**SP1使用手册**

1. 主要功能：

T.Insert(x,y)

在数列x位置之后，顺序插入y个数，过程中已包含读入

T.Delete(x,y)

在数列x位置之后，顺序删除y个数

T.Change\_Same(x,y,z)

修改数列中由位置x开头的连续y个数，全部设置为z

T.Reverse(x,y)

数列中由位置x开头的连续y个数，翻转

T.Get\_Sum(x,y)

返回数列中由位置x开头的连续y个数的和

T.Get\_MaxSum(x,y)

返回数列中由位置x开头的连续y个数，所决定的区间里的，连续子段的最大和

T.Print(T.root)

遍历打印以x为根的子树，可以用来调试

T.erase(r)

删除以r为根的子树

Get\_kth(int r,int k)

返回以r为根的子树中的序号为k的元素，并没有考虑相同元素的并列问题；注意：该返回值，不一定代表区间k排名的值，需要格外注意

Splay(int r,int g)

伸展r到g的孩子

Rotate(int x,int kind)

节点x旋转，kind表示类型，0左1右

Init()

初始化，已包含读入：n个数，而n本身需要调用前已经读入，否则无法正常工作；会自动建树T，结束前会同步更新各数据域

Build(int &x,int l,int r,int f)

建树过程，x为根，l,r为数组a[]的下标范围,f为父节点编号（索引）；注意：a[]不在封装的SPT结构体中定义，必须之前预定义，并且调用前需要将数据读入a[]，过程中不包含读入

push\_down(int r)

信息标记下放

push\_up(int r)

信息更新上传

Update\_Same(int r,int v)

区间置值节点数据更新

Update\_Rev(int r)

区间翻转节点数据更新

NewNode(int &r,int f,int k)

新建一个节点，r为根节点，f为父节点编号，k为值；该过程将会优先从内存池内回收利用空间，确定r值，空间不够则会自动扩展

Travel(int x)

遍历到x

debug()

完整调试过程

1. 用户直接接触的接口

1）T.Insert(x,y)

2）T.Delete(x,y)

3）T.Change\_Same(x,y,z)

4）T.Reverse(x,y)

5）T.Get\_Sum(x,y)

6）T.Get\_MaxSum(x,y)

7）T.Print(T.root)

8）Get\_kth(int r,int k)

9）Init()

1. 可能还会扩展的功能更新
2. 返回打印一个区间所有元素的接口
3. 求区间最值
4. 区间排序