叉积的应用

卓亮

福州一中

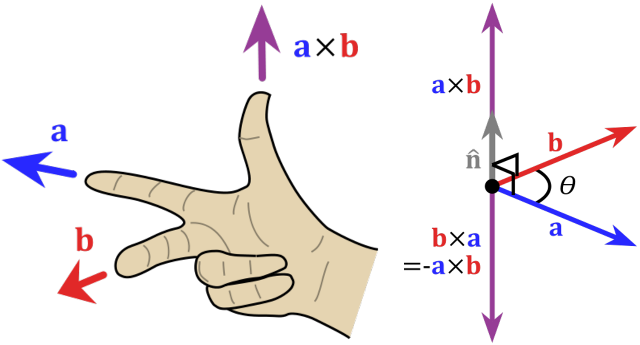
February 2, 2012

目录

什么是叉积

叉积，亦称向量积，是三维空间上两个向量的一种二元运算。两个向量$\vec{a}$与$\vec{b}$的叉积记作$\vec{a}\times\vec{b}$。

$$\vec{a}\times\vec{b}=|\vec{a}||\vec{b}|\sin\theta\vec{n}$$



计算

$\vec{i},\vec{j},\vec{k}$满足

$$\vec{i}\times\vec{j}=\vec{k},\vec{j}\times\vec{k}=\vec{i},\vec{k}\times\vec{i}=\vec{j}$$

类似的，

$$\vec{j}\times\vec{i}=-\vec{k},\vec{k}\times\vec{j}=-\vec{i},\vec{i}\times\vec{k}=-\vec{j}$$

$$\vec{i}\times\vec{i}=\vec{j}\times\vec{j}=\vec{k}\times\vec{k}=\vec{0}$$

若$\vec{a}=a_x\vec{i}+a_y\vec{j}+a_z\vec{k}$，$\vec{b}=b_x\vec{i}+b_y\vec{j}+b_z\vec{k}$，

$$\begin{aligned}
\vec{a}\times\vec{b} &= (a_x\vec{i}+a_y\vec{j}+a_z\vec{k})\times(b_x\vec{i}+b_y\vec{j}+b_z\vec{k}) \\
&= a_xb_x\vec{i}\times\vec{i}+a_xb_y\vec{i}\times\vec{j}+a_xb_z\vec{i}\times\vec{k} \\
& \quad a_yb_x\vec{j}\times\vec{i}+a_yb_y\vec{j}\times\vec{j}+a_yb_z\vec{j}\times\vec{k} \\
& \quad a_zb_x\vec{k}\times\vec{i}+a_zb_y\vec{k}\times\vec{j}+a_zb_z\vec{k}\times\vec{k} \\
&= (a_yb_z-a_zb_y)\vec{i}+(a_zb_x-a_xb_z)\vec{j}+(a_xb_y-a_yb_x)\vec{k}
\end{aligned}$$

计算

$\vec{i},\vec{j},\vec{k}$满足

$$\vec{i}\times\vec{j}=\vec{k},\vec{j}\times\vec{k}=\vec{i},\vec{k}\times\vec{i}=\vec{j}$$

类似的，

$$\vec{j}\times\vec{i}=-\vec{k},\vec{k}\times\vec{j}=-\vec{i},\vec{i}\times\vec{k}=-\vec{j}$$

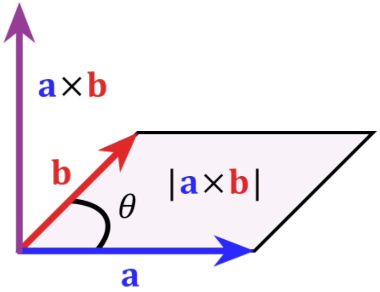
$$\vec{i}\times\vec{i}=\vec{j}\times\vec{j}=\vec{k}\times\vec{k}=\vec{0}$$

若$\vec{a}=a_x\vec{i}+a_y\vec{j}+a_z\vec{k}$，$\vec{b}=b_x\vec{i}+b_y\vec{j}+b_z\vec{k}$，

$$\vec{a}\times\vec{b}=\begin{vmatrix}\vec{i} & \vec{j} & \vec{k}\\ a_x & a_y & a_z\\ b_x & b_y & b_z\end{vmatrix}$$

性质

几何意义



性质

- $\vec{a} \times \vec{b} = -\vec{b} \times \vec{a}$

性质

- $\vec{a} \times \vec{b} = -\vec{b} \times \vec{a}$
- $\vec{a} \times (\vec{b} + \vec{c}) = (\vec{a} \times \vec{b}) + (\vec{a} \times \vec{c})$

性质

- $\vec{a} \times \vec{b} = -\vec{b} \times \vec{a}$
- $\vec{a} \times (\vec{b} + \vec{c}) = (\vec{a} \times \vec{b}) + (\vec{a} \times \vec{c})$
- $(r\vec{a}) \times \vec{b} = \vec{a} \times (r\vec{b}) = r(\vec{a} \times \vec{b})$

作用

在二维计算几何中，

- 多边形的面积

作用

在二维计算几何中，

- 多边形的面积
- 点与线的位置关系

题意简述

一个工厂制造产品，有N个流程。第i个流程的时间系数是T_i。有M个产品要制造，第i个产品的容易程度是F_i。一个产品j，在流程i所需时间为T_iF_j。流程顺序不可颠倒，产品也必须按给定的顺序制作。一旦一个流程完成，就交给下一个流程。此时，下一个流程必须是空闲的，不然就会出错。问完成所有产品需要的时间。[1]

数据满足$1 \le N \le 100000$，$1 \le M \le 100000$。

[1] COCI 2011/2012 Contest 3

参考解法

贪心：能早开始就早开始。

参考解法

贪心：能早开始就早开始。
朴素做法：依次处理每个产品，确定下一个产品的开始时间。

参考解法

贪心：能早开始就早开始。
朴素做法：依次处理每个产品，确定下一个产品的开始时间。
时间复杂度：$O(NM)$。

参考解法

贪心：能早开始就早开始。

改进做法：快速确定下一个产品的开始时间。

参考解法

贪心：能早开始就早开始。

改进做法：快速确定下一个产品的开始时间。
假设产品j的开始时间是b_j。对任一道工序i，都不能出错。

参考解法

贪心：能早开始就早开始。

改进做法：快速确定下一个产品的开始时间。
假设产品j的开始时间是b_j。对任一道工序i，都不能出错。

$$b_j + F_j \sum_{k=1}^{i} T_k \leq b_{j+1} + F_{j+1} \sum_{k=1}^{i-1} T_k \qquad (1)$$

参考解法

记

$$S_i = \sum_{k=1}^{i} T_k$$

式(1)等价于

$$b_{j+1} - b_j \geq F_j S_i - F_{j+1} S_{i-1} \tag{2}$$

参考解法

记

$$S_i = \sum_{k=1}^{i} T_k$$

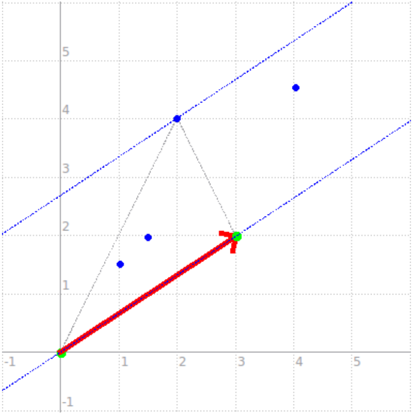
式(1)等价于

$$b_{j+1} - b_j \geq F_j S_i - F_{j+1} S_{i-1} \tag{2}$$

式(2)右端：向量(F_j, F_{j+1})与(S_{i-1}, S_i)的叉积。

参考解法

经过$(0,0),(F_j,F_{j+1}),(S_{i-1},S_i)$的三角形面积的两倍。



参考解法

对所有(S_{i-1}, S_i)求凸包。

参考解法

对所有(S_{i-1}, S_i)求凸包。
依次处理每个产品：在凸壳上二分，求与向量(F_j, F_{j+1})相切的位置。

参考解法

对所有(S_{i-1}, S_i)求凸包。
依次处理每个产品：在凸壳上二分，求与向量(F_j, F_{j+1})相切的位置。
时间复杂度：$O(N + M\log N)$。

题意简述

某人要组织一场比赛。她有N道备选题，这场比赛有K题。每位选手要做这K题。

她想，如果选手把这K题全做出来，选手会觉得这个比赛过于简单，很无趣。但如果选手只做出了很少的题目，又会觉得很难过。因此她想选这K道题，使得解出恰好$K-1$题的概率尽量大。

假设她已经进行了实验，得出了每道题被解出的概率。[2]

一共有不超过20组测试数据，对每组测试数据，$1 \le K \le N \le 36$。

[2] CodeChef CIELQUIZ

参考解法

含0的情况。

参考解法

不含0的情况。

选概率为$a_1, a_2, \ldots, a_K$的题目，恰好做出$K-1$题的概率是

$$(1-a_1)a_2a_3\ldots a_K + a_1(1-a_2)a_3a_4\ldots a_K + \cdots + a_1a_2\ldots a_{K-1}(1-a_K) \tag{3}$$

参考解法

不含0的情况。

选概率为$a_1, a_2, \ldots, a_K$的题目，恰好做出$K-1$题的概率是

$$(1-a_1)a_2a_3\ldots a_K+a_1(1-a_2)a_3a_4\ldots a_K+\cdots+a_1a_2\ldots a_{K-1}(1-a_K) \tag{3}$$

改写式(3)

$$\prod_{i=1}^{K} a_i \sum_{i=1}^{K} \frac{1-a_i}{a_i} \tag{4}$$

参考解法

朴素：枚举所有可能的情形。
时间复杂度：$O(\mathrm{C}_N^K)$。

参考解法

折半搜索。

参考解法

折半搜索。

预处理从p_1到$p_{\lfloor N/2 \rfloor}$中取i项的情况。

参考解法

折半搜索。

预处理从p_1到$p_{\lfloor N/2 \rfloor}$中取i项的情况。

如果集合$A = \{a_1, a_2, \ldots, a_K\}$，$B \subset A$，$C = A \backslash B$。那么

$$\prod_{a \in A} a \sum_{a \in A} \frac{1-a}{a} = \prod_{b \in B} b \prod_{c \in C} c \left(\sum_{b \in B} \frac{1-b}{b} + \sum_{c \in C} \frac{1-c}{c} \right) \quad (5)$$

参考解法

对集合X，我们设

$$S_X = \sum_{x\in X} \frac{1-x}{x}, T_X = \prod_{x\in X} x$$

式(5)改写成

$$T_B T_C (S_B + S_C) = S_B T_B T_C + S_C T_C T_B \tag{6}$$

参考解法

对集合X，我们设

$$S_X = \sum_{x \in X} \frac{1-x}{x}, T_X = \prod_{x \in X} x$$

式(5)改写成

$$T_B T_C (S_B + S_C) = S_B T_B T_C + S_C T_C T_B \tag{6}$$

令$U_X = S_X T_X$，则式(6)改写成

$$U_B T_C + U_C T_B = U_B T_C - U_C (-T_B) \tag{7}$$

式(7)右端：向量$(U_B, -T_B)$与(U_C, T_C)的叉积。

参考解法

折半搜索p_1到$p_{\lfloor N/2 \rfloor}$。

参考解法

折半搜索p_1到$p_{\lfloor N/2 \rfloor}$。
对取i项的情形求出凸壳。

参考解法

折半搜索p_1到$p_{\lfloor N/2 \rfloor}$。
对取i项的情形求出凸壳。
折半搜索$p_{\lfloor N/2 \rfloor + 1}$到$p_N$，在对应项的凸壳中二分，取极值。

参考解法

折半搜索p_1到$p_{\lfloor N/2 \rfloor}$。
对取i项的情形求出凸壳。
折半搜索$p_{\lfloor N/2 \rfloor+1}$到$p_N$，在对应项的凸壳中二分，取极值。
时间复杂度：$O(N \cdot 2^{\lceil N/2 \rceil})$。

参考解法

折半搜索p_1到$p_{\lfloor N/2 \rfloor}$。
对取i项的情形求出凸壳。
折半搜索$p_{\lfloor N/2 \rfloor + 1}$到$p_N$，在对应项的凸壳中二分，取极值。
时间复杂度：$O(N \cdot 2^{\lceil N/2 \rceil})$。
更优的做法：略。

总结

- 形式$ab - cd$。

总结

- 形式$ab - cd$。
- 2个独立的数据。

总结

- 形式$ab-cd$。
- 2个独立的数据。
- 列式，变形，凑配。

参考文献

Cross product from Wikipedia.

谢谢！