



#include<bits/stdc++.h>

#define ll long long

#define inf 1e18

#define mn 1000005

using namespace std;

ll cnt=0,ans=0,n,m,bj[mn],in[mn];

struct w{ll ch[2],f,num;

void st(){memset(ch,0,sizeof(ch));f=0;num=0;}

}a[1000005];

char s[1000005];

queue<ll>q;

void ins(char \*s)

{ll len=strlen(s),x=0,i,j;

for(i=0;i<len;i++)

{

j=s[i]-'0';

if(a[x].ch[j]==0)a[x].ch[j]=++cnt;

x=a[x].ch[j];

}

a[x].num=1;

}

void bui()

{ll i,x;

for(i=0;i<2;i++)

if(a[0].ch[i])

{

a[a[0].ch[i]].f=0;

q.push(a[0].ch[i]);

}

while(q.empty()==0)

{

x=q.front();

q.pop();

for(i=0;i<2;i++)

if(a[x].ch[i])

{

a[a[x].ch[i]].f=a[a[x].f].ch[i];

q.push(a[x].ch[i]);

if(a[a[a[x].f].ch[i]].num)//这两行为重点，就是如果一个节点的fail节点为被标记过，

a[a[x].ch[i]].num=1;// 既然自己后缀都行不通，自己也危险，这两行代码在ac自动机中很常见

}

else a[x].ch[i]=a[a[x].f].ch[i];

}

}

void dfs(ll x)// 通过搜索寻找有没有环

{ll i;

in[x]=1;// 作下路径标记

for(i=0;i<=1;i++)

if(in[a[x].ch[i]]) // 根据路径标记判断是否拥有环

{

printf("TAK");

exit(0); // 找到了环，输出并退出程序

}

else if(a[a[x].ch[i]].num==0&&bj[a[x].ch[i]]==0)

{

//只有下一位不为危险结点并且有可能成环，才递归搜索

bj[a[x].ch[i]]=1;//下一个结点已经被搜索过了

dfs(a[x].ch[i]);

}

in[x]=0;//抹除路径标记

}

int main()

{

ll t,i,j,k,z,y,x;

char ch;

cin>>n;

for(i=1;i<=n;i++)

{

scanf("%s",s);

ins(s);

}

bui();

dfs(0);

printf("NIE");

return 0;

}