Name : Nisarg .k. Amlani
Roll : Ce001
Id : 22ceueg082

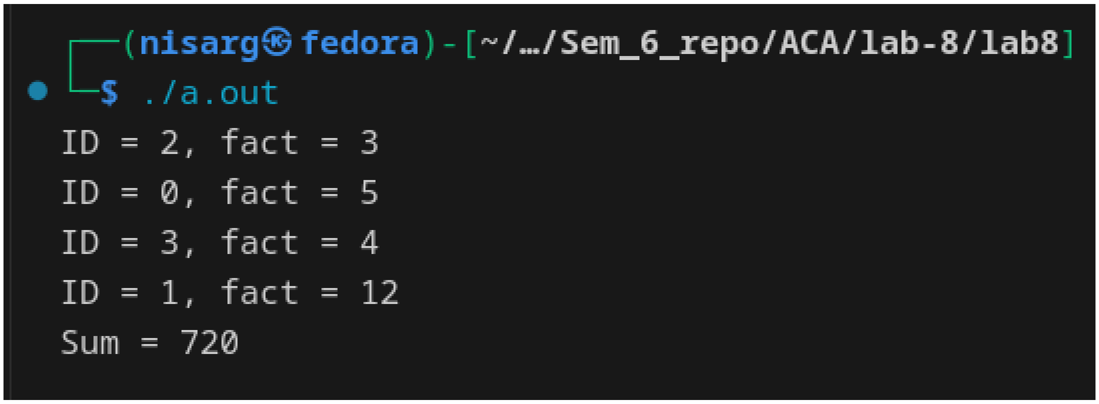
LAB :0 8

---

1. Write an OpenMP program using 4 threads, each thread calculates factorial of its id and then all the factorials respective to threads need to be added to get on final sum. Use shared and private clauses.
Output: Individual threads factorial result with their respective ids and the final sum of all the factorials.

```c
#include<stdio.h>
#include<stdlib.h>
#include"omp.h"

int main(){
   int sum = 1, fact = 1;
   omp_set_num_threads(4);
   #pragma omp parallel private(fact) reduction(*:sum)
   {
       fact = 1;
       int id = omp_get_thread_num();
       int nthrds = omp_get_num_threads();
       for(int i = 1+id; i <= 6; i+=nthrds){
           fact *=  i;
       }
       printf("ID = %d, fact = %d\n",id,fact);
       sum *= fact;
   }
   printf("Sum = %d\n",sum);
}
```



```
┌──(nisarg㉿fedora)-[~/…/Sem_6_repo/ACA/lab-8/lab8]
└─$ ./a.out
ID = 2, fact = 3
ID = 0, fact = 5
ID = 3, fact = 4
ID = 1, fact = 12
Sum = 720
```