**Lab Exercise 4**

**Critical Construct and Reduction Clause**

**Name:** Ayush Sharma

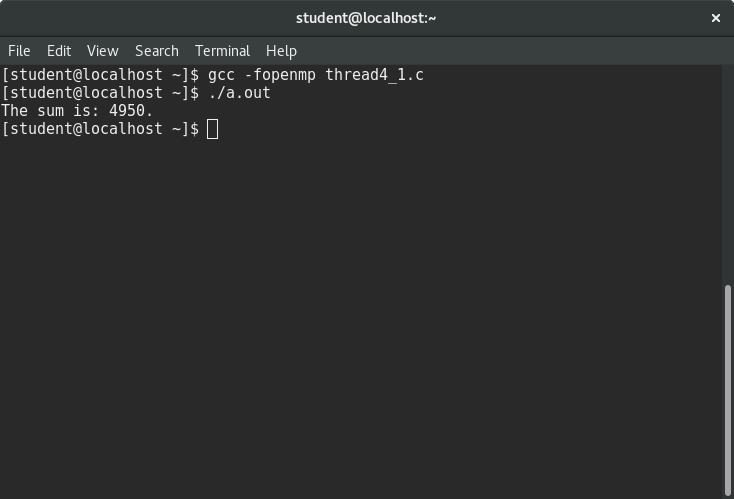
**Reg. No:** 15BCE1335

**Faculty:** Prof. Gayatri. R

1. Critical Construct

int sum=0;  
int lsum=0;  
int array[100];  
for(int i = 0; i < 100; i++){  
 array[i] = i;  
}  
#pragma omp parallel private(lsum)  
{  
 lsum=0;  
 #pragma omp for  
 for(int i = 0; i < 100; i++){  
 lsum += array[i];  
 }  
 #pragma omp critical  
 {   
 sum+=lsum;  
 }  
}  
printf("The sum is: %d.\n",sum);

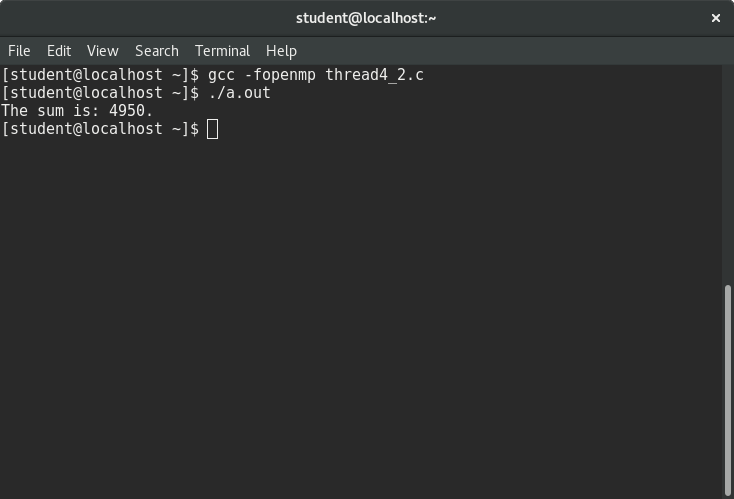
**Output:**

****

2. Reduction Clause

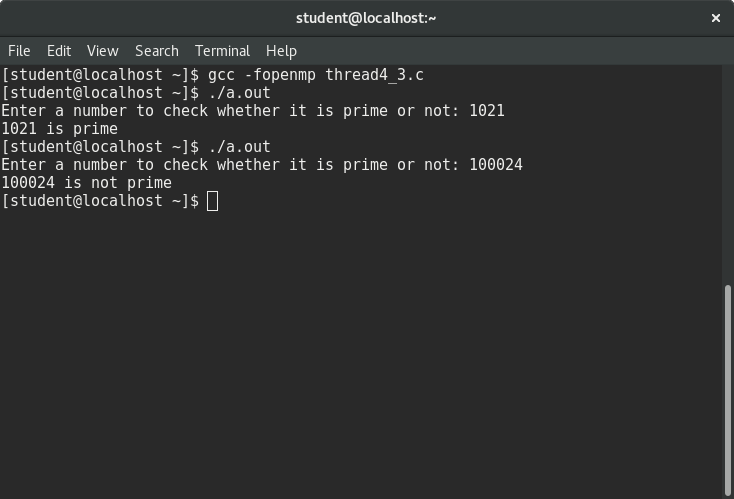
int sum=0;  
int array[100];  
for(int i = 0; i < 100; i++){  
 array[i] = i;  
}  
#pragma omp parallel for reduction (+:sum)  
 for(int i = 0; i < 100; i++){  
 sum += array[i];  
 }  
printf("The sum is: %d.\n",sum);

**Output:**

****

3. Prime - Critical construct

int prime=1,i,N;   
printf("Enter a number to check whether it is prime or not: ");  
scanf("%d",&N);  
int num=(N/2) + 1;  
#pragma omp parallel private (prime)  
 prime=1;  
 #pragma omp for  
 for (i=2; i<num; i++)  
 {  
 if(N%i==0){  
 prime = 0;  
 }  
 }  
 #pragma omp critical  
 if(prime==1)  
 printf("%d is prime\n",N);  
 else  
 printf("%d is not prime\n",N);  
}



4. Prime - Reduction Clause

int prime=1,i,N;   
printf("Enter a number to check whether it is prime or not: ");  
scanf("%d",&N);  
int num=(N/2) + 1;  
#pragma omp parallel for reduction (\*:prime)  
prime=1;  
for (i=2; i<num; i++)  
{  
if(N%i==0){  
prime = 0;  
}  
}  
if(prime==1)  
 printf("%d is prime\n",N);  
else  
 printf("%d is not prime\n",N);  
}

**Output:**

