

希望大家天天开心

——输的时候，呼吸都是错的

博客园

首页

新随笔

联系

订阅

管理

随笔 - 46 文章 - 0 评论 - 11

公告

AmazingCounters.com

昵称： 真是啰嗦

园龄： 1年1个月

粉丝： 8

关注： 0

+加关注

< 2019年9月 >

日	一	二	三	四	五	六
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5
6	7	8	9	10	11	12

搜索

找找看

谷歌搜索

常用链接

- 我的随笔
- 我的评论
- 我的参与
- 最新评论
- 我的标签

随笔分类

- ACM(26)
- Android(1)
- C++(1)
- CSS(4)
- Java Web(4)
- JavaScript(9)
- ST表(2)
- 模拟退火算法(3)
- 树状数组(3)

随笔档案

- 2019年8月(3)
- 2019年7月(9)
- 2019年6月(1)
- 2019年4月(1)

ST表的原理及其实现

ST表类似树状数组，线段树这两种算法，是一种用于解决RMQ(Range Minimum/Maximum Query,即区间最值查询)问题的离线与线段树相比，预处理复杂度同为 $O(n\log n)$,查询时间上，ST表为 $O(1)$,线段树为 $O(n\log n)$

st表的主体是一个二维数组 $st[i][j]$ ，表示需要查询的数组的从下标 i 到下标 $i+2^j-1$ 的最值，这里以最小值为例

预处理函数：

```
1 int a[1010]; //原始输入数组
2 int st[1010][20]; //st表
3
4 void init(int n)
5 {
6     for (int i = 0; i < n; i++)
7         st[i][0] = a[i];
8
9     for (int j = 1; (1 << j) <= n; j++)
10     {
11         for (int i = 0; i + (1 << j) - 1 < n; i++)
12             st[i][j] = min(st[i][j - 1], st[i + (1 << (j - 1))][j - 1]);
13     }
14 }
```

这里首先把从 $0\sim n-1$ 的 2^0 部分进行覆盖，再往下继承

继承这里也很好理解，我们以一个长度为5的数组 $[5,1,2,3,4]$ 为例

2^0 部分覆盖过去自然是5, 4, 3, 2, 1

2^1 部分的长度为4，从0一直到3，因为从下标为4开始后面只有他自己

$st[0][1]$ 是下标为 $0\sim 1$ 的最小值，自然也就是 $st[0][0]$ 和 $st[1][0]$ 的最值

以此往下类推我们可以得出结论：

$$st[i][j] = \min(st[i][j - 1], st[i + 2^{(j - 1)}][j - 1])$$

到这里初始化就完成了，注意下标不要越界，如果你对为什么这么处理有困惑的话，请继续看查询

查询函数这里不太好理解

初始化时，每一个状态对应的区间长度都为 2^j ，由于给出的查询区间长度不一定恰好为 2^j ，所以我们要引出一个定理： $2^{\log(a)} > a/2$ 。

https://blog.csdn.net/Hanks_o/article/details/77547380 这里有一段非常非常好理解的解释，这里超级感谢原作者，我本人不释，他的讲解是这样的：

这个很简单，因为 $\log(a)$ 表示小于等于 a 的2的最大几次方。

比如说 $\log(4)=2, \log(5)=2, \log(6)=2, \log(7)=2, \log(8)=3, \log(9)=3, \dots$

那么我们要查询 x 到 y 的最小值。

设 $len=y-x+1, t=\log(len)$

根据上面的定理： $2^t > len/2$

从位置上来说， $x+2^t$ 越过了 x 到 y 的中间！

因为位置过了一半

2019年3月(1)
2018年12月(3)
2018年11月(2)
2018年10月(5)
2018年9月(6)
2018年8月(5)
2018年7月(10)

文章分类

ACM

相册

QAQ(2)

最新评论

1. Re:ST表的原理及其实现
@ .texas雾草，所以都是线段树卡不过T3滚来学ST表的吗? ...
--Rorschach_XR
2. Re:ST表的原理及其实现
线段树查询时间应该不是nlogn是logn
--.texas
3. Re:ST表的原理及其实现
@ 暗い之殇log函数是以e为底求对数，除以log(2.0) 是根据换底公式转为 以2为底的log2()、...
--茄子Min
4. Re:ST表的原理及其实现
第23行为什么后面要除以log (2.0) 呀?
--暗い之殇
5. Re:ACM-ICPC 2018北京网络赛-A题 Saving Tang Monk II
@ stagelovepig不是啊~~~是如果曾经在这个地方，以同样的氧气瓶数量走过 就pass掉，因为重复了...
--真是啰嗦

阅读排行榜

1. ST表的原理及其实现(5435)
2. IntelliJ IDEA 配置JSP & Servlet 开发环境(3525)
3. 2018 ACM-ICPC 亚洲区域赛青岛现场赛 —— Problem F. Tournament(1024)
4. HDU6300-2018ACM暑假多校联合训练1003-Triangle Partition (247)
5. android 开发 简单的小计算器(235)

评论排行榜

1. ST表的原理及其实现(4)
2. HI~(3)
3. ACM-ICPC 2018北京网络赛-A题 Saving Tang Monk II-优先队列 (2)
4. ACM-ICPC 2018青岛网络赛-H题 Traveling on the Axis(2)

所以x到y的最小值可以表示为min(从x往后2^t的最小值，从y往前2^t的最小值)前面的状态表示为mn[t][x]
设后面（从y往前2^t的最小值）的初始位置是k，
那么k+2^t-1=y，所以k=y-2^t+1
所以后面的状态表示为mn[t][y-2^t+1]
所以x到y的最小值表示为min(mn[t][x],mn[t][y-2^t+1])，所以查询时间复杂度是O (1)

查询函数：

```
1 int search(int l, int r)
2 {
3     int k = (int)(log((double)(r - l + 1)) / log(2.0));
4     return min(st[l][k],st[r - (1 << k) + 1][k]);
5 }
```

示例程序：

```
1 #include <iostream>
2 #include <algorithm>
3
4 using namespace std;
5
6 int a[1010]; //原始输入数组
7 int st[1010][20]; //st表
8
9 void init(int n)
10 {
11     for (int i = 0; i < n; i++)
12         st[i][0] = a[i];
13
14     for (int i = 1; (1 << i) <= n; i++)
15     {
16         for (int j = 0; j + (1 << i) - 1 < n; j++)
17             st[j][i] = min(st[j][i - 1],st[j + (1 << i) - 1][i - 1]);
18     }
19 }
20
21 int search(int l, int r)
22 {
23     int k = (int)(log((double)(r - l + 1)) / log(2.0));
24     return min(st[l][k],st[r - (1 << k) + 1][k]);
25 }
26
27 int main()
28 {
29     int n,m;
30     while (cin >> n >> m)
31     {
32         for (int i = 0; i < n; i++)
33             cin >> a[i];
34
35         init(n);
36
37         while (m--)
38         {
39             int l, r;
40             cin >> l >> r;
41             cout << search(l,r) << endl;;
42         }
43     }
44     return 0;
45 }
```

这里有一个HDU3183的例题大家可以移步看一下具体的使用

<https://www.cnblogs.com/qq965921539/p/9609015.html>

分类: ACM, ST表

推荐排行榜

- 1. ST表的原理及其实现(14)
- 2. CSS的水平居中和垂直居中(1)
- 3. HDU6342-2018ACM暑假多校联合训练4-1011-Problem K. Expression in Memories(1)
- 4. HDU6299-2018ACM暑假多校联合训练1002-Balanced Sequence(1)
- 5. Java Web 学习与总结（三）会话跟踪(1)



真是啰嗦
关注 - 0
粉丝 - 8

+加关注

« 上一篇: HDU6333-2018ACM暑假多校联合训练1002-Harvest of Apples-莫队+费马小定理
» 下一篇: HDU3183 贪心/RMQ-ST表

posted @ 2018-09-08 13:29 真是啰嗦 阅读(5435) 评论(0)

评论列表

#1楼 2019-05-04 08:12 暗い之殇 第23行为什么后面要除以log (2.0) 呀?	支持
#2楼 2019-07-19 14:31 茄子Min @ 暗い之殇 log函数是以e为底求对数，除以log(2.0) 是根据换底公式转为 以2为底的log2()、	支持
#3楼 2019-08-05 15:30 .texas 线段树查询时间应该不是nlogn是logn	支持
#4楼 2019-08-05 16:15 Rorschach_XR @ .texas 雾草，所以都是线段树卡不过T3滚来学ST表的吗?	支持

刷新评论 刷新

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#) [网站首页](#)。

- 【推荐】超50万C++/C#源码: 大型实时仿真组态图形源码
- 【活动】阿里云910会员节多款云产品满减活动火热进行中
- 【推荐】新手上天翼云，数十款云产品、新一代主机0元体验
- 【推荐】零基础轻松玩转华为云产品，获赠礼包加返百元大礼
- 【推荐】华为IoT平台开发者套餐9.9元起，购买即送免费课程

相关博文:
· ST表

- [ST表 \(入门篇\)](#)
- [ST表学习](#)
- [ST 表学习](#)
- [洛谷P3865【模板】ST表ST表](#)

最新 IT 新闻:

- [搭载高通骁龙855移动平台的三星Galaxy A90 5G现已正式发布](#)
- [历经30多年的努力，科学家终于得到了另一种高温超导材料](#)
- [偿还30亿美元债务 退任CEO 贾跃亭宣布FF重大消息](#)
- [Apple Card被发现含有90%的钛和10%的铝](#)
- [我的胃是就是这么坏掉的](#)
- » [更多新闻...](#)