

National Textile University

Department of Computer Science

Name

Gulam Rasool

Reg No:

23-NTU-CS-FL-1159

Semester:

5th

Lab:

03

Submitted to:

Sir Nasir Mehmood

LAB-03

1. Introduction

A process is simply a program in execution. When you type a command in Linux (like ls), the OS creates a process for it.

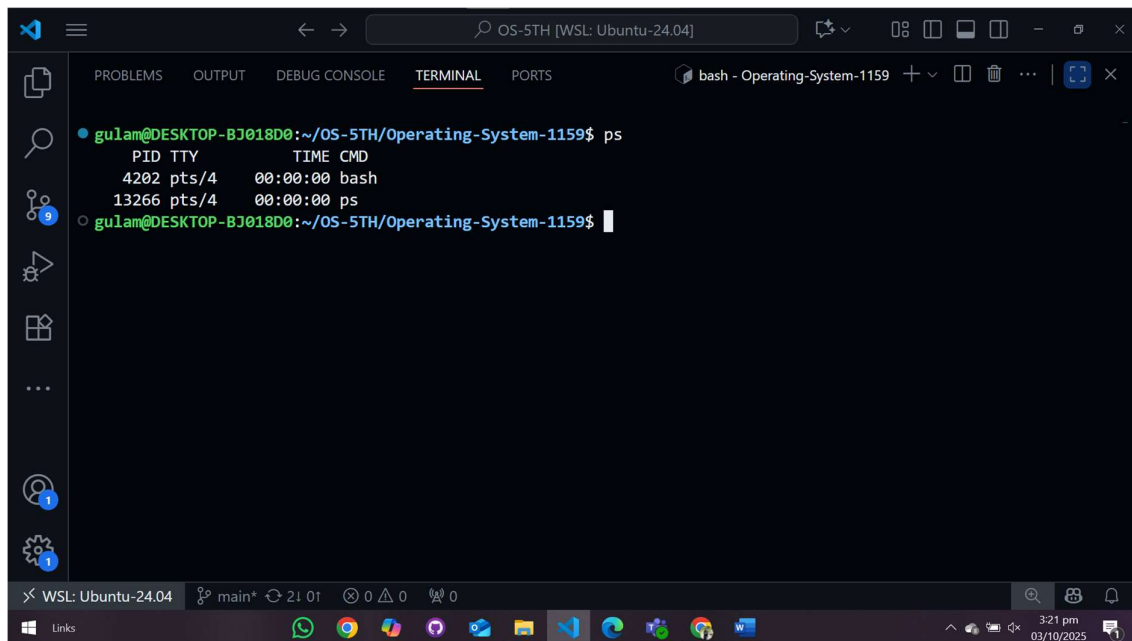
Every process has:

- **PID (Process ID)** → unique number for each process.
- **PPID (Parent Process ID)** → ID of the process that created it.
- **State** → running, sleeping, stopped, zombie, etc

2. Linux Process Commands

2.1 Viewing Processes

ps → *Process Status*

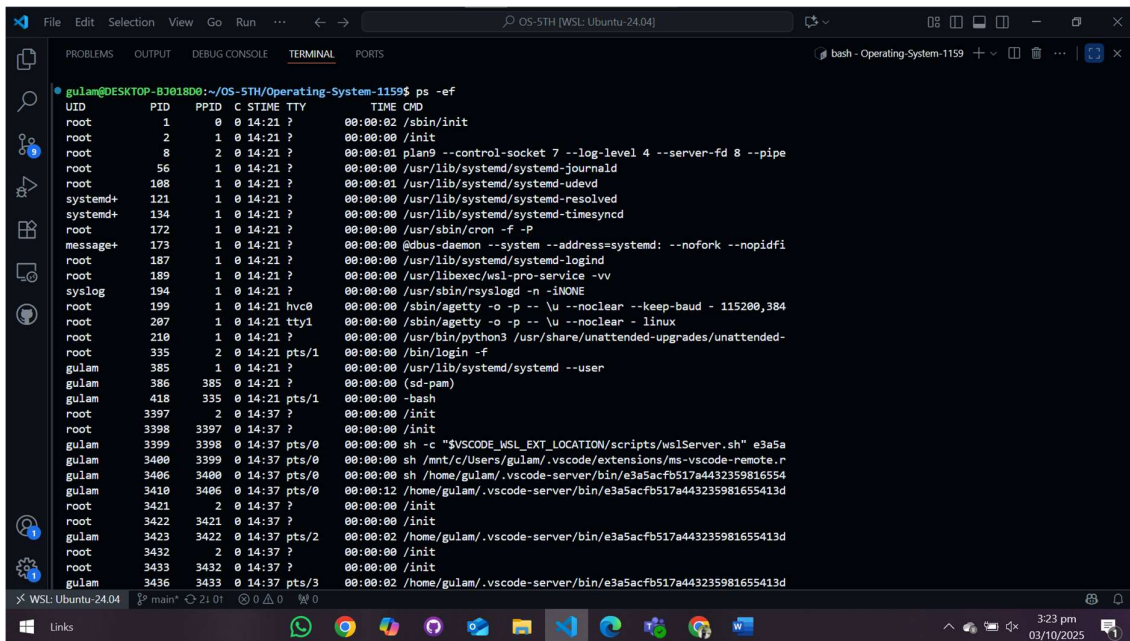


The screenshot shows a Windows terminal window with the title bar "WSL: Ubuntu-24.04". The terminal is running a bash shell. The user has entered the command `ps`, and the output is displayed as follows:

```
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ ps
  PID TTY          TIME CMD
  4202 pts/4    00:00:00 bash
 13266 pts/4    00:00:00 ps
```

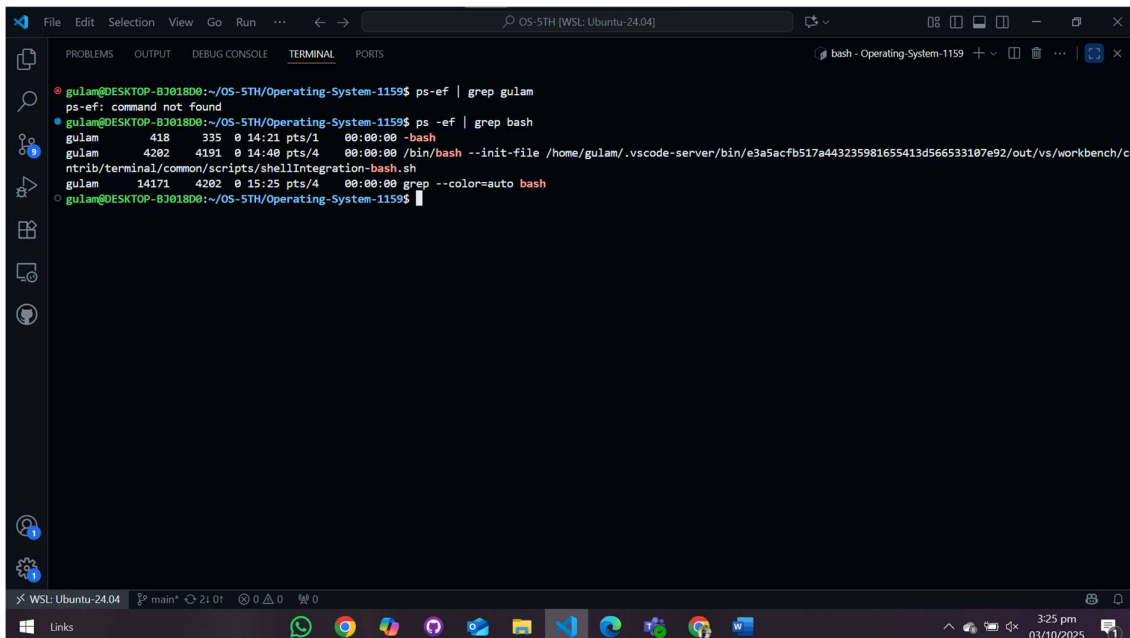
The terminal window also shows a sidebar with icons for file explorer, search, and other tools. The bottom status bar indicates the current time as 3:21 pm on 03/10/2025.

ps -ef → Full list of all processes



```
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ ps -ef
UID          PID    PPID  C STIME TTY          TIME CMD
root           1        0  0 14:21 ?        00:00:02 /sbin/init
root           2        1  0 14:21 ?        00:00:00 /init
root           8        2  0 14:21 ?        00:00:01 plan9 --control-socket 7 --log-level 4 --server-fd 8 --pipe
root          56        1  0 14:21 ?        00:00:00 /usr/lib/systemd/systemd-journald
root         108        1  0 14:21 ?        00:00:01 /usr/lib/systemd/systemd-udev
systemd+     121        1  0 14:21 ?        00:00:00 /usr/lib/systemd/systemd-resolved
systemd+     134        1  0 14:21 ?        00:00:00 /usr/lib/systemd/systemd-timesyncd
root         172        1  0 14:21 ?        00:00:00 /usr/sbin/cron -f -P
message+     173        1  0 14:21 ?        00:00:00 @dbus-daemon --system --address=systemd: --nofork --nopidfi
root         187        1  0 14:21 ?        00:00:00 /usr/lib/systemd/systemd-logind
root         189        1  0 14:21 ?        00:00:00 /usr/libexec/ssh-pro-service -vv
syslog       194        1  0 14:21 ?        00:00:00 /usr/sbin/rsyslogd -n -iNONE
root         199        1  0 14:21 hvc0     00:00:00 /sbin/agetty -o -p -- \u --noclear - 115200,384
root         207        1  0 14:21 tty1     00:00:00 /sbin/agetty -o -p -- \u --noclear - linux
root         210        1  0 14:21 ?        00:00:00 /usr/bin/python3 /usr/share/unattended-upgrades/unattended-
root         335        2  0 14:21 pts/1    00:00:00 /bin/login -f
gulam        385        1  0 14:21 ?        00:00:00 /usr/lib/systemd/systemd --user
gulam        386       385  0 14:21 ?        00:00:00 (sd-pam)
gulam        418       335  0 14:21 pts/1    00:00:00 -bash
root        3397        2  0 14:37 ?        00:00:00 /init
root        3398       3397  0 14:37 ?        00:00:00 sh
gulam        3399       3398  0 14:37 pts/0     00:00:00 sh -c "$VSCODE_WSL_EXT_LOCATION/scripts/wslServer.sh" e3a5a
gulam        3400       3399  0 14:37 pts/0     00:00:00 sh /mnt/c/Users/gulam/.vscode/extensions/ms-vscode-remote.r
gulam        3406       3400  0 14:37 pts/0     00:00:00 sh /home/gulam/.vscode-server/bin/e3a5acfb517a4432359816554
gulam        3410       3406  0 14:37 pts/0     00:00:12 /home/gulam/.vscode-server/bin/e3a5acfb517a443235981655413d
root        3421        2  0 14:37 ?        00:00:00 /init
root        3422       3421  0 14:37 ?        00:00:00 /init
gulam        3423       3422  0 14:37 pts/2     00:00:02 /home/gulam/.vscode-server/bin/e3a5acfb517a443235981655413d
root        3432        2  0 14:37 ?        00:00:00 /init
root        3433       3432  0 14:37 ?        00:00:00 /init
gulam        3436       3433  0 14:37 pts/3     00:00:02 /home/gulam/.vscode-server/bin/e3a5acfb517a443235981655413d
```

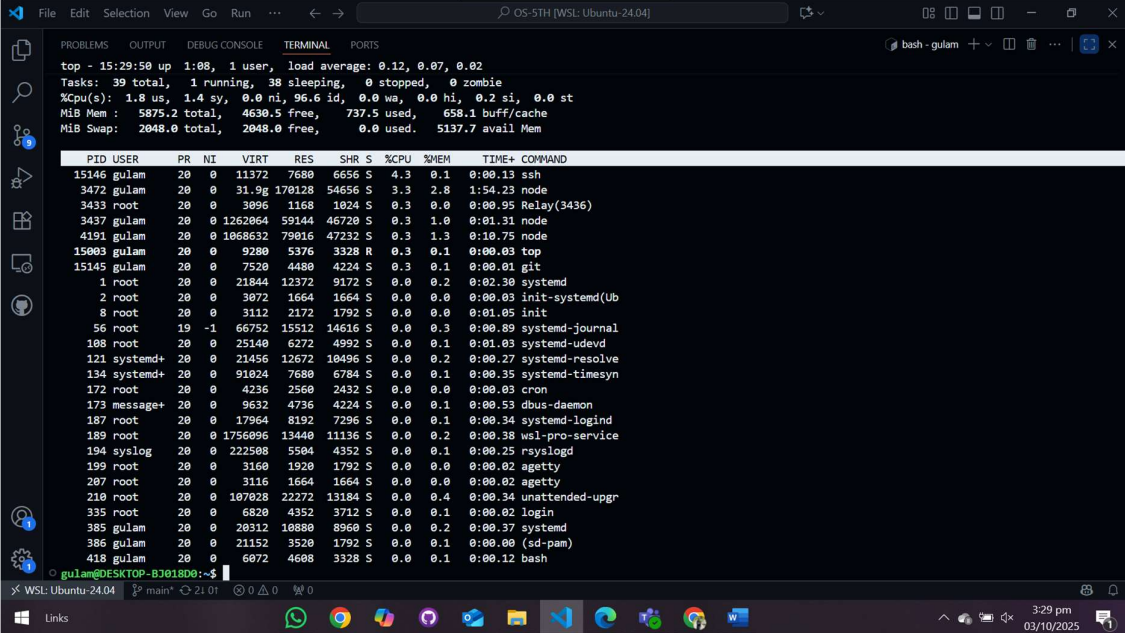
ps -ef | grep bash



```
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ ps -ef | grep bash
ps-ef: command not found
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ ps -ef | grep bash
gulam        418       335  0 14:21 pts/1    00:00:00 -bash
gulam        4202       4191  0 14:40 pts/4     00:00:00 /bin/bash --init-file /home/gulam/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e92/out/vs/workbench/co
ntrib/terminal/common/scripts/shellIntegration-bash.sh
gulam       14171       4202  0 15:25 pts/4     00:00:00 grep --color=auto bash
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$
```

2.2 Monitoring Processes Interactively

top → *Dynamic process viewer*



The screenshot shows a Windows Subsystem for Linux (WSL) terminal window titled "05-5TH [WSL: Ubuntu-24.04]". The terminal displays the output of the `top` command. At the top, it shows system statistics: "top - 15:29:59 up 1:08, 1 user, load average: 0.12, 0.07, 0.02". Below this, it lists tasks: "Tasks: 39 total, 1 running, 38 sleeping, 0 stopped, 0 zombie". It also shows CPU usage: "%Cpu(s): 1.8 us, 1.4 sy, 0.0 ni, 96.6 id, 0.0 wa, 0.0 hi, 0.2 si, 0.0 st". Memory usage is shown as: "MiB Mem : 5875.2 total, 4630.5 free, 737.5 used, 658.1 buff/cache". Swap usage is: "MiB Swap: 2048.0 total, 2048.0 free, 0.0 used, 5137.7 avail Mem".

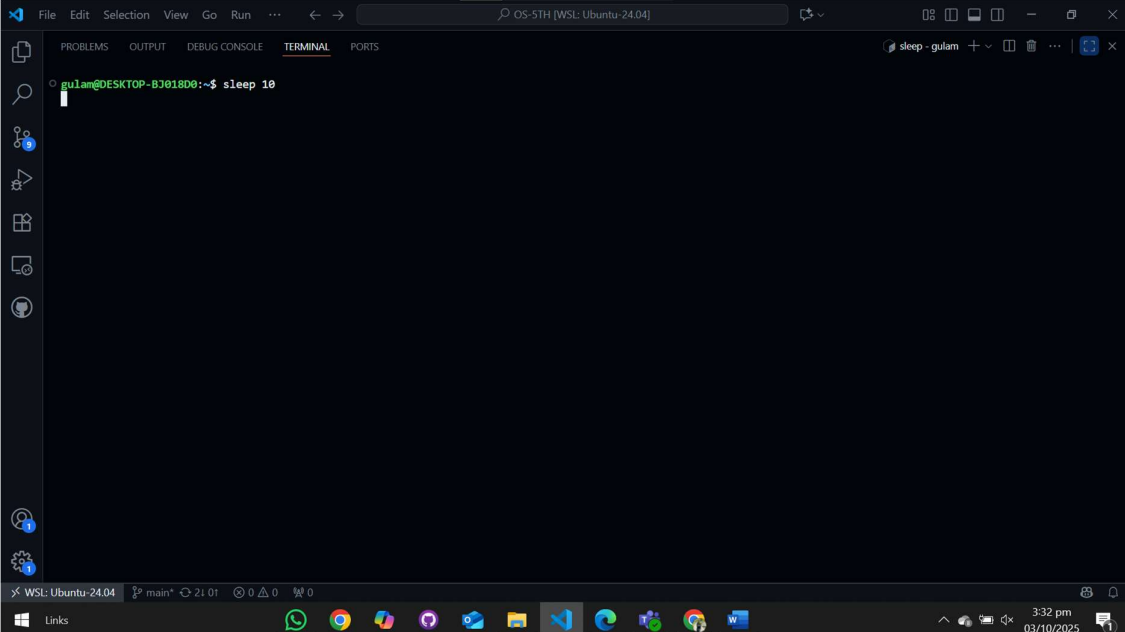
The main part of the output is a table of processes. The header row is: `PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND`. The table lists various system processes and user processes. For example, the first few rows are:

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
15146	gulam	20	0	11372	7680	6656	S	4.3	0.1	0:00.13	ssh
3472	gulam	20	0	31.9g	170128	54656	S	3.3	2.8	1:54.23	node
3433	root	20	0	3096	1168	1024	S	0.3	0.0	0:00.95	Relay(3436)
3437	gulam	20	0	1262064	59144	46720	S	0.3	1.0	0:01.31	node
4191	gulam	20	0	1068632	79016	47232	S	0.3	1.3	0:10.75	node
15003	gulam	20	0	9280	5376	3328	R	0.3	0.1	0:00.03	top
15145	gulam	20	0	7520	4480	4224	S	0.3	0.1	0:00.01	git

The terminal window also shows a taskbar at the bottom with various application icons and a system tray on the right showing the time as 3:29 pm on 03/10/2025.

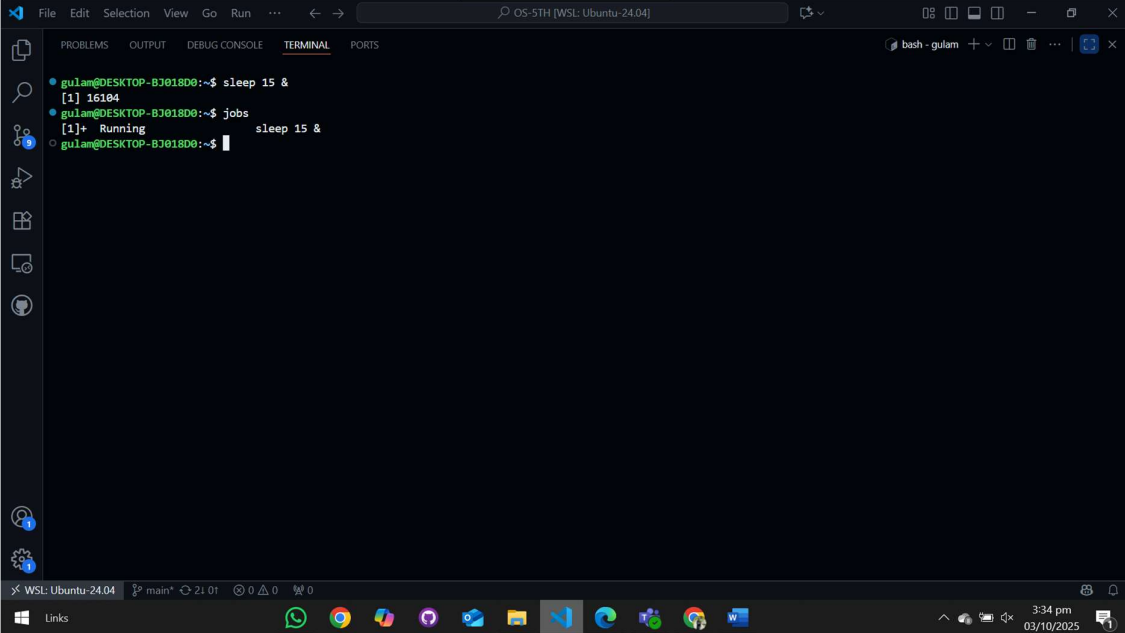
2.3 Foreground and Background Jobs

Foreground:



The screenshot shows a Windows Subsystem for Linux (WSL) terminal window titled "05-5TH [WSL: Ubuntu-24.04]". The terminal displays the output of the `sleep 10` command. The prompt is `gulam@DESKTOP-BJ018D0:~$` and the command entered is `sleep 10`. The cursor is at the end of the command line, indicating that the command is running in the foreground.

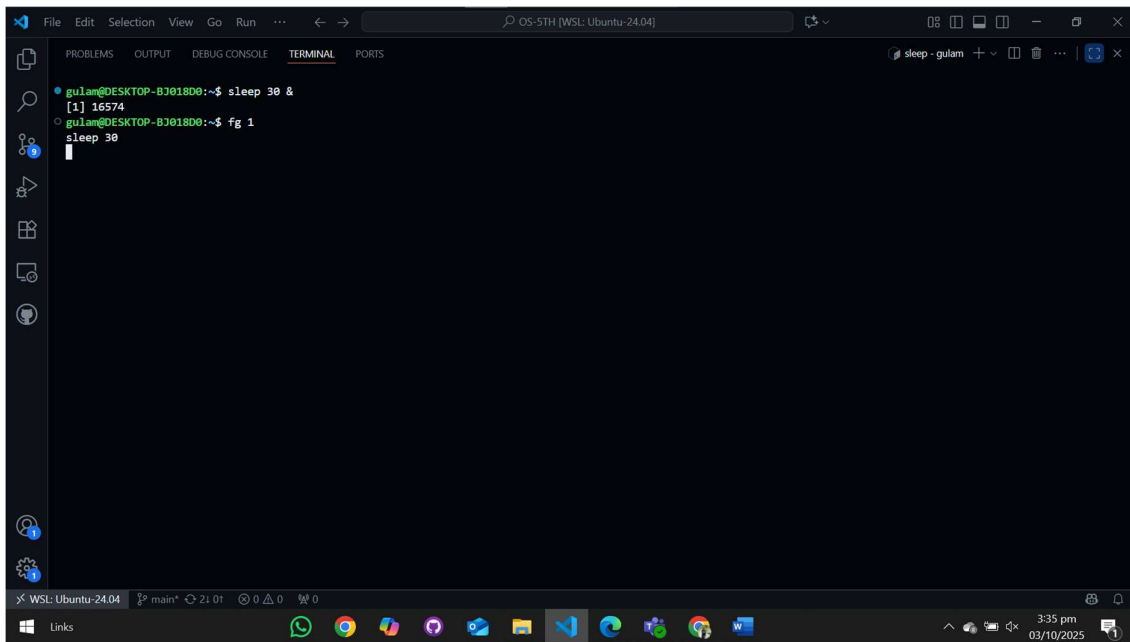
Check background jobs:



The screenshot shows a Visual Studio Code window with a terminal pane open. The terminal is running a bash shell as 'gulam' on a WSL Ubuntu 24.04 environment. The user has executed the command 'sleep 15 &' and then 'jobs'. The output shows a single background job with PID 16104, which is currently 'Running'. The terminal interface includes a sidebar with icons for Explorer, Search, Source Control, and Run and Debug. The bottom status bar shows the file path 'WSL: Ubuntu-24.04', the file name 'main', and the time '2:01'.

```
gulam@DESKTOP-BJ018D0:~$ sleep 15 &
[1] 16104
gulam@DESKTOP-BJ018D0:~$ jobs
[1]+  Running                  sleep 15 &
gulam@DESKTOP-BJ018D0:~$
```

Bring a job to foreground:



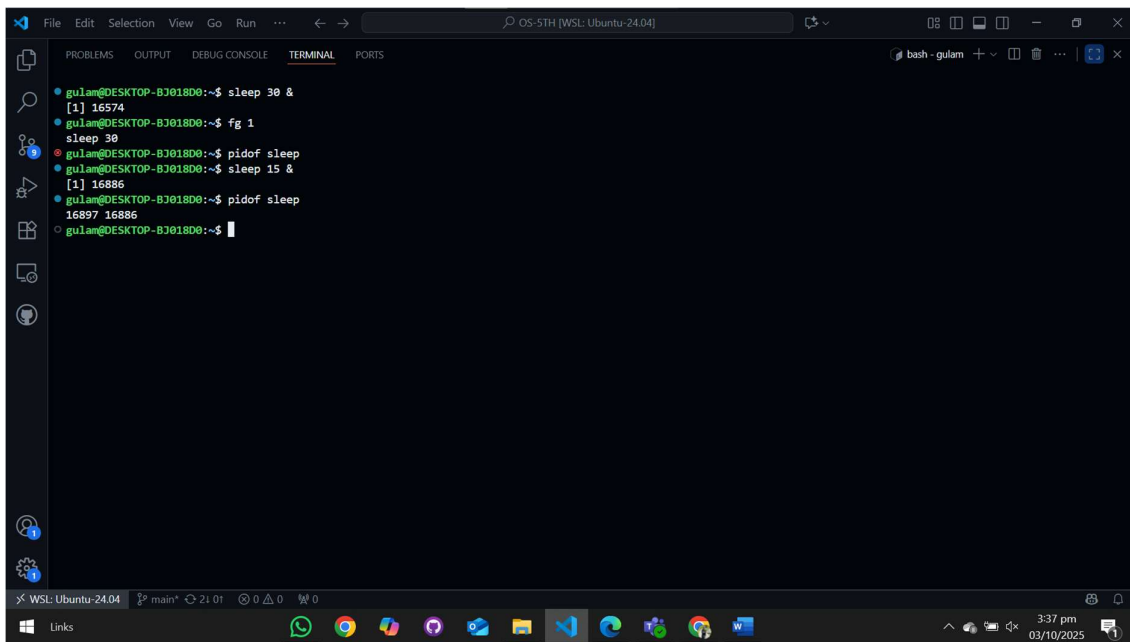
The screenshot shows a Windows terminal window titled "WSL: Ubuntu-24.04". The terminal output is as follows:

```
gulam@DESKTOP-BJ018D0:~$ sleep 30 &
[1] 16574
gulam@DESKTOP-BJ018D0:~$ fg 1
sleep 30
```

The terminal window has a dark theme and a sidebar on the left with various icons. The bottom status bar shows the time as 3:35 pm on 03/10/2025.

2.4 Process Identification

Get PID of a process by name:



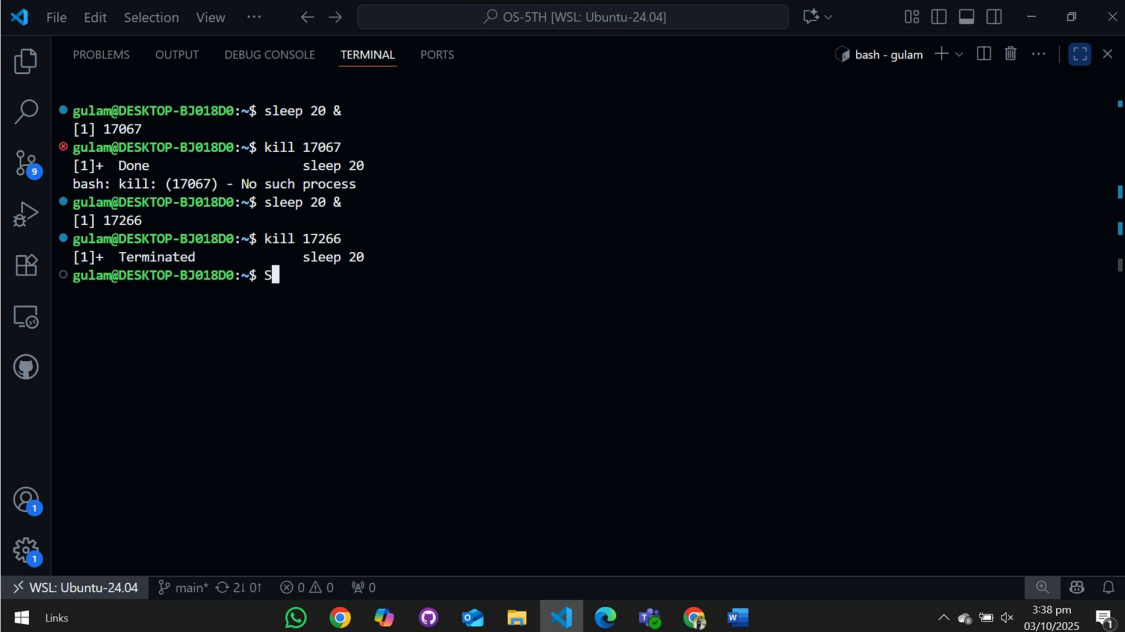
The screenshot shows a Windows terminal window titled "WSL: Ubuntu-24.04". The terminal output is as follows:

```
gulam@DESKTOP-BJ018D0:~$ sleep 30 &
[1] 16574
gulam@DESKTOP-BJ018D0:~$ fg 1
sleep 30
gulam@DESKTOP-BJ018D0:~$ pidof sleep
gulam@DESKTOP-BJ018D0:~$ sleep 15 &
[1] 16886
gulam@DESKTOP-BJ018D0:~$ pidof sleep
16897 16886
gulam@DESKTOP-BJ018D0:~$
```

The terminal window has a dark theme and a sidebar on the left with various icons. The bottom status bar shows the time as 3:37 pm on 03/10/2025.

2.5 Killing Processes

Kill by PID:

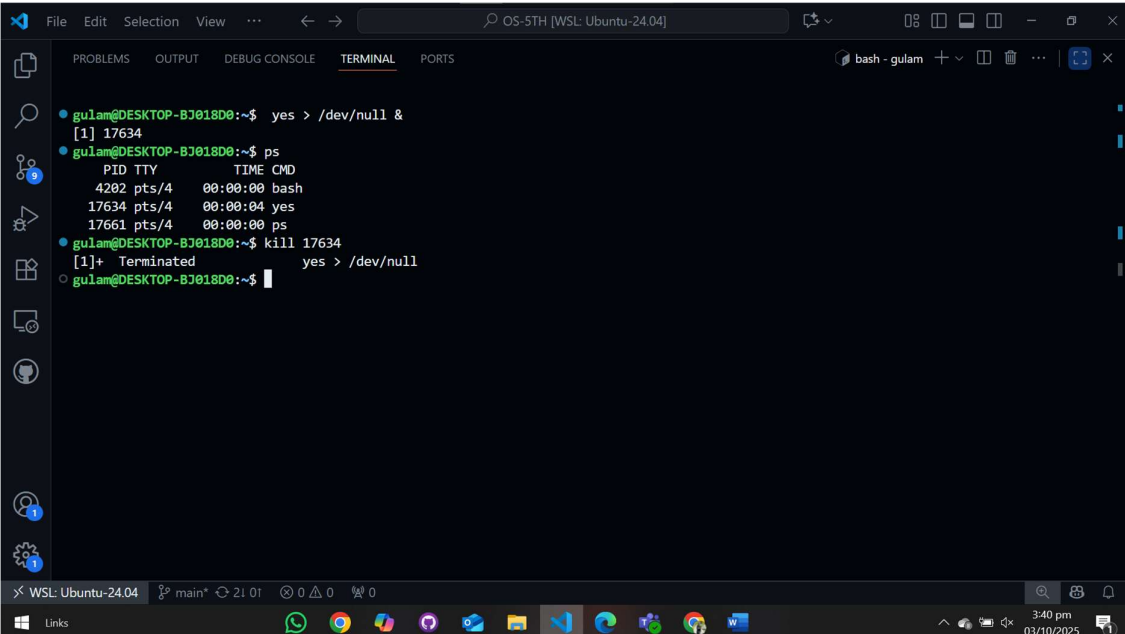


The screenshot shows a WSL terminal window titled "OS-5TH [WSL: Ubuntu-24.04]". The terminal output is as follows:

```
gulam@DESKTOP-BJ018D0:~$ sleep 20 &
[1] 17067
gulam@DESKTOP-BJ018D0:~$ kill 17067
[1]+  Done                  sleep 20
bash: kill: (17067) - No such process
gulam@DESKTOP-BJ018D0:~$ sleep 20 &
[1] 17266
gulam@DESKTOP-BJ018D0:~$ kill 17266
[1]+  Terminated          sleep 20
gulam@DESKTOP-BJ018D0:~$
```

. Run an infinite process:

yes > /dev/null &

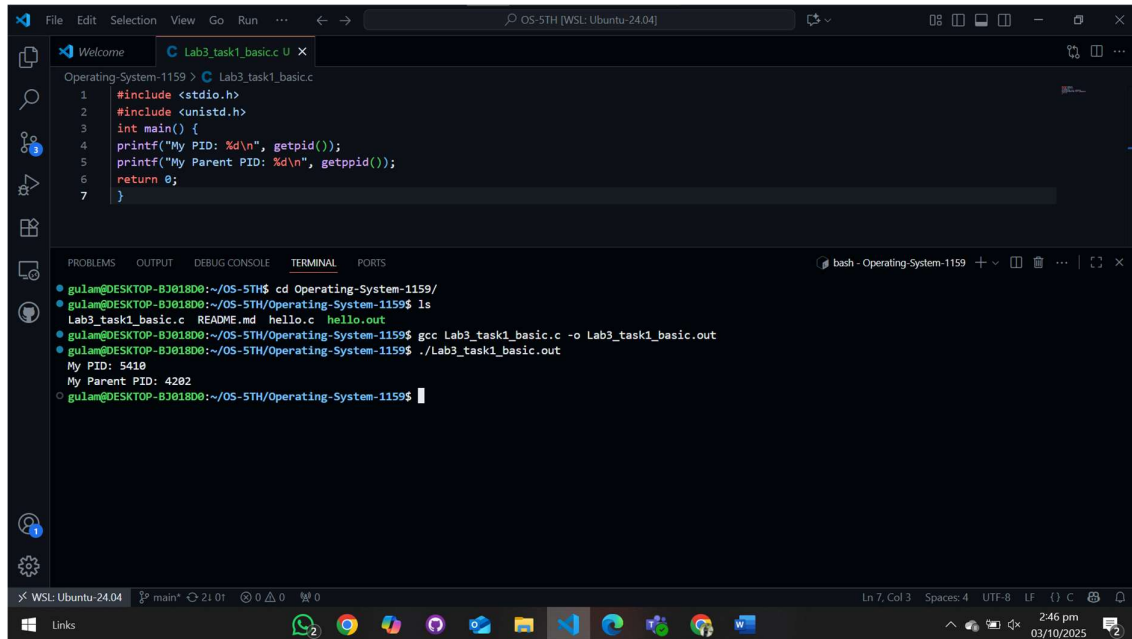


The screenshot shows a WSL terminal window titled "OS-5TH [WSL: Ubuntu-24.04]". The terminal output is as follows:

```
gulam@DESKTOP-BJ018D0:~$ yes > /dev/null &
[1] 17634
gulam@DESKTOP-BJ018D0:~$ ps
  PID TTY          TIME CMD
  4202 pts/4    00:00:00 bash
  17634 pts/4    00:00:04 yes
  17661 pts/4    00:00:00 ps
gulam@DESKTOP-BJ018D0:~$ kill 17634
[1]+  Terminated          yes > /dev/null
gulam@DESKTOP-BJ018D0:~$
```

3.C Programs on Processes

Program 1: Print PID and PPID.



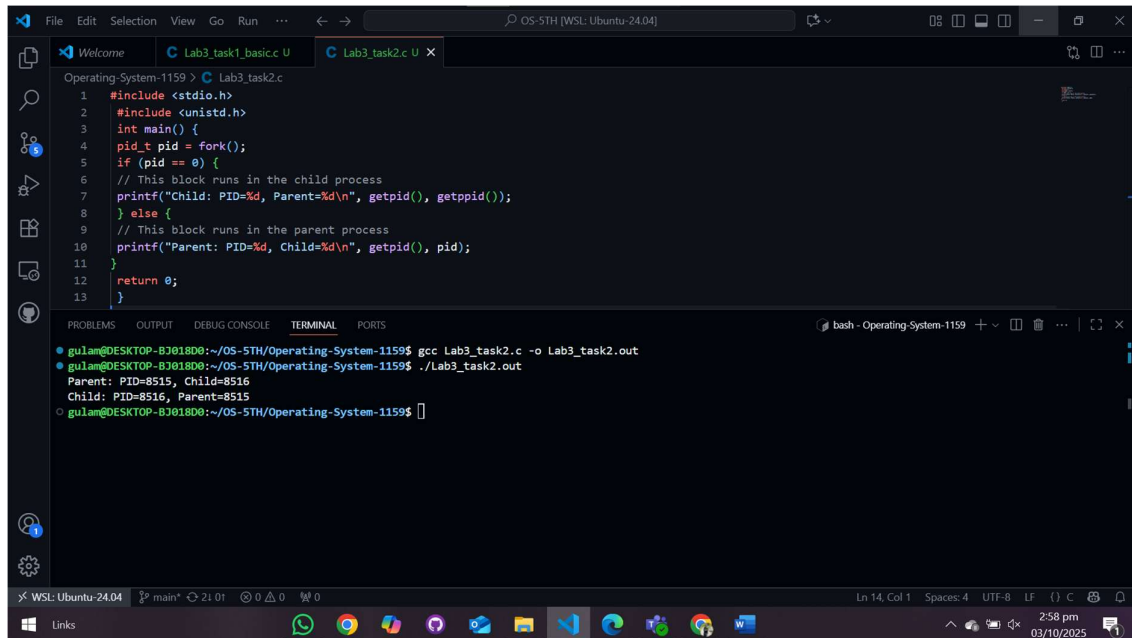
The screenshot shows a VS Code editor with a file named `Lab3_task1_basic.c`. The code is as follows:

```
1 #include <stdio.h>
2 #include <unistd.h>
3 int main() {
4     printf("My PID: %d\n", getpid());
5     printf("My Parent PID: %d\n", getppid());
6     return 0;
7 }
```

The terminal output shows the following commands and results:

```
gulam@DESKTOP-BJ018D0:~/OS-5TH$ cd Operating-System-1159/
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ ls
Lab3_task1_basic.c  README.md  hello.c  hello.out
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ gcc Lab3_task1_basic.c -o Lab3_task1_basic.out
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ ./Lab3_task1_basic.out
My PID: 5418
My Parent PID: 4282
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$
```

Program 2: Fork Creating Child Processes.



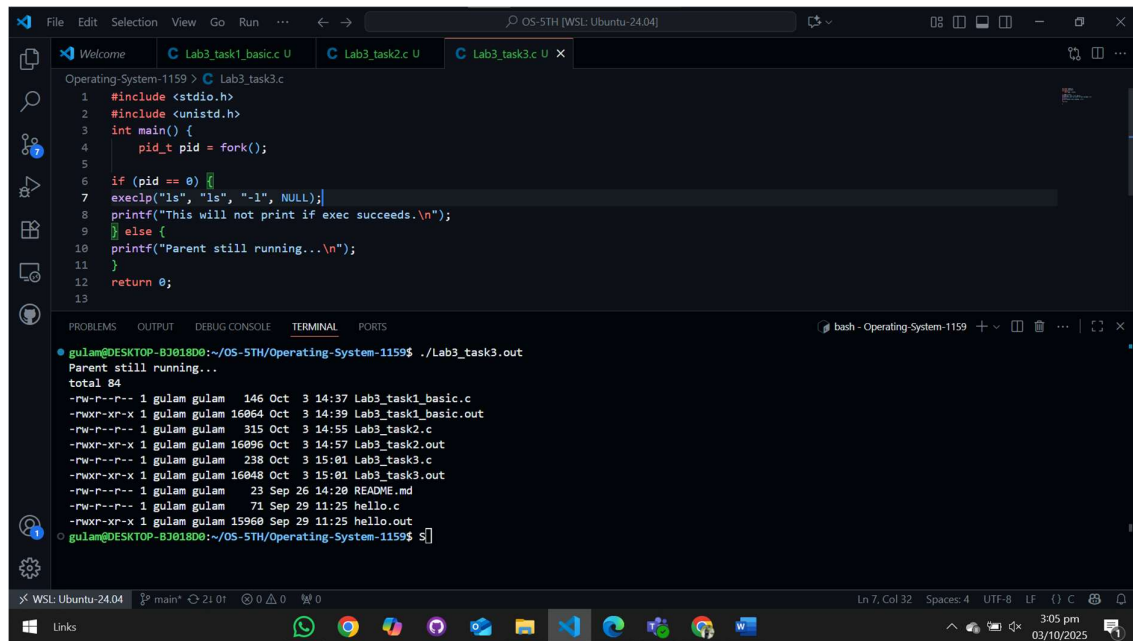
The screenshot shows a VS Code editor with two files: `Lab3_task1_basic.c` and `Lab3_task2.c`. The code in `Lab3_task2.c` is as follows:

```
1 #include <stdio.h>
2 #include <unistd.h>
3 int main() {
4     pid_t pid = fork();
5     if (pid == 0) {
6         // This block runs in the child process
7         printf("Child: PID=%d, Parent=%d\n", getpid(), getppid());
8     } else {
9         // This block runs in the parent process
10        printf("Parent: PID=%d, Child=%d\n", getpid(), pid);
11    }
12    return 0;
13 }
```

The terminal output shows the following commands and results:

```
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ gcc Lab3_task2.c -o Lab3_task2.out
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ ./Lab3_task2.out
Parent: PID=8515, Child=8516
Child: PID=8516, Parent=8515
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$
```

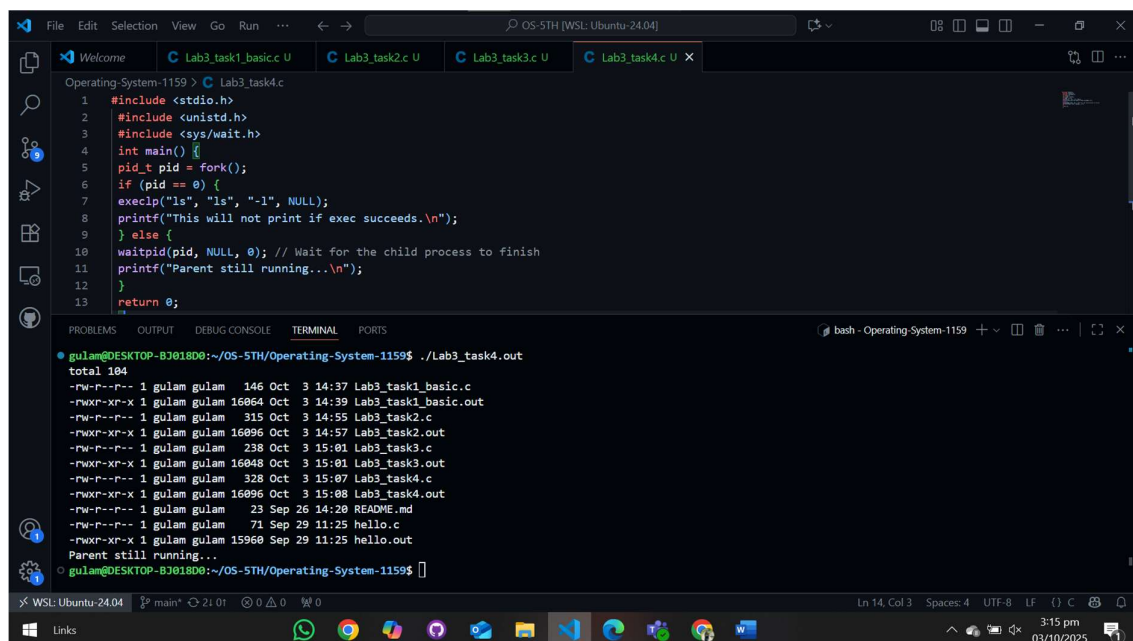

Program 3: Execl – Replacing a New Process



```
Operating-System-1159 > C Lab3_task3.c
1 #include <stdio.h>
2 #include <unistd.h>
3 int main() {
4     pid_t pid = fork();
5
6     if (pid == 0) {
7         execl("ls", "ls", "-l", NULL);
8         printf("This will not print if exec succeeds.\n");
9     } else {
10        printf("Parent still running...\n");
11    }
12    return 0;
13 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ ./Lab3_task3.out
Parent still running...
total 84
-rw-r--r-- 1 gulam gulam 146 Oct 3 14:37 Lab3_task1_basic.c
-rwxr-xr-x 1 gulam gulam 16064 Oct 3 14:39 Lab3_task1_basic.out
-rw-r--r-- 1 gulam gulam 315 Oct 3 14:55 Lab3_task2.c
-rwxr-xr-x 1 gulam gulam 16096 Oct 3 14:57 Lab3_task2.out
-rw-r--r-- 1 gulam gulam 238 Oct 3 15:01 Lab3_task3.c
-rwxr-xr-x 1 gulam gulam 16048 Oct 3 15:01 Lab3_task3.out
-rw-r--r-- 1 gulam gulam 23 Sep 26 14:20 README.md
-rw-r--r-- 1 gulam gulam 71 Sep 29 11:25 hello.c
-rwxr-xr-x 1 gulam gulam 15960 Sep 29 11:25 hello.out
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$
```

Program 4: Wait – Synchronization



```
Operating-System-1159 > C Lab3_task4.c
1 #include <stdio.h>
2 #include <unistd.h>
3 #include <sys/wait.h>
4 int main() {
5     pid_t pid = fork();
6
7     if (pid == 0) {
8         execlp("ls", "ls", "-l", NULL);
9         printf("This will not print if exec succeeds.\n");
10    } else {
11        waitpid(pid, NULL, 0); // Wait for the child process to finish
12        printf("Parent still running...\n");
13    }
14    return 0;
15 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$ ./Lab3_task4.out
total 104
-rw-r--r-- 1 gulam gulam 146 Oct 3 14:37 Lab3_task1_basic.c
-rwxr-xr-x 1 gulam gulam 16064 Oct 3 14:39 Lab3_task1_basic.out
-rw-r--r-- 1 gulam gulam 315 Oct 3 14:55 Lab3_task2.c
-rwxr-xr-x 1 gulam gulam 16096 Oct 3 14:57 Lab3_task2.out
-rw-r--r-- 1 gulam gulam 238 Oct 3 15:01 Lab3_task3.c
-rwxr-xr-x 1 gulam gulam 16048 Oct 3 15:01 Lab3_task3.out
-rw-r--r-- 1 gulam gulam 328 Oct 3 15:07 Lab3_task4.c
-rwxr-xr-x 1 gulam gulam 16096 Oct 3 15:08 Lab3_task4.out
-rw-r--r-- 1 gulam gulam 23 Sep 26 14:20 README.md
-rw-r--r-- 1 gulam gulam 71 Sep 29 11:25 hello.c
-rwxr-xr-x 1 gulam gulam 15960 Sep 29 11:25 hello.out
Parent still running...
gulam@DESKTOP-BJ018D0:~/OS-5TH/Operating-System-1159$
```