CSE4001 - Parallel and Distributed Computing, Fall 2019
Vellore Institute of Technology
Instructor: Prof Deebak B D - SCOPE

Lab report

Title of Lab: Beginning with OpenMP
Assessment #: 2
Date: 02|08|2019

Author's name: Gagan Deep Singh
Registration ID: 17BCI0140
Lab section: Friday L59 + L60

AIM:
Write a simple OpenMP program to demonstrate the use of for, private and shared clause.

a. Sum of 'n' array Using Private and Shared Clause
b. Product of 'n' array using Private and Shared Clause

SOURCE CODE:

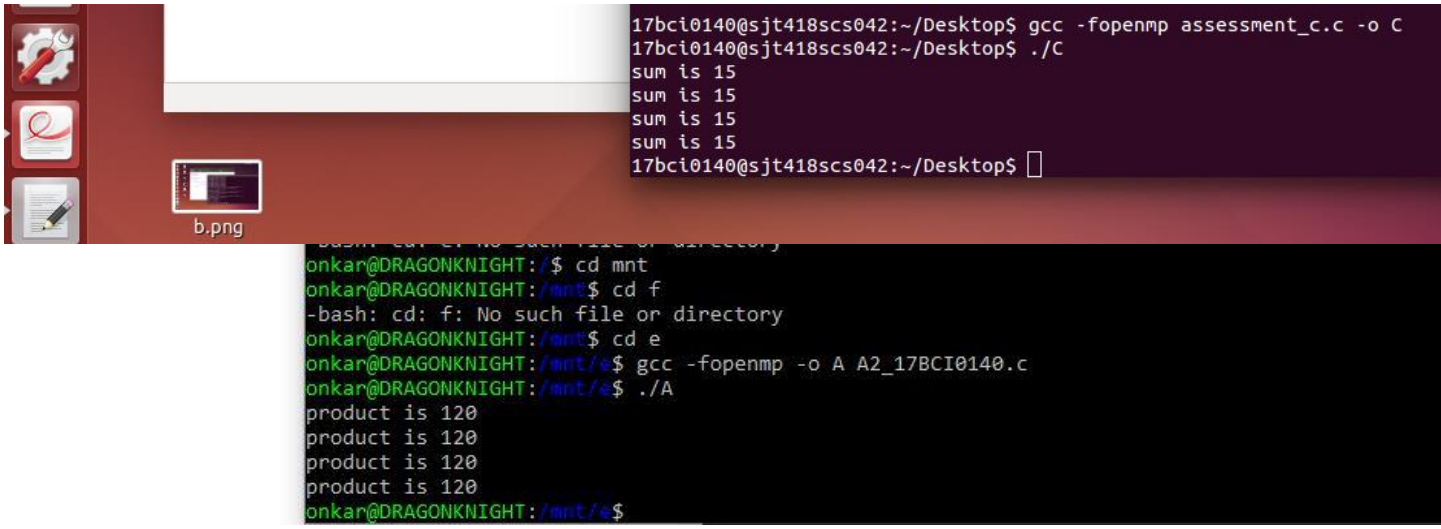
part-a

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


int main(void){

	int arr[5] = {1, 2, 3, 4, 5};
	int sum = 0, i;
	  #pragma omp parallel private(i) shared(sum, arr)
	//for(int i .... it is not possible in pragma
	  {
	      #pragma omp for
	      for(i = 0; i < 5; i++)
	            sum += arr[i];

	      printf("sum is %d\n", sum);
	}
	return 0;
}
```

part-b

```
#include<stdio.h>
#include<omp.h>
int main(void){
	int arr[5] = {1, 2, 3, 4, 5};
	int product = 1, i;
	  #pragma omp parallel private(i) shared(product, arr)
	  {
	      #pragma omp for
	      for(i = 0; i < 5; i++)
	            product *= arr[i];

	      printf("product is %d\n", product);
	}
	return 0;
}
```

EXECUTION:



RESULTS:

A variable in an OpenMP can be either shared or private. If a variable is shared, then there exists one instance of this variable which is shared among all threads. If a variable is private, then each thread in a team of threads has its own local copy of the private variable.