1. Hdu6156(回文数位dp) （2017-8-22）

#include <iostream>

#include <cstring>

#include <cmath>

#include <queue>

#include <stack>

#include <list>

#include <map>

#include <set>

#include <string>

#include <cstdlib>

#include <cstdio>

#include <algorithm>

using namespace std;

int T;

typedef long long LL;

int a[33],t[33];

LL dp[33][33][2];

LL dfs(int len,int cur,int s,int fp,int y)

{

if(!cur)

return s?y:1;

if(!fp && dp[len][cur][s] != -1)

return dp[len][cur][s];

int n = fp?a[cur]:y-1;

LL res = 0;

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

{

t[cur] = i;

if(len == cur && i == 0)

res += dfs(len-1,cur-1,s,fp&&i==n,y);

else if(s && cur <= (len+1)/2)

res += dfs(len,cur-1,t[len-cur+1]==i,fp&&i==n,y);

else

res += dfs(len,cur-1,s,fp&&i==n,y);

}

if(!fp)

dp[len][cur][s] = res;

return res;

}

LL sum(LL x,LL y)

{

int len = 0;

while(x)

{

a[++len] = x % y;

x /= y;

}

//printf("len=%d\n",len);

return dfs(len,len,1,1,y);

}

const int N = 1e5+10;

LL L[N],R[N],l[N],r[N],ans[N];

int main()

{

int Ca=1;

scanf("%d",&T);

for (int ca=1;ca<=T;ca++){

scanf("%I64d%I64d%I64d%I64d",&L[ca],&R[ca],&l[ca],&r[ca]);

}

for (int i=2;i<=36;i++){

memset(dp,-1,sizeof dp);

for (int ca=1;ca<=T;ca++)if (l[ca]<=i && i<=r[ca]){

ans[ca]+=sum(R[ca],i)-sum(L[ca]-1,i);

}

}

for (int ca=1;ca<=T;ca++){

printf("Case #%d: %I64d\n",ca,ans[ca]);

}

return 0;

}

1. BZOJ2115最大xor路径（2017-8-22）

// <BZOJ2115.cpp> - Wed Aug 3 16:26:56 2016

// This file is created by XuYike's black technology automatically.

// Copyright (C) 2015 ChangJun High School, Inc.

// I don't know what this program is.

#include <iostream>

#include <vector>

#include <algorithm>

#include <cstring>

#include <cstdio>

#include <cmath>

using namespace std;

typedef long long lol;

int gi(){

int res=0,fh=1;char ch=getchar();

while((ch>'9'||ch<'0')&&ch!='-')ch=getchar();

if(ch=='-')fh=-1,ch=getchar();

while(ch>='0'&&ch<='9')res=res\*10+ch-'0',ch=getchar();

return fh\*res;

}

lol gl(){

lol res=0,fh=1;char ch=getchar();

while((ch>'9'||ch<'0')&&ch!='-')ch=getchar();

if(ch=='-')fh=-1,ch=getchar();

while(ch>='0'&&ch<='9')res=res\*10+ch-'0',ch=getchar();

return fh\*res;

}

const int MAXN=50010;

const int MAXM=200010;

const int INF=1e9;

int bt=1,b[MAXN],next[MAXM],to[MAXM];lol val[MAXM];

inline void add(lol z,int y,int x){

next[++bt]=b[x];b[x]=bt;to[bt]=y;val[bt]=z;

next[++bt]=b[y];b[y]=bt;to[bt]=x;val[bt]=z;

}

bool vis[MAXN];

lol dis[MAXN],h[MAXM];

int tot;

void dfs(int x,int f){

vis[x]=1;

for(int i=b[x];i;i=next[i]){

if((i^f)==1)continue;

if(vis[to[i]]){

if(i&1)h[++tot]=dis[x]^dis[to[i]]^val[i];

continue;

}

dis[to[i]]=dis[x]^val[i];

dfs(to[i],i);

}

}

int main(){

int n=gi(),m=gi();

for(int i=1;i<=m;i++)add(gl(),gi(),gi());

dfs(1,0);

lol ans=dis[n];

for(lol p=1ll<<62;p;p>>=1){

int fd=0;

for(int i=1;i<=tot;i++)

if(h[i]&p)

if(fd)h[i]^=h[fd];

else fd=i;

if(fd&&!(ans&p))ans^=h[fd];

h[fd]=0;

}

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

return 0;

}