**題目**：unknown switch

**題目大意**：給一組開關，和一堆燈泡，還有一堆開關控制上的結果(比汝說開了某些開關，哪些燈泡會亮那樣)，要求的是這些燈泡被哪些開關控制。

**解法**：先把書入略做修改，因為輸入的開關控制是，”動”了哪些開關，而不是哪些開關是開著的，把開關的狀態改成，是哪些開關開著。接著假設所有的燈泡都是被所有的開關控制著，接著讀每一筆資料，做以下處理，如果資料中燈泡是亮的，把能控制燈泡的開關集合中，沒有開的開關去掉，反之亦然。罪後如果有燈泡的開關控制數量不是1就輸出問號，要不然就輸出是哪個開關

**程式碼**：

#include <iostream>

#include <cstdio>

#include <vector>

#include <cstring>

using namespace std;

int n,m,q;

vector<int> bulb[1001];

char qu[1001], light[1001], last[1001];

vector<int> change\_s, unchange\_s;

vector<int> change\_l, unchange\_l;

int main()

{

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

while(scanf("%d %d %d", &n,&m,&q) == 3){

if(n == 0 && m == 0 && q == 0)break;

for(int i = 0;i < 1001;i++){

bulb[i].clear();

for(int a = 0;a < n;a++){

bulb[i].push\_back(a);

}

}

memset(last, '0', sizeof(last));

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

scanf("%s %s", qu, light);

change\_s.clear();

unchange\_s.clear();

change\_l.clear();

unchange\_l.clear();

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

if(qu[j] == '1')change\_s.push\_back(j);

else unchange\_s.push\_back(j);

}

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

if(light[j] != last[j]){

change\_l.push\_back(j);

}else {

unchange\_l.push\_back(j);

}

}

for(int j = 0;j < unchange\_l.size();j++){

int temp = unchange\_l[j];

if(bulb[temp].size() > 1){

for(int a = 0;a < change\_s.size();a++){

for(int b = 0;b < bulb[temp].size();b++){

if(bulb[temp][b] == change\_s[a]){

bulb[temp].erase(bulb[temp].begin()+b);

break;

}

}

}

}

}

for(int j = 0;j < change\_l.size();j++){

int temp = change\_l[j];

if(bulb[temp].size() > 1){

for(int a = 0;a < unchange\_s.size();a++){

for(int b = 0;b < bulb[temp].size();b++){

if(bulb[temp][b] == unchange\_s[a]){

bulb[temp].erase(bulb[temp].begin()+b);

break;

}

}

}

}

}

//for(int a = 0;a < m;a++){

//cout <<"bulb["<<a<<"] = ";

//for(int b = 0;b < bulb[a].size();b++){

// cout <<bulb[a][b]<<" ";

//}cout <<endl;

//}

strcpy(last, light);

}

char ans[1001];

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

if(bulb[i].size() != 1){

putchar('?');

//ans[i] = '?';

}else {

if(bulb[i][0] <= 9)putchar((char)('0' + bulb[i][0]));

//ans[i] = (char)('0' + bulb[i][0]);

else putchar((char)('A' + (bulb[i][0]-10)));

//ans[i] = (char)('A'+(bulb[i][0]-10));

}

}cout <<endl;

//ans[m] = '\0';

//puts(ans);

}

return 0;

}