小明最近宅在家里无聊，于是他发明了一种有趣的游戏，游戏道具是N张叠在一起的卡片，每张卡片上都有一个数字，数字的范围是0~9，游戏规则如下：   
　　首先取最上方的卡片放到桌子上，然后每次取最上方的卡片，放到桌子上已有卡片序列的最右边或者最左边。当N张卡片全部都放到桌子上后，桌子上的N张卡片构成了一个数。这个数不能有前导0，也就是说最左边的卡片上的数字不能是0。游戏的目标是使这个数最小。   
　　现在你的任务是帮小明写段程序，求出这个最小数。

**Input**

第一行是一个数T，表示有T组测试数据；   
然后下面有T行， 每行是一个只含有0~9的字符串，表示N张叠在一起的卡片，最左边的数字表示最上方的卡片。   
  
[Technical Specification]   
T<=1000   
1 <= N <= 100

**Output**

对于每组测试数据，请在一行内输出能得到的最小数。

**Sample Input**

3

565

9876543210

9876105432

**Sample Output**

556

1234567890

1678905432

就是模拟，但有一点

注意的一点是0不能放在最前面但却不一定不能放在当前决策状态的最前面

思路：先找字符串中的除0外的最小的数，记住坐标j。

然后for处理每一个数，如果

i==j，放到第一个

i>j，加到右边

i<j，比第一个数小就加前面，比第一个数大就加到后面

不能x=s[0]

I从1-l

万一第一位是0呢

#include<bits/stdc++.h>

typedef long long ll;

using namespace std;

#define eps 1e-9

int main()

{

//freopen("input.txt","r",stdin);

int T;

string s;

cin>>T;

while(T--)

{

cin>>s;

int l=s.length();

char x='9';

int p(0);

for(int i=0;i<l;i++)

if(s[i]!='0' && s[i]<=x)

{

x=s[i];

p=i;

}

string t="";

for(int i=0;i<l;i++)

{

if(i==p)

t=s[i]+t;

if(i>p)

t+=s[i];

if(i<p)

{

if(s[i]<=t[0])

t=s[i]+t;

else t+=s[i];

}

}

cout<<t<<"\n";

}

return 0;

}