Computer Science & Engineering

CSE4001

Parallel and Distributed Computing

LAB ASSIGNMENT 5

Submitted to **Prof. DEEBAK B.D.**

TOPIC: PROBLEMS USING OPENMP

NAME: PUNIT MIDDHA

REG.NO: 19BCE2060

SLOT: L55+L56

DATE: 05/10/2021

QUESTION – I

Write a simple OpenMP program to demonstrate the use of pattern generation in schedule clause

```
* * * * * *

* * * * * *

* * * * * *

* * * * * *

* * * * * *
```

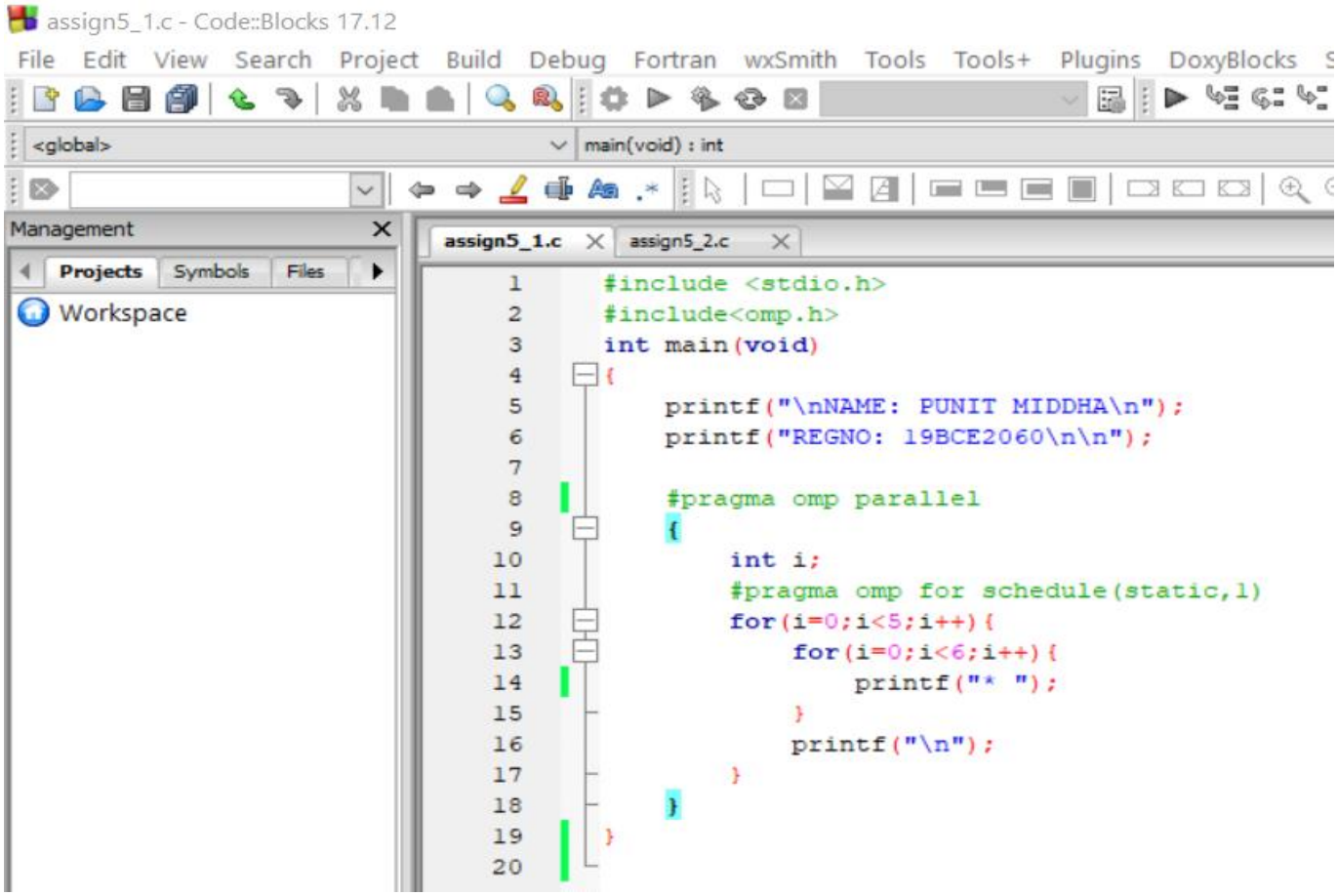
1. **Statically assign the loop iterations to threads**
2. **Dynamically assign one iteration to each thread**

SOURCE CODE:

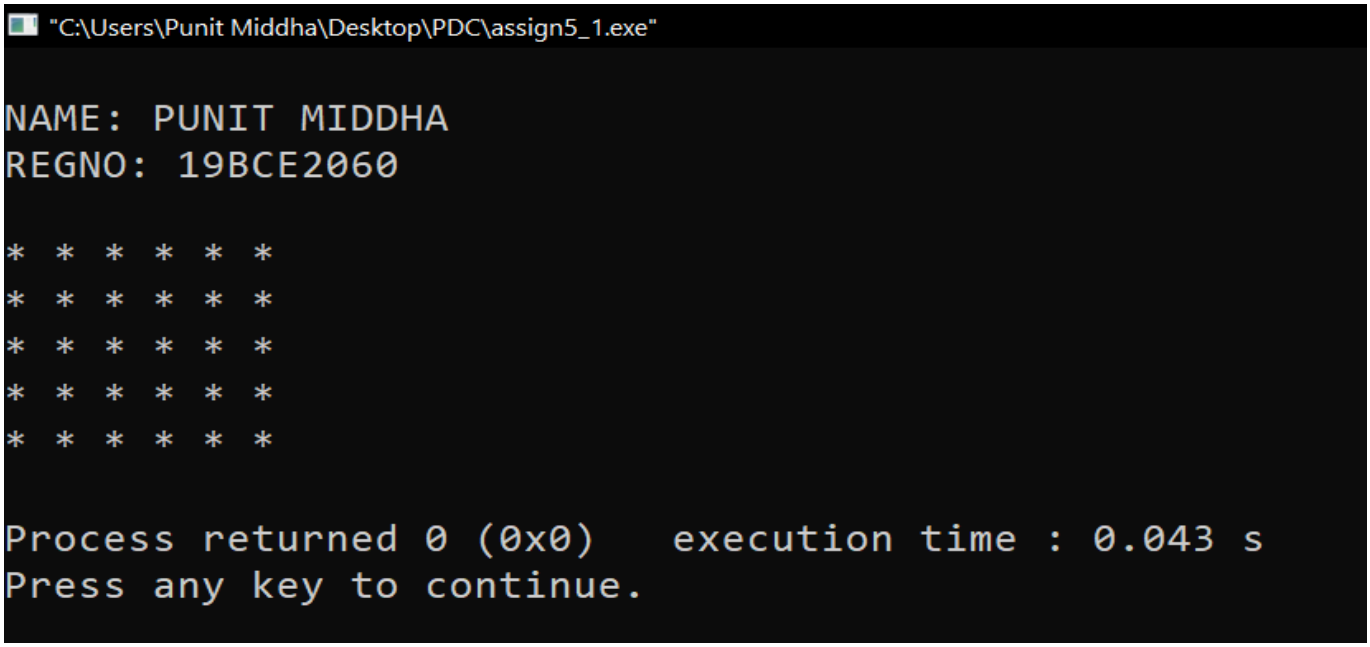
1. Statically assign the loop iterations to threads

```
#include <stdio.h>

#include<omp.h>

int main(void)

{

    printf("\nNAME: PUNIT MIDDHA\n");

    printf("REGNO: 19BCE2060\n\n");


    #pragma omp parallel

    {

        int i;

        #pragma omp for schedule(static,1)

        for(i=0;i<5;i++){

            for(i=0;i<6;i++){

                printf("* ");

            }
```

```
            printf("\n");
        }
    }
}
```



EXECUTION:



2. **Dynamically assign one iteration to each thread**

```
#include <stdio.h>
#include<omp.h>
int main(void)
{
    printf("\nNAME: PUNIT MIDDHA\n");
    printf("REGNO: 19BCE2060\n\n");

    #pragma parallel
    {
        int i;
        #pragma omp for schedule(dynamic,1)
        for(i=0;i<5;i++){
            for(i=0;i<6;i++){
                printf("*");
            }
            printf("\n");
        }
    }
}
```

assign5_2.c - Code::Blocks 17.12

File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Hel

<global> main(void) : int

Management

Projects Symbols Files

Workspace

assign5_1.c assign5_2.c

```c
#include <stdio.h>
#include<omp.h>
int main(void)
{
    printf("\nNAME: PUNIT MIDDHA\n");
    printf("REGNO: 19BCE2060\n\n");

    #pragma parallel
    {
        int i;
        #pragma omp for schedule(dynamic,1)
        for(i=0;i<5;i++){
            for(i=0;i<6;i++){
                printf("*");
            }
            printf("\n");
        }
    }
}
```

EXECUTION:

"C:\Users\Punit Middha\Desktop\PDC\assign5_2.exe"

```
NAME: PUNIT MIDDHA
REGNO: 19BCE2060

******
******
******
******
******

Process returned 0 (0x0)   execution time : 0.033 s
Press any key to continue.
```

REMARKS:

From this experiment, we tend to perceive the utilization of OpenMp threads for scheduling statically and dynamically. In static programming, every thread is allotted a piece of iterations in mounted fashion (round-robin). The iterations are divided among threads equally. In dynamic programming, every thread is initialized with a piece of threads; then, as every thread completes its iterations, it gets allotted an ensuing set of iterations. The parameter part defines the number of contiguous iterations that are allocated to a thread at a time. Also, execution time for static is comparatively more that the dynamic programming.