struct RB\_tree

{

int data;

int colour;

RB\_tree \*left;

RB\_tree \*right;

RB\_tree \*father;

};

void right\_rotate(RB\_tree \*&v,RB\_tree \*&tree\_root)

{

bool check=false;

RB\_tree \*l=v->left;

RB\_tree \*r=v->right;

RB\_tree \*temp=l->right;

if(v==tree\_root) check=true;

l->right=v;

v->left=temp;

if(check) tree\_root=l;

}

void left\_rotate(RB\_tree \*&v)

{

RB\_tree \*l=v->left;

RB\_tree \*r=v->right;

RB\_tree \*temp=r->left;

if(r!=NULL) r->left=v;

v->right=temp;

}

void recolour(RB\_tree \*&v)

{

RB\_tree \*dad=v->father;

RB\_tree \*grandpa=dad->father;

RB\_tree \*uncle=grandpa->left;

dad->colour=0;

grandpa->colour=1;

uncle->colour=0;

if(grandpa->father->colour) recolour(grandpa);

}

void add\_RB\_tree(RB\_tree \*&tree\_root,int data)

{

if(tree\_root==NULL)

{

RB\_tree \*temp=new RB\_tree;

temp->data=data;

temp->left=NULL;

temp->right=NULL;

temp->colour=0;

tree\_root=temp;

}

else

{

RB\_tree \*dad=tree\_root;

bool check=true;

while(check)

{

if(data>dad->data)

{

if(dad->right!=NULL) dad=dad->right;

else

{

RB\_tree \*temp=new RB\_tree;

temp->data=data;

temp->left=NULL;

temp->right=NULL;

temp->colour=1;

temp->father=dad;

dad->right=temp;

RB\_tree \*grandpa=dad->father;

if(dad->colour==1)

{

if(grandpa->right==dad)

{

if(grandpa->left->colour==1) recolour(temp);

else

{

dad->colour=0;

grandpa->colour=1;

left\_rotate(temp);

}

}

else

{

swap(temp->data,dad->data);

dad->left=temp;

dad->right=NULL;

dad->colour=0;

grandpa->colour=1;

left\_rotate(temp);

}

}

check=false;

}

}

else if(data<dad->data)

{

if(dad->left!=NULL) dad=dad->left;

else

{

RB\_tree \*temp=new RB\_tree;

temp->data=data;

temp->left=NULL;

temp->right=NULL;

temp->colour=1;

temp->father=dad;

dad->left=temp;

RB\_tree \*grandpa=dad->father;

if(dad->colour==1)

{

if(grandpa->left==dad)

{

if(grandpa->right->colour==1) recolour(temp);

else

{

dad->colour=0;

grandpa->colour=1;

right\_rotate(temp,tree\_root);

}

}

else

{

swap(temp->data,dad->data);

dad->right=temp;

dad->left=NULL;

dad->colour=0;

grandpa->colour=1;

right\_rotate(temp,tree\_root);

}

}

check=false;

}

}

else check=false;

}

}

}