

Parallel and Distributed Computing
CSE4001
Fall Semester 2020-21

Lab Assignment 2

**ISHAAN OHRI
18BCE0265**

Aim:

Write a simple OpenMP program to demonstrate the use of 'for' clause.

- Print 'n' array elements
- Sum of n' array elements
- Product of n' array elements

Printing using a single thread

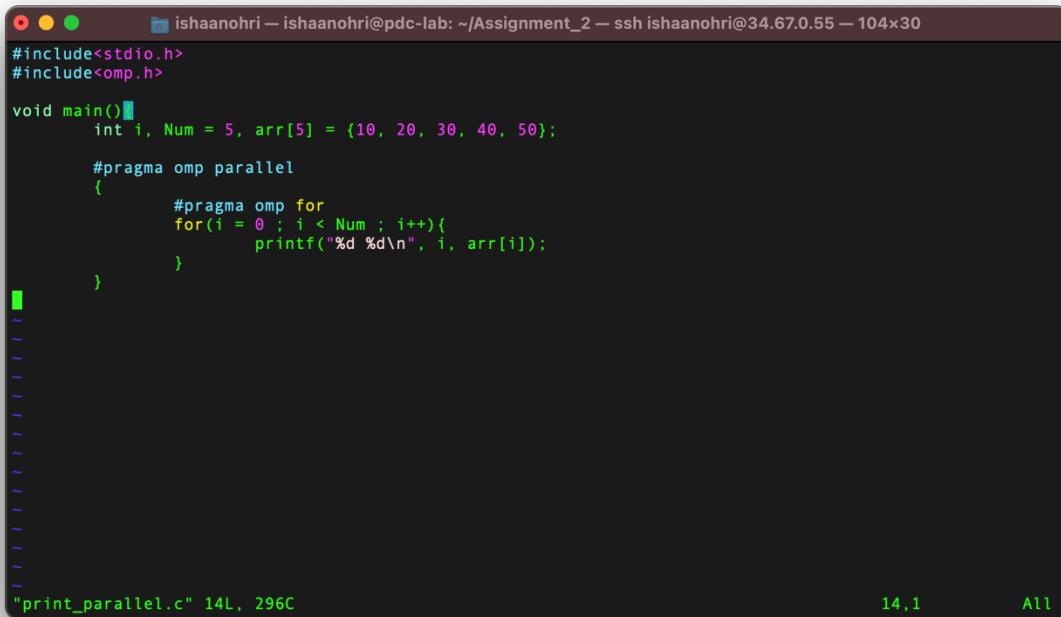
Source Code:

Execution:

```
[ishaanohri@pdc-lab:~/Assignment_2$ gcc -fopenmp print.c -o h  
[ishaanohri@pdc-lab:~/Assignment_2$ ./h  
0 10  
1 20  
2 30  
3 40  
4 50  
ishaanohri@pdc-lab:~/Assignment_2$
```

Printing using multiple threads

Source Code:



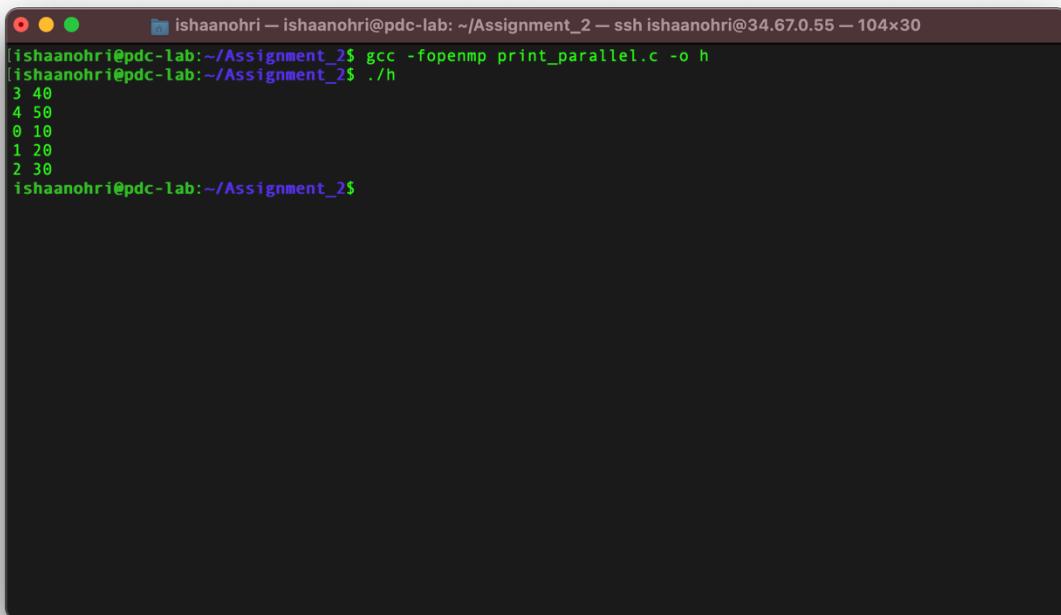
```
#include<stdio.h>
#include<omp.h>

void main()
{
    int i, Num = 5, arr[5] = {10, 20, 30, 40, 50};

    #pragma omp parallel
    {
        #pragma omp for
        for(i = 0 ; i < Num ; i++){
            printf("%d %d\n", i, arr[i]);
        }
    }
}

"print_parallel.c" 14L, 296C 14,1 All
```

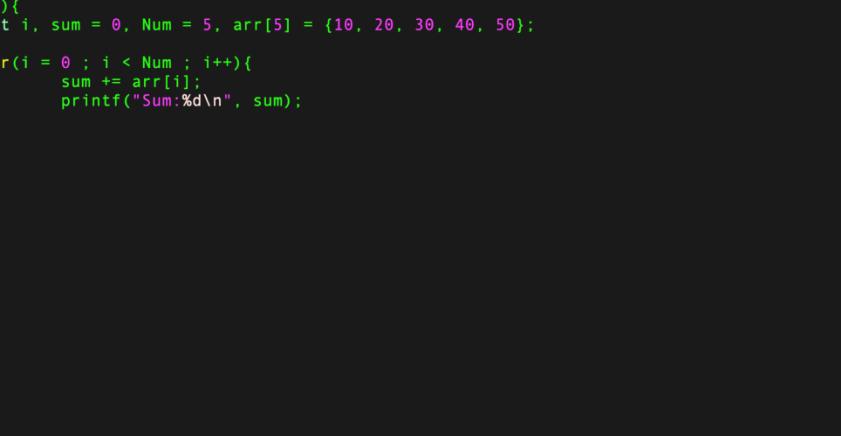
Execution:



```
[ishaanohri@pdc-lab:~/Assignment_2$ gcc -fopenmp print_parallel.c -o h
[ishaanohri@pdc-lab:~/Assignment_2$ ./h
3 40
4 50
0 10
1 20
2 30
ishaanohri@pdc-lab:~/Assignment_2$ ]
```

Sum using a single thread

Source Code:



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

void main(){
    int i, sum = 0, Num = 5, arr[5] = {10, 20, 30, 40, 50};

    for(i = 0 ; i < Num ; i++){
        sum += arr[i];
        printf("Sum:%d\n", sum);
    }
}

"sum.c" 11L, 219C
```

Execution:

```
[ishaanohri@pdc-lab:~/Assignment_2$ ssh ishaanohri@34.67.0.55 - 104x30  
[ishaanohri@pdc-lab:~/Assignment_2$ gcc -fopenmp sum.c -o h  
[ishaanohri@pdc-lab:~/Assignment_2$ ./h  
Sum:10  
Sum:30  
Sum:60  
Sum:100  
Sum:150  
ishaanohri@pdc-lab:~/Assignment_2$
```

Sum using a multiple threads

Source Code:

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

void main(){
    int i, sum = 0, Num = 5, arr[5] = {10, 20, 30, 40, 50};

    #pragma omp parallel
    {
        #pragma omp for
        for(i = 0 ; i < Num ; i++){
            sum += arr[i];
            printf("Sum:%d\n", sum);
        }
    }
}

"sum_parallel.c" 15L, 297C           11,1-8      All
```

Execution:

```
[ishaanohri@pdc-lab:~/Assignment_2$ gcc -fopenmp sum_parallel.c -o h
[ishaanohri@pdc-lab:~/Assignment_2$ ./h
Sum:10
Sum:70
Sum:100
Sum:50
Sum:150
ishaanohri@pdc-lab:~/Assignment_2$ ]
```

Product using a single thread

Source Code:

```
ishanohri - ishanohri@pdc-lab: ~/Assignment_2 - ssh ishanohri@34.67.0.55 - 104x30

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

void main(){
    int i, product = 1, Num = 5, arr[5] = {10, 20, 30, 40, 50};

    for(i = 0 ; i < Num ; i++){
        product *= arr[i];
        printf("Product:%d\n", product);
    }
}

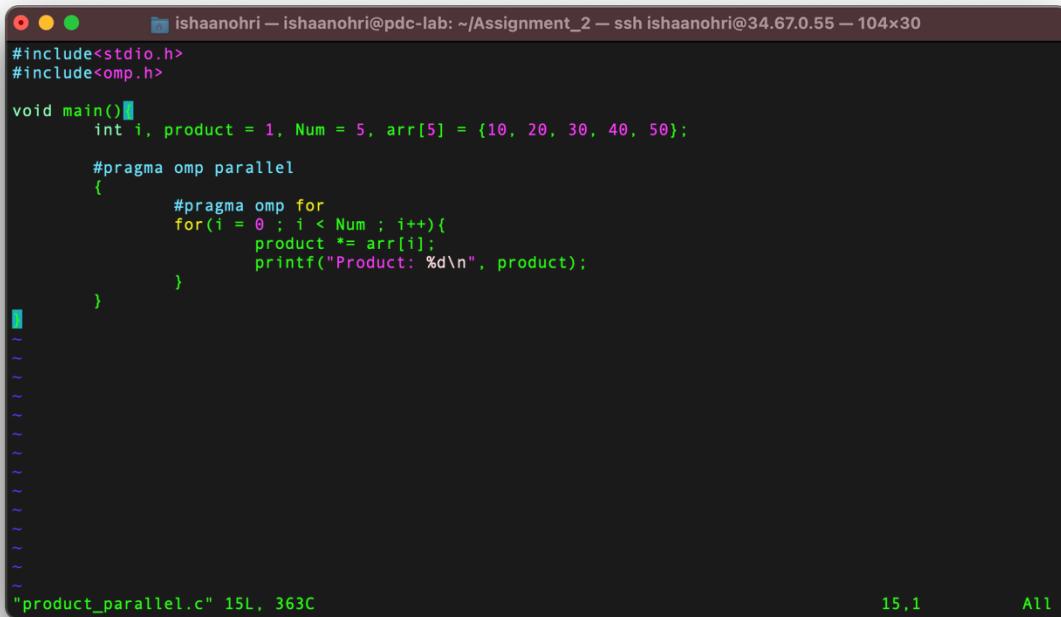
product.c 11L, 249C
```

Execution:

```
ishaanohri - ishaanohri@pdc-lab: ~/Assignment_2 - ssh ishaanohri@34.67.0.55 - 104x30
[ishaanohri@pdc-lab:~/Assignment_2$ gcc -fopenmp product.c -o h
[ishaanohri@pdc-lab:~/Assignment_2$ ./h
Product:10
Product:200
Product:6000
Product:240000
Product:12000000
ishaanohri@pdc-lab:~/Assignment_2$
```

Product using a multiple threads

Source Code:



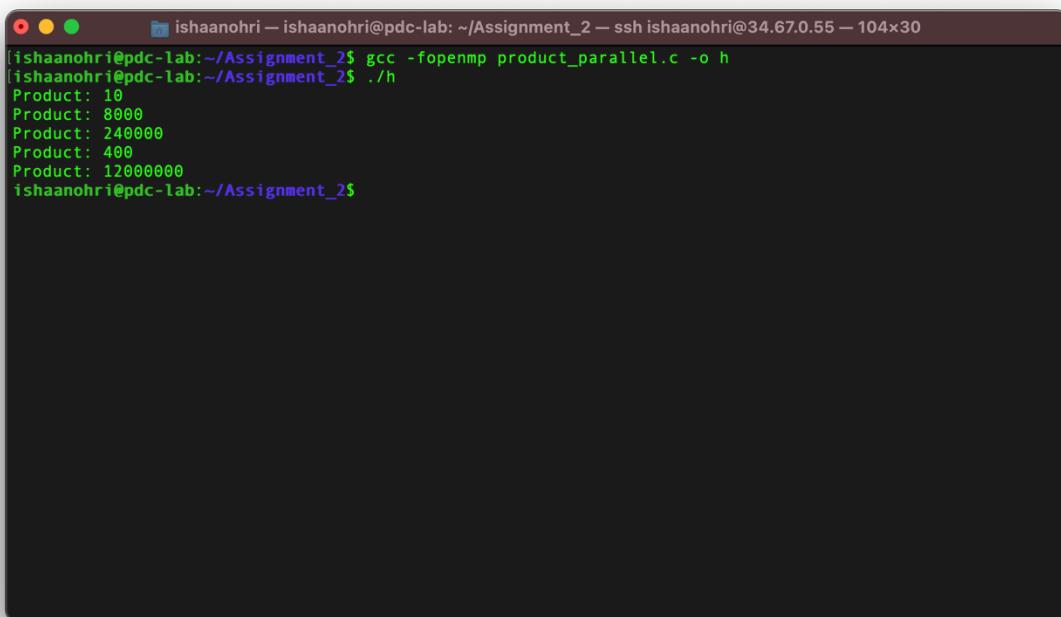
```
#include<stdio.h>
#include<omp.h>

void main()
{
    int i, product = 1, Num = 5, arr[5] = {10, 20, 30, 40, 50};

    #pragma omp parallel
    {
        #pragma omp for
        for(i = 0 ; i < Num ; i++){
            product *= arr[i];
            printf("Product: %d\n", product);
        }
    }
}

"product_parallel.c" 15L, 363C           15,1          All
```

Execution:



```
[isshaanohri@pdc-lab:~/Assignment_2$ gcc -fopenmp product_parallel.c -o h
[isshaanohri@pdc-lab:~/Assignment_2$ ./h
Product: 10
Product: 8000
Product: 240000
Product: 400
Product: 12000000
isshaanohri@pdc-lab:~/Assignment_2$ ]
```

Result:

From this experiment I understood the usage of *for* clause using parallel pragma and arrays by printing of array elements, sum of array elements, and product of array elements. The above experiment was conducted and all results along with the source code have been attached above in the document. The experiment was assisted by Dr Deepak. I thank sir for his assistance.