**上帝造题五分钟**

问题描述

　　第一分钟，上帝说：要有题。于是就有了L，Y，M，C  
　　第二分钟，LYC说：要有向量。于是就有了长度为n写满随机整数的向量  
　　第三分钟，YUHCH说：要有查询。于是就有了Q个查询，查询向量的一段区间内元素的最小值  
　　第四分钟，MZC说：要有限。于是就有了数据范围  
　　第五分钟，CS说：要有做题的。说完众神一哄而散，留你来收拾此题

输入格式

　　第一行两个正整数n和Q，表示向量长度和查询个数  
　　接下来一行n个整数，依次对应向量中元素：a[0]，a[1]，…，a[n-1]  
　　接下来Q行，每行两个正整数lo，hi，表示查询区间[lo, hi]中的最小值，即min(a[lo],a[lo+1],…,a[hi])。

输出格式

　　共Q行，依次对应每个查询的结果，即向量在对应查询区间中的最小值。

样例输入

7 4  
1 -1 -4 8 1 2 -7  
0 0  
1 3  
4 5  
0 6

样例输出

1  
-4  
1  
-7

样例说明

　　第一个查询[0,0]表示求min{a[0]}=min{1}=1  
　　第二个查询[1,3]表示求min{a[1],a[2],a[3]}=min{-1,-4,8}=-4  
　　第三个查询[4,5]表示求min{a[4],a[5]}=min{1,2}=1  
　　第四个查询[0,6]表示查询整个向量，求min{a[0..6]}=min{1,-1,-4,8,1,2,-7}=-7

数据规模和约定

　　1<=n<=1984，1<=Q<=1988，向量中随机整数的绝对值不超过1,000

C++代码：

#include <stdio.h>

int random[1985], query[1989][2]; //存放向量以及查询的区间

int n,Q;//向量的长度，查询的个数

int result[1985][1985];

int main()

{

int i,j;

scanf("%d%d", &n, &Q);

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

{

scanf("%d", &random[i]);

}

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

{

scanf("%d%d", &query[i][0], &query[i][1]);

}

result[0][0] = random[0];

for(j = 1; j < n; j++)

{

if(random[j] < result[0][j - 1])

{

result[0][j] = random[j];

}

else

{

result[0][j] = result[0][j - 1];

}

result[j][0] = result[0][j];

}

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

{

result[i][i] = random[i];

for(j = i + 1; j < n && j > i; j++)

{

if(random[j] < result[i][j - 1])

{

result[i][j] = random[j];

}

else

{

result[i][j] = result[i][j - 1];

}

result[j][i] = result[i][j];

}

}

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

{

printf("%d\n", result[query[i][0]][query[i][1]]);

}

return 0;

}

C代码：

#include<stdio.h>

#include<stdlib.h>

int min(int a,int b,int c[]){

int i;

int mina = c[a];

for(i=a+1;i<=b;i++){

if(c[i]<mina){

mina = c[i];

}

}

return mina;

}

int main(){

int i,n,q,b,c,a[2000];

scanf("%d%d",&n,&q);

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

scanf("%d",&a[i]);

}

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

scanf("%d%d",&b,&c);

printf("%d\n",min(b,c,a));

}

system("pause");

return 0;

}

JAVA代码：

import java.io.\*;

class Main

{

public static void main(String[] args) throws Exception

{

BufferedReader bufr =

new BufferedReader(new InputStreamReader(System.in));

String s = bufr.readLine();

String[] str1 = s.split("\\s");

String st = bufr.readLine();

String[] str = st.split("\\s");

int[] nums = new int[Integer.parseInt(str1[0])];

int[] nums1 = new int[Integer.parseInt(str1[1])];

for(int i = 0; i < str.length; i++)

{

nums[i] = Integer.parseInt(str[i]);

}

for(int i = 0; i < Integer.parseInt(str1[1]); i++)

{

String si = bufr.readLine();

String[] stri = si.split("\\s");

int num = nums[Integer.parseInt(stri[0])];

for(int j = Integer.parseInt(stri[0])+1; j <= Integer.parseInt(stri[1]); j++)

{

if(num > nums[j])

num = nums[j];

}

nums1[i] = num;

}

for(int i = 0; i < nums1.length; i++)

{

System.out.println(nums1[i]);

}

}

}