

**ARYAMAN MISHRA**

**19BCE1027**

**CSE4001 - L15+16**

## **Usage of firstprivate**

```
#include<stdio.h>

#include<omp.h>

int main(void)

{

int i=10;

#pragma omp parallel firstprivate(i)

{

i=10+omp_get_thread_num();

printf("thread %d:i=%d\n",omp_get_thread_num(),i);

}

return 0;

}
```

```
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$ gcc -o lab21 -fopenmp lab21.c
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$ ./lab21
thread 1:i=11
thread 0:i=10
thread 5:i=15
thread 4:i=14
thread 3:i=13
thread 2:i=12
```

## Adding two arrays

```
#include <omp.h>

#include <stdio.h>

int main()
{
    int a[10],b[10],i,sum[10];

    for (i=0; i<10; i++)
    {
        printf("Input in a[%d]: ",i);
        scanf("%d",&a[i]);
    }

    for (i=0; i<10; i++)
    {
        printf("Input in b[%d]: ",i);
        scanf("%d",&b[i]);
    }

    #pragma omp parallel for

    for (i=0; i < 10; i++)
    {

        sum[i] = a[i] + b[i];

        printf("CPU:%d\tThread:%d\tValue:%d\n",sched_getcpu(),omp_get_thread_num(),sum[i]);

    }
```

}

```
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$ gcc -o lab22 -fopenmp lab22.c
lab22.c: In function 'main':
lab22.c:21:40: warning: implicit declaration of function 'sched_getcpu' [-Wimplicit-function-declaration]
   21 | printf("CPU:%d\tThread:%d\tValue:%d\n", sched_getcpu(), omp_get_thread_num(), sum[i]);
      |                                     ^
      |                                     |
      |                                     ~~~~~
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$ ./lab22
Input in a[0]: 1
Input in a[1]: 2
Input in a[2]: 3
Input in a[3]: 4
Input in a[4]: 5
Input in a[5]: 6
Input in a[6]: 7
Input in a[7]: 8
Input in a[8]: 9
Input in a[9]: 10
Input in b[0]: 11
Input in b[1]: 12
Input in b[2]: 13
Input in b[3]: 14
Input in b[4]: 15
Input in b[5]: 16
Input in b[6]: 17
Input in b[7]: 18
Input in b[8]: 19
Input in b[9]: 20
CPU:0 Thread:1 Value:16
CPU:0 Thread:1 Value:18
CPU:0 Thread:0 Value:12
CPU:0 Thread:0 Value:14
CPU:0 Thread:5 Value:30
CPU:0 Thread:4 Value:28
CPU:0 Thread:3 Value:24
CPU:0 Thread:3 Value:26
CPU:0 Thread:2 Value:20
CPU:0 Thread:2 Value:22
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$
```

## Addition of factors of a number

```
#include<omp.h>

#include<stdio.h>

#include<stdlib.h>

int main()

{

int n,m,i,j;

printf("Enter number of factors.\n");

scanf("%d",&n);

printf("Enter a number.\n");

scanf("%d",&m);

int a[n];

for(i=0;i<n;i++)

{

a[i]=i+1;

}

int sum=0;

#pragma omp parallel for firstprivate(sum)

for(i=2;i<m+2;i++)

{

for(j=0;j<n;j++)

{

if(a[j]%i==0)
```

```

{
sum=sum+a[j];
}

}

printf("cpu: %d\tthread: %d\tnumber: %d\tsum:
%d\n",sched_getcpu(),omp_get_thread_num(),i,sum);

sum=0;

}

}

```

```

aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$ gcc -o lab23 -fopenmp lab23.c
lab23.c: In function 'main':
lab23.c:27:53: warning: implicit declaration of function 'sched_getcpu' [-Wimplicit-function-declaration]
   27 | printf("cpu: %d\tthread: %d\tnumber: %d\tsum: %d\n",sched_getcpu(),omp_get_thread_num(),i,sum);
       |
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$ ./lab23
Enter number of factors.
10
Enter a number.
3
cpu: 0 thread: 1      number: 3      sum: 18
cpu: 0 thread: 0      number: 2      sum: 30
cpu: 0 thread: 2      number: 4      sum: 12
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$ ./lab23
Enter number of factors.
10
Enter a number.
5
cpu: 0 thread: 1      number: 3      sum: 18
cpu: 0 thread: 0      number: 2      sum: 30
cpu: 0 thread: 4      number: 6      sum: 6
cpu: 0 thread: 3      number: 5      sum: 15
cpu: 0 thread: 2      number: 4      sum: 12
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$

```

## Addition of odd and even numbers

```

#include <omp.h>

#include <stdio.h>

int main()
{

int n,i,sum_even=0,sum_odd=0;

```

```

printf("Enter numbers.\n");

scanf("%d",&n);

int a[n];

printf("Enter numbers in array.First odd then even or enter in whichever order.\n");

    for (i=0; i<n; i++)

    {

        printf("Input in a[%d]: ",i);

        scanf("%d",&a[i]);

    }

#pragma omp parallel for

    for (i=0; i < n; i++)

    {

        if (a[i] % 2 == 0)

            sum_even += a[i];

        if(i==n-1)

            printf("CPU:%d\tThread:%d\tValue:%d\n",sched_getcpu(),omp_get_thread_num(),sum_even);

    }


#pragma omp parallel for

    for (i=0; i < n; i++)

    {

        if (a[i] % 2 != 0)

            sum_odd+= a[i];

        if(i==n-1)

            printf("CPU:%d\tThread:%d\tValue:%d\n",sched_getcpu(),omp_get_thread_num(),sum_odd);

    }

```

}

```
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$ gcc -o lab24 -fopenmp lab24.c
lab24.c: In function 'main':
lab24.c:21:41: warning: implicit declaration of function 'sched_getcpu' [-Wimplicit-function-declaration]
   21 |     printf("CPU:%d\tThread:%d\tValue:%d\n", sched_getcpu(), omp_get_thread_num(), sum_even);
       |                                         ^
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$ ./lab24
Enter numbers.
5
Enter numbers in array.First odd then even or enter in whichever order.
Input in a[0]: 1
Input in a[1]: 2
Input in a[2]: 3
Input in a[3]: 4
Input in a[4]: 5
CPU:0   Thread:4   Value:2
CPU:0   Thread:4   Value:6
aryaman@aryaman-VirtualBox:~/Desktop/19BCE1027PDC$
```

RESULT:ALL 4 PROGRAMS HAVE BEEN SUCCESFULLY COMPILED AND EXECUTED.