2-3 设 $a[0:n-1]$是已排好序的数组。请改写二分搜索算法，使得当搜索元素 $x$ 不在数组中时，返回小于 $x$ 的最大元素位置 $i$ 和大于 $x$ 的最小元素位置 $j$。当搜索元素在数组中时，$i$ 和 $j$ 相同，均为 $x$ 在数组中的位置。

2-4 给定两个整数 $u$ 和 $v$，它们分别有 $m$ 和 $n$ 位数字，且 $m \leqslant n$。用通常的乘法求 $uv$ 的值需要的 $O(mn)$时间。可以将 $u$ 和 $v$ 均看作是有 $n$ 位数字的大整数，用本章介绍的分治法，在 $O(n^{\log 3})$时间内计算 $uv$ 的值。当 $m$ 比 $n$ 小得多时，用这种方法就显得效率不够高，试设计一个算法，在上述情况下用 $O(nm^{\log(3/2)})$时间求出 $uv$ 的值。

2-5 在用分治法求两个 $n$ 位大整数 $u$ 和 $v$ 的乘积时，将 $u$ 和 $v$ 都分割成长度为 $n/3$ 位的 3 段。证明可以用 5 次 $n/3$ 位整数的乘法求得 $uv$ 的值。按此思想设计一个求两个大整数乘积的分治算法，并分析算法的计算复杂性。（提示：$n$ 位的大整数除以一个常数 $k$ 可以在 $\theta(n)$时间内完成。符号 $\theta$ 所隐含的常数可能依赖于 $k$。）

2-6 对任何非零偶数 $n$，总可以找到奇数 $m$ 和正整数 $k$，使得 $n=m2^k$。为了求出两个 $n$ 阶矩阵的乘积，可以把一个 $n$ 阶矩阵分成 $m \times m$ 个子矩阵，每个子矩阵有 $2^k \times 2^k$ 个元素。当需要求 $2^k \times 2^k$ 的子矩阵的积时，使用 Strassen 算法。设计一个传统方法与 Strassen 算法相结合的矩阵相乘算法，对任何偶数 $n$，都可以求出两个 $n$ 阶矩阵的乘积。并分析算法的计算时间复杂性。

2-7 设 $P(x)=a_0+a_1x+\cdots+a_dx^d$ 是一个 $d$ 次多项式。假设已有一算法能在$O(i)$时间内计算一个 $i$ 次多项式与一个 1 次多项式的乘积，以及一个算法能在$O(i\log i)$时间内计算两个 $i$ 次多项式的乘积。对于任意给定的 $d$ 个整数 $n_1, n_2, \cdots, n_d$，用分治法设计一个有效算法，计算出满足 $P(n_1)=P(n_2)=\cdots=P(n_d)=0$ 且最高次项系数为 1 的 $d$ 次多项式 $P(x)$，并分析算法的效率。

2-8 设 $n$ 个不同的整数排好序后存于 $T[0:n-1]$中。若存在下标 $i$，$0 \leqslant i < n$，使得 $T[i]=i$，设计一个有效算法找到这个下标。要求算法在最坏情况下的计算时间为 $O(\log n)$。

2-9 设 $T[0:n-1]$是 $n$ 个元素的数组。对任一元素 $x$，设 $S(x)=\{i \mid T[i]=x\}$。当 $|S(x)|>n/2$时，称 $x$ 为 $T$ 的主元素。设计一个线性时间算法，确定 $T[0:n-1]$是否有一个主元素。

2-10 若在习题 2-9 中，数组 $T$ 中元素不存在序关系，只能测试任意两个元素是否相等，试

2-24 试用栈来模拟递归，消去算法 qSort 中的递归。并证明所需的栈空间为$O(\log n)$。

2-25 在算法 select 中，输入元素被划分为 5 个一组，如果将它们划分为 7 个一组，该算法仍然是线性时间算法吗？划分成 3 个一组又怎样？

2-26 试说明如何修改快速排序算法，使它在最坏情况下的计算时间为$O(n\log n)$。

2-27 给定由 $n$ 个互不相同的数组成的集合 $S$ 以及正整数 $k \leqslant n$，试设计一个 $O(n)$ 时间算法找出 $S$ 中最接近 $S$ 的中位数的 $k$ 个数。

2-28 设 $X[0:n-1]$和 $Y[0:n-1]$为两个数组，每个数组中含有 $n$ 个已排好序的数。试设计一个$O(\log n)$时间的算法，找出 $X$ 和 $Y$ 的 $2n$ 个数的中位数。

2-29 考查如图 2-13 所示的有两个输入端和两个输出端的两个位置开关。当开关处于位置 1 时，输入 1 和 2 分别产生输出 1 和 2；当开关处于位置 2 时，输入 1 和 2 分别产生

# 上载作业： 第二章作业

## 作业信息

| 截止日期 | 满分 |
| --- | --- |
| **2021年3月29日 星期一** 下午11:59 | **1** |

第二章课后习题：2-3,2-7,2-25,2-28