#include<iostream>

#include <omp.h>

using namespace std;

const int n = 6;

const int m = 8;

int main()

{

int d[n][m], i, j, c\_size = 2;

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

for (j = 0; j < m; j++)

d[i][j] = rand() % 10;

int min = d[0][0], max = d[0][0];

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

{

for (j = 0; j < m; j++)

{

cout << d[i][j] << ' ';

}

cout << endl;

}

cout << endl << endl;

omp\_set\_num\_threads(4);

#pragma omp parallel for schedule(static)

for (int i = 0; i < n; i++)

{

#pragma omp critical

{

for (int j = 0; j < m; j++)

{

if (d[i][j] < min)

min = d[i][j];

if (d[i][j] > max)

max = d[i][j];

}

}

printf("All thread = %d. Thread num = %d.\n",omp\_get\_num\_threads(), omp\_get\_thread\_num());

}

printf("Max=%d\nMin=%d\n", max, min);

return 0;

}