**[终于把RBTree敲出来了， 累.](http://www.cppblog.com/qywyh/archive/2006/09/15/12489.html)**

基本测试没问题， 有bug请指出:)

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  
http://www.cppblog.com/Images/OutliningIndicators/None.gifconst   int  MAXSIZE  =   100 ;  
http://www.cppblog.com/Images/OutliningIndicators/None.gifconst   int  R  =   0 ;  
http://www.cppblog.com/Images/OutliningIndicators/None.gifconst   int  B     =   1 ;  
http://www.cppblog.com/Images/OutliningIndicators/None.gifconst   int  NIL  =   0 ;  
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifclass  RBTree  
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.gifpublic :  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    RBTree();  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  Root();  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  Size();  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  Key( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  Pointer( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  Search( int ,  int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  Minimum( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  Maximum( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  successor( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  predecessor( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     void  LeftRotate( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     void  RightRotate( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     void  Insert( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     void  InsertByPointer( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     void  InsertFixup( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     void  Delete( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     void  DeleteByPointer( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     void  DeleteFixup( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     void  Travel( int );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gifprivate :  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  key[MAXSIZE];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  left[MAXSIZE];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  right[MAXSIZE];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  color[MAXSIZE];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  p[MAXSIZE];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  stack[MAXSIZE];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  root;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  top;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  size;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif} ;  
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifRBTree::RBTree()  
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    root  =  NIL;  // 默认空树root为NIL;   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    top  =   0 ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    size  =   0 ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    memset(right, NIL,  sizeof (right));  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    memset(left, NIL,  sizeof (left));  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    memset(p, NIL,  sizeof (p));  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    color[NIL]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  RBTree::Root()  
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     return  root;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  RBTree::Size()  
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     return  size;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  RBTree::Key( int  x)  
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     return  key[x];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  RBTree::Pointer( int  k)  
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     return  Search(Root(), k);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  RBTree::Minimum( int  x)  
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     while  (left[x]  !=  NIL)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        x  =  left[x];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     return  x;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  RBTree::Maximum( int  x)  
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     while  (right[x]  !=  NIL)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        x  =  right[x];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     return  x;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  RBTree::Search( int  x,  int  k)  
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     while  (x  !=  NIL  &&  k  !=  key[x])  
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         if  (k  <  key[x])  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            x  =  left[x];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         else   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            x  =  right[x];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     return  x;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  RBTree::successor( int  x)  
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  (right[x]  !=  NIL)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         return  Minimum(right[x]);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  y  =  p[x];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     while  (y  !=  NIL  &&  x  ==  right[y])  
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        x  =  y;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        y  =  p[y];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     return  y;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  RBTree::predecessor( int  x)  
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  (left[x]  !=  NIL)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         return  Maximum(left[x]);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  y  =  p[x];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     while  (y  !=  NIL  &&  x  ==  left[y])  
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        x  =  y;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        y  =  p[y];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     return  y;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  RBTree::LeftRotate( int  x)  
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  y  =  right[x];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    right[x]  =  left[y];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    p[left[y]]  =  x;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    p[y]  =  p[x];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     if  (p[x]  ==  NIL)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        root  =  y;  
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         if  (x  ==  left[p[x]])  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            left[p[x]]  =  y;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         else   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            right[p[x]]  =  y;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    left[y]  =  x;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    p[x]  =  y;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  RBTree::RightRotate( int  x)  
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  y  =  left[x];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    left[x]  =  right[y];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    p[right[y]]  =  x;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    p[y]  =  p[x];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     if  (p[x]  ==  NIL)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        root  =  y;  
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         if  (x  ==  left[p[x]])  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            left[p[x]]  =  y;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         else   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            right[p[x]]  =  y;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    right[y]  =  x;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    p[x]  =  y;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  RBTree::Insert( int  k)  
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  z;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     if  (top  >   0 )  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        z  =  stack[ -- top];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     else   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        z  =   ++ size;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    key[z]  =  k;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    InsertByPointer(z);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  RBTree::Delete( int  k)  
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  z  =  Search(Root(), k);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     if  (z  !=  NIL)  
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        stack[top ++ ]  =  z;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        size -- ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        DeleteByPointer(z);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  RBTree::InsertByPointer( int  z)  
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  y  =  NIL;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  x  =  root;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     while  (x  !=  NIL)  
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        y  =  x;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         if  (key[z]  <  key[x])  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            x  =  left[x];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         else   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            x  =  right[x];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    p[z]  =  y;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     if  (y  ==  NIL)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        root  =  z;  
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         if  (key[z]  <  key[y])  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            left[y]  =  z;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         else   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            right[y]  =  z;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    left[z]  =  NIL;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    right[z]  =  NIL;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    color[z]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    InsertFixup(z);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  RBTree::InsertFixup( int  z)  
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  y;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     while  (color[p[z]]  ==  R)  
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         if  (p[z]  ==  left[p[p[z]]])  
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            y  =  right[p[p[z]]];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif             if  (color[y]  ==  R)  
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                color[p[z]]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[y]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[p[p[z]]]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                z  =  p[p[z]];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
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                 if  (z  ==  right[p[z]])  
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                    z  =  p[z];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    LeftRotate(z);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif                }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[p[z]]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[p[p[z]]]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                RightRotate(p[p[z]]);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }   
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            y  =  left[p[p[z]]];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif             if  (color[y]  ==  R)  
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                color[p[z]]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[y]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[p[p[z]]]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                z  =  p[p[z]];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
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                 if  (z  ==  left[p[z]])  
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                    z  =  p[z];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    RightRotate(z);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif                }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[p[z]]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[p[p[z]]]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                LeftRotate(p[p[z]]);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    color[root]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  RBTree::DeleteByPointer( int  z)  
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  x, y;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     if  (left[z]  ==  NIL  ||  right[z]  ==  NIL)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        y  =  z;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     else   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        y  =  successor(z);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     if  (left[y]  !=  NIL)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        x  =  left[y];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     else   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        x  =  right[y];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    p[x]  =  p[y];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     if  (p[y]  ==  NIL)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        root  =  x;  
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         if  (y  ==  left[p[y]])  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            left[p[y]]  =  x;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         else   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            right[p[y]]  =  x;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     if  (y  !=  z)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        key[z]  =  key[y];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     if  (color[y]  ==  B)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        DeleteFixup(x);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  RBTree::DeleteFixup( int  x)  
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  y, w;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     while  (x  !=  root  &&  color[x]  ==  B)  
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         if  (x  ==  left[p[x]])  
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            w  =  right[p[x]];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif             if  (color[w]  =  R)  
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                color[w]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[p[x]]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                LeftRotate(p[x]);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                w  =  right[p[x]];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif             if  (color[left[w]]  ==  B  &&  color[right[w]]  ==  B)  
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                color[w]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                x  =  p[x];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
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                 if  (color[right[w]]  ==  B)  
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                    color[left[w]]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    color[w]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    RightRotate(w);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    w  =  right[p[x]];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif                }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[w]  =  color[p[x]];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[p[x]]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[right[w]]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                LeftRotate(p[x]);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                x  =  root;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }   
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            w  =  left[p[x]];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif             if  (color[w]  =  R)  
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                color[w]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[p[x]]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                RightRotate(p[x]);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                w  =  left[p[x]];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif             if  (color[right[w]]  ==  B  &&  color[left[w]]  ==  B)  
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                color[w]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                x  =  p[x];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
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                 if  (color[left[w]]  ==  B)  
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                    color[right[w]]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    color[w]  =  R;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    LeftRotate(w);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    w  =  left[p[x]];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif                }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[w]  =  color[p[x]];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[p[x]]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                color[left[w]]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                RightRotate(p[x]);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                x  =  root;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    color[x]  =  B;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  RBTree::Travel( int  x)  
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  (x  !=  NIL)  
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        cout  <<   ' ( ' ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        Travel(left[x]);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        cout  <<   '   '   <<  key[x]  <<   '   ' ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        Travel(right[x]);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        cout  <<   ' ) ' ;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.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    RBTree T;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     for  ( int  i = 1 ; i <= 10 ; i ++ )  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        T.Insert(i \* 10 );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     for  ( int  i = 1 ; i <= 10 ; i ++ )  
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.Delete(i \* 10 );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        T.Travel(T.Root());  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        cout  <<  endl;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     return   0 ;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}