



摊还分析

东南大学计算机学院 方效林

本章内容

- 聚合分析
- 核算方法
- 势能方法

摊还分析

- 对一个数据结构执行 n 个操作
 - 有些操作代价高
 - 有些操作代价低
 - 有些操作代价中等
- 将所有操作的代价平摊到每个操作上
 - 不涉及概率
 - 不同于平均情况

聚合分析

■ 栈的三种操作

操作	代价
PUSH	1
POP	1
MULTIPOP(s,k)	$\min(s, k)$

■ 对一个空栈，执行由n个PUSH, POP, MULTIPOP组成的操作，代价是多少？

由于multipop(s,n)的代价最坏为n，因此n个操作最坏为 ~~$O(n^2)$~~ ?

一个元素要么入栈，要么出栈，
一个空栈中，n个PUSH, POP, MULTIPOP组成的操作最多与PUSH次数相当
即最多对n个元素操作，因此为 $O(n)$

聚合分析

最坏情况代价为~~k~~,
n次累加代价是否为 ~~$O(kn)$~~ ?

■ 二进制计数器(k位)

□ 初始值为0，每次加1，累加n次，

n	A[7]	A[6]	A[5]	A[4]	A[3]	A[2]	A[1]	A[0]	总代价
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	1
2	0	0	0	0	0	0	1	0	3
3	0	0	0	0	0	0	1	1	4
4	0	0	0	0	0	1	0	0	7
5	0	0	0	0	0	1	0	1	8
6	0	0	0	0	0	1	1	0	10
7	0	0	0	0	0	1	1	1	11

聚合分析

第0位翻转, 有n次
第1位翻转, 有n/2次
第2位翻转, 有n/2²次

■ 二进制计数器(k位)

□ 初始值为0, 每次加1, 累加n次,

总共 $\sum_{i=0}^k \frac{n}{2^i} \leq 2n$
因此为O(n)

n	A[7]	A[6]	A[5]	A[4]	A[3]	A[2]	A[1]	A[0]	总代价
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	1
2	0	0	0	0	0	0	1	0	3
3	0	0	0	0	0	0	1	1	4
4	0	0	0	0	0	1	0	0	7
5	0	0	0	0	0	1	0	1	8
6	0	0	0	0	0	1	1	0	10
7	0	0	0	0	0	1	1	1	11

核算方法

■ 栈的三种操作

操作	代价
PUSH	1
POP	1
MULTIPOP(s,k)	$\min(s, k)$

■ 对一个空栈，执行由n个PUSH, POP, MULTIPOP组成的操作，代价是多少？

一个元素只有入栈后，才能出栈，将代价全部放到PUSH操作上(预支付将来出栈的代价)，出栈操作代价为0，因此n个操作代价为 $O(n)$

操作	代价
PUSH	2
POP	0
MULTIPOP(s,k)	0

核算方法

置位代价为2(预付将来复位代价)

复位代价为0

n次累加代价 $O(2n)$?

■ 二进制计数器(k位)

□ 初始值为0，每次加1，累加n次，

n	A[7]	A[6]	A[5]	A[4]	A[3]	A[2]	A[1]	A[0]	总代价
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	1
2	0	0	0	0	0	0	1	0	3
3	0	0	0	0	0	0	1	1	4
4	0	0	0	0	0	1	0	0	7
5	0	0	0	0	0	1	0	1	8
6	0	0	0	0	0	1	1	0	10
7	0	0	0	0	0	1	1	1	11

势能方法

- 数据结构初始状态 D_0 ，其势 $\Phi(D_0)$
- 执行第 i 个操作的状态 D_i ，其势 $\Phi(D_i)$
- 第 i 个操作的摊还代价 \hat{c}_i ，真实代价 c_i

$$\hat{c}_i = c_i + \Phi(D_i) - \Phi(D_{i-1})$$

- 总摊还代价

$$\begin{aligned}\sum_{i=1}^n \hat{c}_i &= \sum_{i=1}^n (c_i + \Phi(D_i) - \Phi(D_{i-1})) \\ &= \sum_{i=1}^n c_i + \Phi(D_n) - \Phi(D_0)\end{aligned}$$

势能方法

■ 二进制计数器(k位)

- 初始值为0，每次加1，累加n次，

n	A[7]	A[6]	A[5]	A[4]	A[3]	A[2]	A[1]	A[0]	势
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	1
2	0	0	0	0	0	0	1	0	0
3	0	0	0	0	0	0	1	1	2
4	0	0	0	0	0	1	0	0	0
5	0	0	0	0	0	1	0	1	1
6	0	0	0	0	0	1	1	0	0
7	0	0	0	0	0	1	1	1	3

势能方法

- 二进制计数器(k位)

- 初始值为0，每次加1，累加n次，

- 定义势为计数器中 1 的个数

- 假设第 i 次累加要对 t_i 个位复位，则代价 $1 + t_i$

复位 t_i 位，置位1位

- 若结果k位均为0 (即前一次k位均为1)，则势差

- $$\Phi(D_i) - \Phi(D_{i-1}) = -t_i = -k$$

- 否则，势差 $\Phi(D_i) - \Phi(D_{i-1}) = 1 - t_i$

- 无论哪种情况，摊还代价 $\hat{c}_i = c_i + \Phi(D_i) - \Phi(D_{i-1}) \leq (1 + t_i) + (1 - t_i) = 2$