ctexh宏包是对CJK和xeCJK的进一步封装，包含了ctexart、ctexrep、ctexbook三大标准文档类. TEX users 两个或者以上的回车视为换行，或者使用

来进行换行. 反斜杠的表示方法：\LaTeX中英文标点的表达‘blah’ “blah” 各种横线：- – —

西欧中的重音：naïve cliće

不同符号¶ §

© ™

hello

斜体字

行内数学公式： $a^2 + b^2 = c^2$

行间数学公式：

$$a^2 + b^2 = c^2 \tag{1}$$

$$1 + 1 = 3 \tag{666}$$

$$1 + 1 = 4$$

$\lim_{n\to\infty} \sum_{k=1}^{n^2} \frac{1}{k^2} = \frac{\pi^2}{6}$.

$\frac{\partial x}{\partial y}$

Here I am gonna deal with my homework of database

$$\Pi_{person\_name}(\sigma_{city="Miami"}(employee))$$

$$\Pi_{person\_name}(\sigma_{salary>100{,}000}(works))$$

$$\Pi_{person\_name}(\sigma_{city="Miami" \wedge salary>100{,}000}(employee \bowtie works))$$

$$\Pi_{course\_id,ID}(\sigma_{dept\_name=Comp.Sci.}(takes))$$

$\Pi_{salary}(instructor) - \Pi_{instructor.salary}(\sigma_{instructor.salary<i.salary}(instructor \times \rho_i(instructor)))$

$\Pi_{ID,course\_id}(\sigma_{dept\_name=Comp.Sci.}(student)) \div \Pi_{course\_id}(\sigma_{dept\_name=Comp.Sci.}(course))$

$\Pi_{ID,titile}((instructor \bowtie teaches) \bowtie course) \div \Pi_{title}(\sigma_{title="D.B.S." \wedge title="O.S."}(course))$

$\Pi_{customer\_name,can\_spend}(g_{(limit-credit\_balance)\ as\ can\_spend}(credit\_info))$

$instructor \leftarrow instructor - \sigma_{dept\_name="Physics"}(instructor)$

$instructor \leftarrow \sigma_{salary<4{,}0000 \vee salary>60{,}000}(instructor)$

$\Pi_{title}(\sigma_{depr\_name="Comp.Sci." \wedge credits=3}(course))$

$\Pi_{ID}(students)-(\Pi_{ID}(students)-\Pi_{takes.ID}(\sigma_{name="Einstein"}(\Pi_{ID,t.ID}((takes \bowtie \rho_{teaches}(t))) \bowtie instructor)))$

$\Pi_{sec\_id,count(ID)}(sec\_idg_{count(ID) \wedge (year=2009 \wedge semester="Autumn")}(takes))$

$r \leftarrow \Pi_{sec\_id,count(ID)}(sec\_idg_{count(ID) \wedge (year=2009 \wedge semester="Autumn")}(takes))$

$\Pi_{ENROLLMENT}\rho_{q(SEC\_NAME,ENROLLMENT)}((\sigma_{r.count(ID)<s.count(ID)}(r \times \rho_s(r))))$

假设第i天做实验的次数为b_i,第1至第i天做的实验次数之和为a_i,即$a_i = b_1 + b_2 + \ldots + b_i$.

由题意,$p^\circ \{a_n\}(1 \leq n \leq 50)$各项值域为$[50, 75]$.

令$c_i = a_i + 24$,构造数列c_n,其各项的值域为$[75, 99]$.

而$a_1, a_2, \ldots, a_{50}, c_1, c_2, \ldots, c_{50}$共有100项,由定理2.1.1 可得:$\exists k, j \subseteq [1, 50]$且$k < j$,使得$a_k = c_j$.

$\therefore \sum_{n=k}^{j} b_n = 24$,命题证毕.

要使两个点所连线段中点坐标也是整数,只需要两个点的横坐标之和为偶数且纵坐标之和也为偶数.

而一个点的坐标的情况根据横纵坐标奇偶总共可以分为四种情况(o,e),(o,o),(e,e),(e,o)(用o表示奇数,e表示偶数,以下同).

由定理2.1.1,五个点中必有至少两点符合以上四种情况中的同一情况.

若有至少两点符合(o,e),可知其连线中点坐标也为整数.

其余情况同理.命题证毕.

6的一个完全剩余系为$\{0, 1, 2, 3, 4, 5\}$,由定理2.1.1,必存在a,b,使$(a-b) \mid 6$.同理,在剩余的5个数字中也必存在c,d,使$(c-d) \mid 4$.

$\therefore \exists a, b, c, d, \ddot{} (a-b)(c-d) \mid 24$.

$19 \times 4 + 1 = 77$

$a_n = n!$

$a_n = 8 - \frac{1}{2^n}$

$a_n = 3a_{n-1} + 3^n - 3^0$

$3^1 a_{n-1} = 3^2 a_{n-2} + 3^n - 3^1$

$\vdots$

$3^{n-1} a_1 = 3^n a_0 + 3^n - 3^{n-1}$

将上式累加可得:$a_n = (n - \frac{1}{2})3^n + \frac{1}{2}$.

$O(2^n)$

(1)由题意:$T(n) = 4 + T(n-3), (n \geq 4)$.

$T(2) = 4, T(1) = 2$.

$\therefore T(n) = 2n$.

(2)直觉上,要使时间最短,应该尽可能使烤架被充分使用.

比如在n=3时,按照以上算法,在汉堡只剩1个时,烤架未被完全占用,用时4min.而事实上是可以让烤架被完全占用的,用时为3min.因而该算法并不最优.

(3)一个最优算法为:

$n \geq 5$时,两个汉堡同时烤并翻面，然后对余下的n-2个汉堡递归应用同样的过程

$n \leq 3$时,若n=2则两个汉堡同时烤并翻面,若n=3,则先取1号和2号汉堡同时烤正面;然后将1号汉堡翻面,并取3号汉堡烤正面;最后将3号汉堡翻面,并取2号汉堡烤反面. $n_{*\Gamma Mn} + n_{n/Mn} = n_{-MnpW}$

$\sigma_{dept\_name=Comp.Sci.}(student \bowtie takes)$

$\Pi_{ID}(takes \div \Pi_{course\_id}(\sigma_{dept\_name="Comp.Sci."}(course)))$

先考虑安排总坐在前排和后排的同学,总共$A_8^5 \times A_8^4$种坐法,剩下7位同学随意安排在剩余位置.

$\therefore$共有$A_8^5 \times A_8^4 \times A_7^6$种坐法.

从线排列考虑.所有线排列共有15!种.

而A与B相邻对应的线排列共$(2 \times 14! - 2 \times 13!)$种,A与C相邻对应的线排列数也与之相同.

A与B、C同时相邻的线排列共$2 \times (13! + 4 \times 12!)$种.

综上,排座方法总共有$\frac{15! - 2 \times (2 \times 14! - 2 \times 13!) + 13! + 4 \times 12!}{15}$种.

$M - \{a\}$有$\frac{11!}{2! \times 4! \times 5!}$种11-排列数.

$M - \{b\}$有$\frac{11!}{3! \times 3! \times 5!}$种11-排列数.

$M - \{b\}$有$\frac{11!}{3! \times 4! \times 4!}$种11-排列数.

$\therefore$ 多重集M的11-排列数为:$\frac{11!}{2! \times 4! \times 5!} + \frac{11!}{3! \times 3! \times 5!} + \frac{11!}{3! \times 4! \times 4!}$.

$\binom{17}{10} - \binom{17}{12}$

$\frac{4}{9}$

$cost_{吃加速} - cost_{吃最近的食物} < valve1$,则吃加速

$cost_{吃斜走} - cost_{吃最近的食物} < valve2$,则吃斜走

$cost_{吃第二个斜走} - cost_{吃最近的食物} < valve3$,则吃第二个斜走

$cost_{我吃该食物} - cost_{对方吃该食物} > valve4$,则我放弃该食物,转而寻找下个食物

b^{depth}

$\sum_{k=1}^{n} k^2 \ B(n,k)$