

《数据结构与算法》勘误表

URL: <http://db.pku.edu.cn/mzhang/ds/resource/errata.pdf>

张 铭 王腾蛟 赵海燕

修定于 2009 年 10 月 28 日

目 录

一、教材信息

1. 教材出版信息

张铭，王腾蛟，赵海燕，《数据结构与算法》，高等教育出版社，2008年6月。普通高等教育“十一五”国家级规划教材

高教社购书热线：010-58581118，58581117，58581116。

高教社网上订购 URL: <http://www.landaco.com.cn>

2. 每位作者负责的章节

本书第8、10、11、12章（第194—240页，第260—281页）由张铭教授执笔，第5、6、7、9章（第100—193，第241—259页）由王腾蛟副教授执笔，第1、2、3、4章（第1—99页）由赵海燕副教授执笔。

3. 课程网站（课程讲义、算法源代码等）

<http://www.db.pku.edu.cn/mzhang/DS/>（教育网）

<http://www.jpk.pku.edu.cn/pkujpk/course/sjig/>

4. 课程和教材论坛

http://groups.google.com/group/ds2008_advanced_pkucs/

二、勘误表

页数	具体位置	修改结果
P24	倒数 L5 习题 12(3)	$T(n) = 2T(\lfloor n/2 \rfloor) + n$
P31	算法 2.3 中的 L6(即 for 语句)之前增加一行	$n = \text{curLen};$
P33	算法 2.5 中的 L6 中的 \leq 号中间的空格去掉，	if (curLen \leq 0)
P37	L1（算法 2.8）	Head = tail = new Link <T> (null);
P37	算法 2.9 中 L6(即 while 语句之前一行) 改为	Link<T> *p = head->next;
p87	倒数 L1	while (s2[i] != '\0' && s1[i] != '\0') {
P93	倒数 L9	在图 4.6 的例子中，第 1 趟匹配后，如果模式仅仅右移一位的话，下趟匹配中会有冗余的比
P94	L11	$T(j-i+2 \dots j-1)$ 必相同。依此类推，或可找到某个 k ($0 < k < i$) 值，使得
P94	L13 后增加	如果找不到这样的 k ，则取其为 0。
P94	倒数 L12 开始至倒数 L4 修改为	<p>计算 next[i] 时充分利用了位置 i 之前的各个字符的 next 值，若用 n_i 表示 next[i] 的话，特征数 n_{i+1} ($i > 0$, $0 \leq n_{i+1} \leq i$) 可以采用如下的递归方式来定义：</p> <ol style="list-style-type: none"> 1) $n_0 = -1$; 对于 $i > 0$ 的 n_{i+1}，假定已知前一位置的特征数 $n_i = k$; 2) 如果 $i > 0$, $k \geq 0$ 且 $P_i = P_k$ 则 $n_{i+1} = k+1$; 3) 当 $k \geq 0$, 且 $P_i \neq P_k$ 时，则令 $k = n_k$，并让 3) 循环直到条件不满足(变为 $P_i = P_k$ 或上一步骤 $k =$

		<p>0 而导致 $k=-1$ 了);</p> <p>4) 当 $k = 0$, 且 $P_i \neq P_k$ 时, 则 $n_{i+1} = 0$。</p> <p>一般把模式的第一个字符的特征数设置为-1, 以尽可能地减少冗余比较。此时, 上述条件 2) 和 4) 可以合并处理。在算法中, 简化为</p> <p>1) $n_0 = -1$; 对于 $i > 0$ 的 n_i, 假定已知前一位置的特征数 $n_i = k$;</p> <p>2) 当 $k \geq 0$, 且 $P_i \neq P_k$ 时, 则令 $k = n_k$, 并让 2) 循环直到条件不满足 (变为 $P_i = P_k$ 或 $k = 0$);</p> <p>3) $n_{i+1} = k+1$; // 此时 $k = -1$, 或者 $P_i = P_k$</p>
P95	L1	此时该把模式右移 $i-k$ 位, 即用 P_k 与 T_j 进行比较。如果 $P_i = P_k$, 则 T_j
P95	算法 4.7 的 L9	while (i < m-1) { // 若写成 $i < m$ 会越界
P95	算法 4.7 的 L14	删除 “if (i == m) break” // 此句是跟 “ $i < m$ ” 匹配的, 上面写 “ $i < m-1$ ” 就可免掉
P97	L8	return (j-pLen);
P102	证明部分的第 4 行	将“如果删去一个分支结点”改为“如果删去一个有两个空结点的分支结点”
P114	Parent 函数的 while 循环之后 (代码的倒数第 3 行)	添加: if (pointer == NULL) return NULL;
P117	InsertNode 函数的 while 条件	建议将 while(pointer != NULL) 变为 while(1)
P143	图 6.6	索引值为 2 的行: 将子结点链表中的值由顺次为 6, 7, 5 变为 5, 6, 7 索引值为 6 的行: 将子结点链表中的值由顺次为 10, 10 变为 10, 11
P148	L3-4 替换	<p>原来的</p> <pre>if (pointer == NULL) root = subroot->RightSibling();</pre> <p>替换为:</p> <pre>if (pointer == NULL) { // subroot 无父, 则是某个树根 pointer = root; while (pointer->RightSibling() != subroot) // 顺右链找左邻树根 pointer = pointer->RightSibling(); pointer->setSibling(subroot->RightSibling()); // 前后挂接, 脱链 }</pre>
P163	图 7.4a	第二个子图编号 v_1 改为 v_0 , 即子图为 v_0, v_2, v_3 组成
P163	第 2 段最后一行	n 个顶点 $\{v_0, v_1, v_2, \dots, v_{n-1}\}$
P166	$D(v_i)$	$n-1$ $n-1$

		$D(v_i) = \sum_{j=0} A[i, j] = \sum_{j=0} A[j, i]$
P166	ID(v_j)	$ID(v_j) = \sum_{i=0}^{n-1} A[i, j], OD(v_i) = \sum_{j=0}^{n-1} A[i, j]$
P169	代码 7.3 中 NextEdge 函数	if (preEdge.to < numVertex) 改为 if (preEdge.to < numVertex-1) 更好
P183	第一段	...该算法的基本思想是:在原图的相邻矩阵 adj 上做 n 次迭代,递归地产生一个矩阵序列 $adj^{(0)}, adj^{(1)}, \dots, adj^{(n-1)}$ 。其中, $adj^{(k)}$ 的值等于...
P183	图 7.21	矩阵代号为 adj , $adj^{(0)}$, $adj^{(1)}$, $adj^{(2)}$
P222	L3	收集 r 个队列的时间是 $\Theta(r)$
P245	9.3 节第 4 段第 2 行	归并趟数为 $\lceil \log_k m \rceil$
P251	第 3 段	从图 9.4 中可以方便地看到...
P318	L10	“满足 $4m+8(m+1)$ ” 前面添加 “同层拉链占一个指针”
P328	图 11.25(f)	c 结点应该为 62 d 结点应该为 60
P345	倒数 L7	"首先适配"中间没有逗号
P377	图 12.47	(4,9)YALAAMs\$ 下面结点标号 1; (9,9)\$ 下面结点标号 9

三、鸣谢

感谢同时任教的宋国杰老师、历届助教的大力协助,全体信息学院同学的认真钻研、积极探索,所有热心读者的帮助。