hw3

> 4.2-3，4.2-5；4.4-5，4.4-7；4.5-2，4.5-4；

4.2-3

4.2-3 如何修改 Strassen 算法，使之适应矩阵规模 n 不是 2 的幂的情况？证明：算法的运行时间为 $\Theta(n^{\lg 7})$。

对输入的矩阵进行扩展，使其为2的幂次，并然后运行给定的算法。扩展到下一个最大的2的幂次（称之为m）最多会将n的值翻倍，因为每一个2的幂次都相差两倍。因此，此算法的运行时间为

$$m^{\lg 7} \leq (2n)^{\lg 7} = 7n^{\lg 7} \in O\left(n^{\lg 7}\right)$$

以及

$$m^{\lg 7} \geq n^{\lg 7} \in \Omega\left(n^{\lg 7}\right)$$

综合这两者，得到运行时间是 $\Theta\left(n^{\lg 7}\right)$。

4.2-5

4.2-5 V. Pan 发现一种方法，可以用 132 464 次乘法操作完成 68×68 的矩阵相乘，发现另一种方法，可以用 143 640 次乘法操作完成 70×70 的矩阵相乘，还发现一种方法，可以用 155 424次乘法操作完成 72×72 的矩阵相乘。当用于矩阵相乘的分治算法时，上述哪种方法会得到最佳的渐近运行时间？与 Strassen 算法相比，性能如何？

取这三种算法，并将乘法的次数除以矩阵的边长的lg(7)次幂，可得以下值：
3745
3963
4167
这说明如果作为Strassen算法的基本情况使用，第一个算法将对非常大的矩阵表现得最好。

4.4-5

4.4-5 对递归式 $T(n)=T(n-1)+T(n/2)+n$，利用递归树确定一个好的渐近上界，用代入法进行验证。

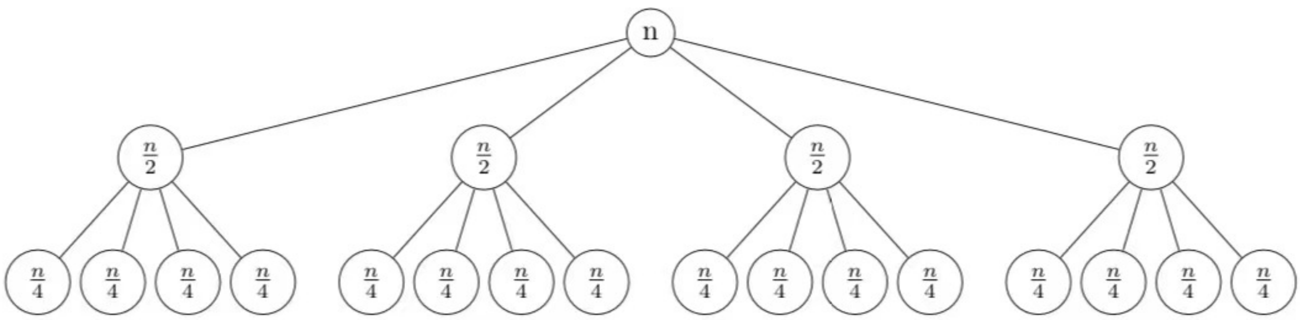
递归树看起来像一个长分支，从中跳出的分支直接跳到一半的位置。这看起来像一个非常满的树，猜测运行时间是 $O\left(n^2\right)$。为了使用代入法来验证这一点，尝试证明 $T(n) \leq 2^n$。

$$
\begin{aligned}
T(n) &= T(n-1) + T\left(\frac{n}{2}\right) + n \\
&\leq 2^{n-1} + \sqrt{2^n} + n \\
&\leq 2^n
\end{aligned}
$$

为了证明它是一个非常紧的界限，将证明不能有任何多项式的上界。即如果有 $T(n) \leq cn^k$，那么将其代入递推式时，得到 n^k 的新系数可以高达 $c\left(1+\frac{1}{2^k}\right)$，无论如何选择 c 都比 c 大。

4.4-7

4.4-7 对递归式 $T(n)=4T(\lfloor n/2 \rfloor)+cn$（$c$ 为常数），画出递归树，并给出其解的一个渐近紧确界。用代入法进行验证。



通过简单的代入法看出答案是 $\Theta\left(n^2\right)$。假设 $T(n) \leq c'n^2$，则

$$
\begin{aligned}
T(n) &= 4T\left(\lfloor n/2 \rfloor\right) + cn \\
&\leq 4c'\left(\frac{n}{2}\right)^2 + cn \\
&= c'n^2 + cn
\end{aligned}
$$

当 $c' + \frac{c}{n} \leq 1$ 时，这是 $\leq c'n^2$，对于足够大的 n 这是成立的，只要 $c' < 1$。可以用类似的方法来证明它也被 n^2 下界所限制。

4.5-2

4.5-2 Caesar 教授想设计一个渐近快于 Strassen 算法的矩阵相乘算法。他的算法使用分治方法，将每个矩阵分解为 $n/4 \times n/4$ 的子矩阵，分解和合并步骤共花费 $\Theta(n^2)$时间。他需要确定，他的算法需要创建多少个子问题，才能击败 Strassen 算法。如果他的算法创建 a 个子问题，则描述运行时间 $T(n)$的递归式为 $T(n)=aT(n/4)+\Theta(n^2)$。Caesar 教授的算法如果要渐近快于 Strassen 算法，a 的最大整数值应是多少？

Strassen算法的运行时间是 $Theta\left(n^{\lg 7}\right)$. 选择 $a=48$，满足 $\log_4(a)<\lg 7$ 的最大整数。由于 $2<\log_4(48)$，因此存在 $\epsilon>0$ 使得 $n^2<n^{\log_4(48)-\epsilon}$. 根据Master定理的第1种情况，$T(n)=\Theta\left(n^{\log_4(48)}\right)$，这比 $\Theta\left(n^{\lg 7}\right)$ 在渐近意义上要好。

4.5-4

4.5-4 主方法能应用于递归式 $T(n)=4T(n/2)+n^2\lg n$ 吗？请说明为什么可以或者为什么不可以。给出这个递归式的一个渐近上界。

在这里不能应用Master方法。注意到 $\log_b a=\log_2 4=2$ 且 $f(n)=n^2\lg n$. 很明显，情况1和2都不适用。此外，虽然 f 在渐近意义上大于 n^2，但它不是多项式级别的更大，所以情况3也不适用。将证明 $T(n)=O\left(n^2\lg^2 n\right)$. 为此将归纳证明 $T(n)\le n^2\lg^2 n$.

$$
\begin{aligned}
T(n) &= 4T\left(\frac{n}{2}\right)+n^2\lg n\\
&\le 4\left(\left(\frac{n}{2}\right)^2\lg^2\left(\frac{n}{2}\right)\right)+n^2\lg n\\
&= n^2(\lg n-\lg 2)^2+n^2\lg n\\
&= n^2\lg^2 n-n^2(2\lg n-1-\lg n)\\
&= n^2\lg^2 n-n^2(\lg n-1)\\
&\le n^2\lg^2 n
\end{aligned}
$$

前提是 $n\ge 2$.