```c
#include<stdlib.h>
#include <stdio.h>
#include <omp.h>
#include <string.h>
#include<time.h>

int main(int argc,char *argv[])
{
  // Define Variables
clock_t begin, end;
double time_spent;

 double **x, *y,*A;
int n=atoi(argv[1]);
int i,j;
y=(double *)malloc(n*sizeof(double));
A=(double *)malloc(n*sizeof(double));
x=(double **)malloc(n*sizeof(double*));
for(i=0;i<n;i++)
x[i]=(double *)malloc(n*sizeof(double));
begin=clock();
#pragma omp parallel for private(i,j) shared(x) schedule(static)
for (i=0; i<n; i++)
        for (j=0; j<n; j++)
                x[i][j]= i;
#pragma omp parallel for shared(y,A) private(i,j) schedule (static)
for (i=0; i<n; i++)
 {
        y[i]= i;
        A[i] = 0;
  }
#pragma omp parallel for shared(x,y,A) private(i,j) schedule (static)
  for (i=0; i<n; i++)
  {

    for (j=0; j<n; j++)
    {
      A[i] += x[i][j] * y[j];
    }
  }
end=clock();
printf("Result vector:\n");
// Print A[n]
 for (i=0; i<n; i++)
    printf("%lf\n", A[i]);
time_spent = (double)(end - begin) / CLOCKS_PER_SEC;
printf("Time=%lf",time_spent);
```