**ASSIGNMENT - X**

**TITLE :**

Write a C program to dynamically adjust the priority of a process using the nice() system call and observe the impact on scheduling.

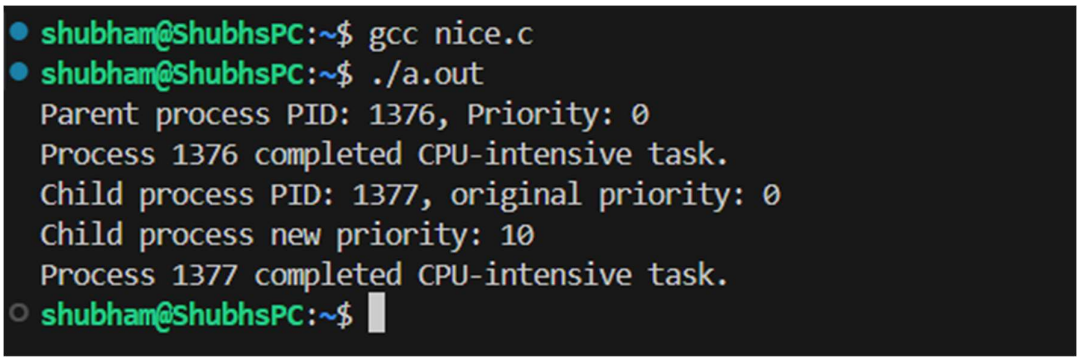
**NAME :** Shinde Shubham Dnyandev, **DIV :** SY-B, **ROLL NO. :** 23107121.

**PROGRAM :**

```
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <sys/resource.h>
#include <errno.h>

void cpu_intensive_task()
{
   long int i, sum =0;
   for (i= 0; i < 100; i++)
   {
      sum +=1;
   }
   printf("Process %d completed CPU-intensive task.\n" , getpid());
}

int main()
{
   pid_t pid = fork();
```

```
    if (pid < 0)
    {
      perror("fork failed");
      return 1;
    }

    if(pid == 0)
    {
      printf("Child process PID: %d, original priority: %d\n" , getpid(),
getpriority(PRIO_PROCESS, 0));
      if (nice(10) == -1)
      {
        perror("Nice failed in child");
      }
      else
      {
        printf("Child process new priority: %d\n" , getpriority(PRIO_PROCESS, 0));
      }
      cpu_intensive_task();
    }
    else
    {
        printf("Parent process PID: %d, Priority: %d\n" , getpid() ,
getpriority(PRIO_PROCESS, 0));
        cpu_intensive_task();
        wait(NULL);
    }
    return 0;
}
```

**OUTPUT :**


shubham@ShubhsPC:~$ gcc nice.c
shubham@ShubhsPC:~$ ./a.out
Parent process PID: 1376, Priority: 0
Process 1376 completed CPU-intensive task.
Child process PID: 1377, original priority: 0
Child process new priority: 10
Process 1377 completed CPU-intensive task.
shubham@ShubhsPC:~$