# Zuiduanlu

2612

#include<iostream>

using namespace std;

#define MAXN 50010

#define MAXM 1010

struct data{

int x;

int v;

data(){

}

data(int \_x,int \_v){

x=\_x;

v=\_v;

}

friend bool operator <(data x,data y){

return x.v>y.v;

}

};

int n;

int a[MAXN];

int mn;

int f[MAXN];

priority\_queue<data>q;

bool vis[MAXN];

void dijkstra(){

int i,x;

memset(f,0x3f,sizeof(f));

f[0]=0;

q.push(data(0,0));

while(!q.empty()){

x=q.top().x;

q.pop();

if(vis[x]){

continue ;

}

vis[x]=1;

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

if(f[x]+a[i]<f[(x+a[i])%mn]){

f[(x+a[i])%mn]=f[x]+a[i];

q.push(data((x+a[i])%mn,f[(x+a[i])%mn]));

}

}

}

}

int main(){

int i,j,x;

scanf("%d",&n);

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

scanf("%d",&a[i]);

}

mn=a[1];

dijkstra();

scanf("%d",&n);

while(n--){

scanf("%d",&x);

printf(x>=f[x%mn]?"TAK\n":"NIE\n");

}

return 0;

}

6071

#include<iostream>

#include<cmath>

#include<algorithm>

#include<cstdlib>

#include<cstdio>

#include<vector>

#include<queue>

#include<cstring>

using namespace std;

#define MAXN 120010

const long long N = LONG\_LONG\_MAX/4;

long long MIN(long long A,long long B){

return A<B?A:B;

}

long long MAX(long long A,long long B){

return A>B?A:B;

}

struct data{

int x,y,v;

data(){}

data(int \_x,int \_y,long long \_v){

x=\_x;y=\_y;v=\_v;

}

friend bool operator <(data x,data y){

return x.v>y.v;

}

};

long long K;

long long a[4];

long long mod;

long long f[MAXN][4];

priority\_queue<data>q;

bool vis[MAXN][4];

void dijkstra(){

q.push(data(0,1,0));

while(!q.empty()){

int x=q.top().x;

int y=q.top().y;

q.pop();

if(vis[x][y]){

continue ;

}

vis[x][y]=1;

int nx=(x+a[y])%mod;

int ny=(y+1)%4;

if(f[x][y]+a[y]<f[nx][ny]){

f[nx][ny]=f[x][y]+a[y];

q.push(data(nx,ny,f[nx][ny]));

}

nx=((x+a[(y-1+4)%4])%mod+mod)%mod;

ny=(y-1+4)%4;

if(f[x][y]+a[(y-1+4)%4]<f[nx][ny]){

f[nx][ny]=f[x][y]+a[(y-1+4)%4];

q.push(data(nx,ny,f[nx][ny]));

}

}

}

long long ans;

long long cnt(long long A,long long B){

long long C=A/B\*B;

if(C<A)C=C+B;

return C;

}

int main(){

freopen("1005.in","r",stdin);

int T;

scanf("%d",&T);

while(T--){

scanf("%lld",&K);

for(int i=0;i<4;i++){

scanf("%lld",&a[i]);

}

mod=2\*MIN(a[0],a[1]);

for(int i=0;i<mod;i++){

for(int j=0;j<4;j++){

f[i][j]=N;vis[i][j]=0;

}

}

f[0][1]=0;

dijkstra();

ans=N;

for(long long i=0;i<mod;i++){

if(f[i][1]>=K)ans=MIN(ans,f[i][1]);

else ans=MIN(ans,f[i][1]+cnt(K-f[i][1],mod));

}

printf("%lld\n",ans);

}

return 0;

}

6166

#include<bits/stdc++.h>

#define mst(a,b) memset(a,b,sizeof(a))

#define F(i,a,b) for(int i=(a);i<=(b);++i)

using namespace std;

typedef long long ll;

const int N=2e5+7;ll inf=1ll<<60;

int g[N],v[N],nxt[N],w[N],ed;

int t,n,m,k,a[N],Q[N],in[N],cas;

ll d[N];

struct EDGE{int a,b,c;}edge[N];

void adg(int x,int y,int z){v[++ed]=y,w[ed]=z,nxt[ed]=g[x],g[x]=ed;}

ll spfa(int S,int T){

F(i,1,T)d[i]=inf,in[i]=0;

d[S]=0,Q[1]=S,in[S]=1;

unsigned short l=1,r=2;

while(l!=r){

int x=Q[l++];in[x]=0;

for(int i=g[x];i;i=nxt[i])

if(d[v[i]]>d[x]+w[i])

{

d[v[i]]=d[x]+w[i];

if(!in[v[i]])in[v[i]]=1,Q[r++]=v[i];

}

}

return d[T];

}

int main(){

scanf("%d",&t);

while(t--) {

scanf("%d%d",&n,&m);

F(i,1,m)scanf("%d%d%d",&edge[i].a,&edge[i].b,&edge[i].c);

scanf("%d",&k);

F(i,1,k)scanf("%d",a+i);

int S=n+1,T=n+2,pre=ed,now;

ll ans=inf;

F(i,0,17)

{

mst(g,0),ed=0,now=1<<i;

F(j,1,m)adg(edge[j].a,edge[j].b,edge[j].c);

F(j,1,k)

{

if(a[j]&now)adg(S,a[j],0);

else adg(a[j],T,0);

}

ans=min(ans,spfa(S,T));

mst(g,0),ed=0;

F(j,1,m)adg(edge[j].a,edge[j].b,edge[j].c);

F(j,1,k)

{

if((a[j]&now)==0)adg(S,a[j],0);

else adg(a[j],T,0);

}

ans=min(ans,spfa(S,T));

}

printf("Case #%d: %lld\n",++cas,ans);

}

return 0;

}