#include<stdio.h>

#include<omp.h>

int main()

{

int tid;

int a=76,b=2,c,res;

double start,end;

omp\_set\_num\_threads(5);

start = omp\_get\_wtime();

#pragma omp parallel

{

tid = omp\_get\_thread\_num();

if(tid==0)

{

printf("Thread %d : Addition\n",tid);

c= a+b;

printf("Result : %d\n",c);

}

else if(tid==1)

{

printf("Thread %d : Subraction\n",tid);

c = a-b;

printf("Result : %d\n",c);

}

else if(tid==2)

{

printf("Thread %d : Multiplication\n",tid);

c= a\*b;

printf("Result : %d\n",c);

}

else if(tid==3)

{

printf("Thread %d : Division\n",tid);

c= a/b;

printf("Result : %d\n",c);

}

else

{

printf("Thread %d : Logical operation\n",tid);

res = ((a<=b)||(a!=b));

printf("Return value of above expression ((a<=b)||(a!=b)) is: %d\n",res);

res= ((a<b)&&(a==b));

printf("Return value of above expression ((a<b)&&(a==b)) is: %d\n",res);

res=!((a<b)&&(a==b));

printf("Return value of above expression !((a<b)&&(a==b)) is: %d\n",res);

}

end= omp\_get\_wtime();

}

printf("Task duration is from %f to %f\n",start,end);

return 0;

}