Experiment 3

**Aim: To implement programs based on scope of the variable and attribute clause.**

**Assignment 1: Write a Program in OpenMP to demonstrate the scope of private variable.**

**Solution:**

**#include <stdio.h>**

**#include <omp.h>**

**int main() {**

**int x = 10;**

**omp\_set\_num\_threads(4);**

**#pragma omp parallel private(x)**

**{**

**// Each thread has its own private copy of x**

**x += omp\_get\_thread\_num();**

**printf("Thread %d: x = %d\n", omp\_get\_thread\_num(), x);**

**}**

**// x outside the parallel region is unaffected**

**printf("Outside parallel region: x = %d\n", x);**

**return 0;**

**}**

***Compilation and execution:***

**gcc -fopenmp private\_example.c -o private\_example**

**./private\_example**

==================================================================================

**Assignment 2: Write a Program in OpenMP to demonstrate the scope of shared variable.**

**Solution:**

**#include <stdio.h>**

**#include <omp.h>**

**int main() {**

**int x = 10;**

**omp\_set\_num\_threads(4);**

**#pragma omp parallel shared(x)**

**{**

**// All threads share the same copy of x**

**x += omp\_get\_thread\_num();**

**printf("Thread %d: x = %d\n", omp\_get\_thread\_num(), x);**

**}**

**// x outside the parallel region is affected by the threads**

**printf("Outside parallel region: x = %d\n", x);**

**return 0;**

**}**

***Compilation and execution:***

**gcc -fopenmp shared\_example.c -o shared\_example**

**./shared\_example**

=====================================================================================

**Assignment 3: Write a Program in OpenMP to demonstrate the scope of firstprivate variable and its difference with private variable.**

**Solution:**

**#include <stdio.h>**

**#include <omp.h>**

**int main() {**

**int n = 5;**

**int x = 0; // Initial value for x**

**int j = 1; // Initial value for j**

**printf("Original x: %d, Original j: %d\n", x, j);**

**#pragma omp parallel for private(x) firstprivate(j)**

**for (int i = 0; i < n; i++) {**

**x = x + i; // Modification to x in each thread**

**j = j + i; // Modification to j in each thread**

**printf("Thread %d: Local x=%d, Local j=%d\n", omp\_get\_thread\_num(), x, j);**

**}**

**printf("Final x: %d, Final j: %d\n", x, j);**

**return 0;**

**}**

**}**

***Compilation and execution:***

**gcc -fopenmp firstprivate\_example.c -o firstprivate\_example**

**./firstprivate\_example**

**Assignment 4: Write a Program in OpenMP to demonstrate the scope of lastprivate variable and its difference with private variable.**

**Solution :**

**#include <stdio.h>**

**#include <omp.h>**

**int main() {**

**int n = 5;**

**int x = 0; // Initial value for x**

**int j = 1; // Initial value for j**

**printf("Original x: %d, Original j: %d\n", x, j);**

**#pragma omp parallel for private(x) lastprivate(j)**

**for (int i = 0; i < n; i++) {**

**x = x + i; // Modification to x in each thread**

**j = j + i; // Modification to j in each thread**

**printf("Thread %d: Local x=%d, Local j=%d\n", omp\_get\_thread\_num(), x, j);**

**}**

**printf("Final x: %d, Final j: %d\n", x, j);**

**return 0;**

**}**

***Compilation and execution:***

**gcc -fopenmp lastprivate\_example.c -o lastprivate\_example**

**./lastprivate\_example**