**算法设计与分析网络学习第二次课程学习指南**

**时间**：2020年2月27日(星期四)上午10:00（3-4节）

**课堂安排**： 10:00-11:15观看视频，做练习题和思考题（https://www.icourse163.org/course/HIT-356006）

11:15-11:45 答疑、讨论

(腾讯会议号码：543119465，密码0227)

**学习内容**：算法设计与分析的数学基础

**1. 授课视频**

算法设计与分析(基础篇) 第二讲 2.1-2.3

**2. 阅读**

算法导论(第三版) 第三章，4.3-4.5节，4.6节(选读)

**3. 练习题**

1. 证明或否证：*f* (*n*)+o(*f*(*n*))=Θ(*f*(*n*))
2. 证明：Θ(*f*(*x*)) + O(*g*(*x*)) = O( max(*f*(*x*), *g*(*x*))).
3. 证明或给出反例：Θ(*f*(*n*))∩o(*f*(*n*))=∅
4. 证明：设*k*是任意常数正整数，则*logkn* = o(*n*)
5. 证明：*logn*!=Θ(*nlogn*)
6. 用迭代法解方程 T(*n*)＝T(9*n*/10)+*n*
7. 解方程 T(*n*)=6T(*n*/3)+*logn*
8. 解方程 T(*n*) =3T(*n*/3 +5) + *n*/2
9. 解方程 T(*n*)= T (⎡*n*/2⎤ )+1
10. 解方程 T(*n*)=9T(*n*/3)+*n*
11. 解方程 T(*n*)=T( ⎣*n*/2⎦ )+*n*3
12. 证明或否证：如果*f*(*n*)=O(*g*(*n*))，那么2*f*(*n*)=O(2*g*(*n*))
13. 令*f*, *g*和*h*为定义在正整数上的正实数函数。假设*f*(*n*)=*O*(*g*(*n*))，*g*(*n*)=*O*(*h*(*n*))，证明*f*(*n*)=*O*(*h*(*n*)).
14. 将下列函数按它们在n→∞时的无穷大阶数，从小到大排序。

*n*, *nlg*7, *n*2.5, *n*3*lgn*, *n*2*lgn*, *nlgn*

1. 一个时间复杂性为O(*n*2)的算法的运行时间一定大于一个时间复杂性为O(n)的算法么？为什么？

**4. 思考题**

维基百科中的Master定理描述和算法导论不同，思考其差别以及哪个正确？(维基百科中的Master定理描述如下)

