20241203题解

A

感觉不用讲。直接输出s[1]s[2]s[0]。

B

从给定点开始直接向四个方向遍历就行了。别忘了加上自己。

C

注意到n是20，所以可以直接搜索（或者说是枚举），因为搜索到一个新数时无非两种决策：加入原来的那一段或者新开一段。新开的时候就将之前那段的答案tmp异或到ans上，并将tmp赋值为现在这个数；加入的话就将这个数按位或到tmp上。最后存ans的最小值即可。

D

万恶的计算几何，虽然公式不难推但是不好写。

首先已知N是偶数且已知第0个和第N/2个点，因此这两个点的连线一定是最长的对角线，也是这个正N边形的外接圆的直径，因此这条连线的中点是外接圆的圆心，记为O’(x’,y’)，则我们有x’=(x[0]+x[N/2])/2，y’=(y[0]+y[N/2])/2。

那么我们将p[0]绕O’逆时针旋转theta=(2\*pi/N)度就得到了p[1]。这里你可以用复数来做，也可以直接推旋转公式来做。

如果绕的是原点旋转的话，那么旋转后的点坐标自然是(x[0]\*cos(theta)-y[0]\*sin(theta),y[0]\*cos(theta)+x[0]\*sin(theta))，但我们绕的是(x’,y’)，因此要先将(x’,y’)平移到(0,0)，再平移回去，得到最终式子是：

((x[0]-x’)\*cos(theta)-(y[0]-y’)\*sin(theta)+x’,(y[0]-y’)\*cos(theta)+(x[0]-x’)\*sin(theta)+y’)

E

是个DP。

首先对于编号相同的球，显然是直接连着拿最好，因此把编号相同的球的最左边和最右边坐标先处理出来。那我们实际上关注的状态就是取完这个编号之后是停在最左边还是最右边。

状态：dp[i][j]表示取完编号为i的球后停在j端的最小距离（j=0表示左，j=1表示右）。再记录pos[i][j]表示编号为i的球第j端的坐标，那么我们有：

dp[i][0]=min(dp[i-1][0]+|pos[i][1]-pos[i-1][0]|,dp[i-1][1]+|pos[i][1]-pos[i-1][1])+pos[i][1]-pos[i][0]

dp[i][1]=min(dp[i-1][0]+|pos[i][0]-pos[i-1][0]|,dp[i-1][1]+|pos[i][0]-pos[i-1][1])+pos[i][1]-pos[i][0]

注意编号不一定连续，建议先离散化编号。

答案为min(dp[k][0]+pos[k][0],dp[k][1]+pos[k][1])，因为最后还要走回0，k是编号的种类数。

F

首先是回文，那可以想到双向搜索。

过程大概是：从两边当前搜到的点出发，每次两个位置各选出一条相连的边满足边上的字母相同。最终两个顶点重合或者相邻的时候就是答案。于是我们可以直接BFS，把两个结点打包成一个搜索时的状态入队即可，同时记录这个状态的步数。

对于最终两点重合的状态，直接返回步数\*2；对于最终两点相邻的状态，返回步数\*2+1。