

算法设计与分析作业 06

正心 44

Prof. Hong Gao

TA: 郭环宇

发布日期: 2019/05/10

截至日期: 2019/05/18

作业提交 PDF 版本, 作业命名格式为学号 _ 姓名 _xx 次作业 _version.pdf (方便大家更新自己的作业), 代码单独提交任一编程语言文件 (*.c *.cpp *.h *.cs *.java *....), 与作业一起打包并命名为学号 _ 姓名 _XX 次作业 _version.xxx 发送至邮箱 DBLB_2019algorithm@163.com

Exercise 1. (5)

对某个数据结构执行大小为 n 的一个操作序列, 若 i 为 2 的整数幂, 则第 i 个操作的代价为 i , 否则为 1。请利用会计方法分析每次操作的平摊代价

Exercise 2. (15)

Bill 提出了一种叫做翻转堆栈的数据结构, 翻转堆栈只支持 Flipping-Push() 函数。在每次 Flipping-Push() 中, 首先压栈, 并检查堆栈中的对象数目数是否是 2 的幂 (2^i)。如果是, 则堆栈中的所有对象将翻转。例如我们使用 Flipping-Push() 将对象 1,2,3,4 压入堆栈中, 堆栈中的内容 (从底向上看) 在每次压栈后为:

$(1) \Rightarrow (2, 1) \Rightarrow (2, 1, 3) \Rightarrow (4, 3, 1, 2)$

Bill 请求你分别使用聚集分析、会计方法、势能方法分析 Flipping-Push() 函数的平摊代价 (amortized cost), 堆栈反转的代价等于堆栈现有对象数目

Exercise 3. (8)

有两个堆栈 A 和 B, 都可以使用以下 5 种操作 (A 大小为 n , B 大小为 m):

PushA(x):x 压入堆栈 A, 实际代价 =1

PushB(x):x 压入堆栈 B, 实际代价 =1

MultiPopA(k): 从 A 中弹出 $\min\{k, n\}$ 个元素, 实际代价 = $\min\{k, n\}$

MultiPopB(k): 从 B 中弹出 $\min\{k, m\}$ 个元素, 实际代价 = $\min\{k, m\}$

Transfer(k): 从 A 中弹出元素并压入 B 中, 直到转移 k 个元素或 A 为空, 实际代价 = 转移元素数目

(a) MultiPopA(k), MultiPopB(k), Transfer(k) 三种操作最坏情况下的时间复杂度为多少?

(b) 定义一个势能函数 $\Phi(n, m)$, 用它证明以上操作平摊代价为 $O(1)$