

STL算法

二分查找

STL容器

测速

bitset

map

首尾访问

erase fast lower find

string

字符串hash

找最小字符串O(n)

(1).字符串找到特定子串修改成别的字符串

字典树

后缀SA

set

erase find *lower sort

iterator

vector

insert in mid

unique sort+去重

find maxx

STL算法

二分查找

```
binary_search(a.begin(), a.end(), x)    return true or false
```

STL容器

测速

```
#include <stdio.h>
#include <chrono>

int main () {
    double sum = 0;
    double add = 1;

    // Start measuring time
    auto begin = std::chrono::high_resolution_clock::now();

    int iterations = 1000*1000*200;
    for (int i=0; i<iterations; i++) {
        sum += add;
    }
```

```

        add /= 2.0;
    }

    // Stop measuring time and calculate the elapsed time
    auto end = std::chrono::high_resolution_clock::now();
    auto elapsed = std::chrono::duration_cast<std::chrono::nanoseconds>(end -
begin);

    printf("Result: %.20f\n", sum);

    printf("Time measured: %.3f seconds.\n", elapsed.count() * 1e-9);

    return 0;
}

```

bitset

```

#include <bits/stdc++.h>
using namespace std;
int n,m;
int ans=0;
int a[10000];
int count_one(int x)
{
    int cnt=0;
    for(int i=0;i<n;i++)
    {
        if(x&(1<<i))cnt++;
    }
    return cnt;
}
int main()
{
    cin>>n>>m;
    for(int i=0;i<n;i++)
    {
        cin>>a[i];
    }
    for(int i=0;i<=(1<<n)-1;i++)
    {
        if(count_one(i)==(n-m))
        {
            bitset<2023>b;
            b[0]=1;
            for(int j=0;j<n;j++)
            {
                if(i&(1<<j))
                    b=b|b<<a[j];
            }
        }
    }
}

```

```

        }
        int temp=b.count();
        ans=max(ans,temp);
    }
}
cout<<ans-1;
return 0;
}

```

```

i64 x;
cin >> x;

bitset<63> b(x);

```

map

首尾访问

```

#include <bits/stdc++.h>
using namespace std;
#define ll long long
const int inf=0x3f3f3f3f;
const int N=1e5+1;
string s[N];
int a[N];
int main()
{
    map<int, int>mp;
    int n;
    cin>>n;
    for(int i=1;i<=n;i++)
    {
        int x;
        cin>>x;
        mp[x]++;
    }
    auto i=mp.begin();
    cout<<i->first<<" "<<i->second<<endl;
    auto it=mp.rbegin();
    cout<<it->first<<" "<<it->second;
    return 0;
}

```

erase fast lower find

```
#include <bits/stdc++.h>
#define ll long long
ll sum;
int main()
{
    int n,m;
    std::cin>>n>>m;
    std::map<int, int>mp;
    for(int i=1;i<=n;i++)
    {
        int x;
        std::cin>>x;
        mp[i]=x;
        sum=sum+x;
    }
    mp[n+1]=1;
    mp[0]=1;
    while(m--)
    {
        int chose;
        std::cin>>chose;
        if(chose==2)
        {std::cout<<sum<<std::endl;}
        else
        {
            int l,r,k;
            std::cin>>l>>r>>k;
            for(auto it=mp.lower_bound(l);it->first<=r;it++)
            {
                for(int i=1;i<=k;i++)
                {
                    int x=round(sqrt(it->second)*10);
                    sum-=it->second;
                    sum+=x;
                    if(x==it->second)
                    {
                        it=mp.erase(it);
                        it--;
                        break;
                    }
                    it->second=x;
                }
            }
        }
    }
    return 0;
}
```

string

字符串hash

```
using std::string;

const int M = 1e9 + 7;
const int B = 233;

typedef long long ll;

int get_hash(const string& s) {
    int res = 0;
    for (int i = 0; i < s.size(); ++i) {
        res = ((ll)res * B + s[i]) % M;
    }
    return res;
}

bool cmp(const string& s, const string& t) {
    return get_hash(s) == get_hash(t);
}
```

```
#define ll long long    // 双Hash方法，不同的Base和MOD，相当于两次 单Hash
ll Base1 = 29;
ll Base2 = 131;
ll MOD1 = 1e9 + 7;
ll MOD2 = 1e9 + 9;
const int MAXN = 2e4 + 50;

class Solution {
public:
    set< pair <ll, ll> > H;    // 因为是一个二元组，所以可以用 pair 容器。
    ll h1[MAXN], h2[MAXN], p1[MAXN], p2[MAXN];

    int distinctEchoSubstrings(string text) {
        int n = text.size();
        h1[0] = 0, h2[0] = 0, p1[0] = 1, p2[0] = 1;
        for(int i = 0; i < n; i++)
        {
            h1[i+1] = (h1[i]*Base1 + (text[i] - 'a' + 1)) % MOD1;
            h2[i+1] = (h2[i]*Base2 + (text[i] - 'a' + 1)) % MOD2;
        }

        for(int i = 1; i < n; i++)
        {
            p1[i] = (p1[i-1]*Base1) % MOD1;
```

```

        p2[i] = (p2[i-1]*Base2) % MOD2;
    }

    for(int len = 2; len <= n; len += 2)
    {
        for(int i = 0; i + len - 1 < n; i++)
        {
            int x1 = i, y1 = i + len/2 - 1;
            int x2 = i + len/2, y2 = i + len - 1;
            ll left1 = ((h1[y1 + 1] - h1[x1] * p1[y1 + 1 - x1])%MOD1+MOD1) %
MOD1;
            ll right1 = ((h1[y2 + 1] - h1[x2] * p1[y2 + 1 - x2])%MOD1+MOD1) %
MOD1;
            ll left2 = ((h2[y1 + 1] - h2[x1] * p2[y1 + 1 - x1])%MOD2 + MOD2) %
MOD2;
            ll right2 = ((h2[y2 + 1] - h2[x2] * p2[y2 + 1 - x2])%MOD2 + MOD2) %
MOD2;

            if(left1 == right1 && left2 == right2) H.insert(make_pair(left1,
left2));
        }
    }
    return H.size();
}
};

```

找最小字符串O(n)

```

#include <bits/stdc++.h>
using namespace std;
#define ll long long
const int inf=0x3f3f3f3f;
const int N=2e6+1;
const int M=1e3+10;
string s;

int getMin(){
    int i=0, j=1, k=0;
    int len=s.size();
    while(i<len && j<len && k<len){
        int ti=(i+k)%len, tj=(j+k)%len;
        if(s[ti]==s[tj]) k++;
        else{
            if(s[ti]>s[tj]){
                i=i+k+1;
                k=0;
            }
            else{

```

```

        j=j+k+1;
        k=0;
    }
    if(i==j) j++;
}
}
return i<j?i:j;
}
int main()
{
    ios::sync_with_stdio(false);
    cin.tie(0);cout.tie(0);
    cin>>s;
    int k=getMin();
    string temp="";
    s=s+s;
    for(int i=0;i<s.size()/2;i++)
    {
        temp=temp+s[(i+k)%s.size()];
    }
    s=temp;
    cout<<s<<"\n";
    return 0;
}

```

(1).字符串找到特定子串修改成别的字符串

```

#include<iostream>
using namespace std;
string a,b;

int ch(string s,int x)
{
    if(s[x]>='a' && s[x]<='z') return 1;
    else if(s[x]>='A' && s[x]<='Z') return 2;
    else if(s[x]>='0' && s[x]<='9') return 3;
    else if(s[x]==' ') return 4;
    else return 5;
}

int check(int x,int y)
{
    if((x<0||b[x]!=' '||ch(b,x)==5)&&(y>=b.size()||b[y]!=' '||ch(b,y)==5))
        return 1;
    else
        return 0;
}

int main()

```

```

{
    int N;
    cin>>N;
    getchar();
    while(N-->0)
    {
        getline(cin,a);
        cout << a << endl << "AI: ";
        int l = 0,r = a.size() - 1;
        while(a[l]!=' ') l++;
        while(a[r]!=' ') r--;
        for(int i=l; i<=r; i++)
        {
            if(ch(a,i) == 2 && a[i] != 'I')
                b+=a[i]+32;
            else if(a[i] == '?')
                b+='!';
            else if(a[i] == ' ' && (a[i+1] == ' ' || ch(a,i+1) == 5))
                continue;
            else
                b+=a[i];
        }
        for(int i=0; i<b.size(); i++)
        {
            if(b[i]=='I' && check(i-1,i+1))
                cout<<"you";
            else if(b.substr(i,2) == "me" && check(i-1,i+2))
                cout<<"you", i++;
            else if(b.substr(i,7) == "can you" && check(i-1,i+7))
                cout<<"I can", i+=6;
            else if(b.substr(i,9) == "could you" && check(i-1,i+9))
                cout<<"I could", i+=8;
            else
                cout<<b[i];
        }
        cout<<endl;
        b="";
    }
}

```

字典树

```

#include <bits/stdc++.h>
using namespace std;
#define ll long long
const int inf=0x3f3f3f3f;
const int N=3e6+1;
int son[N][70];
int cnt[N];
int idx=0;
int getnum(char x){

```



```

        if(x>='A'&&x<='Z')
            return x-'A';
        else if(x>='a'&&x<='z')
            return x-'a'+26;
        else
            return x-'0'+52;
    }
void insert(string s)
{
    int p=0;
    for(int i=0;i<s.size();i++)
    {
        int u=getnum(s[i]);
        if(!son[p][u])
        {
            son[p][u]=++idx;
        }
        p=son[p][u];
        cnt[p]++;
    }
}
int find(string s)
{
    int p=0;
    for(int i=0;i<s.size();i++)
    {
        int u=getnum(s[i]);
        if(!son[p][u])
        {
            return 0;
        }
        p=son[p][u];
    }
    return cnt[p];
}
void solve()
{
    int n,m;
    cin>>n>>m;
    string str;
    while(n--)
    {
        cin>>str;
        insert(str);
    }
    while(m--)
    {
        cin>>str;
        cout<<find(str)<<"\n";
    }
}
int main()

```

```

{
    ios::sync_with_stdio(false);
    int t;
    cin>>t;
    while(t--)
    {

        for(int i=0;i<=idx;i++)
        {
            cnt[i]=0;
            for(int j=0;j<=69;j++)
            {
                son[i][j]=0;
            }
        }
        idx=0;
        solve();
    }
    return 0;
}

```

后缀SA

```

#include <bits/stdc++.h>
using namespace std;
#define ll long long
constexpr ll inf=0x3f3f3f3f;
constexpr ll N=2e6+10;
constexpr ll M=1e3+10;
using namespace std;

int fir[N],rnk[N],sa[N],sec[N],s[N],h[N];
int st[N],top;
ll val[N],ans[N];
string str;
void MakeSA(int n,int m)
{
    int *x=rnk,*y=sec;
    for(int i=1;i<=n;i++)s[x[i]=str[i]-'a'+1]++;
    for(int i=1;i<=m;i++)s[i]+=s[i-1];
    for(int i=n;i;i--)sa[s[x[i]]--]=i;
    for(int j=1,p=0;p<n?(p=0,1):0;j<=1,m=p)
    {
        for(int i=n-j+1;i<=n;i++)y[++p]=i;
        for(int i=1;i<=n;i++)if(sa[i]>j)y[++p]=sa[i]-j;
        for(int i=1;i<=m;i++)s[i]=0;
        for(int i=1;i<=n;i++)s[fir[i]=x[y[i]]]++;
        for(int i=1;i<=m;i++)s[i]+=s[i-1];
        for(int i=n;i;i--)sa[s[fir[i]]--]=y[i];
        swap(x,y),x[sa[p=1]]=1;
    }
}

```

```

        for(int i=2;i<=n;i++)x[sa[i]]=(p+=(y[sa[i]]!=y[sa[i-1]]||y[sa[i]+j]!=y[sa[i-1]+j])));
    }
    for(int i=1;i<=n;i++)rnk[sa[i]]=i;
    // sa[x]=y
    // 第x小的后缀串的编号是y

    for(int i=1,j=0;i<=n;i++)
    {
        if(rnk[i]==1)h[rnk[i]]=0;
        else
        {
            j=max(j-1,0);
            while(str[i+j]==str[sa[rnk[i]-1]+j])j++;
            h[rnk[i]]=j;
        }
    }
    top=0,st[0]=1;
    for(int i=2;i<=n;i++)
    {
        while(top&&h[st[top]]>=h[i])top--;
        st[++top]=i,val[top]=val[top-1]+1ll*(i-st[top-1])*h[i];
        ans[i]+=val[top];
    }
    top=0,st[0]=n+1;
    for(int i=n;i>=2;i--)
    {
        while(top&&h[st[top]]>=h[i])top--;
        st[++top]=i,val[top]=val[top-1]+1ll*(st[top-1]-i)*h[i];
        ans[i-1]+=val[top];
    }
    for(int i=1;i<=n;i++)printf("%lld\n",ans[rnk[i]]+(n-i+1));
}

int main(){
    int n;
    cin>>n;
    cin>>str;
    str=" "+str;
    MakesA(n,26);
    for(int i=1;i<=n;i++)
    {
        cout<<sa[i]<<' '<<rnk[i]<<" "<<sa[rnk[i]]<<endl;
    }
}

```

```

#include <bits/stdc++.h>
using namespace std;
#define ll long long

```

```

const int inf=0x3f3f3f3f;
const int N=4e6+1;
const int M=1e3+10;
int son[N][70];
int cnt[N];
int idx=0;
int val[N];
string strings[N];
int getnum(char x){
    if(x>='A'&&x<='Z')
        return x-'A';
    else if(x>='a'&&x<='z')
        return x-'a'+26;
    else
        return x-'0'+52;
}
void insert(string s,int v)
{
    int p=0;
    for(int i=0;i<s.size();i++)
    {
        int u=getnum(s[i]);
        if(!son[p][u])
        {
            son[p][u]=++idx;
        }
        p=son[p][u];
        cnt[p]++;
    }
    val[p]+=v;
}
int find(string s)
{
    int p=0;
    for(int i=0;i<s.size();i++)
    {
        int u=getnum(s[i]);
        if(!son[p][u])
        {
            return -1;
        }
        p=son[p][u];
    }
    if(val[p])return val[p];
    else return -1;
}
char temp[1005];

void solve()
{
    int n,m;
    cin>>n>>m;

```

```

for(int i=1;i<=n;i++)
{
    cin>>strings[i];
}
string str;int v;
for(int i=1;i<=m;i++)
{
    cin>>str>>v;
    insert(str,v);
}
ll ans=0;int f=0;
for(int i=1;i<=n;i++)
{
    for(int j=0;j<n;j++)
    {
        int k=0;
        if(strings[i][j]=='#')continue;
        while(strings[i][j]!='#'&&j<n)
        {
            temp[k++]=strings[i][j++];
        }
        temp[k]=0;
        int val=find(temp);
        if(val==-1)f=1;
        else ans+=val;
    }
}
for(int j=0;j<n;j++)
{
    for(int i=1;i<=n;i++)
    {
        int k=0;
        if(strings[i][j]=='#')continue;
        while(strings[i][j]!='#'&&i<=n)
        {
            temp[k++]=strings[i++][j];
        }
        temp[k]=0;
        int val=find(temp);
        if(val==-1)f=1;
        else ans+=val;
    }
}
if(f)cout<<-1<<"\n";
else cout<<ans<<"\n";
}
int main()
{
    ios::sync_with_stdio(false);
    int t;
    cin>>t;
    while(t--)

```

```

{
    for(int i=0;i<=idx;i++)
    {
        cnt[i]=0;
        for(int j=0;j<=69;j++)
        {
            son[i][j]=0;
        }
        val[i]=0;
    }
    idx=0;

    solve();
    // cout<<find("bd")<<endl;
}
return 0;
}

```

set

erase find *lower sort

```

#include <bits/stdc++.h>
using namespace std;
#define ll long long
const int inf=0x3f3f3f3f;
const int N=5e5+1;
int a[N];
int b[N];
int p[N];
void solve()
{
    int n;
    cin>>n;
    for(int i=1;i<=n;i++)
    {
        cin>>a[i]>>b[i];
        p[i]=i;
    }
    sort(p+1,p+n+1,[&](int i,int j){return a[i]>a[j];});
    multiset<int>st;
    st.insert(inf);st.insert(-inf);
    for(int i=1;i<=n;i++)
    {
        st.insert(b[i]);
    }
    int maxx=-inf;int minn=inf;
    for(int i=1;i<=n;i++)
    {

```

```

        st.erase(st.find(b[p[i]]));
        minn=min(minn,abs(a[p[i]]-max(maxx,*st.lower_bound(a[p[i]]))));
        minn=min(minn,abs(a[p[i]]-max(maxx,*prev(st.upper_bound(a[p[i]]))));
        maxx=max(maxx,b[p[i]]); //前i个的最大b【i】值
                                //这是必须选的
    }
    cout<<minn<<"\n";
    return ;
}
int main()
{
    ios::sync_with_stdio(false);
    int t;
    cin>>t;
    while(t-->0)
    {
        solve();
    }
    return 0;
}

```

iterator

```

typedef long long LL;
const int N=5e5;
int a[N];
int cnt;
bool st[N];
void solve()
{
    int n,k;
    cin>>n>>k;
    set<int>s;
    for(int i=1;i<=n;i++)s.insert(i);
    while(k-->0)
    {
        int cho;
        cin>>cho;
        if(cho==1)
        {
            int x;
            cin>>x;
            s.erase(x);
        }
        else
        {
            int x;
            cin>>x;
            set<int>::iterator l;
            l=s.find(x);
            if(l==s.begin())

```

```

        {
            cout<<0<<endl;
        }
        else
        {
            l--;
            cout<<*l<<endl;
        }
    }
}
signed main()
{
    ios::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
    int T;
    T=1;
    while(T--)
    {
        solve();
    }

    return 0;
}

```

vector

insert in mid

```

#include <bits/stdc++.h>
using namespace std;
#define ll long long
const int inf=0x3f3f3f3f;
const int N=1e5+1;
int a[N];
vector<int>v;
void solve()
{
    string op;
    int num;
    cin>>op;
    //cout<<v.size()<<"size\n";
    if(op=="add")
    {
        cin>>num;
        v.insert(upper_bound(v.begin(), v.end(), num), num);
    }
}

```



```

else if(op=="mid")
{
    if(v.size()&1)
    {
        cout<<v[v.size()>>1]<<"\n";
    }
    else
    {
        cout<<v[(v.size()>>1)-1]<<"\n";
    }
}
return ;
}
int main()
{
    ios::sync_with_stdio(false);
    cin.tie(0);cout.tie(0);
    int n;
    cin>>n;
    for(int i=1;i<=n;i++)
    {
        int x;
        cin>>x;
        //v.push_back(x);
        v.insert(upper_bound(v.begin(), v.end(),x),x);
    }
    int t;
    cin>>t;
    while(t--)
    {
        solve();
    }
    return 0;
}

```

unique sort+去重

```

sort(v.begin(), v.end());
v.erase(unique(v.begin(), v.end()),v.end());

```

find maxx

```

int x=*max_element(a.begin(),a.begin()+m);

```