

[比赛链接](#)

补题情况

题目	蒋贤蒙	王赵安	王智彪
B	0	0	0
C	0	0	0
D	0	0	0
E	0	0	0
G	0	0	0
J	0	0	0
K	0	0	0
M	0	0	0

题解

G. Fascination Street

题意

给定 n 个位置，选择位置 i 的费用为 w_i 。一个合法方案要求每个位置自己被选中或者自己左右两边相邻的格子至少有一个被选中。

允许 k 次操作，每次可以交换位置 i 和位置 j 的费用。问合法方案的最小费用。

题解

直接处理交换显然不可作，但交换可以等价于对方案中选中的 x 个位置不付费，再对方案中没有选中的 x 个位置付费。

设 $\text{dp}(i, 0/1, 0/1, k_1, k_2)$ 表示只考虑前 i 个位置，保证前 $i-1$ 个位置已经合法，其中第 $i-1, i$ 个位置是否被选中。

然后有 k_1 个位置选中但没付费，有 k_2 个位置没选中但付费的方案的最小费用。不难想到 dp 转移。

边界为 $\text{dp}(0, 1, 0, 0, 0) = 0$ 。最终答案为 $\min_{j_1+j_2 \neq 0, 0 \leq t \leq k} \text{dp}(n, j_1, j_2, t, t)$ 。总时间复杂度 $O(nk^2)$ 。

```
const int MAXN=2.5e5+5,MAXK=11;
const LL inf=1e18;
int a[MAXN];
LL dp[2][2][2][MAXK][MAXK];
void set_min(LL &x,LL y){
    x=min(x,y);
}
```

```
}
int main(){
    int n=read_int(),m=read_int(),pos=0;
    _rep(i,1,n)
    a[i]=read_int();
    mem(dp[pos],127);
    dp[pos][1][0][0][0]=0;
    _rep(i,1,n){
        pos=!pos;
        mem(dp[pos],127);
        _for(j1,0,2)_for(j2,0,2)
        _rep(k1,0,m)_rep(k2,0,m){
            LL t=dp[!pos][j1][j2][k1][k2];
            set_min(dp[pos][j2][1][k1][k2],t+a[i]);
            set_min(dp[pos][j2][1][k1+1][k2],t);
            if(j1||j2){
                set_min(dp[pos][j2][0][k1][k2],t);
                set_min(dp[pos][j2][0][k1][k2+1],t+a[i]);
            }
        }
    }
    LL ans=inf;
    _for(j1,0,2)_for(j2,0,2){
        if(j1||j2){
            _rep(k,0,m)
            set_min(ans,dp[pos][j1][j2][k][k]);
        }
    }
    enter(ans);
    return 0;
}
```

From: <https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link: https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal_string:%E7%BB%84%E9%98%9F%E8%AE%AD%E7%BB%83%E6%AF%94%E8%B5%9B%E8%AE%B0%E5%BD%95:contest14&rev=1628771242

Last update: 2021/08/12 20:27