**Assignment 2**

**Parallel Bubble Sort**

#include<iostream>

#include<omp.h>

using namespace std;

void swap(int &a, int &b)

{

int temp;

temp=a;

a=b;

b=temp;

}

void bubble(int \*a, int n)

{

double start=omp\_get\_wtime();

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

{

#pragma omp parallel

for(int j=i+1;j<n;j++)

{

if(a[j]<a[i])

{

swap(a[j],a[i]);

}

}

}

double end=omp\_get\_wtime();

double time=end-start;

cout<<"\nTime taken => "<<time<<endl;

}

int main()

{

omp\_set\_num\_threads(4);

double start,end;

int \*a,n;

cout<<"\nEnter total number of elements => ";

cin>>n;

a=new int[n];

cout<<"\nEnter elements => ";

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

{

cin>>a[i];

}

bubble(a,n);

cout<<"\nSorted Array => ";

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

{

cout<<a[i]<<" ";

}

return 0;

}

**Output –**

Enter total number of elements => 5

Enter elements => 5 4 3 2 1

Time taken => 0.00200009

Sorted Array => 1 2 3 4 5