《算法设计与分析》第2次作业参考答案

题目1：求下列递推关系表示的算法复杂度。

(1)$T(n) = 9T(\frac{n}{3}) + n$

(2)$T(n) = 8T(n/6) + n3/2logn$

(3)$T(n) = 7T(n/7) + n$

答：

(1)$T(n) = \Theta(n^2)$

由master定理：

$a = 9, b = 3, f(n) = n, n^{\log_b a} = n^2$

$\because f(n) = n = O(n^{(\log_b a)-\epsilon})$，即$\epsilon = 1$

$\therefore T(n) = \Theta(n^{(\log_b a)}) = \Theta(n^2)$

(2)$T(n) = \Theta(n^{\frac{3}{2}} \log n)$

由master定理：

$a = 8, b = 6, f(n) = n^{\frac{3}{2}} \log n, n^{\log_b a} = n^{\log_6 8}$

$\because f(n) = O(n^{(\log_b a)+\epsilon})$

$\therefore T(n) = \Theta(n^{\frac{3}{2}} \log n)$

(3)$T(n) = \Theta(n \log n)$

由master定理：

$a = 7, b = 7, f(n) = n, n^{\log_b a} = n$

$\because f(n) = n = \Theta(n^{\log_b a}) = \Theta(n)$

$\therefore T(n) = \Theta(n^{\log_b a} \log n) = \Theta(n \log n)$

题目2：设A[0:n-1]是一个元素个数为n的未排序的数组，运用分治算法找到第 k个最大的元素。请注意，你需要找的是数组排序后的第 k 个最大的元素，而不是第 k 个不同的元素。你需要给出具体的**算法思路**、**伪代码**，并设计一个时间复杂度为$\mathbf{O(n)}$的算法（可以是平均或最坏情况。若能清晰写出最坏情况复杂度为$\mathbf{O(n)}$的算法，有额外分）。

答：

算法思路：

可以利用快速排序解决这个问题，先对原数组排序，再返回倒数第k个位置，这样平均时间复杂度是 $O(n \log n)$，但是还可以更快。

回顾一下快速排序，将数组$a[l:r]$划分成两个子数组$a[l:q-1]$和$a[q+1:r]$，每一轮划分都会确定一个元素的最终位置，假设元素x的最终位置是q，那么数组$a[l:q-1]$ 中的每个元素小于等于 $a[q] = x$，且 $a[q] = x$ 小于等于 $a[q+1:r]$ 中的每个元素。

因此可以改进快速排序算法来解决这个问题：在分解的过程当中，对子数组进行划分，

如果划分得到的 q 正好就是需要的下标，就直接返回 $a[q]$；否则，如果 q 比目标下标小，就递归右子区间，否则递归左子区间。这样就可以把原来递归两个区间变成只递归一个区间，提高效率。快速排序的性能和划分出的子数组的长度密切相关，最坏情况是$O(n^2)$，但是如果引入随机化的过程，期望的时间代价是$O(n)$。

伪代码：

Algorithm 1 分治法求两有序数组的中位数

Input: 数组 A, 整数 k;

Output: 数组 A中第 k大的元素;

function PARTITION($int\ X[], int\ l, int\ r$)
    $x \leftarrow X[r],\ i \leftarrow l-1$
    for $j = l \rightarrow r$ **do**
        if $X[j] <= x$ **then**
            $swap(X[++i],\ a[j])$
        end if
    end for
    $swap(X[i+1],\ a[j])$
    return $i+1$
end function
function QUICKSELECT($int\ X[], int\ l, int\ r, int\ index$)
    $q \leftarrow l;$
    if $q == index$ **then**
        return $X[q];$
    else
        return $q < index?$ QUICKSELECT$(X, q+1, r, index)$: QUICKSELECT$(X, l, q-1, index)$
    end if
end function

下面给出最坏时间复杂度为O(n)的算法$Select(A,k)$求第k小元素的思路（类似，同学们可思考求第k大元素的思路）：

(1)将n个元素分为$\lceil \frac{n}{5} \rceil$组，每组5个元素，每组排序；代价($O(n)$)

(2)将每组第3个元素取出，得到大小为$\lceil \frac{n}{5} \rceil$的数组$M$；

(3)最后一组中的元素可能不足5个（1个取出，2个取较小；3个取中间，4个取第二小）；

(4)$m = Select(M, \left\lceil \frac{|M|}{2} \right\rceil)$；代价$T(\lceil \frac{n}{5} \rceil)$

(5)A中至少有$3(\lceil \frac{1}{2} \lceil \frac{n}{5} \rceil \rceil - 2) \geq \frac{3n}{10} - 6$个元素大于等于$m$；

(6)类似地，A中至少有$\frac{3n}{10} - 6$个元素小于m；

(7)依次扫描整个数组；($A[i] < m$时放入A_1，$A[i] = m$时放入A_2,$A[i] > m$时放入A_3)；代价($O(n)$)

(8)当$k \leq |A_1|$时，调用$Select(A_1, k)$；(A_1大小最多为$n - (\frac{3n}{10} - 6) = \frac{7n}{10} + 6$，代价($T(\frac{7n}{10}+6)$))

(9)当$|A_1| < k \leq |A_1| + |A_2|$时，返回$m$；

(10)当$k > |A_1| + |A_2|$时，调用$Select(A_3, k - |A_1| - |A_2|)$；($A_3$大小最多为$n - (\frac{3n}{10} - 6) = \frac{7n}{10} + 6$)；代价($T(\frac{7n}{10} + 6)$)

复杂度：

$$T(n) = T(\left\lceil \frac{n}{5} \right\rceil) + T(\frac{7n}{6}) + O(n) \leq O(n)$$

题目3：动手设计并实现一个可以进行两个n位大整数的乘法运算的算法。你需要给出具体的**算法思路、伪代码**，并对你设计的算法进行**复杂度分析**，除此之外，你还需要给出**算法运行结果截图**，并用你熟悉的图形库**画出输入规模n与运行时间的关系图**。

答：答案写在这里

算法思路：

参考此blog

伪代码：

Algorithm 2 分治法求大整数乘法(假设X和Y都是n位十进制整数)

Input: 大整数 X, 大整数 Y, 长度 n;

Output: X和Y相乘的乘积;

function $\text{Sign}(long\ A)$
    return $A > 0?\ 1 : -1$
end function
function $\text{calculateSame}(long\ X,\ long\ Y,\ int\ n)$
    $sign \leftarrow \text{Sign}(X) * \text{Sign}(Y)$
    $X = abs(X)$
    $Y = abs(Y)$
    if $X == 0\ ||\ Y == 0$ **then**
        return 0
    else if $n == 1$ **then**
        return $sign * X * Y$
    else
        $A = (X/10^{\frac{n}{2}})$
        $B = (X\%10^{\frac{n}{2}})$
        $C = (Y/10^{\frac{n}{2}})$
        $D = (Y\%10^{\frac{n}{2}})$
        $AC = \text{calculateSame}(A, C,\ \frac{n}{2})$
        $BD = \text{calculateSame}(B, D,\ \frac{n}{2})$
        $ABCD = \text{calculateSame}((A - B), (D - C),\ \frac{n}{2}) + AC + BD$
        return $sign * (AC * 10^n + ABCD * 10^{\frac{n}{2}} + BD)$
    end if
end function

Algorithm 3 分治法求大整数乘法(假设X和Y位数不相等)

Input: 大整数 X, 大整数 Y, X位数 xn, Y位数 yn;

Output: X和Y相乘的乘积;

function CALCULATEUNSAME($long\ X,\ long\ Y,\ int\ xn,\ int\ yn$)
    if $X == 0 \mid\mid Y == 0$ **then**
        return 0
    else if $((xn == 1\ \&\&\ yn == 1) \mid\mid xn == 1 \mid\mid yn == 1)$ **then**
        return $X * Y$
    else
        $xn0 = xn/2,\ yn0 = yn/2$
        $xn1 = xn - xn0,\ yn1 = yn - yn0$
        $A = (X/10^{xn0})$
        $B = (X\%10^{xn0})$
        $C = (Y/10^{yn0})$
        $D = (Y\%10^{yn0})$
        $AC =$ CALCULATEUNSAME($A, C,\ xn1,\ yn1$)
        $BD =$ CALCULATEUNSAME($B, D,\ xn0,\ yn0$)
        $ABCD =$ CALCULATEUNSAME($(A * 10^{xn0} - B), (D - C * 10^{yn0}), xn1, yn1$)
        return $2 * AC * 10^{xn0+yn0} + ABCD + 2 * BD$
    end if
end function

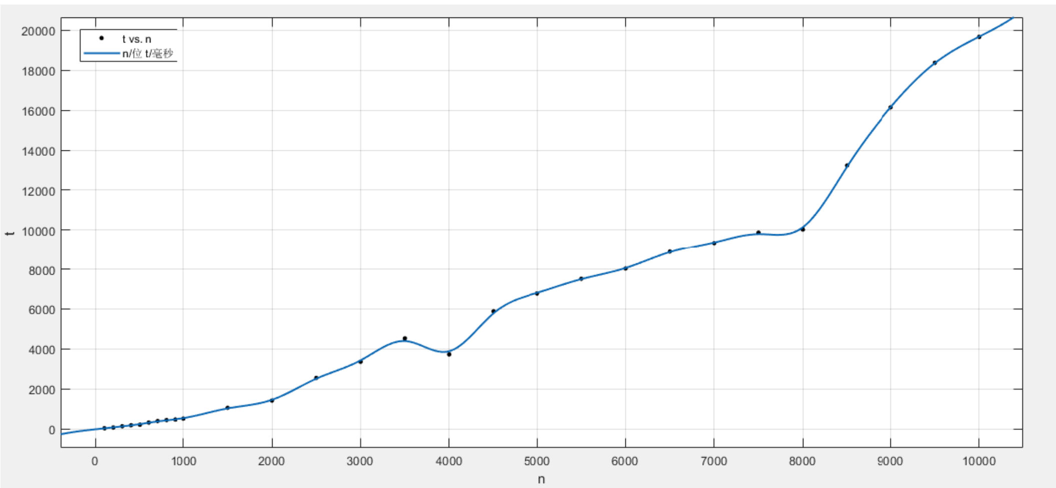

Figure 1: (图为刘诺昂同学所画关系图（供参考）)