**ARYAMAN MISHRA**

**19BCE1027**

**LAB 3**

**Print largest cube value of element of array using lastprivate.**

#include<stdio.h>

#include<omp.h>

void main()

{

int a[]={1,2,3,4,5};

int cube=0; int i,n =5;

#pragma omp parallel for lastprivate(cube)

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

{

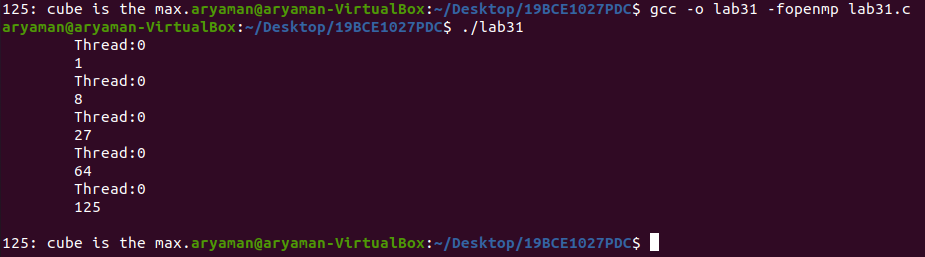
cube = a[i]\*a[i]\*a[i];

printf("\tThread:%d\n\t%d\n",omp\_get\_thread\_num(),cube);

}

printf("\n%d: cube is the max.",cube);

}



**Print sum of squares in array using Reduction.**

#include<stdio.h>

#include<omp.h>

void main(){

int a[]={1,2,3,4,5,6};

int b=0;

#pragma omp parallel for reduction(+:b)

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

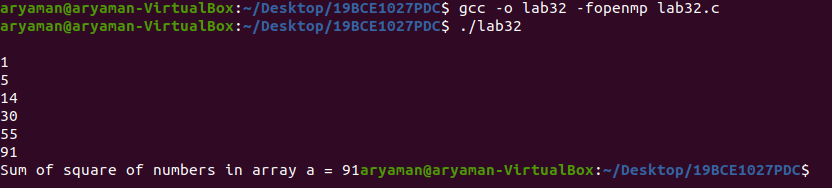
b=(a[i]\*a[i])+b;

printf("\n%d",b);

}

printf("\nSum of square of numbers in array a = %d",b);

}



**Perform addition, multiplication and subtraction of two array elements and store in different arrays using pragma omp sections.**

#include<omp.h>

#include<stdio.h>

#define N 3

void main()

{

int i;

float a[N],b[N],c[N],d[N],e[N];

for(i=0;i<N;i++)

{

a[i]=i\*1.5;

b[i]=i+22.35;

}

#pragma omp sections

{

#pragma omp section

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

c[i]=a[i]+b[i];

printf("\n addition of %0.2f + %0.2f = %0.2f ",a[i],b[i],c[i]);

}

#pragma omp section

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

d[i]=a[i]\*b[i];

printf("\n Multiplication of %0.2f and %0.2f = %0.2f",a[i],b[i],d[i]);

}

#pragma omp section

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

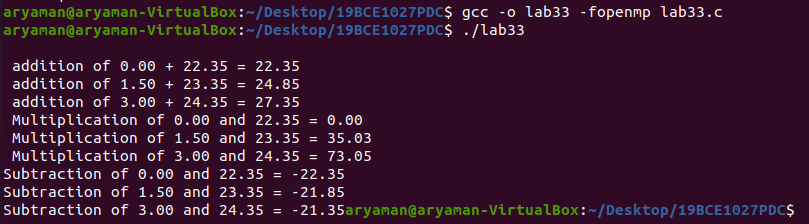
e[i]=a[i]-b[i];

printf("\nSubtraction of %0.2f and %0.2f = %0.2f",a[i],b[i],e[i]);

}

}

}



**Fibonacci series, second task - Factorial of n numbers, third task**

**-- prime number generation -- Parallelize using openmp.**

#include<omp.h>

#include<stdio.h>

#define N 6

void main(){

int a[] ={1,2,3,4,5,6};

int n1=0,n2=1,n3,i,f1,f0,f2; int fact=1;

#pragma omp sections

{

#pragma omp section

for(i=1;i<N;i++)

{

if(i==1){

printf("fibonaci series:\n%d %d",n1,n2);//printing 0 and 1

}

else{ n3=n1+n2;

printf(" %d",n3); n1=n2;

n2=n3;

}

}

#pragma omp section

for(int j=1;j<=N;j++){ fact=j\*fact;

printf("\n Factorial of first %d numbers: %d",N,fact);

}

#pragma omp section

for(i=2;i<N;i++){

int c=0;

for(int j=1;j<=i;j++){ if(i%j==0)

c++;

}

if(c==2)

printf("\nPrime numbers upto %d natural numbers: %d",N,i);

}

}

}

