TAXI

Đón i, trả i+N, phải đi qua toàn bộ 2xN điểm mỗi điểm 1 lần và quay về điểm xuất phát, với chi phí MIN. N<=10

* Bài toán người du lịch: người du lịch cần tham quan N thành phố, mỗi thành phố 1 lần và quay trở về tp xuất phát 🡺 (1) duyệt nhánh cận, (2) QHĐ+bitmask

QHĐ bitmask:

Tp 0..n-1. Gọi dp(i, S) là chi phí MIN là người du lịch đã đi qua tất cả những tp trong tập S và đang đứng tại tp i:

* Res= MIN(dp(i, S=mọi đỉnh)+C[i,0])
* dp(i,S)=MIN\_j (dp(j,S+{j})+C[i,j]), với mọi j không thuộc tập S

Cài đặt: tập S được biểu diễn bởi một số nguyên 32 bít, bít thứ i = 0 nghĩa là i không thuộc S, còn i=1 nghĩa là I thuộc S.

mem[i,S]

int dp(int i; int S) {

if (S==1<<N) return C[i,0];

if (mem[i,S]!=0) return mem[i,S]; 🡪 đảm bảo không gọi quá N x 2^N lần dp

int tmp=vô cùng;

for (j=0..N-1) 🡪 O(N)

if (S & (1<<j) == 0) {

tmp=MIN(tmp,dp(j,S | (1<<j))+C[i,j]);

}

mem[i,S]=tmp;

return tmp;

}

ĐPT = số lần gọi dp x ĐPT 1 lần dp = O(N^2 x 2^N)