解題報告

題目： Bit String Reordering

給一個字串，做為輸入，然後再給一串字串，然後求最少的轉換次數

解法：

一開始使用BFS求解，不過，TLE了，再來改用另一種類似greedy的策略。

先把目標的數字字串，轉換成bit string(分別有從1開頭和從0開頭的兩種答案)，然後去找最近的1，和答案的1，去比較位置，然後把答案加起來，最後，再來比較最少步數的答案，就是所求。

程式碼：

#include <iostream>

#include <cstdio>

#include <queue>

#include <vector>

#include <cstdlib>

using namespace std;

int n,m;

int ans[20];

struct bitstring{

int bits[20];

int one;

bitstring(){}

bitstring(int input[], int start){

one = 0;

int j = 0;

int target = start;

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

for(int a = 0;j < n, a < input[i];j++,a++){

bits[j] = target;

if(target == 1)

one++;

}

target = 1 - target;

}

}

};

int main()

{

freopen("A\_40\_88.in", "r", stdin);

while(scanf("%d %d", &n,&m) == 2){

bitstring a;

a.one = 0;

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

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

if(a.bits[i] == 1)

a.one++;

}

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

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

}

bitstring ansone(ans, 0);

//cout <<"ans 1 = \n";

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

// cout <<ansone.bits[i]<<" ";

//}cout <<endl;

bitstring anstwo(ans, 1);

//cout <<"ans 2 = \n";

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

// cout <<anstwo.bits[i]<<" ";

//}cout <<endl;

int minimum;

if(a.one == ansone.one){

minimum = 0;

for(int i = 0,j = 0;i < n,j < n;){

if(a.bits[i] == 1 && ansone.bits[j] == 1){

minimum += abs(i-j);

i++,j++;

}else if(a.bits[i] != 1){

i++;

}else if(ansone.bits[j] != 1){

j++;

}

}

}else {

minimum = 999999999;

}

int temp = 0;

if(a.one == anstwo.one){

for(int i = 0,j = 0;i < n,j < n;){

if(a.bits[i] == 1 && anstwo.bits[j] == 1){

temp += abs(i-j);

i++,j++;

}else if(a.bits[i] != 1){

i++;

}else if(anstwo.bits[j] != 1){

j++;

}

}

}else temp = 99999999;

if(temp < minimum){

printf("%d\n", temp);

}else printf("%d\n", minimum);

}

return 0;

}