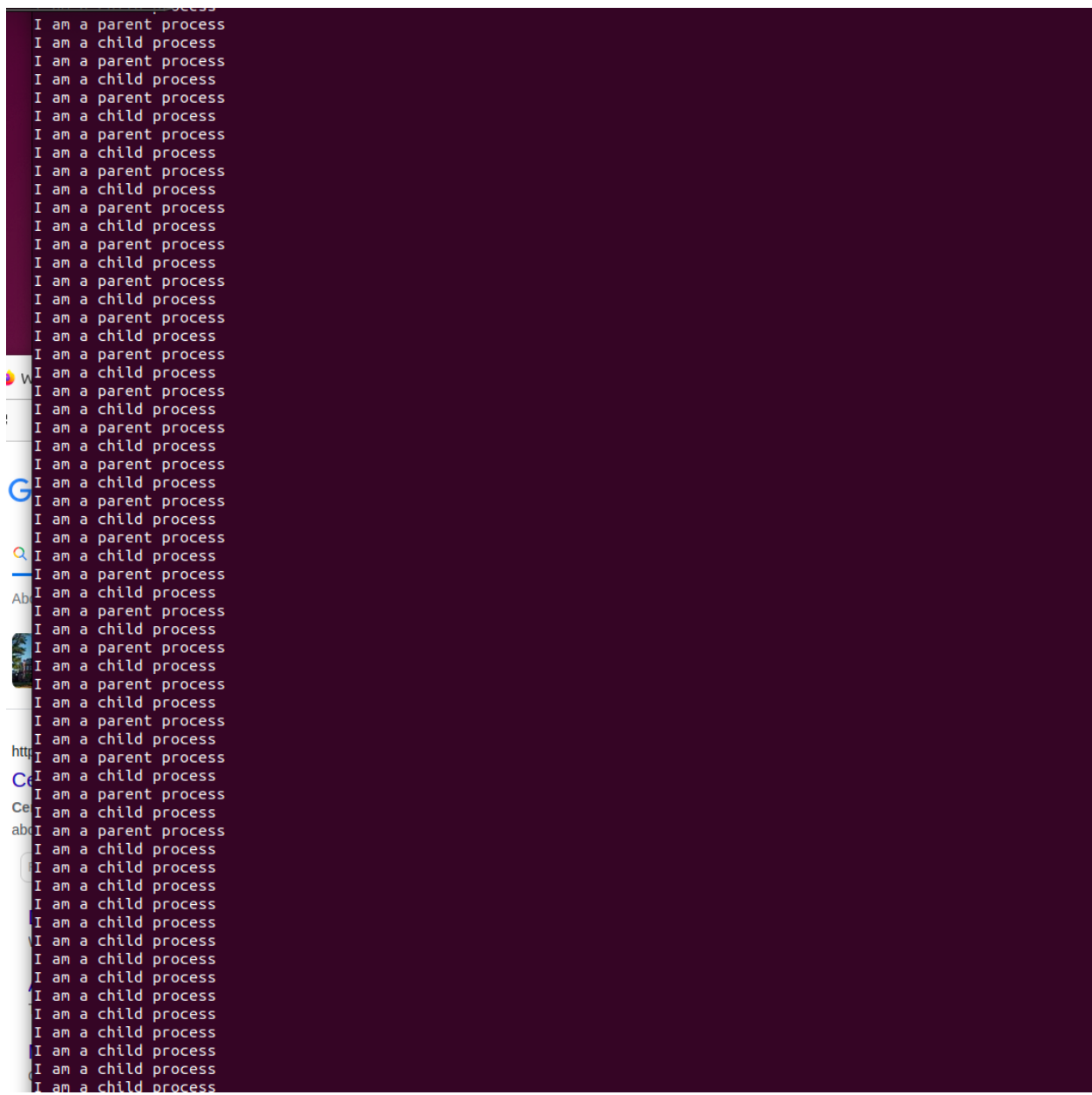1. 4 child processes
2. 3.0 % of CPU and 5.0% of MEM
3. 7959108 KiB
4. Gnome Shell
5. Gnome Shell
6. Apt-get is a Debian-based system package manager (used to install, remove, and update packages). Yum is also a package manager for Red Hat-based systems (used to install, remove and update packages). Wget is command-line utility for downloading files from internet. Gzip is a file compression utility for unix-like systems. Tar is a utility for archiving files and directories into a sinfle archive file. Rar is a proprietary file archiving utility for windows and unix like systems.
7.

```c
#include <stdio.h>
#include <unistd.h>

int main(){
        pid_t pid = fork();
        if(pid == 0){
        for(int i = 0; i <200; i++){
                printf("I am a child process\n");
        }
        }else{
                for (int i = 0; i<200; i++){
                printf("I am a parent process\n");
                }
        }
        return 0;
}
```

```
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a parent process
I am a child process
I am a child process
I am a child process
I am a child process
I am a child process
I am a child process
I am a child process
I am a child process
I am a child process
I am a child process
I am a child process
I am a child process
I am a child process
```

8.

```c
1 #include <sys/types.h>
2 #include <stdio.h>
3 #include <unistd.h>
4 #include <sys/wait.h>
5 #include <stdlib.h>
6
7 int main(){
8        pid_t pid;
9        char *buf;
10       pid=fork();
11       if (pid == 0){
12               printf("I am the child process\n");
13       }else{
14               wait(NULL);
15               printf("child process complete\n");
16               printf("I am the parent process\n");
17               buf = (char *)malloc(100*sizeof(char));
18               getcwd(buf, 100);
19               printf("\n %s \n", buf);
20       }
21       return 0;
22 }
```

```
samaad@Ubuntu:~$ ./practice2
I am the child process
child process complete
I am the parent process
```

9.

```
I am a child process
I am a child process
```

```c
1 #include <stdio.h>
2 #include <unistd.h>
3
4 int main(){
5        pid_t pid;
6        pid = fork();
7
8        if (pid == 0){
9                printf("I am the child process, my PID is %d, my parent's PID
  is %d\n", getpid(), getppid());
10               }else{
11                printf("I am the parent process, my PID is %d, my child's PID
  is %d\n", getpid(), pid);
12
13       }
14       return 0;
15 }
```

```
samaad@Ubuntu:~$ gcc g practicees.c  o practices
samaad@Ubuntu:~$ ./practice3
I am the parent process, my PID is 5135, my child's PID is 5136
I am the child process, my PID is 5136, my parent's PID is 5135
samaad@Ubuntu:~$ gedit practice3.c
```