VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

CSE4001 Parallel and Distributed Computing

Digital Assignment-5 (ELA)

Name: Nakul Jadeja

Reg no.: 19BCE0660

School of Computer Science and Engineering

Course Code: CSE4001

Slot: L11+L12

Winter Semester 2021-22

Professor: Narayanamoorthi M

(1) Assume a 4-digit number and perform the following tasks using Open MP and MPI programming approach.

- To find the sum of individual digits

- To find the sum of even and odd digits and count of even and odd digits

- To check the number is prime or not

- To check the 4-digit number is palindrome or not

CODE: -

MPI CODE:

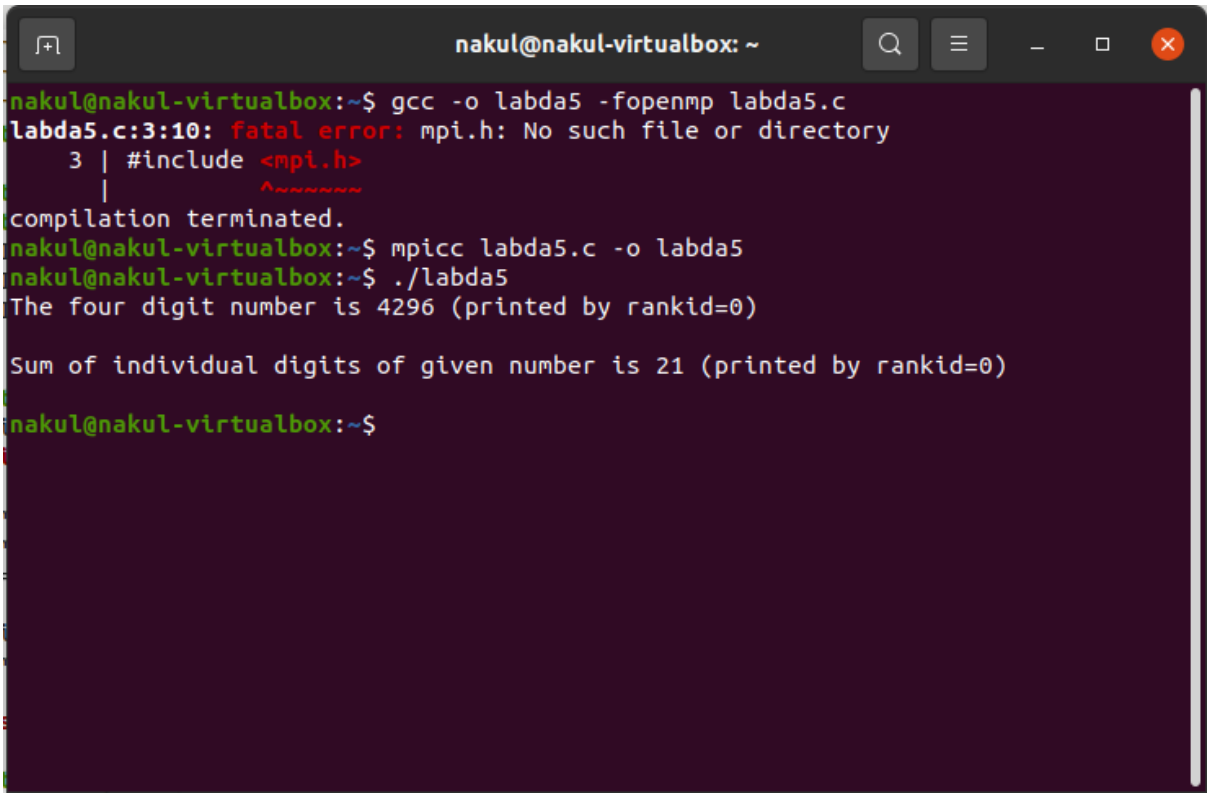
```
#include <stdio.h>
#include <mpi.h>
#include <math.h>
int main(int args, char **argv)
{
int size, rankid;
int a = 4296;
MPI_Init(NULL, NULL);
MPI_Comm_size(MPI_COMM_WORLD, &size);
MPI_Comm_rank(MPI_COMM_WORLD, &rankid);
if (rankid == 0)
{
int num, sum = 0, b = a;
printf("The four digit number is %d (printed by rankid=%d)\n\n", a, rankid);
while (b > 0)
{
```

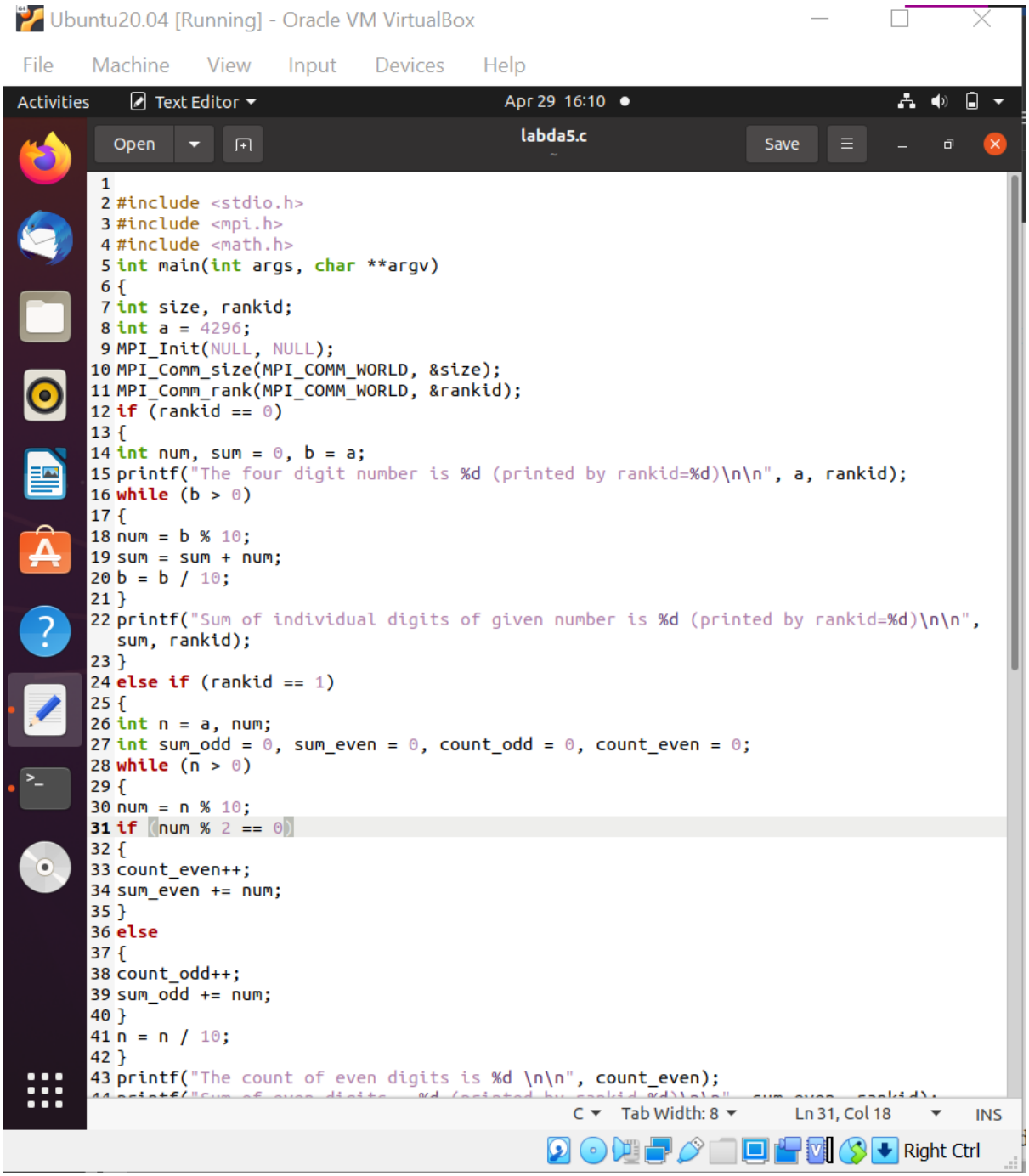
```
num = b % 10;
sum = sum + num;
b = b / 10;
}
printf("Sum of individual digits of given number is %d (printed by
rankid=%d)\n\n", sum, rankid);
}
else if (rankid == 1)
{
int n = a, num;
int sum_odd = 0, sum_even = 0, count_odd = 0, count_even = 0;
while (n > 0)
{
num = n % 10;
if (num % 2 == 0)
{
count_even++;
sum_even += num;
}
else
{
count_odd++;
sum_odd += num;
}
n = n / 10;
}
printf("The count of even digits is %d \n\n", count_even);
```

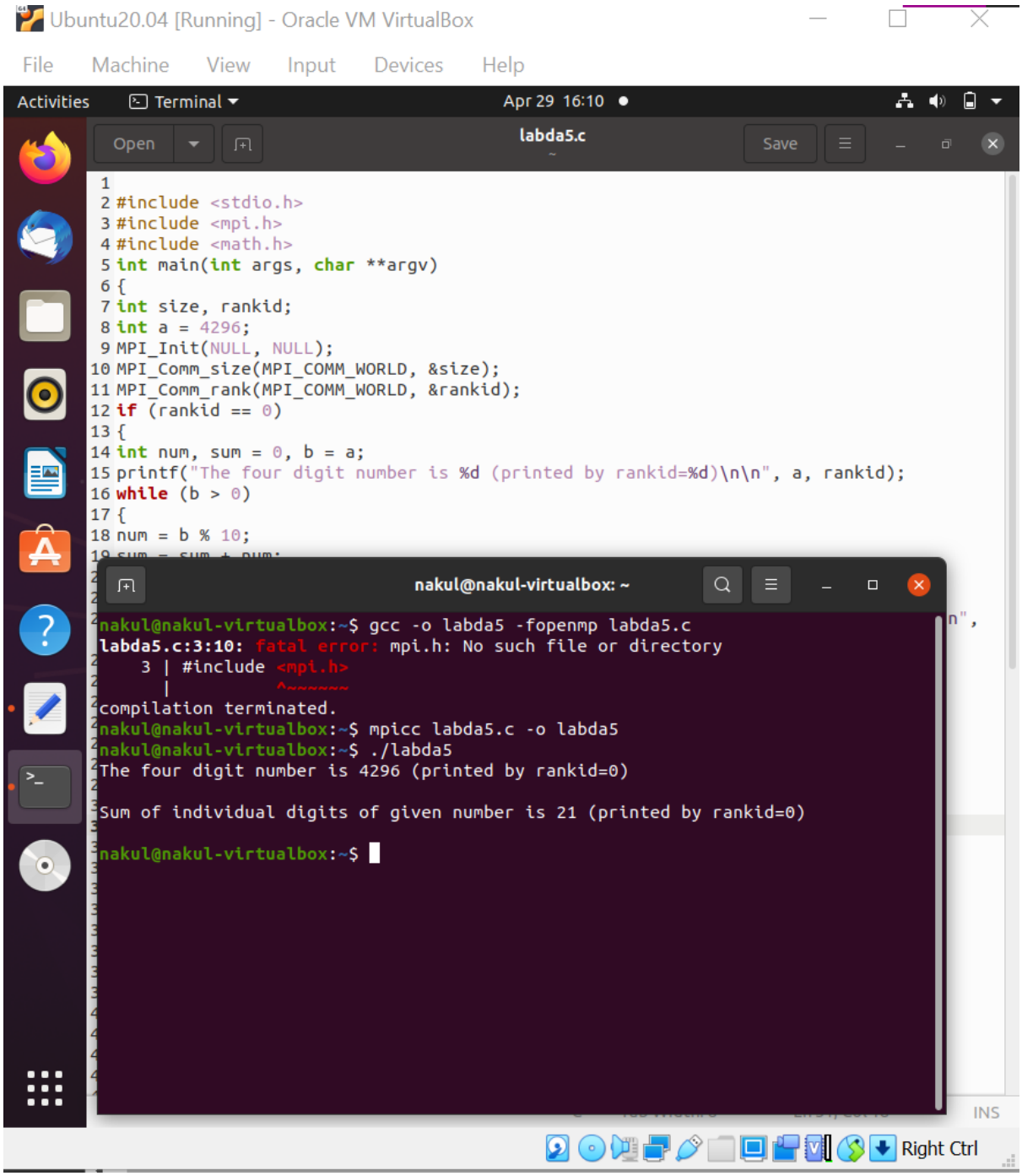
```
printf("Sum of even digits = %d (printed by rankid=%d)\n\n", sum_even,
rankid);
printf("The count of odd digits = %d \n\n", count_odd);
printf("Sum of odd digits = %d (printed by rankid=%d)\n\n", sum_odd, rankid);
}
else if (rankid == 2)
{
int flag = 0;
for (int i = 2; i <= a / 2; ++i)
{
if (a % i == 0)
{
flag = 1;
break;
}
}
if (flag == 0)
printf("%d is a prime number (printed by rankid=%d)\n\n", a, rankid);
else
printf("%d is not a prime number (printed by rankid=%d)\n\n", a, rankid);
}
else if (rankid == 3)
{
int number = a;
int reversed, remainder = 0;
while (number != 0)
{
remainder = number % 10;
```

```
reversed = reversed * 10 + remainder;
number /= 10;
}
if (a == reversed)
printf("%d is a palindrome. (printed by rankid=%d)\n\n", a, rankid);
else
printf("%d is not a palindrome. (printed by rankid=%d)\n\n", a, rankid);
}
MPI_Finalize();
}
```

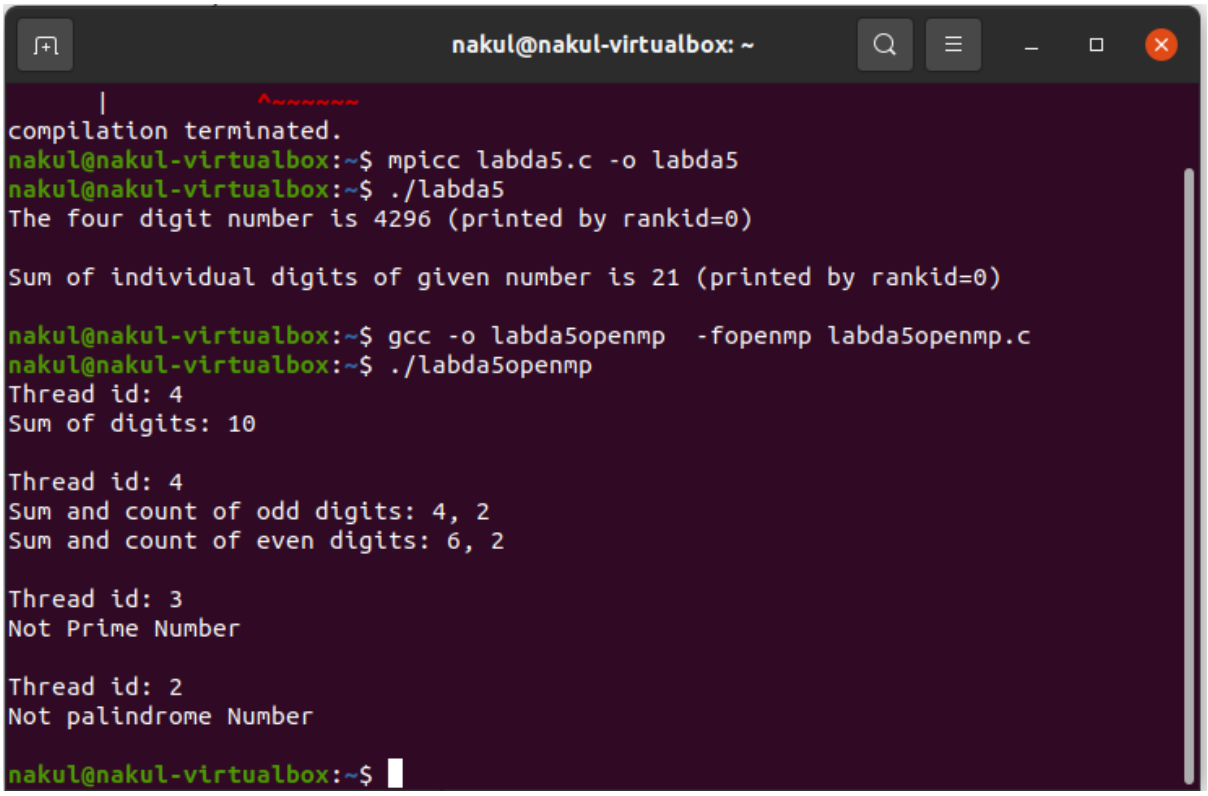
SCREENSHOT OF THE OUTPUT: -

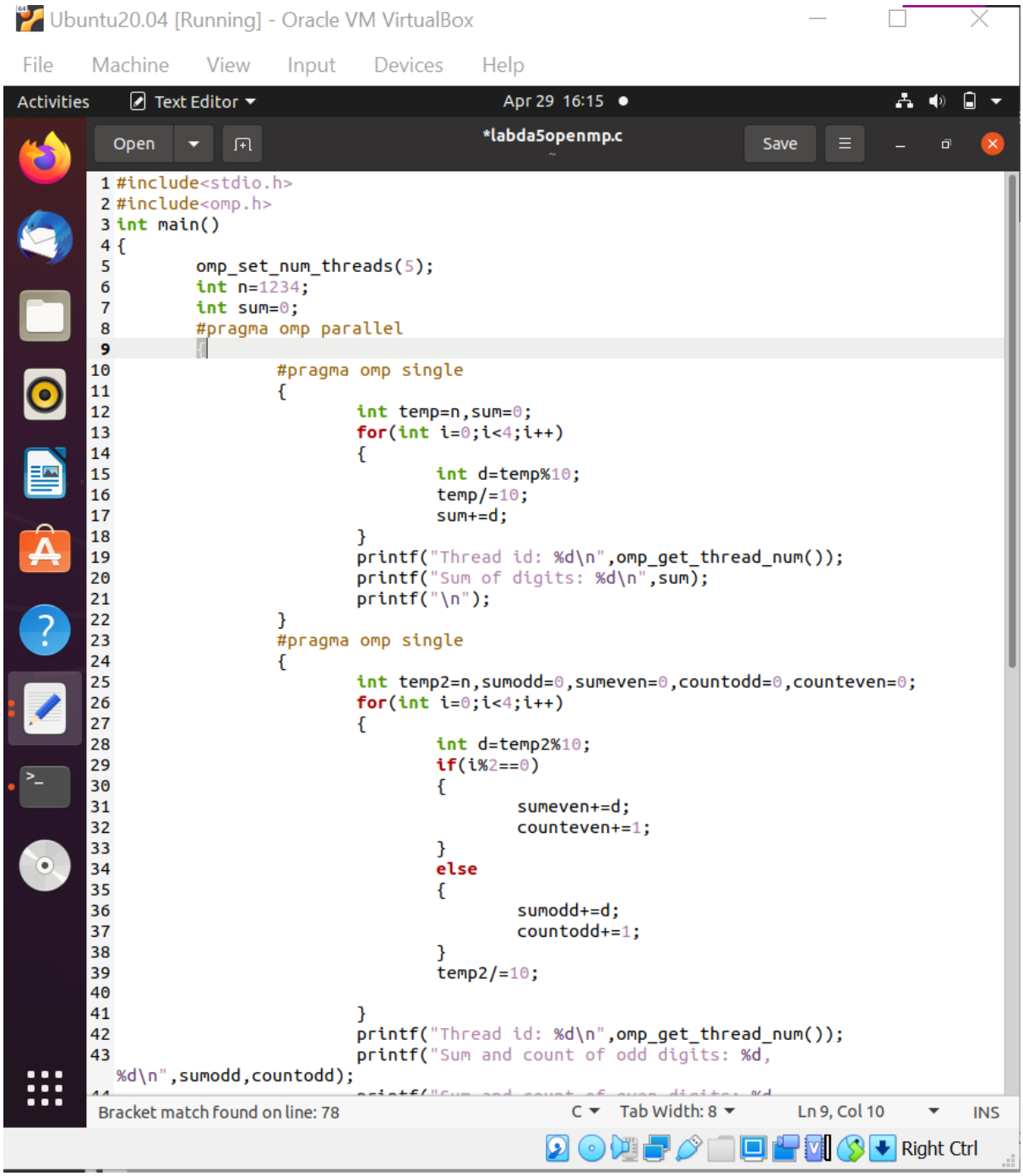


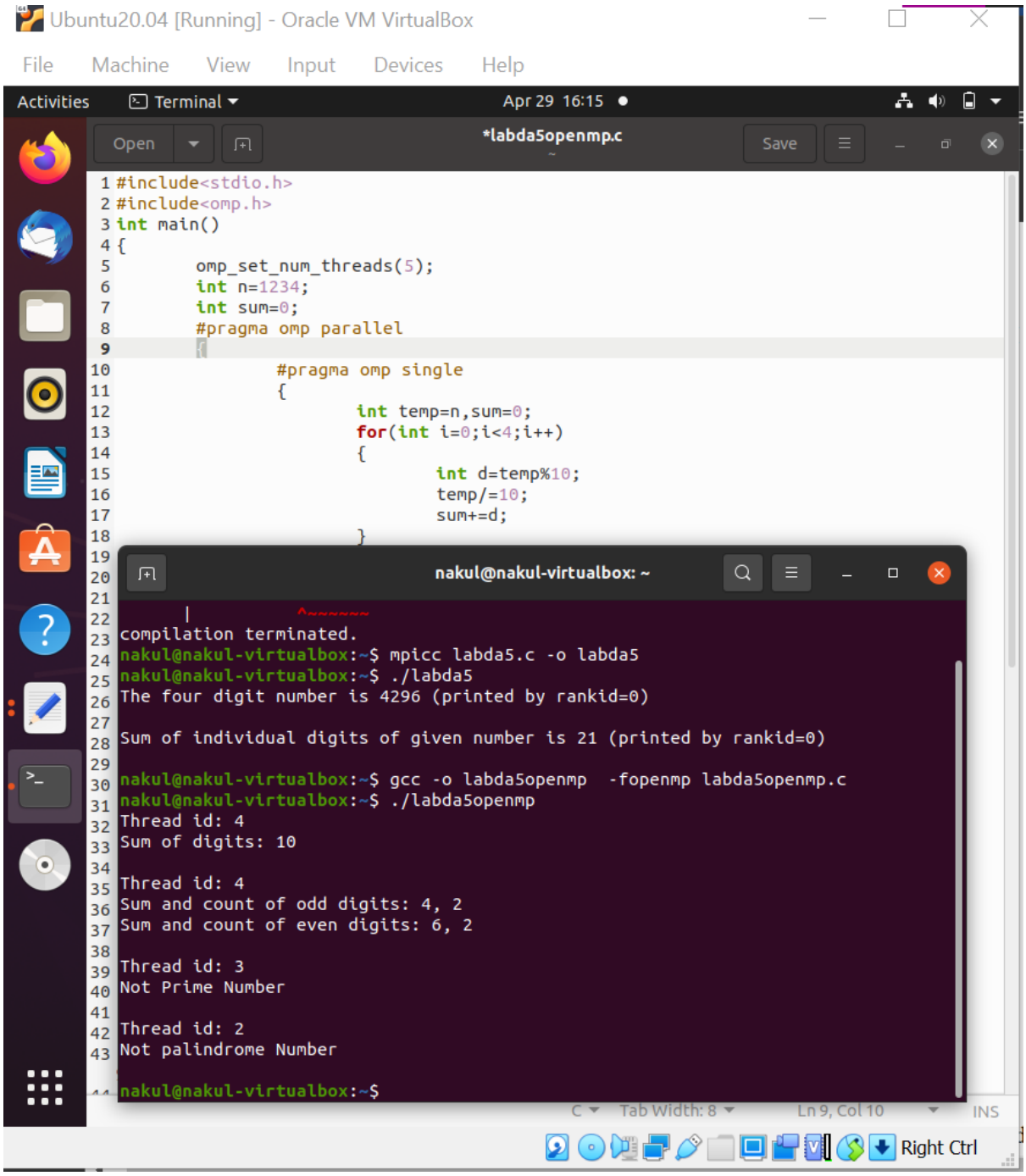

Ubuntu20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Text Editor Apr 29 16:10
Open labda5.c Save
1
2 #include <stdio.h>
3 #include <mpi.h>
4 #include <math.h>
5 int main(int args, char **argv)
6 {
7 int size, rankid;
8 int a = 4296;
9 MPI_Init(NULL, NULL);
10 MPI_Comm_size(MPI_COMM_WORLD, &size);
11 MPI_Comm_rank(MPI_COMM_WORLD, &rankid);
12 if (rankid == 0)
13 {
14 int num, sum = 0, b = a;
15 printf("The four digit number is %d (printed by rankid=%d)\n\n", a, rankid);
16 while (b > 0)
17 {
18 num = b % 10;
19 sum = sum + num;
20 b = b / 10;
21 }
22 printf("Sum of individual digits of given number is %d (printed by rankid=%d)\n\n",
sum, rankid);
23 }
24 else if (rankid == 1)
25 {
26 int n = a, num;
27 int sum_odd = 0, sum_even = 0, count_odd = 0, count_even = 0;
28 while (n > 0)
29 {
30 num = n % 10;
31 if (num % 2 == 0)
32 {
33 count_even++;
34 sum_even += num;
35 }
36 else
37 {
38 count_odd++;
39 sum_odd += num;
40 }
41 n = n / 10;
42 }
43 printf("The count of even digits is %d \n\n", count_even);
C Tab Width: 8 Ln 31, Col 18 INS
Right Ctrl


Ubuntu20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Apr 29 16:10
Open labda5.c Save
#include <stdio.h>
#include <mpi.h>
#include <math.h>
int main(int args, char **argv)
{
int size, rankid;
int a = 4296;
MPI_Init(NULL, NULL);
MPI_Comm_size(MPI_COMM_WORLD, &size);
MPI_Comm_rank(MPI_COMM_WORLD, &rankid);
if (rankid == 0)
{
int num, sum = 0, b = a;
printf("The four digit number is %d (printed by rankid=%d)\n\n", a, rankid);
while (b > 0)
{
num = b % 10;
nakul@nakul-virtualbox: ~
nakul@nakul-virtualbox:~$ gcc -o labda5 -fopenmp labda5.c
labda5.c:3:10: fatal error: mpi.h: No such file or directory
3 | #include <mpi.h>
| ^~~~~~~
compilation terminated.
nakul@nakul-virtualbox:~$ mpicc labda5.c -o labda5
nakul@nakul-virtualbox:~$ ./labda5
The four digit number is 4296 (printed by rankid=0)
Sum of individual digits of given number is 21 (printed by rankid=0)
nakul@nakul-virtualbox:~$
INS
Right Ctrl

OPEN MP CODE: -


nakul@nakul-virtualbox: ~
| ^~~~~~~
compilation terminated.
nakul@nakul-virtualbox:~$ mpicc labda5.c -o labda5
nakul@nakul-virtualbox:~$ ./labda5
The four digit number is 4296 (printed by rankid=0)
Sum of individual digits of given number is 21 (printed by rankid=0)
nakul@nakul-virtualbox:~$ gcc -o labda5openmp -fopenmp labda5openmp.c
nakul@nakul-virtualbox:~$ ./labda5openmp
Thread id: 4
Sum of digits: 10
Thread id: 4
Sum and count of odd digits: 4, 2
Sum and count of even digits: 6, 2
Thread id: 3
Not Prime Number
Thread id: 2
Not palindrome Number
nakul@nakul-virtualbox:~$


Ubuntu20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Text Editor
Apr 29 16:15
Open
*labda5openmp.c
Save
#include<stdio.h>
#include<omp.h>
int main()
{
omp_set_num_threads(5);
int n=1234;
int sum=0;
#pragma omp parallel
{
#pragma omp single
{
int temp=n,sum=0;
for(int i=0;i<4;i++)
{
int d=temp%10;
temp/=10;
sum+=d;
}
printf("Thread id: %d\n",omp_get_thread_num());
printf("Sum of digits: %d\n",sum);
printf("\n");
}
#pragma omp single
{
int temp2=n,sumodd=0,sumeven=0,countodd=0,counteven=0;
for(int i=0;i<4;i++)
{
int d=temp2%10;
if(i%2==0)
{
sumeven+=d;
counteven+=1;
}
else
{
sumodd+=d;
countodd+=1;
}
temp2/=10;
}
printf("Thread id: %d\n",omp_get_thread_num());
printf("Sum and count of odd digits: %d,
%d\n",sumodd,countodd);
Bracket match found on line: 78
C
Tab Width: 8
Ln 9, Col 10
INS
Right Ctrl


Ubuntu20.04 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Apr 29 16:15
Open *labda5openmp.c Save
#include<stdio.h>
#include<omp.h>
int main()
{
omp_set_num_threads(5);
int n=1234;
int sum=0;
#pragma omp parallel
{
#pragma omp single
{
int temp=n,sum=0;
for(int i=0;i<4;i++)
{
int d=temp%10;
temp/=10;
sum+=d;
}
nakul@nakul-virtualbox: ~
^~~~~~~
compilation terminated.
nakul@nakul-virtualbox:~$ mpicc labda5.c -o labda5
nakul@nakul-virtualbox:~$ ./labda5
The four digit number is 4296 (printed by rankid=0)
Sum of individual digits of given number is 21 (printed by rankid=0)
nakul@nakul-virtualbox:~$ gcc -o labda5openmp -fopenmp labda5openmp.c
nakul@nakul-virtualbox:~$ ./labda5openmp
Thread id: 4
Sum of digits: 10
Thread id: 4
Sum and count of odd digits: 4, 2
Sum and count of even digits: 6, 2
Thread id: 3
Not Prime Number
Thread id: 2
Not palindrome Number
nakul@nakul-virtualbox:~$
C Tab Width: 8 Ln 9, Col 10 INS
Right Ctrl

THE END