HIT CS&E

第3章 球和箱子模型

骆吉洲
计算机科学与技术学院

HIT CS&E

提要

HIT CS&E

参考文献

- 《概率与计算》
  - 第5章
- 《Randomized Algorithms》
  - 第8章

HIT CS&E

3.1 与硬币投掷相关的三种分布

3.1.1 Bernoulli实验、几何分布和二项分布
3.1.2 桶排序及其时间复杂度分析
3.1.3 跳表及其操作复杂度分析

自觉复习

HIT CS&E

3.1.1 Bernoulli实验，几何分布和二项分布

课下自觉复习

HIT CS&E

硬币投掷的相关概率术语

投掷一枚有偏硬币

- 一次投掷结果的分布
  两点分布(Bernoulli trial)
- 首次头面向上所需的投掷次数
  几何分布(Geometric)
- *n*次投掷中头面向上的总次数
  二项分布 (Binomial)

HIT CS&E

Bernoulli实验

投掷一枚有偏硬币一次，实验结果的分布

X—表示头面向上还是背面向上

X=1—头面向上　　X=0—背面向上

$\Pr[X=1] = p$　　$\Pr[X=0] = 1-p$

随机变量X称为Bernoulli实验，X的分布称为两点分布

X=1称为实验成功，X=0称为实验失败

$E[X] = p$　　$\mathrm{Var}[X]=p(1-p)$

HIT CS&E

几何分布

实验次数

- 独立同分布重复Bernoulli实验直到实验成功

  i.i.d (independent and identical distribution)
  前后实验无关联，使用同样的硬币

- X—实验次数或者投掷次数

  $\Pr[X=k] = (1-p)^{k-1}p$

- X服从参数为p的几何分布

HIT CS&E

X服从几何分布　$\Pr[X=k] = (1-p)^{k-1}p$

$$\Pr[X=n+k \mid X>k] = \frac{\Pr[X=n+k \wedge X>k]}{\Pr[X>k]}$$

$$= \frac{\Pr[X=n+k]}{\Pr[X>k]}$$

$$= \frac{(1-p)^{n+k-1}p}{\sum_{i=k+1}^{+\infty}(1-p)^{i-1}p}$$

$$= \frac{(1-p)^{n+k-1}p}{p(1-p)^k\sum_{i=0}^{+\infty}(1-p)^{i}} = \frac{(1-p)^{n+k-1}p}{(1-p)^k}$$

$$=(1-p)^{n-1}p$$

$$=\Pr[X=n]$$

无后效性！

HIT CS&E

X服从几何分布　$\Pr[X=k] = (1-p)^{k-1}p$

$$E[X] = \sum_{k=1}^{+\infty} k\cdot\Pr[X=k]$$

$$= \sum_{k=1}^{+\infty} k\cdot p(1-p)^{k-1}$$

$$= p\sum_{k=1}^{+\infty} k\cdot (1-p)^{k-1}$$

$$= p\frac{1}{[1-(1-p)]^2}$$

$$= \frac{1}{p}$$

$$\frac{1}{1-x}=1+x+x^2+\ldots+x^n+\ldots$$

$$\frac{1}{(1-x)^2}=\left(\frac{1}{1-x}\right)'=1+2x+3x^2+\ldots+nx^{n-1}+\ldots$$

HIT CS&E

X服从几何分布　$\Pr[X=k] = (1-p)^{k-1}p$

$$\mathrm{Var}[X] = \sum_{k=1}^{+\infty}(k-E[X])^2\cdot\Pr[X=k]$$

$$= \sum_{k=1}^{+\infty}\left(k-\frac{1}{p}\right)^2\cdot p(1-p)^{k-1}$$

$$= \sum_{k=1}^{+\infty}k(k-1)\cdot p(1-p)^{k-1} + \sum_{k=1}^{+\infty}k\cdot p(1-p)^{k-1} - 2\sum_{k=1}^{+\infty}k\cdot(1-p)^{k-1} + \frac{1}{p}\sum_{k=1}^{+\infty}(1-p)^{k-1}$$

$$= p(1-p)\sum_{k=1}^{+\infty}k(k-1)\cdot(1-p)^{k-2} + (p-2)\sum_{k=1}^{+\infty}k\cdot(1-p)^{k-1} + \frac{1}{p}\sum_{k=1}^{+\infty}(1-p)^{k-1}$$

$$= p(1-p)\frac{2}{[1-(1-p)]^3} + (p-2)\frac{1}{[1-(1-p)]^2} + \frac{1}{p^2}$$

$$= \frac{1-p}{p^2}$$

$$\frac{1}{1-x}=1+x+x^2+\ldots+x^n+\ldots$$

$$\frac{1}{(1-x)^2}=1+2x+3x^2+\ldots+nx^{n-1}+\ldots$$

$$\frac{2}{(1-x)^3}=1\cdot2+2\cdot3x+\ldots+n(n-1)x^{n-2}+\ldots$$

HIT CS&E

二项分布

成功实验的总次数

- 独立同分布重复Bernoulli实验n次
- X—成功实验的总次数

  $\Pr[X=k] =\binom{n}{k}(1-p)^{n-k}p^k$

- X服从参数为p,n的二项分布

HIT CS&E

二项分布

二项分布 $\Pr[X=k] = \binom{n}{k}(1-p)^{n-k}p^k$

X_i=1表示第i次实验成功, 否则X_i=0

$$X = \sum_{i=1}^{n} X_i$$

$$E[X] = \sum_{i=1}^{n} E[X_i] = pn$$ 期望线性性质

$$Var[X] = \sum_{i=1}^{n} Var[X_i] = p(1-p)n$$ 独立性

HIT CS&E

3.1.2 桶排序及其时间复杂度分析

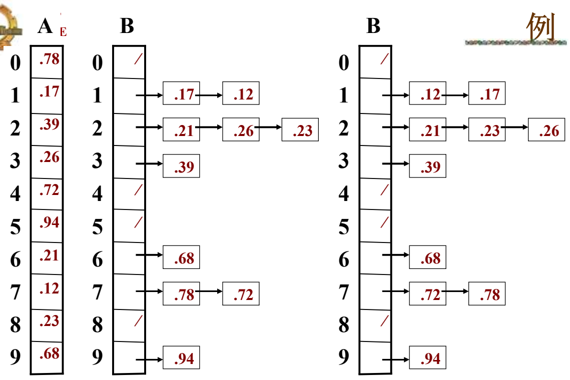
HIT CS&E

桶排序

- 基本思想
  - 假设所有输入值均匀等可能地取自[0,1);
  - 初始化n个空桶，编号介于0到n-1之间;
  - 扫描输入，将数值$A[i]$放入编号为$\lfloor nA[i] \rfloor$的桶中;
  - 将各个桶内的数据各自排序
  - 依编号递增顺序输出各个桶内的数据
- 需要一系列桶,需要排序的值变换为桶的索引
  - 不需要比较操作

HIT CS&E

例



直观上，只要每个桶的数据不多，总时间可以是线性

HIT CS&E

桶排序算法

算法BucketSort(A)

Input:数组A[0:n-1]，0≤A[i]<1

Output: 排序后的数组A

1. for $j \leftarrow 0$ to n-1 do // 初始化 n个桶
2. $B[j] \leftarrow$ NULL;
3. for $i \leftarrow 0$ to n-1 do
4. 将元素$A[i]$插入桶 $B[\lfloor nA[i] \rfloor]$中 //链表维护
5. for $i \leftarrow 0$ to n-1 do
6. 用InsertionSort排序桶$B[i]$内的数据
7. 依编号递增顺序将各个桶内的数据回填到A中

HIT CS&E

时间复杂度分析

散列过程需要$O(n)$时间将n个数据项散列到桶中

散列过程可以视为将n个球投入n个箱子中

$X_0,\ldots,X_{n-1}$ —各个桶中数据项的个数

X_i —服从参数为n,1/n的二项分布，各X_i不独立

$E[X_i] = pn = 1$

$Var[X_i] = p(1-p)n = 1-1/n$

$Var[X_i] = E[X_i^2]-(E[X_i])^2$

$E[X_i^2] = 2-1/n$

$E[T(n)] = O(n)$

$E[\sum_{i=0}^{n-1} X_i^2/2] = \sum_{i=0}^{n-1} E[X_i^2/2] = (2n-1)/2 = n-1/2$ 期望的线性性质

InsertSort排序的最坏时间复杂度为$n^2/2=O(n^2)$

散列完成之后，桶内排序总时间的期望不超过n-1/2

收集排序结果的时间为$O(n)$

HIT CS&E

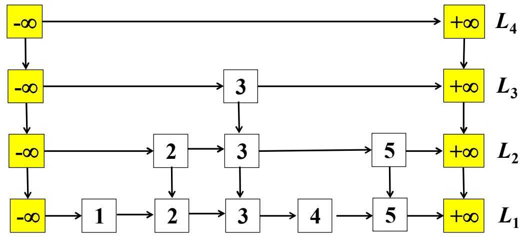
3.1.3 跳表及其操作复杂度分析

HIT CS&E

跳表(Skip Table)

在全序集$S=\{-\infty<x_1<x_2<...<x_n<+\infty\}$支持查找、插入、删除

- 分层有序链表结构，最大层数记为r
- $L_{i+1}\subseteq L_i$, $L_1=S$， $L_r=\{-\infty,+\infty\}$
- $\forall x\in L_{i+1}\subseteq L_i$,则从$x\in L_{i+1}$有指针指向$x\in L_i$

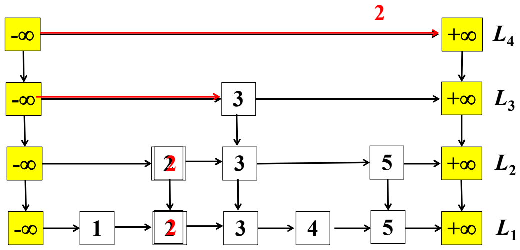


HIT CS&E

跳表中的元素查找

Find(x)

- 从L_r开始逐层用顺序比较定位x所在区间直到L_i
- “查找区间”的平均长度记为E[Interval]
- 最坏时间复杂度为$r\cdot$ E[Interval]

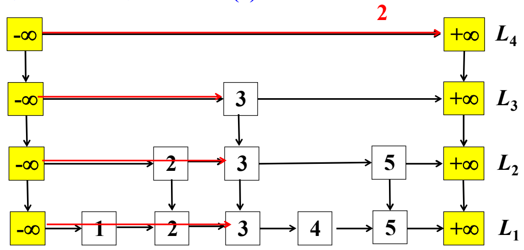


HIT CS&E

跳表中的元素删除

Delete(x)

- 从L_r开始逐层用顺序比较定位x所在区间直到
- 若在L_i的某个区间中找到x，则沿x的层间指针删除x
- 最坏时间复杂度与Find(x)相同



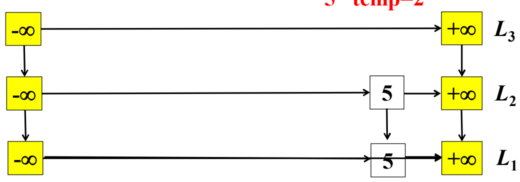
HIT CS&E

跳表中的建立和插入操作

Insert(x)

- 以参数p=1/2产生符合几何分布的随机数temp
- 若temp>r,
  - 初始化L_{temp+1}
  - 初始化L_{r+1} L_{r+2} …L_{temp}并将x插入其中
- 在L_r ,…,L_1中定位x所在区间并将x插入L_i(i≤temp)
- 若$temp>r$,则更新r为$temp+1$

5 temp=2



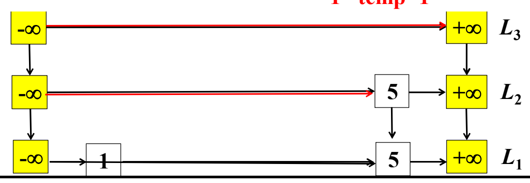
HIT CS&E

跳表中的建立和插入操作(续)

Insert(x)

- 以参数p=1/2产生符合几何分布的随机数temp
- 若temp>r,
  - 初始化L_{temp+1}
  - 初始化L_{r+1} L_{r+2} …L_{temp}并将x插入其中
- 在L_r ,…,L_1中定位x所在区间并将x插入L_i(i≤temp)
- 若$temp>r$,则更新r为$temp+1$

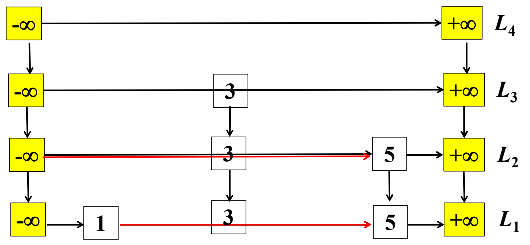
1 temp=1



HIT CS&E

跳表中的建立和插入操作(续)

3 temp=3



HIT CS&E

空间复杂度

$e_i\in S$的层数X_i服从p=1/2的几何分布

所有元素的总存储次数为 $X=\sum_{i=1}^{n}(X_i+1)$

每个元素的每次存储需要$O(1)$空间，$E(X)$是空间复杂度

$E[X] = n+\sum_{i=1}^{n}E[X_i]$

$=n+n/p$ $\qquad E[X_i]=1/p$

$=3n$ $\qquad p=1/2$

跳表的空间复杂度为$O(n)$

HIT CS&E

最大层数

$e_i\in S$的层数X_i服从p=1/2的几何分布

跳表的层数$r=1+\max_i X_i$

$\Pr[X_i\le k] = p+p(1-p)+\ldots+p(1-p)^{k-1}= 1-(1-p)^k = 1-2^{-k}$

$\Pr[\max_i X_i\le k] = \Pr[X_1\le k\wedge\ldots\wedge X_n\le k] =(1-2^{-k})^n$ (独立性)

$\Pr[\max_i X_i\le k-1] = \Pr[X_1\le k-1\wedge\ldots\wedge X_n\le k-1] =(1-2^{-(k-1)})^n$

$\Pr[\max_i X_i=k] = \Pr[\wedge_i X_n\le k] - \Pr[\wedge_i X_n\le k-1] =(1-2^{-k})^n-(1-2^{-(k-1)})^n$

$E[\max_i X_i] = \sum_k k\cdot\Pr[\max_i X_i=k] = (1-2^{-1})^n - (1-2^{0})^n$
$+ 2(1-2^{-2})^n - 2(1-2^{-1})^n$
$+ 3(1-2^{-3})^n - 3(1-2^{-2})^n+\ldots$

$= n(1-2^{-n})^n - (1-2^{0})^n - (1-2^{-1})^n - \ldots - (1-2^{-n})^n\}$

$=O(\log n)$

跳表的期望层数$E[r]=O(\log n)$

HIT CS&E

最大层数

$e_i\in S$的层数X_i服从p=1/2的几何分布

$\Pr[X_i>t] = \sum_{x=t+1}^{+\infty}\Pr[X_i=x] = \sum_{x=t+1}^{+\infty}(1-p)^{x-1}p = (1-p)^t$

$\Pr[\max_{i=1}^{n} X_i>t] = \Pr[(X_1>t)\vee(X_2>t)\vee\ldots\vee(X_n>t)]$

$\le n\Pr[X_1>t]$ 合并界+对称性

$\le n(1-p)^t$

$\le n(1-p)^t$

$= \frac{n}{2^t}$ $\qquad p=1/2$

跳表的层数$r=1+\max_i X_i$

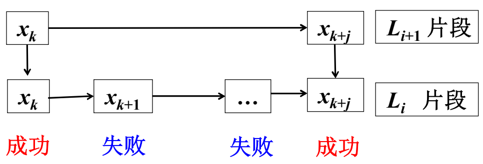
$\Pr[r>a\log n] \le \frac{n}{2^{a\log n}} = 1/n^{a-1}$

跳表的层数高概率取值$O(\log n)$

HIT CS&E

区间间隔

L_i的每个元素以概率1/2出现于L_{i+1}



$E[\text{ Interval }] = E[j] = 1/p = 2$

跳表中区间的期望长度是2

应用到每种操作上

- 平均处理$E[r]=O(\log n)$层
- 每层平均进行E[Interval] =2次比较

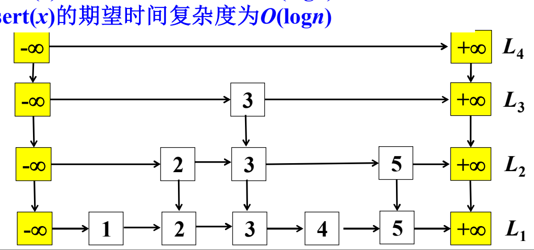
分析结论

应用到每种操作上

- 平均处理$E[r]=O(\log n)$层
- 每层平均进行E[Interval] =2次比较

操作的时间复杂度

- Find(x)的期望时间复杂度为$O(\log n)$
- Delete(x)的期望时间复杂度为$O(\log n)$
- Insert(x)的期望时间复杂度为$O(\log n)$



HIT CS&E

3.2 球与箱子模型

HIT CS&E

3.2.1 模型概述

HIT CS&E

球和箱子模型

m个球

均匀独立地将球投入箱子

n个箱子

很多分析工具都源于球和箱子模型
- 生日悖论
- 赠券收集
- 负载均衡问题
- ……

HIT CS&E

随机函数

球和箱子模型

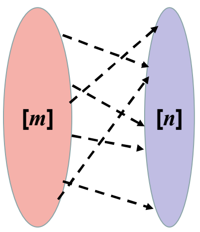
Pr[每种投掷结果]= $\underbrace{\frac{1}{n}\cdot\frac{1}{n}\cdot\dots\cdot\frac{1}{n}}_{m\text{个}} = \frac{1}{n^m}$

每个球可投入n个箱子中的任何一个

共有m个球

HIT CS&E

随机函数



均匀随机函数

球和箱子模型

Pr[每种投掷结果]= $\underbrace{\frac{1}{n}\cdot\frac{1}{n}\cdot\dots\cdot\frac{1}{n}}_{m\text{个}} = \frac{1}{n^m}$

随机函数

Pr[每个函数]= $\frac{1}{|[m]\to[n]|} = \frac{1}{n^m}$

单射	生日悖论
满射	赠券收集
原像	最大负载

HIT CS&E

4.2.2 生日悖论

HIT CS&E

生日悖论

悖论
(1) 引发某种矛盾的结论
(2) 与直觉相矛盾的现象

生日悖论
- 假设：所有人的生日均匀独立地分布
- 直觉：多大的人群才能确保有相同生日的人呢？
- 事实：任意随机选出m(>57)人，则其中有两人具有相同生日的概率大于99%

用球和箱子模型思考
将m个球放入n个箱子
事件$\mathcal{E}$：不存在含有2个球的箱子

HIT CS&E

将m个球放入n个箱子
事件$\mathcal{E}$：不存在含有2个球的箱子

事件$\mathcal{E}$：随机函数f:[m]→[n]是单射

$$\mathrm{Pr}[\mathcal{E}] = \frac{\|[m]\xrightarrow{1\text{-}1}[n]\|}{\|[m]\to[n]\|} = \frac{P_n^m}{n^m} = \frac{n\cdot(n-1)\cdot\ldots\cdot(n-m+1)}{n^m} = \prod_{k=0}^{m-1}\left(1-\frac{k}{n}\right)$$

HIT CS&E

将m个球放入n个箱子
事件$\mathcal{E}$：不存在含有2个球的箱子

$$\mathrm{Pr}[\mathcal{E}] = \prod_{k=0}^{m-1}\left(1-\frac{k}{n}\right)$$

对任意$\mathcal{E}_1,\mathcal{E}_2,\ldots,\mathcal{E}_n$有：$\Pr\left[\bigwedge_{i=1}^{n}\mathcal{E}_i\right] = \prod_{k=1}^{n}\Pr\left[\mathcal{E}_k \mid \bigwedge_{i<k}\mathcal{E}_i\right]$

Pr[没有两个球被投入同一个箱子]
$=\prod_k$Pr[第k+1个球投入空箱子 | 前k个球没有两个投入同一箱子]

HIT CS&E

将m个球放入n个箱子
事件$\mathcal{E}$：不存在含有2个球的箱子

$$\mathrm{Pr}[\mathcal{E}] = \prod_{k=0}^{m-1}\left(1-\frac{k}{n}\right)$$

由泰勒展开可知：$e^{-k/n} \approx 1-\frac{k}{n}$

$$\begin{aligned}\mathrm{Pr}[\mathcal{E}] &= \prod_{k=0}^{m-1}\left(1-\frac{k}{n}\right)\\ &\approx \prod_{k=0}^{m-1} e^{-k/n}\\ &= \exp\left(-\sum_{k=0}^{m-1}\frac{k}{n}\right)\\ &= e^{-m(m-1)/2n}\\ &\approx e^{-m^2/2n}\end{aligned}$$

HIT CS&E

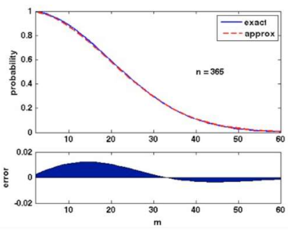
将m个球放入n个箱子
事件$\mathcal{E}$：不存在含有2个球的箱子

$$\mathrm{Pr}[\mathcal{E}] \approx e^{-m^2/2n}$$

当$m=\sqrt{2n\ln\frac{1}{\epsilon}}$,

$\mathrm{Pr}[\mathcal{E}] \approx \varepsilon$

在生日悖论中，
n=365, ε=0.01, m≈57

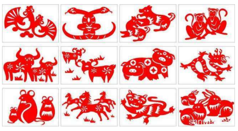


HIT CS&E

3.2.2 赠券收集

赠券收集与任务完成时间

“香香瓜子”中的赠券



集齐n种赠券购买m件

确保n个箱子均不空
需要投入m个球

每种赠券在每包瓜子中出现的概率相等

要集齐n种赠券，需要买多少包瓜子？

确保n个箱子均不空
需要投入X个球

$X=\sum_{i=1}^{n} X_i$

X_i: 已有i-1个箱子非空
空箱子减少一个
需要再投入的球数

X_i: 服从几何分布

$p_i=1-\frac{i-1}{n}$

球落入每个箱子的概率均为$1/n$

$E[X_i]=\frac{1}{p_i}=\frac{n}{n-i+1}$

箱子

非空箱子
i-1个

空箱子
n-i+1个

确保n个箱子均不空
需要投入X个球

X_i: 已有i-1个箱子非空
空箱子减少一个
需要再投入的球数

$X=\sum_{i=1}^{n} X_i$

$p_i=1-\frac{i-1}{n}$

$E[X_i]=\frac{1}{p_i}=\frac{n}{n-i+1}$

$$
\begin{aligned}
E[X] &= \sum_{i=1}^{n} E[X_i] \\
&= \sum_{i=1}^{n} \frac{n}{n-i+1} \\
&= n\sum_{i=1}^{n} \frac{1}{i} \\
&= nH_n
\end{aligned}
$$

期望的线性性质

从期望角度看，平均需要投$n\ln n+O(n)$个球就能确保无空箱子

确保n个箱子均不空
需要投入X个球

定理：$\Pr[X \ge n\ln n+cn] < e^{-c}$
对任意c>0成立

证明：Y_i—投入$n\ln n+cn$个球之后，第i个箱子是空箱子

$\Pr[Y_i] = (1-\frac{1}{n})^{n(\ln n+c)} < e^{-(\ln n+c)} = e^{-c}/n$

Pr[投入$n\ln n+cn$个球之后仍有空箱子]

$=\Pr[Y_1 \vee Y_2 \vee Y_3 \vee \ldots \vee Y_n]$

$\le \Pr[Y_1]+\Pr[Y_2]+\ldots+\Pr[Y_n]$ The Union Bound

$< e^{-c}$

3.2.3 负载均衡问题

负载均衡问题

m个球

均匀独立地将球投入箱子

n个箱子

X_1 X_2 X_3 ...... X_{n-1} X_n

所有箱子中，球最多的那个箱子有多少球？

$\max_{1\le i\le n} X_i$

HIT CS&E

m个球 $X_1\ X_2\ X_3\ \ldots\ldots X_{n-1}\ X_n$

n箱子 各个箱子的负载

$\max\limits_{1\le i\le n} \mathrm{E}[X_i] = ?$

$\sum_{i=1}^{n} X_i = m \Rightarrow \sum_{i=1}^{n} \mathrm{E}[X_i] = \mathrm{E}[\sum_{i=1}^{n} X_i] = m$

对称性
箱子无差别 $\Rightarrow \mathrm{E}[X_1] = \mathrm{E}[X_2] = \ldots = \mathrm{E}[X_n]$

$\max\limits_{1\le i\le n} \mathrm{E}[X_i] = m/n$ 毫无用处

HIT CS&E

占用问题-情形1(m=n)

n个球 $X_1\ X_2\ X_3\ \ldots\ldots X_{n-1}\ X_n$

n箱子 各个箱子的负载

$\max\limits_{1\le i\le n} X_i = ?$

定理：若m=n, 则 $\max\limits_{1\le i\le n} X_i = O(\frac{\ln n}{\ln\ln n})$高概率地成立

高概率成立(with high probability 或w.h.p.)

$\Pr = 1 - O(\frac{1}{n^c})$ or $\Pr = 1 - o(1)$

HIT CS&E

情形1的证明

n个球 $X_1\ X_2\ X_3\ \ldots\ldots X_{n-1}\ X_n$

n箱子 各个箱子的负载

$\Pr[X_i \ge M] \le \Pr[$ n个球中有M个球全部落入第i个箱子 $]$

$\binom{n}{M}$ $\quad\left(\frac{1}{n}\right)^M$

$$= \binom{n}{M}\left(\frac{1}{n}\right)^M = \frac{n(n-1)(n-2)\ldots(n-M+1)}{M!\,n^M}$$

$$= \frac{1}{M!}\prod_{k=0}^{M-1}(1-\frac{k}{n})$$

$$\le \frac{1}{M!}$$

$$\le (\frac{e}{M})^M$$

Striling公式

HIT CS&E

情形1的证明(续)

n个球 $X_1\ X_2\ X_3\ \ldots\ldots X_{n-1}\ X_n$

n箱子 各个箱子的负载

$\Pr[X_j \ge M] \le (\frac{e}{M})^M$

$$\Pr[\max_{1\le i\le n} X_i \ge M] = \Pr[\exists j: X_j \ge M]$$

$$= \Pr[\ (X_1 \ge M) \vee (X_2 \ge M) \vee \ldots \vee (X_n \ge M)\ ]$$

$$\le \sum_{j=1}^{n} \Pr[\ (X_j \ge M)]$$

The Union Bound

$$\le n(\frac{e}{M})^M$$

令 $n(\frac{e}{M})^M = \frac{1}{n}$，解得 $M \approx \frac{3\ln n}{\ln\ln n}$

HIT CS&E

占用问题-情形1(m=$\Theta(n)$)

$\Theta(n)$个球 $X_1\ X_2\ X_3\ \ldots\ldots X_{n-1}\ X_n$

n箱子 各个箱子的负载

$\max\limits_{1\le i\le n} X_i = ?$

定理：若m=$\Theta(n)$, 则 $\max\limits_{1\le i\le n} X_i = O(\frac{\ln n}{\ln\ln n})$高概率地成立

证明概要：通过计数证明$X_i \ge M$的概率
运用Union Bound

HIT CS&E

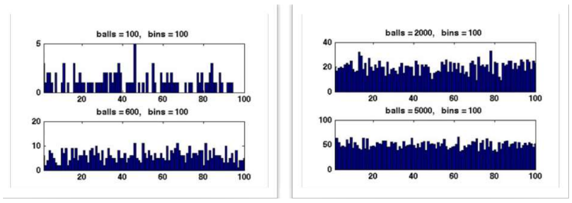
占用问题-情形2(m=$\Omega(n\log n)$)

$\Omega(n\log n)$个球 $X_1\ X_2\ X_3\ \ldots\ldots X_{n-1}\ X_n$

n箱子 各个箱子的负载

$\max\limits_{1\le i\le n} X_i = ?$

定理：若m=$\Omega(n\log n)$, 则 $\max\limits_{1\le i\le n} X_i = O(\frac{m}{n})$高概率地成立



HIT CS&E

球和箱子模型(总结)

将m个球均匀随机投入n个箱子

随机函数f:[m]→[n]

单射	生日悖论
满射	赠券收集
原像	负载问题

为确保f是单射，至多投入m个球

$$m = \Theta(\sqrt{n})$$

为确保f是满射，至少投入m个球

$$m = n \ln n + O(n)$$

最大负载max $f^{-1}(x)$是

$$\begin{cases} O(\frac{\ln n}{\ln \ln n}) & \text{for } m = \Theta(n), \\ O(\frac{m}{n}) & \text{for } m = \Omega(n \ln n) \end{cases}$$

HIT CS&E

实验题目

编写程序完成随机实验

- 验证生日悖论结论的有效性
- 验证赠券收集的期望时间的有效性
- 验证占用问题的最大负载的有效性

HIT CS&E

3.3 生日悖论的应用

3.3.1 生日攻击
3.3.2 Leader选举

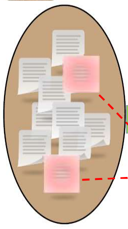
自学

HIT CS&E

3.3.1 生日攻击

HIT CS&E

应用1:生日攻击(Birthday Attack)



加密哈希函数

h → [n]

实例：MD5,SHA-1,SHA-2

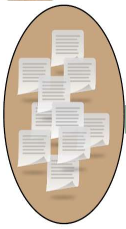
h是确定型且可以被高效计算

- 单向的
  难以由$h(x)$计算x
- 弱防冲突性
  给定x,难以找到y使得$h(x)=h(y)$
- 强防冲突性
  难以找出<x,y>使得$h(x)=h(y)$

冲突对

HIT CS&E

生日攻击(续)



加密哈希函数

h → [n]

生日攻击
均匀独立地随机选取m个文档
$X_1,X_2,\ldots,X_m$

假设：h是均匀的
Can Be Removed
$|h^{-1}(\cdot)|$相等

m个球被均匀随机地投入n个箱子
$h(X_1),h(X_2),\ldots,h(X_m)\in[n]$

只要$m=(2\ln 2 \cdot n)^{1/2}$
Pr[发生冲突] ≥ 1/2

3.3.2 Leader选举

应用2:环上的Leader选举

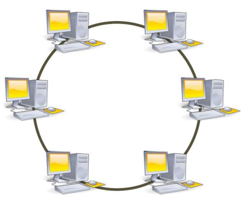
Leader选举问题
- 从一组处理器中选出一个处理器作为Leader
- 所有处理器的初始配置均相同(运行相同的局部算法)
- 有一个处理器最终将自己标记为Leader
- 其他处理器最终将自己标记为非Leader

用途
- 协调处理器之间的通讯和计算过程，以便完成计算任务
- 容错和节省资源

例
- 当死锁出现时，可通过Leader选举来破死锁
- 通过Leader选举，简化广播算法的实现

选举时机
- 分布式算法需要Leader,但计算环境中没有优先的Leader

Leader选举(续)



环拓扑结构
- 处理器连接成环
- 处理器仅能与相邻节点通信

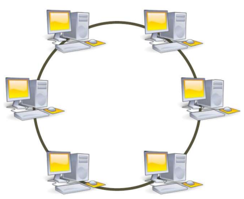
匿名环
- 处理器没有唯一标识
- 所有处理器具有相同的状态机

一致匿名环
- 算法不知道处理器的数目 n
- 所有处理器具有相同的状态机

非一致匿名环
- 算法知道处理器的数目 n
- 所有处理器具有相同的状态机
- n不同，状态机不同

同步算法
- 算法执行按轮进行
- 每轮先通信后计算

Leader选举(续)



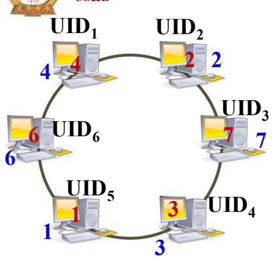
同步匿名环不存在Leader选举算法
- 所有处理器初始状态相同
- 每轮通信信息相同
- 每轮执行的计算也相同
- 归纳重复通信、计算
- 无法区分任何处理器

定理(Angluin 1980)：同步匿名环不存在Leader选举算法

证明：对轮数k进行数学归纳

必须依赖额外信息或能力来打破对称性

Leader选举(续)



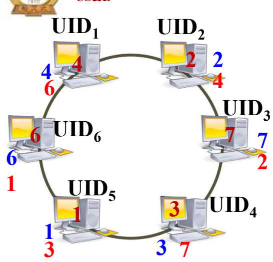
非匿名环
- 每个处理器具有唯一标识UID
- 每个处理器运行相同算法

随机数产生能力
- h:UID→[2^k]
- $h(\cdot)$均匀分布于[2^k]

第1轮

1.处理器将标记为非Leader
2.处理器计算h(UID)
3.max_i=h(UID)
3.处理器发送h(UID)给右邻居

Leader选举(续)

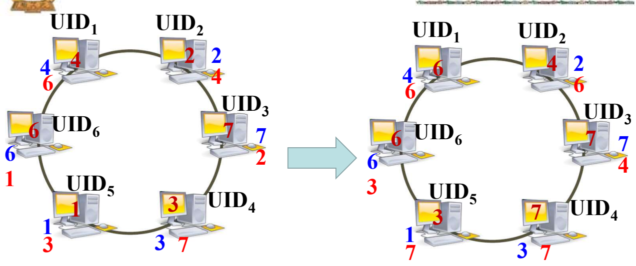


第2轮(所有结点)

1.将接收到的MSG与h(UID)比较
2.如果MSG>h(UID)则
  若MSG>max_i则max_i=MSG
  将MSG转发送给右邻居
3.如果MSG=h(UID)则
  标记自己是Leader
4.如果MSG<h(UID)则
  将h(UID)发送给右邻居

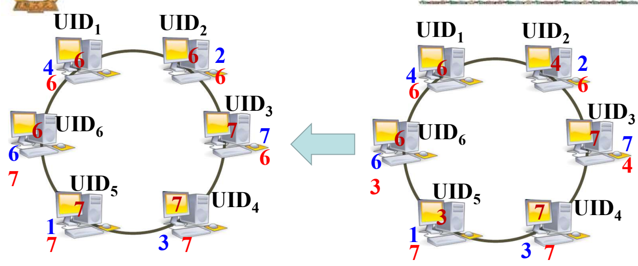
HIT CS&E

Leader选举(续)



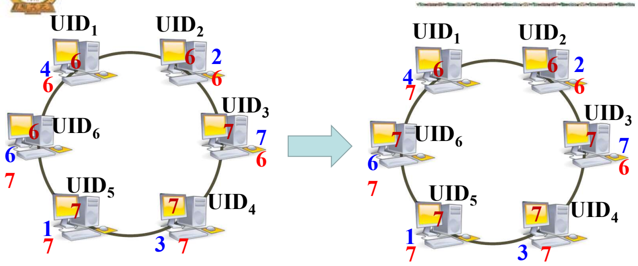
HIT CS&E

Leader选举(续)



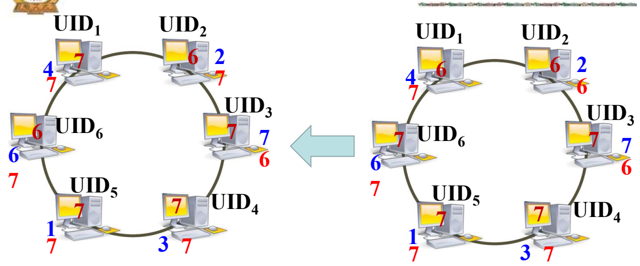
HIT CS&E

Leader选举(续)



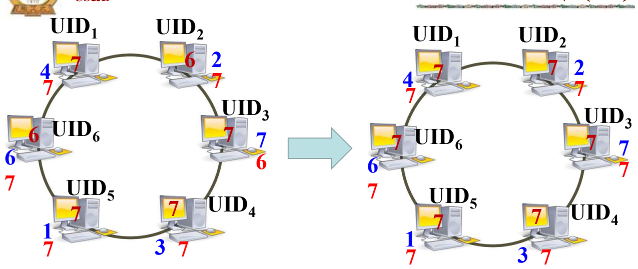
HIT CS&E

Leader选举(续)



HIT CS&E

Leader选举(续)

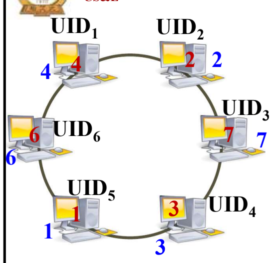


在非一致网络中，经过n轮后，选举产生Leader

在一致网络中，还需n轮转发确认消息Leader

HIT CS&E

Leader选举(续)



随机数

- h:UID→[2^k]
- $h(\cdot)$均匀分布于[2^k]

结论

- 只要$n=O(2^{k/2})$，$h(.)$是单射
- 仅需$O(n)$轮选举，即可得Leader
- 无论是一致环还是非一致环

HIT CS&E

3.4 通用散列函数

3.4.1 两两独立
3.4.2 通用散列函数族

HIT CS&E

3.4.1 两两独立

HIT CS&E

相互独立

定义：事件的相互独立性

- 随机事件$E_1,E_2,\ldots,E_n$
- 对于任意$I\subseteq\{1,2,\ldots,n\}$均有

$$\Pr\Big[\bigcap_{i\in I} E_i\Big] = \prod_{i\in I}\Pr[E_i]$$

则称$E_1,E_2,\ldots,E_n$相互独立

定义：随机变量的相互独立性

- 随机变量$X_1,X_2,\ldots,X_n$
- 对于任意$I\subseteq\{1,2,\ldots,n\}$和任意$x_i$均有

$$\Pr\Big[\bigcap_{i\in I}(X_i=x_i)\Big] = \prod_{i\in I}\Pr[X_i=x_i]$$

则称$X_1,X_2,\ldots,X_n$相互独立

HIT CS&E

k-独立

定义：事件的k-独立性

- 随机事件$E_1,E_2,\ldots,E_n$
- 对于任意$I\subseteq\{1,2,\ldots,n\}$, $|I|\le k$均有

$$\Pr\Big[\bigcap_{i\in I} E_i\Big] = \prod_{i\in I}\Pr[E_i]$$

则称$E_1,E_2,\ldots,E_n$是k-独立的

定义：随机变量的k-独立性

- 随机变量$X_1,X_2,\ldots,X_n$
- 对于任意$I\subseteq\{1,2,\ldots,n\}$ ($|I|\le k$) 和任意x_i均有

$$\Pr\Big[\bigcap_{i\in I}(X_i=x_i)\Big] = \prod_{i\in I}\Pr[X_i=x_i]$$

则称$X_1,X_2,\ldots,X_n$是k-独立的

HIT CS&E

两两独立

定义：事件的两两独立性

- 随机事件$E_1,E_2,\ldots,E_n$
- 对于任意E_i,E_j均有

$$\Pr[E_i\cap E_j] = \Pr[E_i]\cdot\Pr[E_j]$$

则称$E_1,E_2,\ldots,E_n$是两两独立的

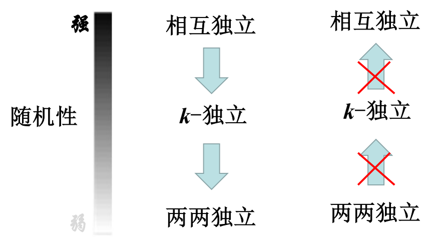
定义：随机变量的两两独立性

- 随机变量$X_1,X_2,\ldots,X_n$
- 对于任意X_i,X_j和x_i,x_j均有

$$\Pr[(X_i=x_i)\cap(X_j=x_j)] = \Pr[X_i=x_i]\cdot\Pr[X_j=x_j]$$

则称$X_1,X_2,\ldots,X_n$是两两独立的

HIT CS&E



HIT CS&E

素数模构造两两独立

定理：设X_1,X_2是$[p]$(p是素数)上的均匀独立随机变量

$Y_i = X_1 + iX_2 \bmod p \qquad i=0,1,2,\ldots,p-1$

则$Y_0, Y_1, \ldots, Y_{p-1}$是$[p]$上均匀的两两独立随机变量

证明：(1)均匀性

$Y_i = X_1 + iX_2 \bmod p$

X_2的值任意取定后，均可由X_1的均匀性确保Y_i均匀地取遍{0,1,2,…,p-1}. （延迟决策原则）

(2)两两独立性

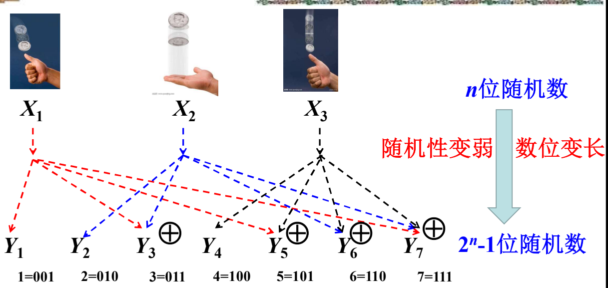
$Y_i = y_i \Leftrightarrow y_i = X_1 + iX_2 \bmod p$ 　 域 　 $X_2 = (y_i - y_j)/(i-j) \bmod p$

$Y_j = y_j \Leftrightarrow y_j = X_1 + jX_2 \bmod p$ 　　　 $X_1 = (y_j - y_i)/(j-i) \bmod p$

$\Pr[(Y_i = y_i) \cap (Y_j = y_j)] = 1/p^2 = \Pr[X_1 = x_1] \cdot \Pr[X_2 = x_2]$

HIT CS&E

两两独立的随机二进制位



$Y_1, \ldots, Y_7$是两两独立的随机二进制位

HIT CS&E

均匀性 　 $Y_i = (X_{j1} \oplus \ldots \oplus X_{j_{i-1}}) \oplus X_{j_i}$ 　 (延迟决策原则)

括弧内的值取定后，Y_i的值取决于X_{j_i}，而X_{j_i}是均匀的

两两独立性 　 ($i<k$)

$Y_i = X_{j1} \oplus \ldots \oplus X_{j_i}$

$Y_k = (X_{l_1} \oplus \ldots \oplus X_{l_{k-1}}) \oplus X_{l_k}$

X_{l_k}仅出现在Y_k中而未出现在Y_i中 　 (必存在)

所以，红色变量取定后 　 (延迟决策原则)

— Y_i的值唯一确定

— Y_k取决于X_{l_k}，而Xl_k是均匀的

— Y_k仍以1/2的概率取1

$\Pr[Y_k = c \cap Y_i = d] = \Pr[Y_k = c \mid Y_i = d] \cdot \Pr[Y_i = d] = (1/2) \cdot (1/2) = ¼$

$= \Pr[Y_k = c] \cdot \Pr[Y_i = d]$

HIT CS&E

3.4.2 通用散列函数

HIT CS&E

通用散列函数族

定义：集合$U \rightarrow \{0,1,2, \ldots, n-1\}$的一族函数$\mathcal{H}$满足：

任意$x_1, x_2, \ldots, x_k \in \{1,2,\ldots,n\}$

均匀随机选取的$h \in \mathcal{H}$

$\Pr[\ h(x_1) = h(x_2) = \ldots = h(x_k)\ ] \le 1/n^{k-1}$

则称是一个k-通用散列函数族

定义：集合$U \rightarrow \{0,1,2, \ldots, n-1\}$的一族函数$\mathcal{H}$满足：

任意$x_1, x_2, \ldots, x_k \in U$

任意$y_1, y_2, \ldots, y_k \in \{0,1,2,\ldots,n-1\}$

均匀随机选取的$h \in \mathcal{H}$

$\Pr[\ (h(x_1) = y_1) \cap \ldots \cap (h(x_k) = y_k)\ ] \le 1/n^k$

则称是一个k-强通用散列函数族

k-强通用蕴含k-通用

HIT CS&E

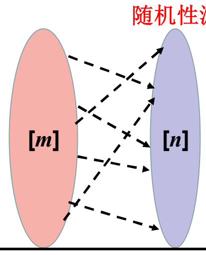
与球和箱子模型的关系

m个球

均匀独立地将球投入箱子

n个箱子

随机性源于投掷方案的选择



一个函数对应一种投掷方案

随机性源于函数的选择

在通用散列函数族中任选一个函数

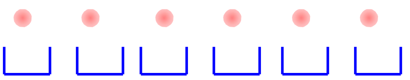
投掷方案看上去都像随机投掷方案

HIT CS&E

与球和箱子模型的关系

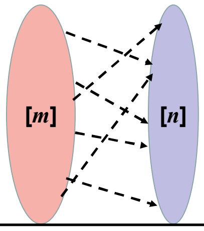
m个球

n个箱子



第i个球和第j个球同时落入同一个箱子 $X_{ij}=1$

否则 $X_{ij}=0$



从2-通用散列函数族中选择散列函数h

$E[X_{ij}] = \Pr[h(i)=h(j)] \le 1/n$

冲突总数 $X = \sum_{i<j} X_{ij}$

$E[X] = \sum_{i<j} E[X_{ij}] = \binom{m}{2} E[X_{ij}] \le \frac{m^2}{2n}$

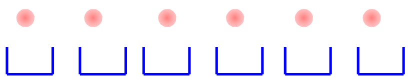
$\Pr[X > \frac{m^2}{n}] \le ½$ Markov不等式

HIT CS&E

与球和箱子模型的关系

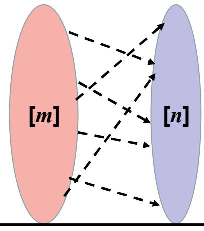
m个球

n个箱子



第i个球和第j个球同时落入同一个箱子 $X_{ij}=1$

否则 $X_{ij}=0$



从2-通用散列函数族中选择散列函数h

冲突总数 $X = \sum_{i<j} X_{ij}$ $\Pr[X > \frac{m^2}{n}] \le ½$

各个桶中最多有Y个球，则$X \ge \binom{Y}{2}$

$\Pr[\binom{Y}{2} > \frac{m^2}{n}] \le \Pr[X > \frac{m^2}{n}] \le ½$

$\Pr(Y \geqslant m\sqrt{2/n}) \leqslant \frac{1}{2}$

HIT CS&E

例1：2-通用散列函数族

定理：设 $h_{a,b}(x)$: $\{0,1,2\ldots,m-1\} \to \{0,1,2, \ldots, n-1\}$定义为

$h_{a,b}(x) = (ax+b \bmod p) \bmod n$ $p \ge m$ 是素数

则$\mathcal{H}=\{h_{a,b}(x) \mid 1 \le a \le p-1, 0 \le b \le p-1\}$是2-通用散列函数族

- 任取$x_1 \ne x_2$，任取 $a,b(a \ne 0)$, 则 $ax_1+b \ne ax_2+b$
  $ax_1+b = ax_2+b \mod p$ $\Leftrightarrow$ $a(x_1-x_2)=0 \mod p$
- 任取$x_1 \ne x_2$ $y_1 \ne y_2$，存在唯一 a,b, 使得 $ax_1+b=y_1$ $ax_2+b=y_2$
  $ax_1+b=y_1 \mod p$, $ax_2+b=y_2 \mod p$ 域 $a=(y_1-y_2)/(x_1-x_2) \mod p$, $b=y_1-x_1(y_1-y_2)/(x_1-x_2) \mod p$
- 任取y_1,在$\{0,1,\ldots,p-1\}$中至多存在p/n个y_2使$y_1=y_2 \mod n$
- 任取$x_1 \ne x_2$，均匀随机选取$h_{a,b} \in \mathcal{H}$
  $\Pr[h_{a,b}(x_1)=h_{a,b}(x_2)] = \frac{p(p-1)/n}{p(p-1)} = 1/n$

HIT CS&E

例2：2-强通用散列函数族

定理：设 $h_{a,b}(x)$: $\{0,1,2\ldots,p-1\} \to \{0,1,2, \ldots, p-1\}$定义为

$h_{a,b}(x) = ax+b \bmod p$ p是素数

则$\mathcal{H}=\{h_{a,b}(x) \mid 0 \le a,b \le p-1\}$是2-强通用散列函数族

- 任取$x_1 \ne x_2$，任取 a,b
  $ax_1+b = ax_2+b \mod p$ $\Leftrightarrow$ $a(x_1-x_2)=0 \mod p$ $\Leftrightarrow$ $a=0$
- 任取$x_1 \ne x_2$ y_1,y_2，至多有一个 a,b, 使得 $ax_1+b=y_1$ $ax_2+b=y_2$
  $ax_1+b=y_1 \mod p$, $ax_2+b=y_2 \mod p$ 域 $a=(y_1-y_2)/(x_1-x_2) \mod p$, $b=y_1-x_1(y_1-y_2)/(x_1-x_2) \mod p$
- 任取$x_1 \ne x_2$，任取y_1,y_2, 均匀随机选取$h_{a,b} \in \mathcal{H}$
  $\Pr[(h_{a,b}(x_1)=y_1) \cap (h_{a,b}(x_2)=y_2)] \le 1/p^2$

HIT CS&E

例3：2-强通用散列函数族

有限域$\{0,1,2\ldots,p^k-1\}$的基本性质

$x \in \{0,1,2\ldots,p^k-1\}$

$\Leftrightarrow$

$\exists u_0,u_1,\ldots,u_{k-1} \in \{0,1,2, \ldots, p-1\}$使得 $x= u_0p^0+u_1p^1+\ldots+u_{k-1}p^{k-1}$

有限域$\{0,1,2\ldots,p^k-1\}$ $\Leftrightarrow$ 向量空间$\{0,1,2, \ldots, p-1\}^k$

任给向量 $a=(a_0,a_1,\ldots,a_{k-1})$ 和 b， 其中$a_i,b \in \{0,1,2, \ldots, p-1\}$

$h_{a,b}(u) = a_0u_0 + a_1u_1 + \ldots + a_{k-1}u_{k-1} + b \mod p$

定义$\{0,1,2\ldots,p^k-1\} \to \{0,1,2, \ldots, p-1\}$的函数

$\mathcal{H} = \{h_{a,b}(x) \mid a \in \{0,1,2, \ldots, p-1\}^k, b \in \{0,1,2, \ldots, p-1\}\}$

HIT CS&E

定理： $\mathcal{H} = \{h_{a,b}(x) \mid a \in \{0,1,2, \ldots, p-1\}^k, b \in \{0,1,2, \ldots, p-1\}\}$
是从$\{0,1,2,\ldots,p^k-1\}$到$\{0,1,2, \ldots, p-1\}$的2-强通用散列函数族

- 任取$u_1 \ne u_2$ y_1,y_2，至多有一个 a,b, 使得 $au+b=y_1$ $au_2+b=y_2$

$au_1+b=y_1 \mod p^k$, $au_2+b=y_2 \mod p^k$ 域 数域上的二元一次方程有唯一解

$au_1+b=y_1 \mod p$, $au_2+b=y_2 \mod p$ 在$\{0,1,\ldots,p^k-1\}$上有p^{k-1}个解 延迟决策原理

- 任取$u_1 \ne u_2$，任取y_1,y_2, 均匀随机选取$h_{a,b} \in \mathcal{H}$

$\Pr[(h_{a,b}(u_1)=y_1) \cap (h_{a,b}(u_2)=y_2)] = p^{k-1}/p^{k+1}=1/p^2$

HIT CS&E

3.5 综合应用

3.5.1 散列法
3.5.2 Bloom Filter

HIT CS&E

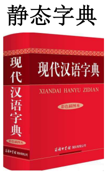
3.5.1 散列表

散列表及其分析
拉链技术
线性空间常数时间散列表

HIT CS&E

数据查找相关数据结构

静态字典



目标：快速查找数据对象

x

未找到x

n个数据对象构成的集合S
- 不宜作为密码的字符串
- 反映疾病特征的DNA片段
- 稳定客户信息（master Data）
- ……

在S中查找给定的x
- 数组
- 链表
- 哈希表
- 搜索树
- ……

HIT CS&E

哈希表

哈希表（一种数据结构）
- 一种线性结构，每个数据项有固定大小的存储空间
- 以$O(1)$的期望时间支持Find, Insert, Delete操作
- 重要假设：数据项之间是无序的
- 若要以某种方式考虑数据间的顺序，哈希表无用

哈希函数
- h(数据项)→非负整数

数据项　数据项的值

哈希下标

John → h("John") → 1

数据项　哈希函数

0	
1	*John* M
2	
3	*Dave* M
4	
5	*Mary* F

Size

应用中该如何确定h和*size*呢？

HIT CS&E

哈希表上的操作

- Insert(x)
  $T[h(x)] = (x, value_x)$
- Delete(x)
  $T[h(x)]$=NULL
- Find(x)
  return $T[h(x)]$

哈希下标

John → h("John") → 1

数据项　哈希函数

0	
1	*John* M
2	
3	*Dave* M
4	
5	*Mary* F

Size

HIT CS&E

冲突

- 冲突(Collision)
  $x \neq y$ 但 $h(x) = h(y)$
- 完美哈希(Perfect Hash)
  不存在冲突的哈希
  Find(x)的时间复杂度为$O(1)$

Joe → h("Joe") → 1

John → h("John") → 1

数据项　哈希函数

0	
1	*John* M
2	
3	*Dave* M
4	
5	*Mary* F

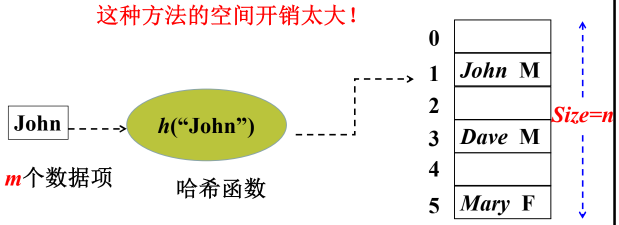
Size

HIT CS&E

完美哈希

一种可能的方法
- $n = \Omega(m^2)$
- h：$[m]\to[n]$是均匀的
- m个数据项视为人
- n个存储空间视为生日数量
- 由生日悖论可知，$h(x)$此时高概率地是单射

这种方法的空间开销太大！



HIT CS&E

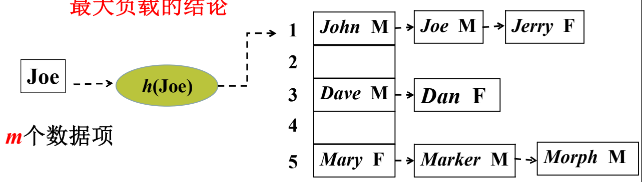
冲突处理-拉链技术

拉链技术
- 将哈希值相同的元素组织成链表
- $Find(x)$—在$h(x)$对应的链表中查找x
  - 最好时间复杂度$O(1)$
  - 最坏时间复杂度$O(\ln n/\ln\ln n)$
  - 最坏时间复杂度$O(m/n)$

$m=o(n\log n)$

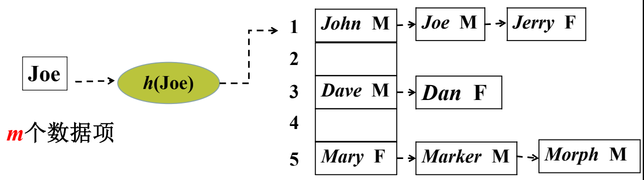
$m=\Omega(n\log n)$

最大负载的结论



HIT CS&E

问题 是否存在满足如下条件的散列表呢？
- m个数据项的静态字典
- 存储空间为$O(m)$
- 查找时间的期望时间为$O(1)$

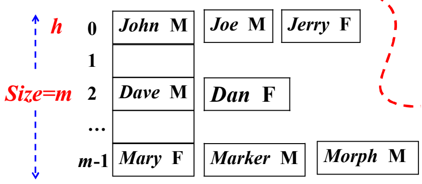


HIT CS&E

构造$O(m)$空间$O(1)$时间的完美散列表

第一阶段
1. 选择一个$[m]\to[m]$的2-通用散列函数族H
2. While (true)
3. 从H中均匀随机地选取一个函数h
4. If 用h散列字典的冲突总数≤m then 返回h

平均需要两次



2-通用散列函数族H

$\Pr[X>\frac{m^2}{n}]\le ½$

$n=m$

Pr[冲突总数> m] ≤ ½

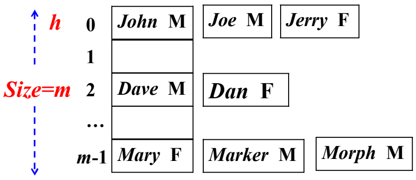
HIT CS&E

构造$O(m)$空间$O(1)$时间的完美散列表

各个桶内的数据项数依次记为$c_0,c_1,\ldots,c_{m-1}$

冲突总数 ≤ m ⇔ $\binom{c_0}{2}+\binom{c_1}{2}+\ldots+\binom{c_{m-1}}{2}\le m$

$$(c_0)^2+(c_1)^2+\ldots+(c_{m-1})^2 = 2[\binom{c_0}{2}+\binom{c_1}{2}+\ldots+\binom{c_{m-1}}{2}] + (c_0+\ldots+c_{m-1})$$
$$\le 2m+m$$
$$= 3m$$



$c_0=3$ 冲突数= $\binom{3}{2}=3$

$c_2=2$ 冲突数= $\binom{2}{2}=1$

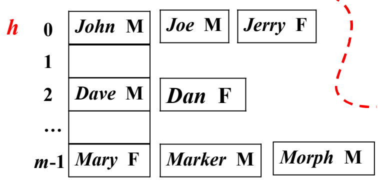
$c_{m-1}=3$ 冲突数= $\binom{3}{2}=3$

HIT CS&E

构造$O(m)$空间$O(1)$时间的完美散列表

第二阶段
1. For i=0 To m-1
2. If $c_i\le 1$ then continue
3. 均匀随机选择2-通用散列函数h_i: $[c_i]\to[(c_i)^2]$ 使桶内不产生冲突
4. 用第3步选中的h_i在桶内建立二级散列表

平均需要两次



2-通用散列函数族H

$\Pr[X>\frac{m^2}{n}]\le ½$

$n=m^2$

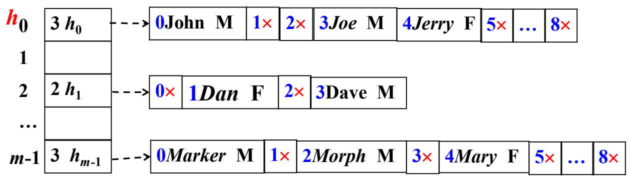
Pr[冲突总数> 1] ≤ ½

HIT CS&E

构造$O(m)$空间$O(1)$时间的完美散列表

第二阶段

1. For i=0 To m-1
2. If $c_i \le 1$ then continue
3. 均匀随机选择2-通用散列函数h_i: $[c_i] \to [(c_i)^2]$ 使桶内不产生冲突
4. 用第3步选中的h_i在桶内建立二级散列表



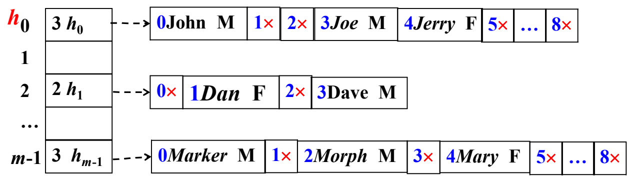
HIT CS&E

构造$O(m)$空间$O(1)$时间的完美散列表

空间开销$O(m)$

- 第一级哈希$O(m)$
  - 各个桶内数据项的数量$c_0,\dots,c_{m-1}$
  - 冲突总数不超过m
- 第二级哈希$O(m)$
  - 第i个桶内需要$(c_i)^2$个存储单元

$(c_0)^2+(c_1)^2+\dots+(c_{m-1})^2 \le 3m$

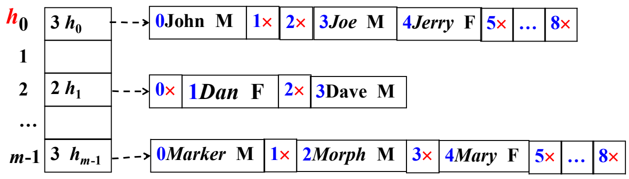


HIT CS&E

构造$O(m)$空间$O(1)$时间的完美散列表

搜索时间$O(1)$

- 第一级哈希$O(1)$时间
- 第二级哈希$O(1)$时间



HIT CS&E

3.5.2 数字指纹与Bloom Filter

HIT CS&E

计算问题

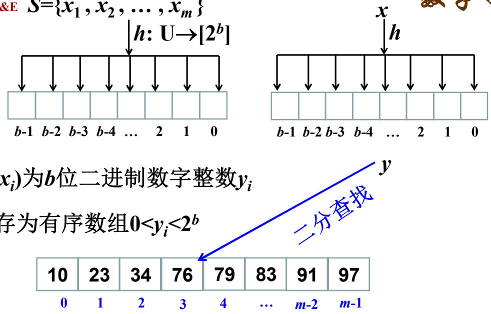
- 对象全集U
- 对象子集S ($S \subseteq U$)

给定$x \in U$,如何快速判断$x \in S$是否成立?

HIT CS&E

数字指纹

$S=\{x_1, x_2, \dots, x_m\}$



视$h(x_i)$为b位二进制数字整数y_i

将y_i存为有序数组$0<y_i<2^b$

找到,判定$x \in S$ 未必可靠　　未找到,判定$x \notin S$ 可靠

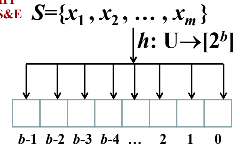
结论可靠吗? false Positive 这种错误的概率与b什么关系?

HIT CS&E

数字指纹

$S=\{x_1,x_2,\dots,x_m\}$

$h: U\to[2^b]$



若希望Pr[假阳性]≤ε，则

$1-2^{-m/2^b}\le\varepsilon$

$b\ge\log_2\frac{m}{-\ln(1-\varepsilon)}$

假设: $h(x)$均匀独立地取$[2^b]$中的每个数值

球：U中的元素

箱子：2^b

假阳性: $x\notin S$但$h(x)=h(x_i)$对某个i成立

Pr[假阳性]$=\Pr[h(x)=h(x_1)\vee\dots\vee h(x)=h(x_m)]$

$=1-\Pr[h(x)\neq h(x_1)\wedge\dots\wedge h(x)\neq h(x_m)]$

$=1-\Pr[h(x)\neq h(x_1)]\cdot\dots\cdot\Pr[h(x)\neq h(x_m)]$

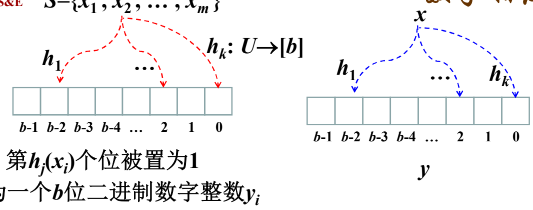
$=1-(1-2^{-b})^m$

$\le 1-2^{-m/2^b}$

HIT CS&E

数字指纹

$S=\{x_1,x_2,\dots,x_m\}$



第$h_j(x_i)$个位被置为1

视为一个b位二进制数字整数y_i

将y_i存为有序数组$0<y_i<2^b$

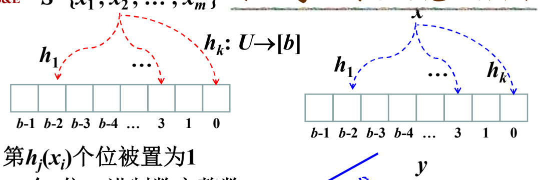
10	23	34	76	79	83	91	97
0	1	2	3	4	…	m-2	m-1

你能用k,b,m表示假阳性发生的概率吗？

HIT CS&E

用数字指纹避免假阳性

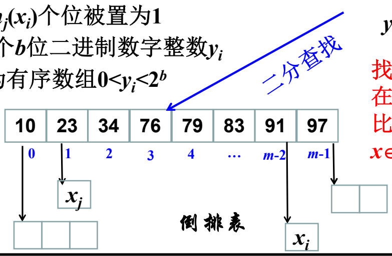
$S=\{x_1,x_2,\dots,x_m\}$



第$h_j(x_i)$个位被置为1

视为一个b位二进制数字整数y_i

将y_i存为有序数组$0<y_i<2^b$



找到

在倒排表比对

比对成功

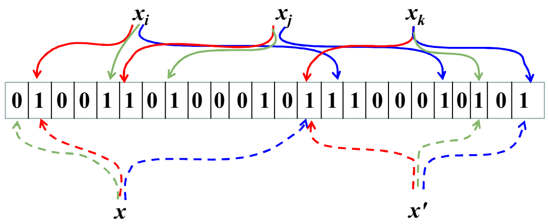
$x\in S$

HIT CS&E

Bloom Filter

$S=\{x_1,x_2,\dots,x_m\}$

$h_1,h_2,h_3: U\to[n]$



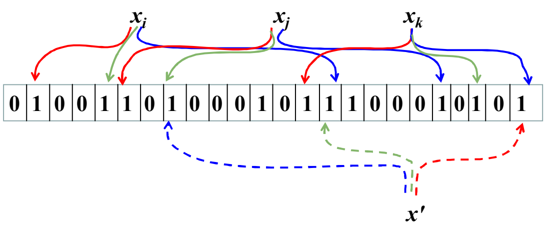
判定$x\notin S$ 可靠　　判定$x'\in S$ 正确

HIT CS&E

Bloom Filter

$S=\{x_1,x_2,\dots,x_m\}$

$h_1,h_2,h_3: U\to[n]$



判定$x'\in S$ 假阳性

假阳性与m,n,函数个数之间的关系是什么？

HIT CS&E

$S=\{x_1,x_2,\dots,x_m\}$　　m个数据项

$h_1,h_2,\dots,h_k: U\to[n]$　　k个哈希函数，UHA

0	1	0	0	1	1	0	1	0	0	0	1	0	1	1	1	0

n个二进制位

相当于：向n个箱子中投入km个球

任意一个位: $\boxed{0}\to\begin{cases}\boxed{0} & p=(1-\frac{1}{n})^{km}\approx e^{-km/n}\\ \boxed{1} & 1-p\end{cases}$

HIT CS&E

任意一个位: $\boxed{0} \rightarrow \begin{cases} \boxed{0} & p=(1-\frac{1}{n})^{km} \approx e^{-km/n} \\ \boxed{1} & 1-p \end{cases}$

0	1	0	0	1	1	0	1	0	0	0	1	0	1	1	1	0

$x \notin S$

这k个位中，若有一个位是0,则不会出现假阳性

Pr[假阳性] = $(1-p)^k \approx (1-e^{-km/n})^k$

HIT CS&E

Pr[假阳性] = $(1-p)^k \approx (1-e^{-km/n})^k$

将m,n视为已知，假阳性是k的函数

- k越大，0的比例越小，假阳性的可能性越高
- k越小，利用散列函数的机会就越小

能否以最小化假阳性概率为目标优化地选择k?

ln(Pr[假阳性]) = $k\ln(1-e^{-km/n}) = g(k)$

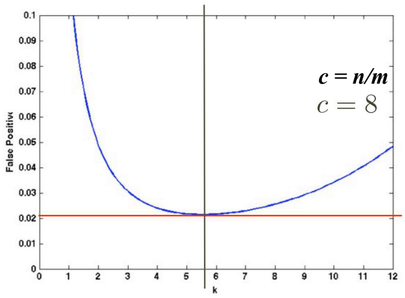
最小化假阳性概率 ⇔ 最小化$g(k)$

求$g(k)$的一阶导数，并令导数等于0

$\ln(1-e^{-km/n}) + \frac{km}{n}\frac{e^{-km/n}}{1-e^{-km/n}} = 0$ 解得 $k=\ln 2\cdot(n/m)$

Pr[假阳性] = $(1/2)^k \approx (0.618)^{n/m}$

HIT CS&E



HIT CS&E

Bloom Filter的应用场景

典型应用场景

- 黑名单管理
- 爬虫URL重复名单管理
- 字典纠错
- 磁盘文件检测
- CDN代理缓存技术

文献:

Network application of Bloom Filter: A Survey, Internet Mathematics, 1(4): 485-509, 2005.

HIT CS&E

用Bloom Filter追踪IP

场景

- 网络服务器在日志中发现了一个恶意数据包
- 追踪该数据包的来源
- 将来源列为恶意攻击点和不信任名单

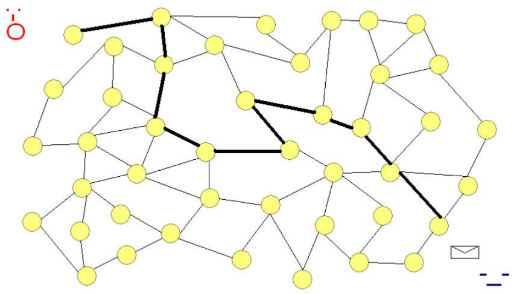
方法1

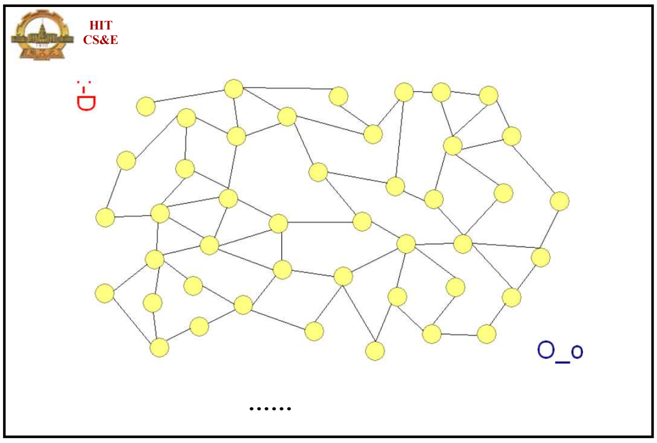
- 每个服务器记录每个数据包的转发地址
- 巨量数据包
  - 巨大存储开销
  - 低效地址查询

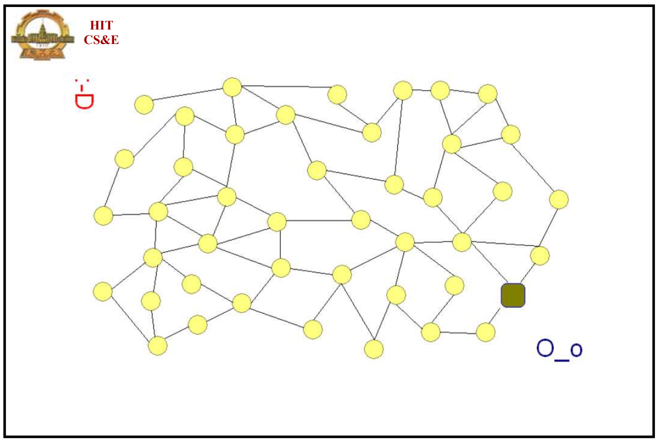
方法2

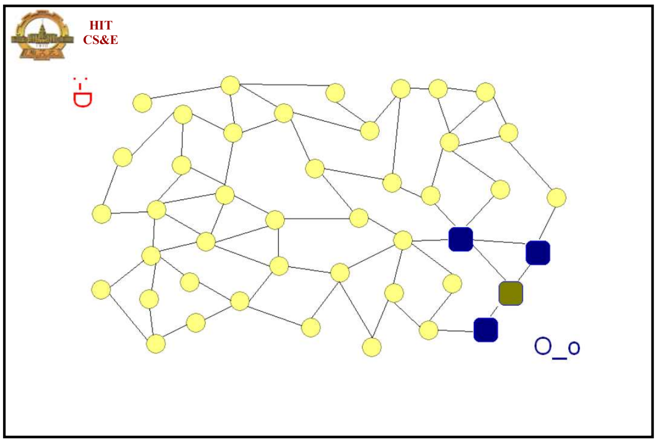
- 每个服务器用Bloom Filter记录看到的IP地址
- 各服务器递归查询邻居谁见过目标地址
  - 高效存储开销
  - 高效地址查询

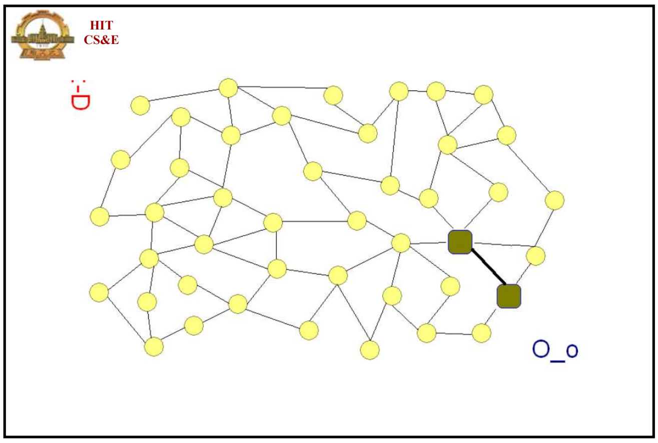
HIT CS&E

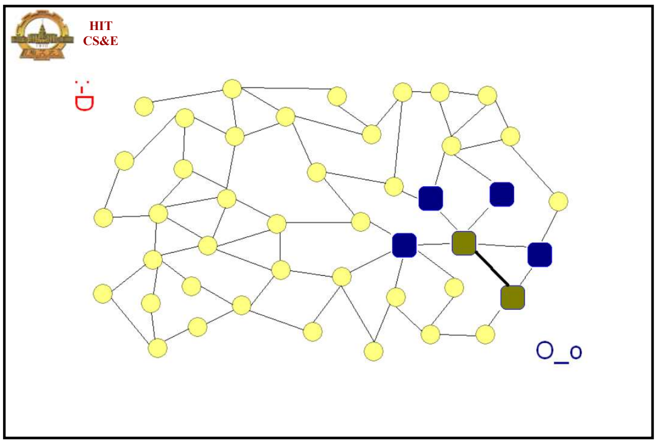


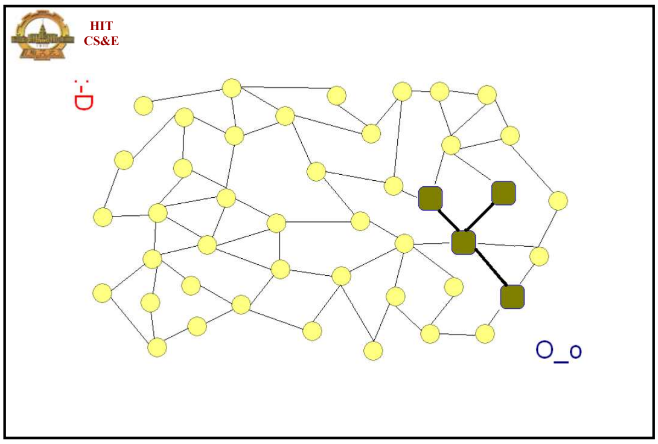

HIT
CS&E
:-D
O_o
......


HIT
CS&E
:-D
O_o

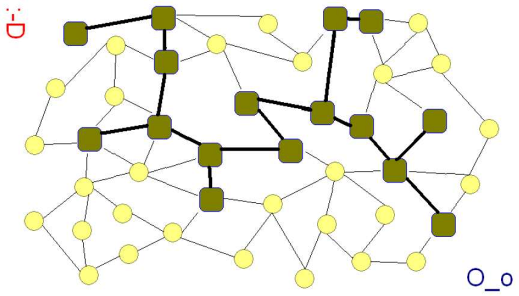

HIT
CS&E
:-D
O_o


HIT
CS&E
:-D
O_o


HIT
CS&E
:-D
O_o


HIT
CS&E
:-D
O_o

HIT CS&E



HIT CS&E

实验3:随机数据结构及其性能

实验目的

- 进一步理解和巩固三种分析工具
- 进一步理解本章引入的两种随机数据结构
- 通过比较理解随机数据结构的优势
- 观察随机数据结构对算法性能的影响
- 调整参数，比较算法在各种参数设置下的性能，理解理论分析结果在参数设置中的作用
- 规范书写实验报告

HIT CS&E

实验内容

二选一

一、BloomFilter

1. 查阅资料了解BloomFilter的典型应用
2. 选择一种有意义(interesting)的应用加以实现
3. 对比用和不用BloomFilter时的时、空性能
4. 调整BloomFilter的参数，对比性能
5. 撰写实验报告

二、随机跳表

1. 查阅资料了解随机跳表的典型应用
2. 选择一种有意义(interesting)的应用加以实现
3. 对比用和不用随机跳表时的时、空性能
4. 调整层间数据留存参数p，对比时、空性能
5. 撰写实验报告