6.3 ① 试分析算法 6.3 的复杂度。

② 令 $A=(0,1,2,7,9,11,16,17,18,19,23,24,25,27,28,30,33,34)$，$B=(3,4,5,6,8,10,12,13,14,15,20,21,22,26,29,31)$。试按算法 6.3，将其进行对数划分，并最终将它们归并之。

6.6 ① 试分析算法 6.9 的总运算量 $W(n)=?$

② 假定序列为 $(1,2,3,4,5,6,7,8)$，试用算法 6.9 求其前缀和。

6.9 试解释在一维心动阵列上计算卷积时，序列 x 和 y 为何要各间隔一拍进入阵列。

6.10 顶点倒塌法是非常有名的求图的连通分量的算法，其基本思想是：连通的相邻的顶点可以合并成一个超顶点，并以它们中最小标号者标记之；此过程可继续在已合并的超顶点之间进行。在下列的算法中，$C(i)$ 表示与 i 相邻的最小的超顶点号码；$D(i)$ 表示顶点 i 所属连通分量的最小标号的顶点；$C(i)=\min\limits_{j}\{D(j)\mid_{A(i,j)=1,D(i)\neq D(j)}\}$ 语句为每个顶点 i 找与它不属于相同分量的相邻的最小号码的顶点 j；语句 $C(i)=\min\limits_{j}\{C(j)\mid_{D(j)=i,C(j)\neq i}\}$ 表示把每个超顶点的根连到最小号码的相邻的超顶点的根上。Hirschberg 的求连通分量算法如下：

算法 6.12　PRAM-CREW 上 Hirschberg 求连通分量算法

输入： 邻接矩阵 $\mathbf{A}_{n\times n}$

输出： 向量 $\mathbf{D}[0:n-1]$，其中 $D(i)$ 表示向量 $\mathbf{D}$ 的分量

Begin

(1) **for all** $i: 0 \leqslant i \leqslant n-1$ **par-do** /* 初始化 */

$D(i) = i$

end for

do step (2) through (6) for $\lceil \log n \rceil$ iterations:

(2) **for all** $i, j: 0 \leqslant i, j \leqslant n-1$ **par-do** /* 找相邻的最小者 */

(2.1) $C(i) = \min_j \{ D(j) \mid_{A(i,j)=1 \text{ and } D(i) \neq D(j)} \}$

(2.2) **if** none **then** $C(i) = D(i)$ **endif**

end for

(3) **for all** $i, j: 0 \leqslant i, j \leqslant n-1$ **par-do** /* 找每个超顶点的最小相邻超顶点 */

(3.1) $C(i) = \min_j \{ C(j) \mid_{D(j)=i \text{ and } C(j) \neq i} \}$

(3.2) **if** none **then** $C(i) = D(i)$ **endif**

end for

(4) **for all** $i: 0 \leqslant i \leqslant n-1$ **par-do** $D(i) = C(i)$ **end for**

(5) **for** $\lceil \log n \rceil$ iterations **do** /* 指针跳越，找各顶点新的超顶点 */

for all $i: 0 \leqslant i \leqslant n-1$ **par-do** $C(i) = C(C(i))$ **end for**

end for

(6) **for all** $i: 0 \leqslant i \leqslant n-1$ **par-do**

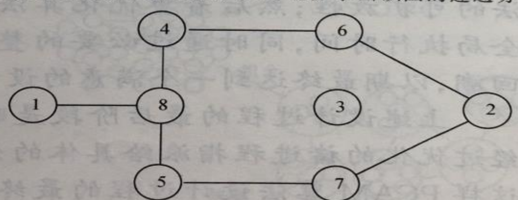
$D(i) = \min\{ C(i), D(C(i)) \}$

end for

End

① 试分析算法 6.12 的复杂度 $t(n) = ?$ 和 $p(n) = ?$

② 给定如图 6.11 所示无向图，试用算法 6.12 逐步求出该图的连通分量。



顶点	1	2	3	4	5	6	7	8
分量	1	2	3	4	5	6	7	8

图 6.11 待求连通分量的无向图