

```
#include<iostream>
#include <omp.h>
#include<sys/time.h>
using namespace std;
typedef long long ll;
const ll TIME = 1e7 ;
#define threads 2
int main(){

    // omp_set_num_threads(10);
    double start,stop;
    struct timeval tv;
    gettimeofday(&tv, NULL);
    start = (tv.tv_sec) * 1000 + (tv.tv_usec) / 1000;
    ll total = 0,i;
    double x,y;
    srand(time(NULL));
    #pragma omp parallel for reduction(+:total) private(x,y)
    num_threads(threads)
        for(i=0;i<TIME;i++){
            x = (double)rand()/(double)RAND_MAX;
            y = (double)rand()/(double)RAND_MAX;
            if( x*x + y*y <= 1.0)total++;
        }

    gettimeofday(&tv, NULL);
    stop = (tv.tv_sec) * 1000 + (tv.tv_usec) / 1000;

    cout<<(double)total / (double)TIME * (double)4<<endl;
    cout<<"total time is: "<<stop-start<<" ms"<<endl;

    return 0;
}
```