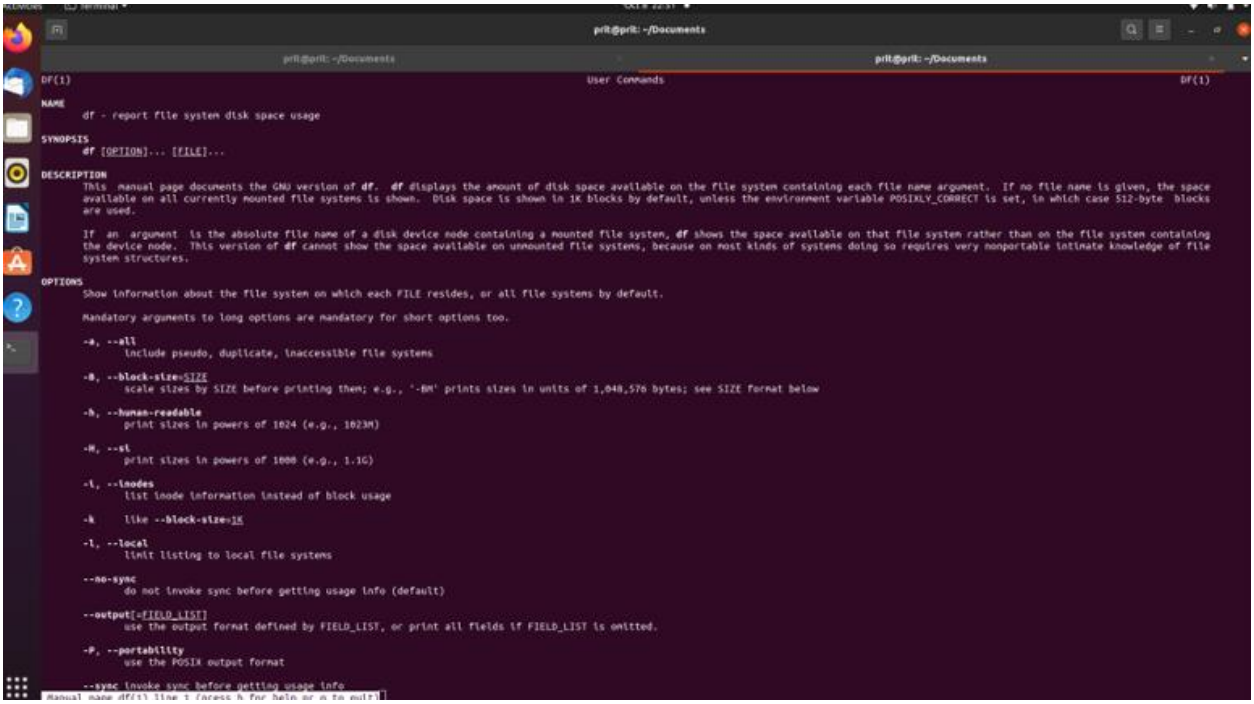
Therefore, optimal page size = $(2 * 2^8 * 31121.72 * 1024 * 8)^{1/2}$ = 361294.83 bytes



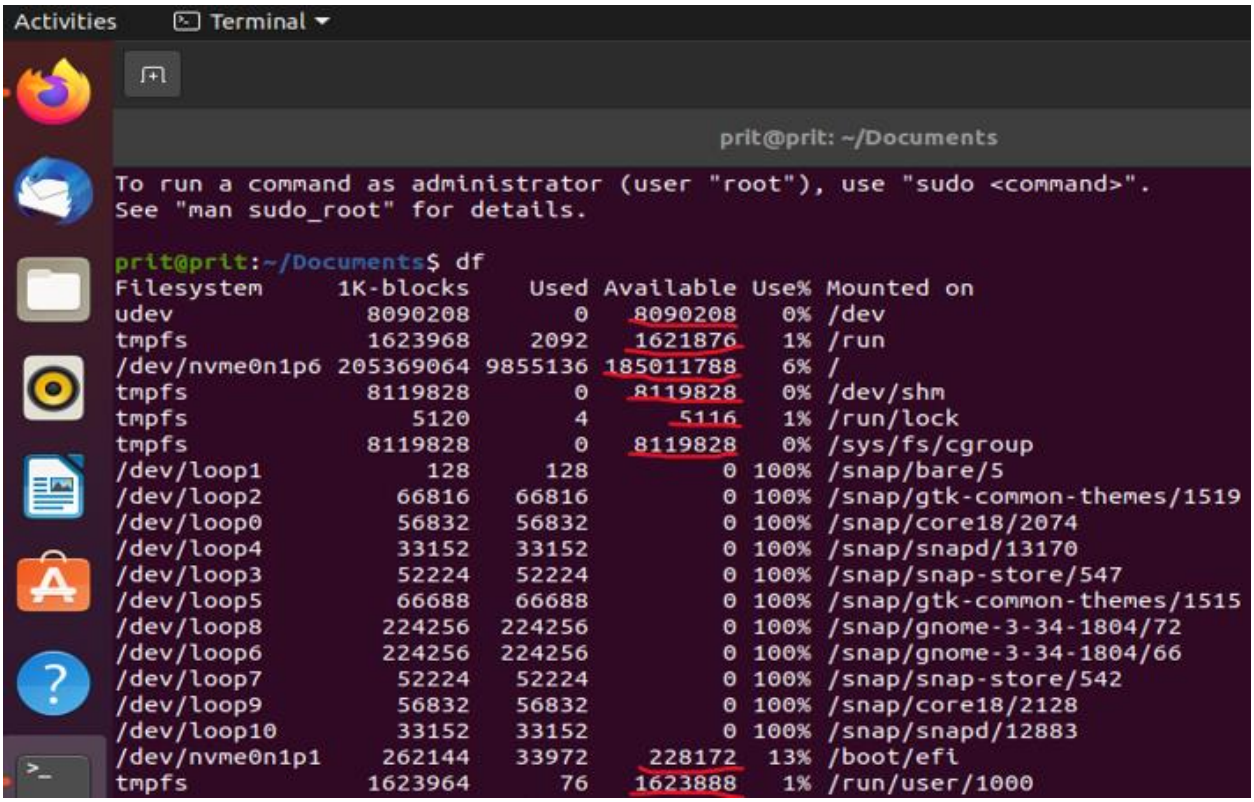
The text files containing all the .exe and .dll file names and sizes are uploaded separately inside the zip file.

Problem 3

Man page of df command:

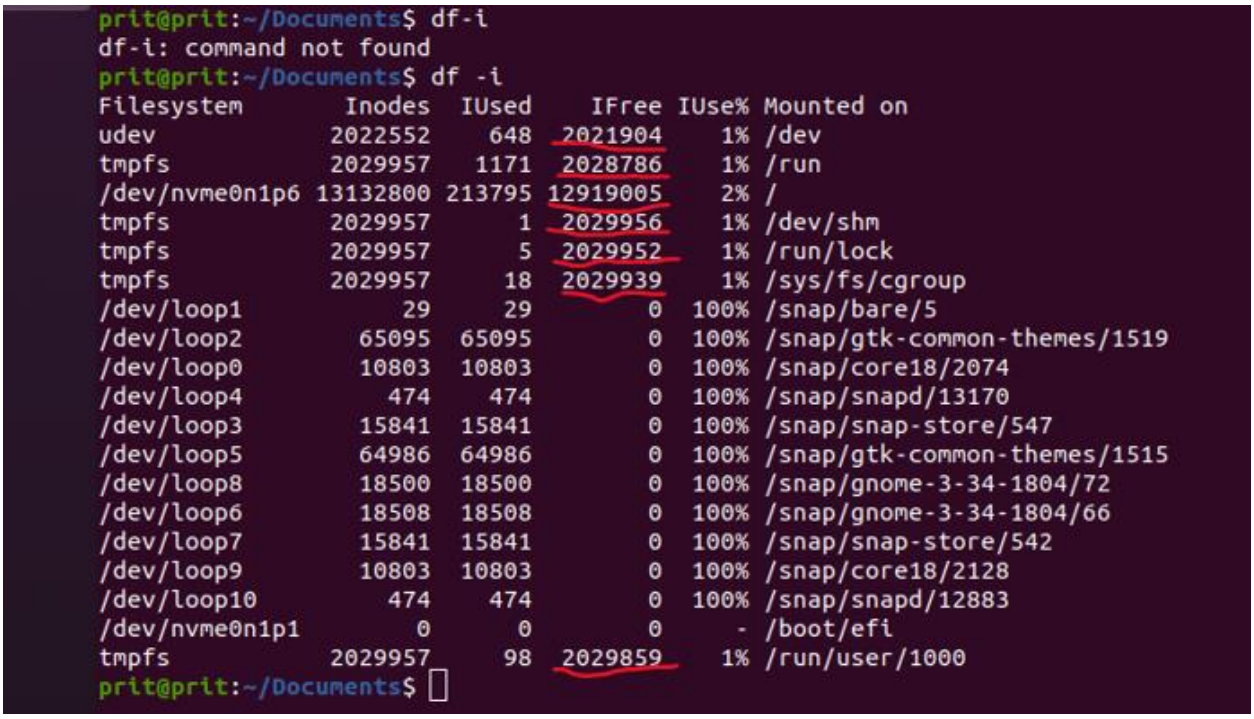


Execution of df Command:



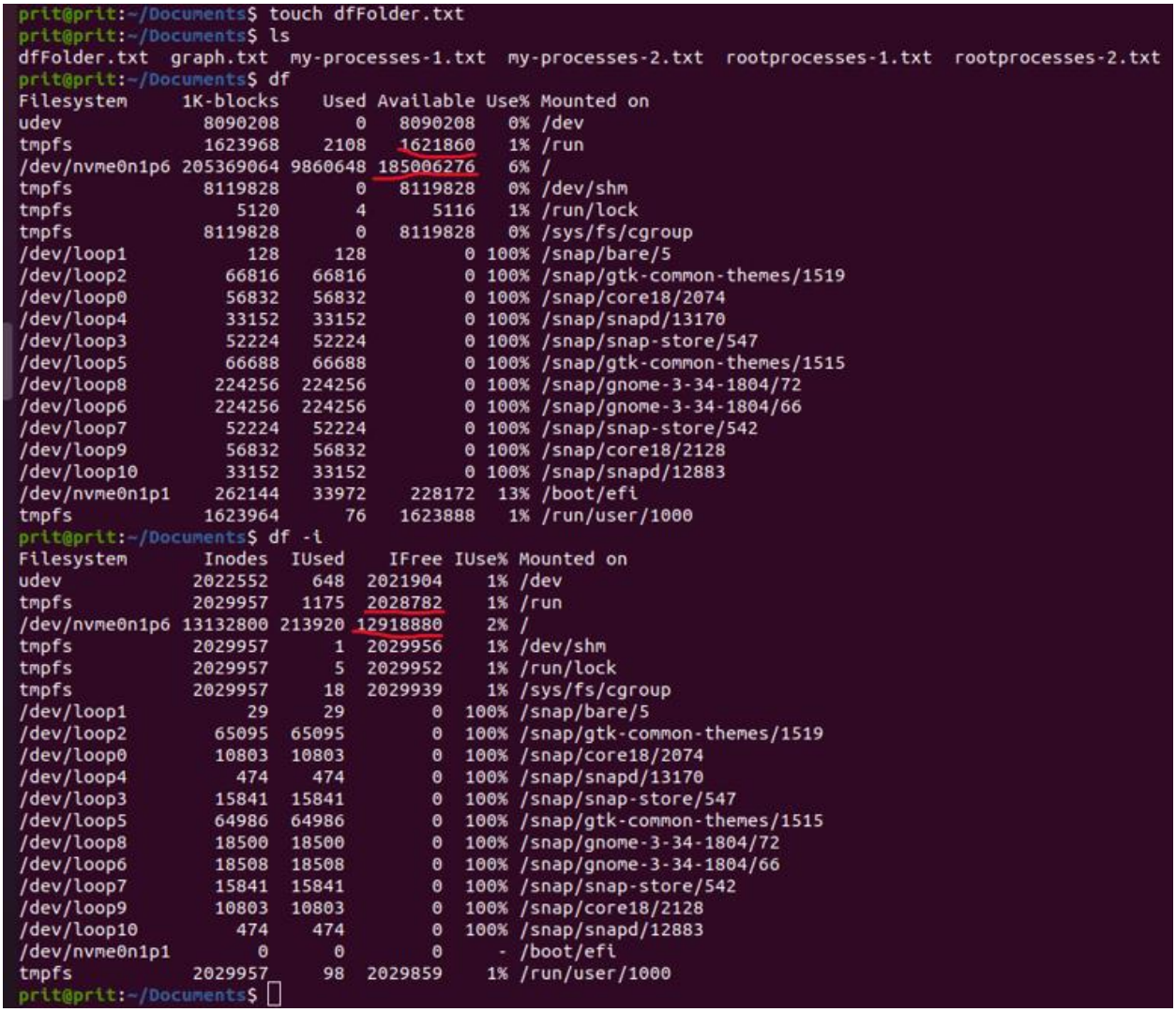
The underlined numbers under “Available” column are the available disk sizes of the file system. The other non-underlined disk sizes are already 100 % used.

Execution of df-i command:



In the above diagram the total number of I nodes are shown under the column of Inodes. The underlined numbers under the ‘Ifree’ colume are the Inodes which are available, the used I nodes are shown under the Iused column.

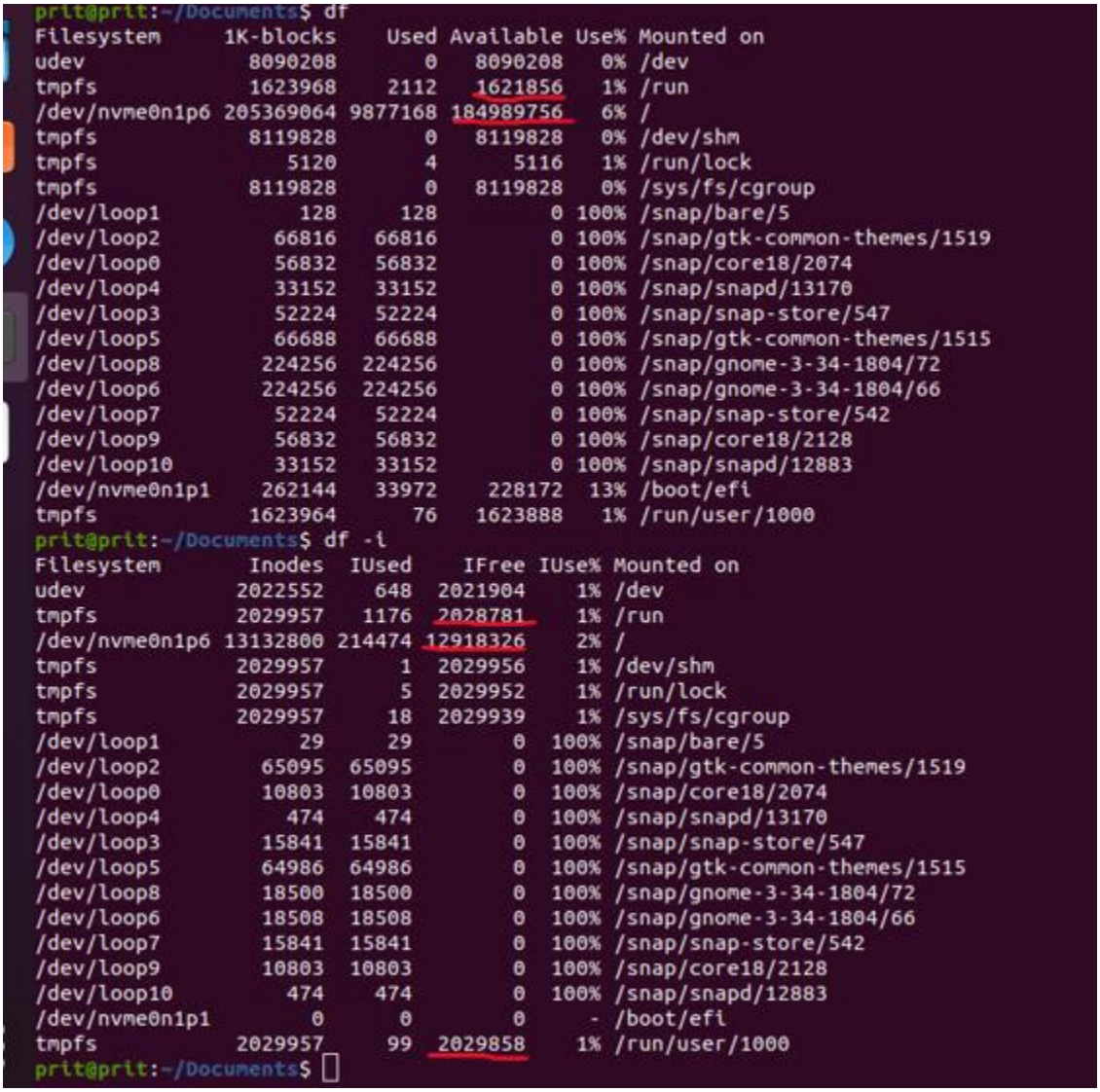
df and df-I command execution after creating a new file with just a few characters in it:



```
prit@prit:~/Documents$ touch dfFolder.txt
prit@prit:~/Documents$ ls
dfFolder.txt  graph.txt  my-processes-1.txt  my-processes-2.txt  rootprocesses-1.txt  rootprocesses-2.txt
prit@prit:~/Documents$ df
Filesystem     1K-blocks    Used Available Use% Mounted on
udev             8090208       0   8090208   0% /dev
tmpfs            1623968    2108   1621860   1% /run
/dev/nvme0n1p6 205369064 9860648 185006276   6% /
tmpfs            8119828       0   8119828   0% /dev/shm
tmpfs               5120       4      5116   1% /run/lock
tmpfs            8119828       0   8119828   0% /sys/fs/cgroup
/dev/loop1           128     128         0 100% /snap/bare/5
/dev/loop2         66816   66816         0 100% /snap/gtk-common-themes/1519
/dev/loop0         56832   56832         0 100% /snap/core18/2074
/dev/loop4         33152   33152         0 100% /snap/snapd/13170
/dev/loop3         52224   52224         0 100% /snap/snap-store/547
/dev/loop5         66688   66688         0 100% /snap/gtk-common-themes/1515
/dev/loop8        224256  224256         0 100% /snap/gnome-3-34-1804/72
/dev/loop6        224256  224256         0 100% /snap/gnome-3-34-1804/66
/dev/loop7         52224   52224         0 100% /snap/snap-store/542
/dev/loop9         56832   56832         0 100% /snap/core18/2128
/dev/loop10        33152   33152         0 100% /snap/snapd/12883
/dev/nvme0n1p1    262144   33972    228172  13% /boot/efi
tmpfs            1623964      76   1623888   1% /run/user/1000
prit@prit:~/Documents$ df -i
Filesystem       Inodes  IUsed    IFree IUse% Mounted on
udev            2022552    648  2021904    1% /dev
tmpfs           2029957   1175  2028782    1% /run
/dev/nvme0n1p6 13132800 213920 12918880    2% /
tmpfs           2029957      1  2029956    1% /dev/shm
tmpfs           2029957      5  2029952    1% /run/lock
tmpfs           2029957     18  2029939    1% /sys/fs/cgroup
/dev/loop1           29     29        0  100% /snap/bare/5
/dev/loop2        65095  65095        0  100% /snap/gtk-common-themes/1519
/dev/loop0        10803  10803        0  100% /snap/core18/2074
/dev/loop4          474    474        0  100% /snap/snapd/13170
/dev/loop3        15841  15841        0  100% /snap/snap-store/547
/dev/loop5        64986  64986        0  100% /snap/gtk-common-themes/1515
/dev/loop8        18500  18500        0  100% /snap/gnome-3-34-1804/72
/dev/loop6        18508  18508        0  100% /snap/gnome-3-34-1804/66
/dev/loop7        15841  15841        0  100% /snap/snap-store/542
/dev/loop9        10803  10803        0  100% /snap/core18/2128
/dev/loop10         474    474        0  100% /snap/snapd/12883
/dev/nvme0n1p1        0      0        0     - /boot/efi
tmpfs           2029957     98  2029859    1% /run/user/1000
prit@prit:~/Documents$
```

Changes in available disk space are underlined in df command after few characters.

df and df-I command execution after an increasing character size over 5000:

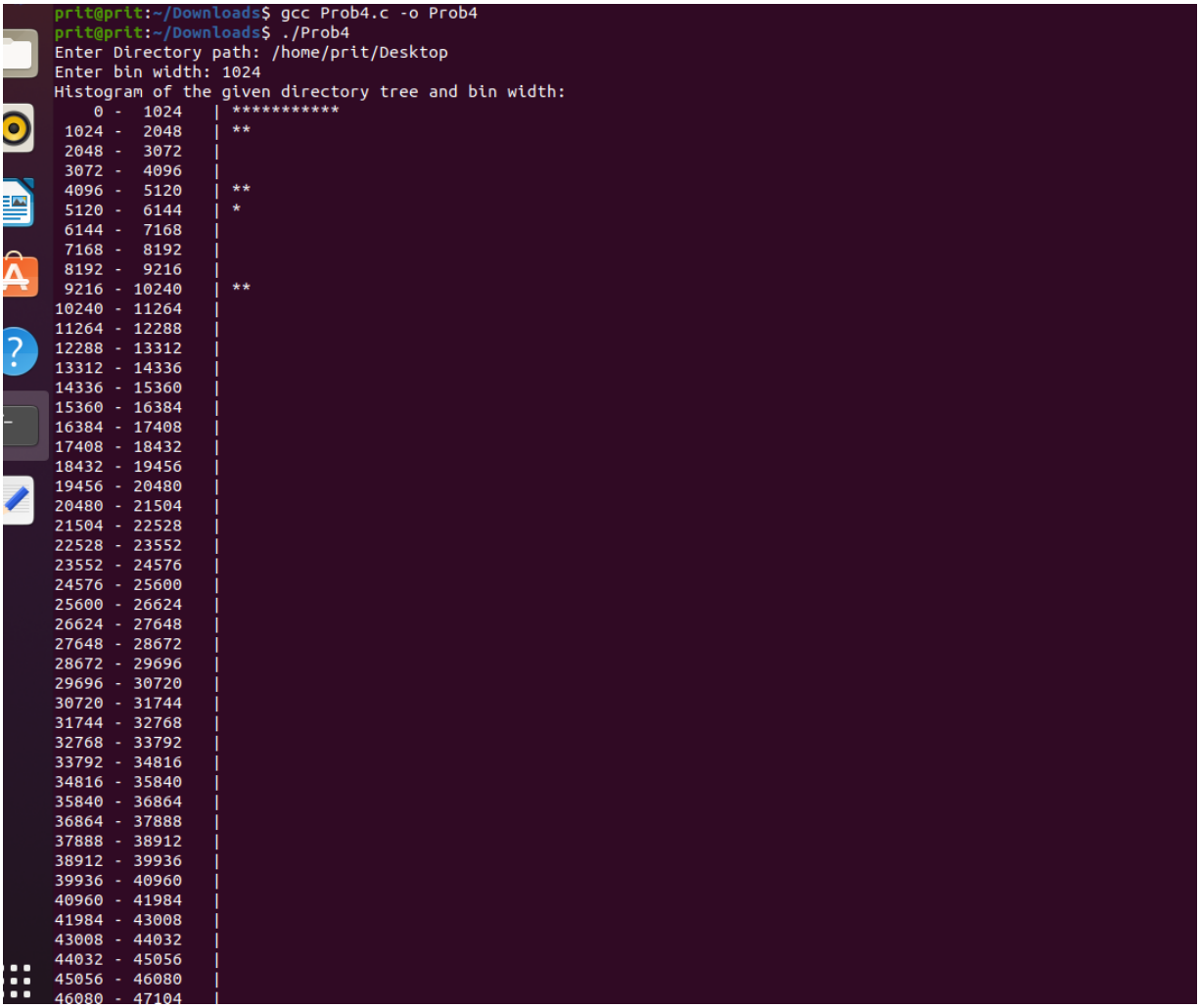


```
prit@prit:~/Documents$ df
Filesystem     1K-blocks    Used Available Use% Mounted on
udev             8090208       0   8090208   0% /dev
tmpfs            1623968    2112   1621856   1% /run
/dev/nvme0n1p6 205369064 9877168 184989756   6% /
tmpfs            8119828       0   8119828   0% /dev/shm
tmpfs               5120       4      5116   1% /run/lock
tmpfs            8119828       0   8119828   0% /sys/fs/cgroup
/dev/loop1           128     128         0 100% /snap/bare/5
/dev/loop2         66816   66816         0 100% /snap/gtk-common-themes/1519
/dev/loop0         56832   56832         0 100% /snap/core18/2074
/dev/loop4         33152   33152         0 100% /snap/snapd/13170
/dev/loop3         52224   52224         0 100% /snap/snap-store/547
/dev/loop5         66688   66688         0 100% /snap/gtk-common-themes/1515
/dev/loop8        224256  224256         0 100% /snap/gnome-3-34-1804/72
/dev/loop6        224256  224256         0 100% /snap/gnome-3-34-1804/66
/dev/loop7         52224   52224         0 100% /snap/snap-store/542
/dev/loop9         56832   56832         0 100% /snap/core18/2128
/dev/loop10        33152   33152         0 100% /snap/snapd/12883
/dev/nvme0n1p1    262144   33972    228172  13% /boot/efi
tmpfs            1623964      76   1623888   1% /run/user/1000
prit@prit:~/Documents$ df -i
Filesystem       Inodes  IUsed    IFree IUse% Mounted on
udev            2022552    648  2021904    1% /dev
tmpfs           2029957   1176  2028781    1% /run
/dev/nvme0n1p6 13132800 214474 12918326    2% /
tmpfs           2029957      1  2029956    1% /dev/shm
tmpfs           2029957      5  2029952    1% /run/lock
tmpfs           2029957     18  2029939    1% /sys/fs/cgroup
/dev/loop1           29     29        0  100% /snap/bare/5
/dev/loop2        65095  65095        0  100% /snap/gtk-common-themes/1519
/dev/loop0        10803  10803        0  100% /snap/core18/2074
/dev/loop4          474    474        0  100% /snap/snapd/13170
/dev/loop3        15841  15841        0  100% /snap/snap-store/547
/dev/loop5        64986  64986        0  100% /snap/gtk-common-themes/1515
/dev/loop8        18500  18500        0  100% /snap/gnome-3-34-1804/72
/dev/loop6        18508  18508        0  100% /snap/gnome-3-34-1804/66
/dev/loop7        15841  15841        0  100% /snap/snap-store/542
/dev/loop9        10803  10803        0  100% /snap/core18/2128
/dev/loop10         474    474        0  100% /snap/snapd/12883
/dev/nvme0n1p1        0      0        0     - /boot/efi
tmpfs           2029957     99  2029858    1% /run/user/1000
prit@prit:~/Documents$
```

Changes in available disk space is underlined in df and df-I command after increasing the number of characters to over 5000.

Problem 4

Output of the code



The c code file is uploaded in the zip file.