NAME: MOUNVI PODAPATI
REG. NO: 19BCE0396
SLOT: L15 + L16
FACULTY: PROF. DEEPAK B.D
DATED: 1ST SEPTEMBER 2021

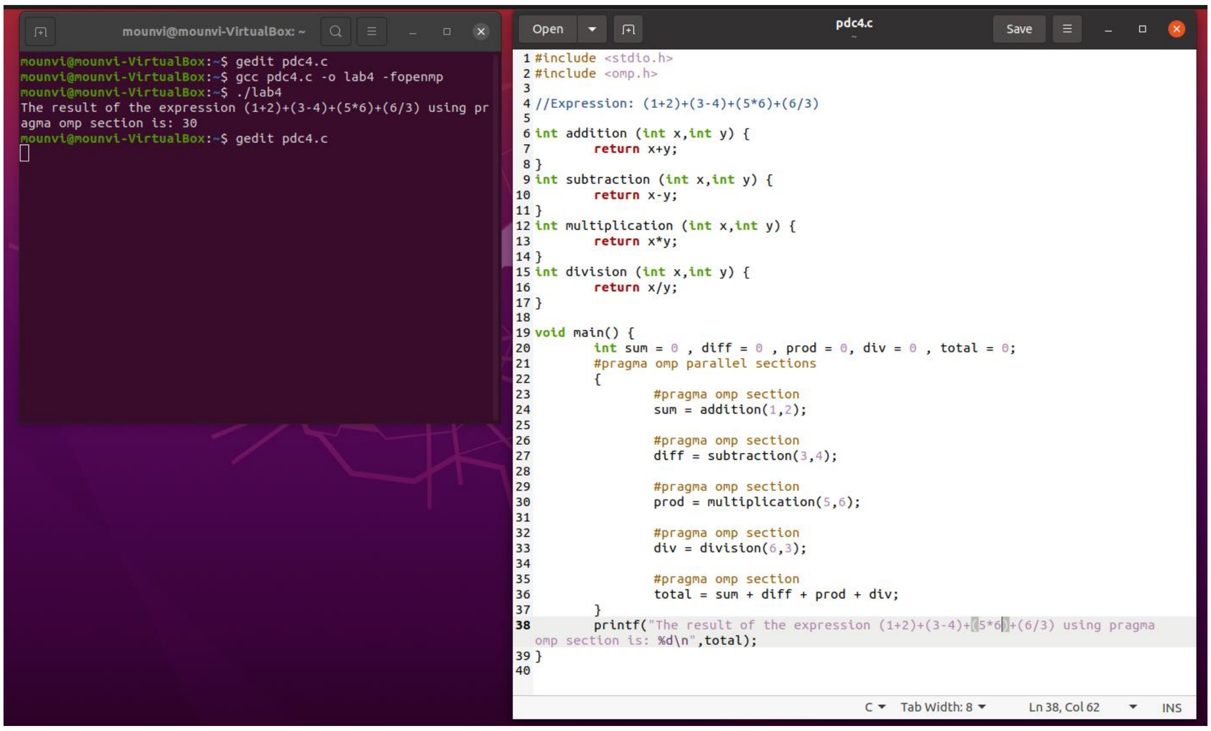
LAB ASSESSMENT – 3

AIM: Write a simple OpenMP program to demonstrate Arithmetic Operation using Section Clause.

SOURCE CODE:

```
#include <stdio.h>
#include <omp.h>

//Expression: (1+2)+(3-4)+(5*6)+(6/3)

int addition (int x,int y) {
	return x+y;
}
int subtraction (int x,int y) {
	return x-y;
}
int multiplication (int x,int y) {
	return x*y;
}
int division (int x,int y) {
	return x/y;
}

void main() {
	int sum = 0 , diff = 0 , prod = 0, div = 0 , total = 0;
	#pragma omp parallel sections
	{
		#pragma omp section
		sum = addition(1,2);

		#pragma omp section
		diff = subtraction(3,4);

		#pragma omp section
		prod = multiplication(5,6);

		#pragma omp section
		div = division(6,3);

		#pragma omp section
		total = sum + diff + prod + div;
	}
	printf("The result of the expression (1+2)+(3-4)+(5*6)+(6/3) using pragma omp section is: %d\n",total);
}
```

EXECUTION:



```
mounvi@mounvi-VirtualBox:~$ gedit pdc4.c
mounvi@mounvi-VirtualBox:~$ gcc pdc4.c -o lab4 -fopenmp
mounvi@mounvi-VirtualBox:~$ ./lab4
The result of the expression (1+2)+(3-4)+(5*6)+(6/3) using pr
agma omp section is: 30
mounvi@mounvi-VirtualBox:~$ gedit pdc4.c
```

```
#include <stdio.h>
#include <omp.h>

//Expression: (1+2)+(3-4)+(5*6)+(6/3)

int addition (int x,int y) {
	return x+y;
}
int subtraction (int x,int y) {
	return x-y;
}
int multiplication (int x,int y) {
	return x*y;
}
int division (int x,int y) {
	return x/y;
}

void main() {
	int sum = 0 , diff = 0 , prod = 0, div = 0 , total = 0;
	#pragma omp parallel sections
	{
		#pragma omp section
		sum = addition(1,2);

		#pragma omp section
		diff = subtraction(3,4);

		#pragma omp section
		prod = multiplication(5,6);

		#pragma omp section
		div = division(6,3);

		#pragma omp section
		total = sum + diff + prod + div;
	}
	printf("The result of the expression (1+2)+(3-4)+(5*6)+(6/3) using pragma omp section is: %d\n",total);
}
```

REMARKS:

Section clause has been exploited and using this I have verified arithmetic operations like addition, subtraction, multiplication and division. The results were calculated separately by using pragma section. The section clause is used to indicate that the particular parallel section will be having multiple subsections that are executed in parallel. The section clause identifies these subsections and the section clause can be very useful when different parts of the same logic can be conducted in parallel, enhancing time efficiency.