矩阵与基本运算

线性代数（1）

Robert_JYH

一、矩阵与矩阵的基本运算

主要应用：矩阵快速幂加速

一、矩阵与矩阵的基本运算

1.1 矩阵介绍

矩阵的定义

- 由$m \times n$个数$a_{ij}(i=1,2,\cdots,m;j=1,2,\cdots,n)$排成的一个$m$行$n$列的数表

$$A=\begin{pmatrix} a_{11}a_{12}\cdots a_{1n} \\ a_{21}a_{22}\cdots a_{2n} \\ \cdots\cdots \\ a_{m1}a_{m2}\cdots a_{mn} \end{pmatrix}$$

- 称为一个m行n列矩阵或$m \times n$矩阵
- 当$m=n$时，称$A=\left(a_{ij}\right)_{n\times n}$为n阶矩阵或n阶方阵
- 元素全为零的矩阵称为零矩阵，用$O_{m\times n}$或O表示
- 例如：图论中的邻接矩阵
- 行向量：矩阵的某一行所形成的向量。
- 列向量：矩阵的某一列所形成的向量。

特殊方阵

①n 阶对角矩阵是指形如 $A=\begin{pmatrix} a_{11} & 0 & \cdots 0 \\ 0 & a_{22} & \cdots 0 \\ & \cdots\cdots & \\ 0 & 0 & \cdots a_{nn} \end{pmatrix}$ 的矩阵

②n 阶单位方阵是指形如 $E_n=\begin{pmatrix} 1 & 0\cdots 0 \\ 0 & 1\cdots 0 \\ & \cdots\cdots \\ 0 & 0\cdots 1 \end{pmatrix}$ 的矩阵

③n 阶三角矩阵是指形如 $\begin{pmatrix} a_{11} & a_{12} \cdots a_{1n} \\ 0 & a_{22} \cdots a_{2n} \\ & \cdots\cdots \\ 0 & 0 \cdots a_{nn} \end{pmatrix}, \begin{pmatrix} a_{11} & 0 & \cdots 0 \\ a_{21} & a_{22} & \cdots 0 \\ & \cdots\cdots & \\ a_{n1} & a_{n2} & \cdots a_{nn} \end{pmatrix}$ 的矩阵

线性运算：矩阵的加法与减法

- 对应元素相加减，要求两矩阵的行数与列数分别相等

例如

$$\begin{pmatrix} 12 & 3 & -5 \\ 1 & -9 & 0 \\ 3 & 6 & 8 \end{pmatrix} + \begin{pmatrix} 1 & 8 & 9 \\ 6 & 5 & 4 \\ 3 & 2 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 12+1 & 3+8 & -5+9 \\ 1+6 & -9+5 & 0+4 \\ 3+3 & 6+2 & 8+1 \end{pmatrix} = \begin{pmatrix} 13 & 11 & 4 \\ 7 & -4 & 4 \\ 6 & 8 & 9 \end{pmatrix}.$$

- 有交换律，结合律

线性运算：矩阵的数乘

- 设$A=(a_{ij})_{m\times n}$，k为任一个数，则规定$kA=(ka_{ij})_{m\times n}$
- 故数k与矩阵A的乘积就是A中所有元素都乘以k

$$\lambda A = A\lambda = \begin{pmatrix} \lambda a_{11} & \lambda a_{12} & \cdots & \lambda a_{1n} \\ \lambda a_{21} & \lambda a_{22} & \cdots & \lambda a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ \lambda a_{m1} & \lambda a_{m1} & \cdots & \lambda a_{mn} \end{pmatrix}.$$

- 有普通数字乘法运算具有的性质

矩阵的乘法

- 设$A=(a_{ij})_{m\times k}$，$B=(b_{ij})_{k\times n}$，则规定$AB=(c_{ij})_{m\times n}$
- 其中$c_{ij}=a_{i1}b_{1j}+a_{i2}b_{2j}+\cdots+a_{ik}b_{kj}\qquad(i=1,2,\cdots,m;j=1,2,\cdots,n)$
- 由此定义可知，只有当左矩阵A的列数与右矩阵B的行数相等时，AB才有意义，而且矩阵AB的行数为A的行数，AB的列数为B的列数，而矩阵AB中的元素是由左矩阵A中某一行元素与右矩阵B中某一列元素对应相乘再相加而得到.

$$C=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1s} \\ \vdots & \vdots & & \vdots \\ \boxed{a_{i1} \quad a_{i2} \quad \cdots \quad a_{is}} & & & \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{ms} \end{pmatrix}\begin{pmatrix} b_{11} & \cdots & b_{1j} & \cdots & b_{1n} \\ \vdots & & \vdots & & \vdots \\ b_{i1} & \cdots & b_{ij} & \cdots & b_{in} \\ \vdots & & \vdots & & \vdots \\ b_{s1} & \cdots & b_{sj} & \cdots & b_{sn} \end{pmatrix}$$

$$=(c_{ij})_{m\times n}$$

矩阵的乘法

- 设$A=(a_{ij})_{m\times k}$，$B=(b_{ij})_{k\times n}$，则规定$AB=(c_{ij})_{m\times n}$
- 其中$c_{ij}=a_{i1}b_{1j}+a_{i2}b_{2j}+\cdots+a_{ik}b_{kj} \qquad (i=1,2,\cdots,m;j=1,2,\cdots,n)$

$$A=\begin{pmatrix} 1 & 0 & -1 & 2 \\ -1 & 1 & 3 & 0 \\ 0 & 5 & -1 & 4 \end{pmatrix}, B=\begin{pmatrix} 0 & 3 & 4 \\ 1 & 2 & 1 \\ 3 & 1 & -1 \\ -1 & 2 & 1 \end{pmatrix}$$

矩阵的乘法

- 设$A=(a_{ij})_{m\times k}$，$B=(b_{ij})_{k\times n}$，则规定$AB=(c_{ij})_{m\times n}$
- 其中$c_{ij}=a_{i1}b_{1j}+a_{i2}b_{2j}+\cdots+a_{ik}b_{kj}\qquad(i=1,2,\cdots,m;j=1,2,\cdots,n)$

$$A=\begin{pmatrix}1 & 0 & -1 & 2\\ -1 & 1 & 3 & 0\\ 0 & 5 & -1 & 4\end{pmatrix},B=\begin{pmatrix}0 & 3 & 4\\ 1 & 2 & 1\\ 3 & 1 & -1\\ -1 & 2 & 1\end{pmatrix}$$

$$AB=\begin{pmatrix}-5 & 6 & 7\\ 10 & 2 & -6\\ -2 & 17 & 10\end{pmatrix}$$

矩阵的乘法

- 设$A=(a_{ij})_{m\times k}$，$B=(b_{ij})_{k\times n}$，则规定$AB=(c_{ij})_{m\times n}$
- 其中$c_{ij}=a_{i1}b_{1j}+a_{i2}b_{2j}+\cdots+a_{ik}b_{kj}$ $\quad (i=1,2,\cdots,m;j=1,2,\cdots,n)$

$$\begin{pmatrix}1 & 2 & 3\end{pmatrix}\begin{pmatrix}3\\2\\1\end{pmatrix}=(10) \qquad \begin{pmatrix}3\\2\\1\end{pmatrix}\begin{pmatrix}1 & 2 & 3\end{pmatrix}=\begin{pmatrix}3 & 6 & 9\\2 & 4 & 6\\1 & 2 & 3\end{pmatrix}$$

$$\begin{pmatrix}-2 & 4\\1 & -2\end{pmatrix}_{2\times 2}\begin{pmatrix}2 & 4\\-3 & -6\end{pmatrix}_{2\times 2}=\begin{pmatrix}-16 & -32\\8 & 16\end{pmatrix}_{2\times 2}$$

$$\begin{pmatrix}2 & 4\\-3 & -6\end{pmatrix}_{2\times 2}\begin{pmatrix}-2 & 4\\1 & -2\end{pmatrix}_{2\times 2}=\begin{pmatrix}0 & 0\\0 & 0\end{pmatrix}_{2\times 2}$$

矩阵的乘法

- 故矩阵乘法与普通数的乘法有所不同，一般地：
- ①不满足交换律，即$AB \neq BA$
- ②在$AB = 0$时，不能推出$A = 0$或$B = 0$，因而也不满足消去律.
- 特别，若矩阵A与B满足$AB = BA$，则称A与B可交换，此时A与B必为同阶方阵.
- 矩阵乘法满足结合律，分配律及与数乘的结合律.

矩阵乘法的实现

```
struct Mat{
    int m[N][N];
};
Mat Mul(Mat x,Mat y){
    Mat c;
    for(int i=1;i<=n;i++)
        for(int j=1;j<=n;j++)
            c.m[i][j]=0;
    for(int i=1;i<=n;i++)
        for(int j=1;j<=n;j++)
            for(int k=1;k<=n;k++)
                c.m[i][j]=c.m[i][j]%mod+x.m[i][k]*y.m[k][j]%mod;
    return c;
}
```

矩阵的幂

定义　设 A 是n 阶方阵，定义

$$A^k = \underbrace{AA\cdots A}_{k}$$

显然　$A^kA^l = A^{k+l}$,　　$(A^k)^l = A^{kl}$

思考：下列等式在什么时候成立？

$$\left.\begin{aligned}&(AB)^k = A^kB^k\\&(A+B)^2 = A^2+2AB+B^2\\&(A+B)(A-B) = A^2-B^2\end{aligned}\right\}$$

A、B可交换时成立

$$AB = BA$$

矩阵的转置

- 定义 把矩阵A 的行换成同序数的列得到的新矩阵，叫做 A 的转置矩阵，记作 A^{T} .

例 $A=\begin{pmatrix}1&2&2\\4&5&8\end{pmatrix}_{2\times 3}$, $A^{T}=\begin{pmatrix}1&4\\2&5\\2&8\end{pmatrix}_{3\times 2}$.

- 转置矩阵的运算性质

(1) $(A^{T})^{T}=A$;

(2) $(A+B)^{T}=A^{T}+B^{T}$;

(3) $(\lambda A)^{T}=\lambda A^{T}$;

(4) $(AB)^{T}=B^{T}A^{T}$.

一、矩阵与矩阵的基本运算

1.2 矩阵快速幂加速递推与DP

矩阵快速幂

```
Mat quick(Mat base,ll y){
    Mat ans;
    for(int i=1;i<=n;i++)
        ans.m[i][i]=1;
    while(y){
        if(y&1)
            ans=Mul(ans,base);
        base=Mul(base,base);
        y>>=1;
    }
    return ans;
}
```

矩阵快速幂加速递推 P1962 斐波那契数列

- 求出Fibonacci数列的第 n 项。
- $n \leq 2^{63}$

矩阵快速幂加速递推 P1962 斐波那契数列

- 求出Fibonacci数列的第 n 项。
- $n \le 2^{63}$
- 考虑递推过程，我们有

$$\begin{bmatrix} F_{n-1} & F_{n-2} \end{bmatrix} \times \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} F_n & F_{n-1} \end{bmatrix}$$

- 令$ans = [F_2 \quad F_1], base = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix},$
- 则答案为 $ans \times base^{n-2}$
- **矩阵快速幂加速问题关键在构造ans和$base$矩阵**

矩阵快速幂加速递推 P1707 刷题比赛

- 求出每个人的刷题数量。

1、nodgd 同学第 $k+2$ 天刷题数量

$$a_{k+2}=pa_{k+1}+qa_k+b_{k+1}+c_{k+1}+rk^2+tk+1$$

2、Ciocio 同学第 $k+2$ 天刷题数量

$$b_{k+2}=ub_{k+1}+vb_k+a_{k+1}+c_{k+1}+w^k$$

3、Nicole 同学第 $k+2$ 天刷题数量

$$c_{k+2}=xc_{k+1}+yc_k+a_{k+1}+b_{k+1}+z^k+k+2$$

(以上的字母 p,q,r,t,u,v,w,x,y,z 都是给定的常数，并保证是正整数)

矩阵快速幂加速递推 P1707 刷题比赛

- 求出每个人的刷题数量。

1、nodgd 同学第 $k+2$ 天刷题数量

$$a_{k+2}=pa_{k+1}+qa_k+b_{k+1}+c_{k+1}+rk^2+tk+1$$

2、Ciocio 同学第 $k+2$ 天刷题数量

$$b_{k+2}=ub_{k+1}+vb_k+a_{k+1}+c_{k+1}+w^k$$

3、Nicole 同学第 $k+2$ 天刷题数量

$$c_{k+2}=xc_{k+1}+yc_k+a_{k+1}+b_{k+1}+z^k+k+2$$

（以上的字母 p,q,r,t,u,v,w,x,y,z 都是给定的常数，并保证是正整数）

矩阵快速幂加速递推 P1707 刷题比赛

- 求出每个人的刷题数量。

$$a_{k+2} = pa_{k+1} + qa_k + b_{k+1} + c_{k+1} + rk^2 + tk + 1$$

$$b_{k+2} = ub_{k+1} + vb_k + a_{k+1} + c_{k+1} + w^k$$

$$c_{k+2} = xc_{k+1} + yc_k + a_{k+1} + b_{k+1} + z^k + k + 2$$

$$\begin{bmatrix} p & 1 & 1 & q & 0 & 0 & r & t & 1 & 0 & 0 \\ 1 & u & 1 & 0 & v & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & x & 0 & 0 & y & 0 & 1 & 2 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & w & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & z \end{bmatrix} \times \begin{bmatrix} a_{k+1} \\ b_{k+1} \\ c_{k+1} \\ a_k \\ b_k \\ c_k \\ k^2 \\ k \\ 1 \\ w^k \\ z^k \end{bmatrix} = \begin{bmatrix} a_{k+2} \\ b_{k+2} \\ c_{k+2} \\ a_{k+1} \\ b_{k+1} \\ c_{k+1} \\ (k+1)^2 \\ k+1 \\ 1 \\ w^{k+1} \\ z^{k+1} \end{bmatrix}$$

矩阵快速幂加速DP P4910 帕秋莉的手环

- 给定一个长度为 n 的环，每个位置可以填 1 或 2。相邻两个不能同时填 1，求方案数。
- $n \leq 10^{18}$

矩阵快速幂加速DP P4910 帕秋莉的手环

- 给定一个长度为 n 的环，每个位置可以填 1 或 2。相邻两个不能同时填 1，求方案数。
- $n \le 10^{18}$
- $f[i][2] = f[i-1][1] + f[i-1][2]$
- $f[i][1] = f[i-1][2]$
- 最后贡献需要根据1号位的情况分类讨论。
- 转移方式与前面的Fibonacci数列的转移相似

矩阵快速幂加速DP P3216 [HNOI2011] 数学作业

- 求将 $1\sim n$ 所有数字顺序连接后模m的值。
- $n \le 10^{18}$

矩阵快速幂加速DP P3216 [HNOI2011] 数学作业

- 求将 $1 \sim n$ 所有数字顺序连接后模m的值。
- $n \le 10^{18}$

$$dp[i] = \left[\left(dp[i-1] \times 10^{\lfloor 1+\lg i \rfloor} \bmod m\right) + i\right] \bmod m$$

- $ans = \begin{bmatrix} dp[i] \\ i+1 \\ 1 \end{bmatrix}$

- $base = \begin{bmatrix} 10^k & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$

一、矩阵与矩阵的基本运算

1.3 矩阵加速图上问题

P2233 [HNOI2002] 公交车路线

- 有8个车站A~H围成一圈，相邻的车站可以互相到达（比如A和H，B和C），求从A车站出发换n次车到达E车站有多少种方案。注意，到达E车站后将不会继续行动。

P2233 [HNOI2002] 公交车路线

- 考虑邻接矩阵
- $A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$
- 这个矩阵除了表示图的可达性以外，还表示 i 经过 1 条边到达 j 的方案数。

P2233 [HNOI2002] 公交车路线

- 接着我们考虑
- $A=\begin{bmatrix}0&1&0&0&0&0&0&1\\1&0&1&0&0&0&0&0\\0&1&0&1&0&0&0&0\\0&0&1&0&1&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&1&0&1&0\\0&0&0&0&0&1&0&1\\1&0&0&0&0&0&1&0\end{bmatrix}$ $A^2=\begin{bmatrix}2&0&1&0&0&0&1&0\\0&2&0&1&0&0&0&1\\1&0&2&0&1&0&0&0\\0&1&0&1&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&1&0&1\\1&0&0&0&1&0&2&0\\0&1&0&0&0&1&0&2\end{bmatrix}$
- 这个矩阵表示 i 经过 2 条边到达 j 的方案数。

P2233 [HNOI2002] 公交车路线

- 也可以用floyd的方式来理解
- $a_{i,j}^2=\sum_{k=1}^{n} a_{i,k}\cdot a_{k,j}$
- $a_{i,j}^3=\sum_{x=1}^{n} a_{i,x}\cdot\left(\sum_{y=1}^{n} a_{x,y}\cdot a_{y,j}\right)=\sum_{x=1}^{n}\sum_{y=1}^{n} a_{i,x}\cdot a_{x,y}\cdot a_{y,j}$
- 推广一下, A^n矩阵表示 i 经过 n 条边到达 j 的方案数。
- 于是对邻接矩阵进行矩阵快速幂即可。

P4159 [SCOI2009] 迷路

- 给定一个带边权的无向图，求从点1到点n，长度为t的路径数量。
- $n \leq 10, t \leq 10^9$,边权小于等于9

拆点：P4159 [SCOI2009] 迷路

- 给定一个带边权的无向图，求从点1到点n，长度为t的路径数量。
- $n \le 10, t \le 10^9$,边权小于等于9

可以拆点，将每个点拆做9个状态（1到9），第k个状态的i点用(i,k)表示，代表还要经过$i-1$个长度的边才能到达点i的状态，这样就好办了。

对于原图中的边(i,j,k)，$(i,0)$连向$(j,k-1)$（从点i出发时必须到达了点i，所以是$(i,0)$，在转移的时候默认边长为1，所以k要减一）
对于每个点i，它所有状态需要相互连边，(i,k)连向$(i,k-1)$（转移时走过一个长度的边）

P2151 [SDOI2009] HH去散步

- 给定一个无向图，求从点s到点t，长度为k的路径数量。不可以走过一条边后立即走这条边回来。

点边互换：P2151 [SDOI2009] HH去散步

- 给定一个无向图，求从点s到点t，长度为k的路径数量。不可以走过一条边后立即走这条边回来。
- 把所有有向边看做点（无向边转两条有向边），并连出一些边。**两条边代表的点连起来，当且仅当一条边的终点为另一条边的起点。**
- 所求为：s**射出的边在图中的所有结点到射向**t**的所有边在图中的所有结点之间的路径长度为**$\boldsymbol{k-1}$。
- 走一条边不能立即走回来，发现在点边互换的有向图中很好处理，只需要将反边代表的结点之间的边去掉就可以了。

P2579 [ZJOI2005] 沼泽鳄鱼

题意：给定一个n个点m条边的无向图，其中会有一些食人鱼按照$p1 \to p2 \to p1 \to \cdots$，或$p1 \to p2 \to p3 \to p1 \to \cdots$，或$p1 \to p2 \to p3 \to p4 \to p1 \to \cdots$的顺序进行行动（每次行动时间需要1单位时间），每条边需要花费1单位时间，求从s到t，长度为k，且到的每个点都没有食人鱼的路径数量。

数据范围：$1 \leqslant n \leqslant 50$，$1 \leqslant k \leqslant 2 \cdot 10^9$，食人鱼的数量不超过20。

P2579 [ZJOI2005] 沼泽鳄鱼

题意：给定一个n个点m条边的无向图，其中会有一些食人鱼按照$p1 \to p2 \to p1 \to \cdots$，或$p1 \to p2 \to p3 \to p1 \to \cdots$，或$p1 \to p2 \to p3 \to p4 \to p1 \to \cdots$的顺序进行行动（每次行动时间需要1单位时间），每条边需要花费1单位时间，求从s到t，长度为k，且到的每个点都没有食人鱼的路径数量。

数据范围：$1 \leqslant n \leqslant 50$，$1 \leqslant k \leqslant 2 \cdot 10^9$，食人鱼的数量不超过20。

- 食人鱼的运动周期最多为12,每个周期内会有一些位置不可达，即邻接矩阵不同。我们把前12个时刻得到的不同邻接矩阵相乘后做矩阵快速幂。剩余部分需要单独处理。

广义矩阵乘法 P2886 [USACO07NOV]Cow Relays G

- 给定一个带边权无向图，求从点s到点t，经过k条边的最短路。

广义矩阵乘法 P2886 [USACO07NOV]Cow Relays G

- 给定一个带边权无向图，求从点s到点t，经过k条边的最短路。
- 首先考虑做k次每次扩展一条边的floyd，单次的式子为：

$$dis'_{i,j} = \min_{k=1}^{n}\left(dis_{i,k} + g_{k,j}\right)$$

- 即：

$$c_{i,j} = \min_{k=1}^{n}\left(a_{i,k} + b_{k,j}\right)$$

- 矩阵乘法为:

$$c_{i,j} = \sum_{k=1}^{n}\left(a_{i,k} \cdot b_{k,j}\right)$$

- 只要$(\min, +)$与$(+,\times)$两个代数系统性质类似，这样的“矩阵乘法”也有结合律，即第二个运算对第一个运算也有分配律，那么我们就可以应用矩阵快速幂来加速这样的广义矩阵乘法。

CF576D Flights for Regular Customers

- 给定一个有向图，求从点1到点n的最短路。第 i 条边只有在已经走过 d_i 条边后才可通行。

CF576D Flights for Regular Customers

- 给定一个有向图，求从点1到点n的最短路。第 i 条边只有在已经走过 d_i 条边后才可通行。
- 首先对边排序，然后从小到大依次加入每条边。
- 考虑只有当前存在的边时，答案怎么求。
- 假设此时加入的边的 d 值为 t，首先要求的是经过恰好 t 条边时可以到达哪些点。
- 这个可以从加入上一条边时的答案递推过来，这个递推式可以矩阵加速。
- 求出恰好 t条边时可以到达哪些点后，对整个图进行一次 bfs 即可求出当前的答案。
- 这样时间复杂度是 $O(n^3\ mlogd)$ 的，无法通过。
- 需要再用bitset优化一下。

P6569 [NOI Online #3 提高组]魔法值

- 给定一个n个点，m条边的无向图（带点权），每一天，每个节点的点权会异或上与它相邻的所有点昨天的点权，q次询问，每次询问节点1在第k天的点权。

二进制拆分：P6569 [NOI Online #3 提高组]魔法值

- 给定一个n个点，m条边的无向图（带点权），每一天，每个节点的点权会异或上与它相邻的所有点昨天的点权，q次询问，每次询问节点1在第k天的点权。
- 首先应用广义矩阵乘法：

$$c_{i,j} = \bigoplus_{k=1}^{n} \left(a_{i,k} \times b_{k,j}\right)$$

- 实现时要对邻接矩阵 $B_{n,n}$ 做矩阵快速幂，时间复杂度为 $O(qn^3\log k)$，不能通过本题。
- 我们预处理所有2^p次幂的邻接矩阵，询问时对 k 二进制拆分。
- 原本是 $n \times n$ 的两个矩阵相乘，现在是 $1 \times n$ 与 $n \times n$ 的矩阵相乘。
- 时间复杂度降为$O(n^3\log k + qn^2\log k)$，可以通过本题。

练习 P6772 [NOI2020] 美食家

精灵王国共有 n 座城市，城市从 1 到 n 编号，其中城市 i 的美食能为小 W 提供 c_i 的愉悦值。精灵王国的城市通过 m 条**单向道路**连接，道路从 1 到 m 编号，其中道路 i 的起点为城市 u_i，终点为城市 v_i，沿它通行需要花费 w_i 天。也就是说，若小 W 在第 d 天从城市 u_i 沿道路 i 通行，那么他会在第 $d+w_i$ 天到达城市 v_i。

小 W 计划在精灵王国进行一场为期 T 天的旅行，更具体地：他会在第 0 天从城市 1 出发，经过 T 天的旅行，最终在**恰好第 T 天**回到城市 1 结束旅行。由于小 W 是一位美食家，每当他到达一座城市时（包括第 0 天和第 T 天的城市 1），他都会品尝该城市的美食并获得其所提供的愉悦值，若小 W 多次到达同一座城市，他将**获得多次愉悦值**。注意旅行途中小 W **不能在任何城市停留**，即当他到达一座城市且还未结束旅行时，他当天必须立即从该城市出发前往其他城市。

此外，精灵王国会在**不同**的时间举办 k 次美食节。具体来说，第 i 次美食节将于第 t_i 天在城市 x_i 举办，若小 W 第 t_i 天时恰好在城市 x_i，那么他在品尝城市 x_i 的美食时会**额外得到** y_i 的愉悦值。现在小 W 想请作为精灵王国接待使者的你帮他算出，他在旅行中能获得的愉悦值之和的**最大值**。

一、矩阵与矩阵的基本运算

1.4 动态DP：线段树维护矩阵乘法

动态DP（DDP）

- 动态DP是指一类可以修改参数，多次询问的DP问题。
- 我们借助矩阵+数据结构来维护。

Can you answer these queries III

- 给定一个长度为 n 的序列，你需要维护两种操作：
- 单点修改权值
- 查询一个区间的最大子段和
- $n, q \le 10^5$
- 考虑暴力
  - $f[i] = a[i] + \max\{f[i-1], 0\}$
  - $g[i] = \max\{g[i-1], f[i]\}$

Can you answer these queries III

- 由于加法对max有分配律，我们改写DP式
  - $f[i] = \max\{f[i-1] + a[i], a[i]\}$
  - $g[i] = \max\{g[i-1], f[i-1] + a[i], a[i]\}$
- 类似于定义于$(+,\times)$运算的矩阵乘法，我们可以定义$(\max,+)$运算的矩阵乘法
- $C_{i,j} = \max\{A_{i,k} + B_{k,j}\}$
- $\begin{bmatrix} a[i] & -\infty & a[i] \\ a[i] & 0 & a[i] \\ -\infty & -\infty & 0 \end{bmatrix} \begin{bmatrix} f[i-1] \\ g[i-1] \\ 0 \end{bmatrix} = \begin{bmatrix} f[i] \\ g[i] \\ 0 \end{bmatrix}$
- 线段树上每个点维护一个矩阵，查询即转化为查询区间矩阵积。

P4719 动态维护最大独立集

- 给定一个 n 个点的树，你需要维护两种操作：
- 单点修改点权
- 查询当前最大独立集（没有上司的舞会）
- $n, q \le 10^5$
- 考虑暴力
  - $f[x][0] = \sum \max(f[v][1], f[v][0])$
  - $f[x][1] = \sum f[v][0] + a[x]$

P4719 动态维护最大独立集

- 我们考虑用树剖来优化，设$g[x][0/1]$分别表示，除去重儿子的答案后对应的上述两个DP值。
- $f[x][0] = \max(g[x][0] + f[son][0], g[x][0] + f[son][1])$
- $f[x][1] = g[x][1] + f[son][0]$
- 对应矩阵为：
- $\begin{bmatrix} g[x][0] & g[x][0] \\ g[x][1] & -\infty \end{bmatrix} \begin{bmatrix} f[son][0] \\ f[son][1] \end{bmatrix} = \begin{bmatrix} f[x][0] \\ f[x][1] \end{bmatrix}$
- 我们修改对应点的矩阵，然后在链上做矩阵乘即可（线段树维护）。

一、矩阵与矩阵的基本运算

1.5 矩阵基本性质

P1224 [NOI2013] 向量内积

两个 d 维向量 $A=[a_1,a_2,\ldots,a_d]$ 与 $B=[b_1,b_2,\ldots,b_d]$ 的内积为其相对应维度的权值的乘积和，即：

$$(A,B)=\sum_{i=1}^{d}a_ib_i=a_1b_1+a_2b_2+\ldots+a_db_d$$

现有 n 个 d 维向量 $x_1,\ldots,x_n$，小喵喵想知道是否存在两个向量的内积为 k 的倍数。请帮助她解决这个问题。

- $n\le 10^5, d\le 30, k=2/3, x_{i,j}\le 10$

P1224 [NOI2013] 向量内积

- 可以发现本题 k 只有 2, 3，首先考虑 $k=2$ 时怎么处理。
- 我们将给出的向量构建成一个 $n \times d$ 的矩阵 A，可以发现本题实际上就是询问 $A \times A^T$是否存在一个非对角线位置的值是 k 的倍数。退一步想，如果我们可以确定某一个向量和一个向量的内积是 k 的倍数，那么这个时候暴力计算的复杂度是可以接受的。
- 我们构造一个矩阵 D，使得

$$D_{i,j} = \begin{cases} \langle \boldsymbol{\alpha}_i, \boldsymbol{\alpha}_j \rangle & i = j \\ 1 & \text{otherwise} \end{cases}$$

- 可以发现当且仅当 $A \times A^T = D$ 时不存在答案。

P1224 [NOI2013] 向量内积

- 我们构造一个随机 0/1 竖向量 W，可以发现对于 $A \times A^T \times W$ 与 $D \times W$ 来说，如果存在某一行满足其对应的值不想等，那么我们就可以确定这一行的该向量存在另一个向量可以使得其内积为 0。
- 根据矩阵乘法的结合律，我们可以计算 $A \times (A^T \times W)$，这样可以控制复杂度为 $O(nd)$。

P1224 [NOI2013] 向量内积

- 考虑在 $k = 3$ 时，不为倍数时的值可能为 1, 2，那么矩阵 D 的构造就是困难的。
- 进一步考虑可以发现 $1^2 \equiv 2^2 \equiv 1 \pmod 3$，所以我们可以构造点积的平方来判定。将长度为 d 的向量扩展为 d^2 的向量即可。

参考与引用内容

[矩阵加速图上问题题型总结 - 作业部落 Cmd Markdown 编辑阅读器 (zybuluo.com)]

[题解 CF576D 【Flights for Regular Customers】 - 洛谷专栏 (luogu.com.cn)]

[P3216 [HNOI2011] 数学作业 题解 - 洛谷专栏 (luogu.com.cn)]

[题解 P3232 【[HNOI2013]游走】 - 洛谷专栏 (luogu.com.cn)]

[题解——P3211 [HNOI2011]XOR和路径 - 洛谷专栏 (luogu.com.cn)]

[学习笔记 - BEST定理 | Lucky_Glass's Blog (luckyglass.github.io)]

[矩阵树定理&BEST定理学习笔记 - tzc_wk - 博客园 (cnblogs.com)]

ORIE 6334 Bridging Continuous and Discrete Optimization Sep 30, 2019 Lecture 8

[线性基学习笔记 | Menci's OI Blog]

Thanks