**Name: Varun Menon** **Course Code: CSE4001**

**Reg No: 19BCE1438** **Faculty: Dr. Harini S**

---

**Lab Experiment 1**

1. Write a parallel program to print “Hello World” with 1 thread

Code:

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

int main() {
    omp_set_num_threads(1);
    #pragma omp parallel {
        int thNum = omp_get_thread_num();
        printf("Hello World %d\n", thNum);
 }
}
```

Output:

```
C:\Users\menon\Desktop>gcc -fopenmp hello.cpp

C:\Users\menon\Desktop>a.exe
Hello World 0

C:\Users\menon\Desktop>
```

2. Write a parallel program to print “Hello World” with 2 Threads

Code:

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

int main()
{
    omp_set_num_threads(2);
    #pragma omp parallel
    {
        int thNum = omp_get_thread_num();
        printf("Hello World %d\n", thNum);
    }
}
```

Output:

```
C:\Users\menon\Desktop>gcc -fopenmp hello2.cpp

C:\Users\menon\Desktop>a.exe
Hello World 0
Hello World 1

C:\Users\menon\Desktop>
```

3. Write a parallel program to print “Hello World” with 4 Threads

Code:

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

int main()
{
    omp_set_num_threads(4);
    #pragma omp parallel
    {
        int thNum = omp_get_thread_num();
        printf("Hello World %d\n", thNum);
    }
}
```

Output:

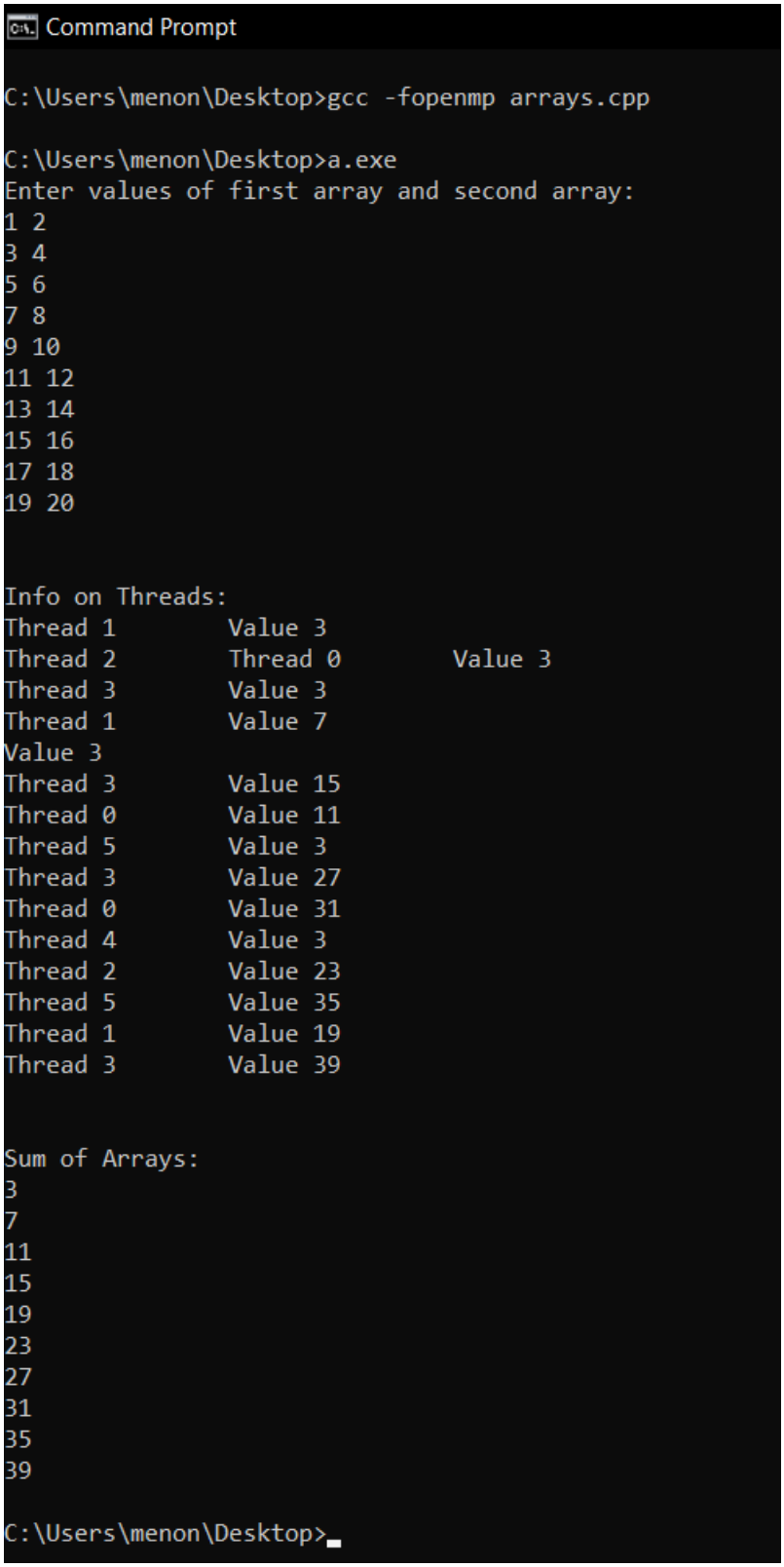
```
C:\Users\menon\Desktop>gcc -fopenmp hello4.cpp

C:\Users\menon\Desktop>a.exe
Hello World 1
Hello World 2
Hello World 0
Hello World 3

C:\Users\menon\Desktop>
```

4. Write a parallel program to add 2 arrays

Code:

```
#define _GNU_SOURCE
#include <stdio.h>
#include <pthread.h>
#include <stdlib.h>
#include <omp.h>
#include <sched.h>
int main()
{
    int a[10], b[10], c[10];
    int i;
    printf("Enter values of first array and second array:\n");
    for(i = 0; i < 10; i++)
    {
        scanf("%d %d", &a[i], &b[i]);
    }
    printf("\n\n");
    printf("Info on Threads:\n");
    #pragma omp parallel
    {
        for (i = 0; i < 10; i++)
        {
            c[i] = a[i] + b[i];
            printf("Thread %d\tValue %d\n", omp_get_thread_num(),
c[i]);
        }
    }
    printf("\n\n");
    printf("Sum of Arrays:\n");
    for(i = 0; i < 10; i++)
    {
        printf("%d\n", c[i]);
    }
    return 0;
}
```

Output:


Command Prompt
C:\Users\menon\Desktop>gcc -fopenmp arrays.cpp
C:\Users\menon\Desktop>a.exe
Enter values of first array and second array:
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
Info on Threads:
Thread 1 Value 3
Thread 2 Thread 0 Value 3
Thread 3 Value 3
Thread 1 Value 7
Value 3
Thread 3 Value 15
Thread 0 Value 11
Thread 5 Value 3
Thread 3 Value 27
Thread 0 Value 31
Thread 4 Value 3
Thread 2 Value 23
Thread 5 Value 35
Thread 1 Value 19
Thread 3 Value 39
Sum of Arrays:
3
7
11
15
19
23
27
31
35
39
C:\Users\menon\Desktop>