[**线段树**](http://www.cppblog.com/reiks/archive/2009/08/28/94620.html)

Posted on 2009-08-28 09:11 [reiks](http://www.cppblog.com/reiks/) 阅读(20) [评论(0)](http://www.cppblog.com/reiks/archive/2009/08/28/94620.html#Post)  [编辑](http://www.cppblog.com/reiks/admin/EditPosts.aspx?postid=94620) [收藏](http://www.cppblog.com/reiks/AddToFavorite.aspx?id=94620) [引用](http://www.cppblog.com/reiks/services/trackbacks/94620.aspx) 所属分类: [算法与数据结构](http://www.cppblog.com/reiks/category/11589.html) http://www.cppblog.com/reiks/aggbug/94620.html?webview=1

http://www.cppblog.com/Images/OutliningIndicators/None.gif#include <iostream>     
http://www.cppblog.com/Images/OutliningIndicators/None.gifusing namespace std;     
http://www.cppblog.com/Images/OutliningIndicators/None.gif#define N 100     
http://www.cppblog.com/Images/OutliningIndicators/None.gifstruct TNode     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    int left, right;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    int n;     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif};     
http://www.cppblog.com/Images/OutliningIndicators/None.gifTNode T[N\*2+1];     
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid build(int s, int t, int step)     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    T[step].left = s;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    T[step].right = t;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    T[step].n = 0;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    if(s == t)     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        return;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    int mid = (s + t) / 2;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    build(s, mid, step\*2);     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    build(mid+1, t, step\*2+1);     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}     
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid insert(int s, int t, int step)     
http://www.cppblog.com/Images/OutliningIndicators/None.gif// insert [s, t] in the tree     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    if(s == T[step].left && t == T[step].right)     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif    http://www.cppblog.com/Images/dot.gif{     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        T[step].n++;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        return;     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    int mid = (T[step].left + T[step].right) / 2;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    if(t <= mid)     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        insert(s, t, step\*2);     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    else if(s > mid)     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        insert(s, t, step\*2+1);     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    else    
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif    http://www.cppblog.com/Images/dot.gif{     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        insert(s,mid, step\*2);     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        insert(mid+1, t, step\*2+1);     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}     
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid calculate(int s, int t, int step, int target, int& count)     
http://www.cppblog.com/Images/OutliningIndicators/None.gif// caculate target in the tree     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    count += T[step].n;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    if(s == t)     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        return;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    int mid = (s + t) / 2;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    if(target <= mid)     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        calculate(s, mid, step\*2, target, count);     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    else    
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        calculate(mid+1, t, step\*2+1, target, count);     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}     
http://www.cppblog.com/Images/OutliningIndicators/None.gifint main()     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    build(0, 7, 1);     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    insert(2, 5, 1);     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    insert(4, 6, 1);     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    insert(0, 7, 1);     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    int count = 0;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    calculate(0, 7, 1, 4, count);     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    cout  << count << endl;     
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    return 0;     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}    
http://www.cppblog.com/Images/OutliningIndicators/None.gif