**OPERATING**

**SYSTEM**

**LAB – 8**

**Overlay Concepts.**

**Submitted by: ~**

**Ayushi Kumari**

**RA1911003010865.**

**Objective:**

To understand the overlay concepts and practice how to overlay the current process to new process in Linux using C.

**Overlay:** Overlay is the concept which enables the user to run another new process from the currently running process address space.

**Practice questions:**

1. **Execute the Following Program and write the output.**

**CODE: -**

$vi ex1.c

#include <stdio.h>

#include <unistd.h>

int main()

{

printf("Transfer to execlp function \n");

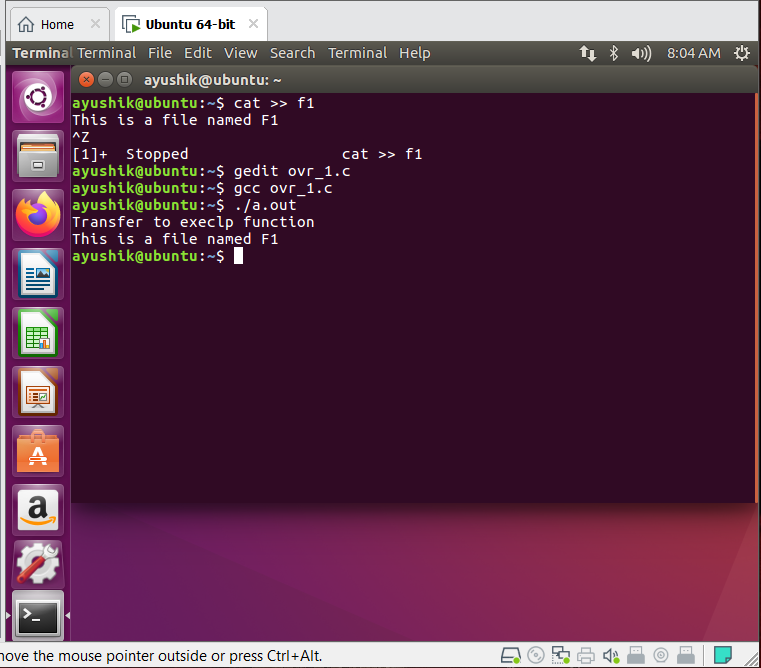
execlp("head", "head","-2","f1",NULL);

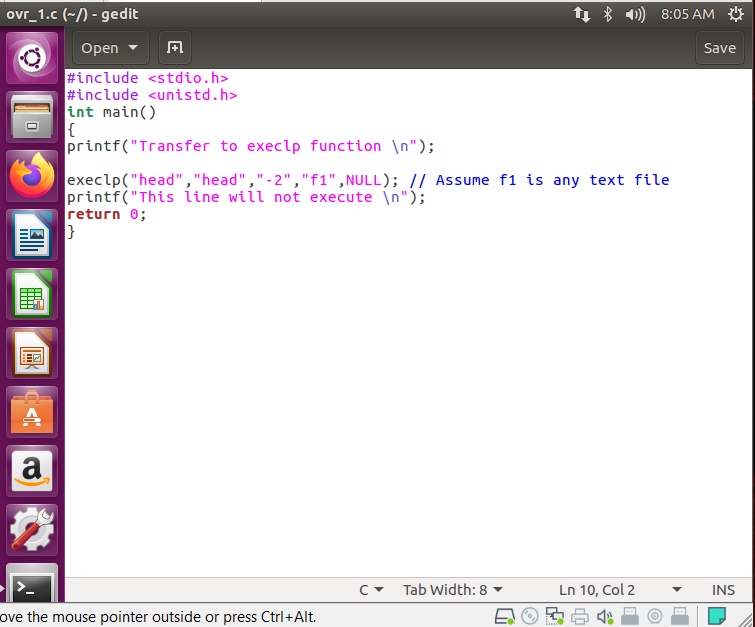
printf("This line will not execute \n");

return 0;

}

**OUTPUT:**





* **Why second printf statement is not executing?**

Executable function is the first printf statement.

When OS executes **execlp** function the control transfers to the **execlp** function due to which the control will never return back to the calling place, hence on successful execution of **execlp** function, it will never return back. So, this program will not execute the **second printf statement**.

1. **Rewrite question 1 with execl() function. Pass the 3rd and 4th argument of the function execl() through command line arguments.**

**$vi ex2.c**

**Input: ./a.out -3 f1**

**CODE-**

#include<stdio.h>

#include<unistd.h>

int main(int argc, char \*argv[])

{

printf("Transfer to execlp function\n");

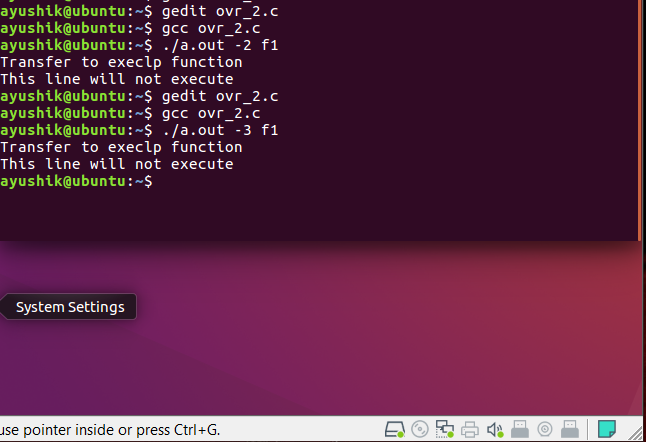
execl("/user/bin/head", "head", argv[1], argv[2], NULL);

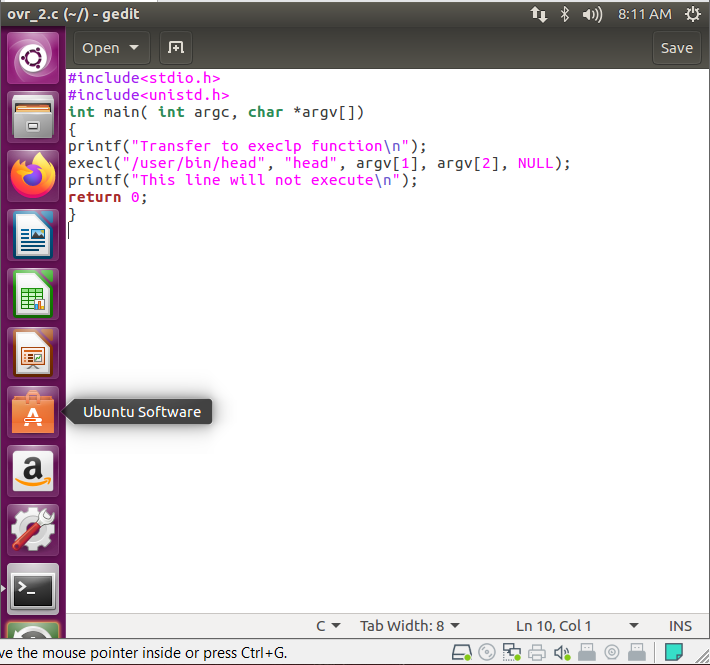
printf("This line will not execute\n");

return 0;

}

**OUTPUT:**





**Outcome: Learned and implemented the overlay concept in Linux using C.**