PRACTICAL NO. 2A

TITLE: Write a program to implement Parallel Bubble Sort using OpenMP

CODE:

```
#include<iostream>
#include<stdlib.h>
#include<omp.h>
using namespace std;
void bubble(int *, int);
void swap(int &, int &);
void bubble(int *a, int n)
{
    for(  int i = 0;  i < n;  i++ )
     {
          int first = i % 2;

          #pragma omp parallel for shared(a,first)
          for(  int j = first;  j < n-1;  j += 2  )
            {
                  if(  a[ j ]  >  a[ j+1 ]  )
                   {
                         swap(  a[ j ],  a[ j+1 ]  );
                   }
                    }
     }
}
void swap(int &a, int &b)
{

    int test;
    test=a;
    a=b;
    b=test;

}
int main()
{

    int *a,n;
    cout<<"\n enter total no of elements=>";
    cin>>n;
    a=new int[n];
    cout<<"\n enter elements=>";
    for(int i=0;i<n;i++)
```

```
    {
        cin>>a[i];
    }
    bubble(a,n);
    cout<<"\n sorted array is=>";
    for(int i=0;i<n;i++)
    {
        cout<<a[i]<<endl;
    }
return 0;
}
```

OUTPUT: