1054. The Dominant Color (20)

时间限制

100 ms

内存限制

65536 kB

代码长度限制

16000 B

判题程序

Standard

作者

CHEN, Yue

Behind the scenes in the computer's memory, color is always talked about as a series of 24 bits of information for each pixel. In an image, the color with the largest proportional area is called the dominant color. A *strictly* dominant color takes more than half of the total area. Now given an image of resolution M by N (for example, 800x600), you are supposed to point out the strictly dominant color.

**Input Specification:**

Each input file contains one test case. For each case, the first line contains 2 positive numbers: M (<=800) and N (<=600) which are the resolutions of the image. Then N lines follow, each contains M digital colors in the range [0, 224). It is guaranteed that the strictly dominant color exists for each input image. All the numbers in a line are separated by a space.

**Output Specification:**

For each test case, simply print the dominant color in a line.

**Sample Input:**

5 3

0 0 255 16777215 24

24 24 0 0 24

24 0 24 24 24

**Sample Output:**

24

[提交代码](https://www.patest.cn/contests/pat-a-practise/1054)

水题，没什么好说的，但是有个挑战，能否在不使用STL的情况下完成要求呢。

#include<iostream>

#include<string>

#include<algorithm>

#include<queue>

#include<vector>

#include<sstream>

#include<stack>

#define MAX 100000000

using namespace std;

int a[MAX] = { 0 };

struct col{

int id;

int time;

};

vector < col >list;

int find(int num)

{

for (int i = 0; i < list.size(); i++)

{

if (list[i].id == num)

return i;

}

return -1;

}

void push(int num)

{

col temp;

temp.id = num;

temp.time = 1;

list.push\_back(temp);

}

int comp(col a, col b)

{

return a.time > b.time;

}

int main()

{

int m, n;

cin >> m>> n;

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

{

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

{

int temp;

scanf("%d",& temp);

int pos = find(temp);

if (pos == -1)

{

push(temp);

}

else

list[pos].time++;

}

}

sort(list.begin(),list.end(), comp);

cout << list[0].id;

}