Q1.（$5 \times 7 = 35$ 分）判断以下命题的正误。若正确，请给出证明；若错误，请给出反例。若无额外说明，本题中出现的所有函数均渐近非负。

1. 若 $\forall i \in \mathbb{N}^+, f_i(n) = O(g(n))$，定义 $F_m(n) = \sum_{i=1}^{m} f_i(n)$，则 $\forall m \in \mathbb{N}^+, F_m(n) = O(g(n))$
2. 若 $\forall i \in \mathbb{N}^+, f_i(n) = O(g(n))$，定义 $F(n) = \sum_{i=1}^{n} f_i(n)$，则 $F(n) = O(g(n))$
3. 若 $f_1(n), g(n)$ 各点函数值均为正，且 $f_1(n) = o(g(n))$，则存在无穷函数列 $\{f_1, f_2, \dots\}$，使得 $\forall i \in \mathbb{N}^+, f_i(n) = o(f_{i+1}(n))$ 且 $f_i(n) = o(g(n))$
4. 若 $f(n) + g(n) = \Omega(h(n))$，则 $f(n), g(n)$ 中至少有一个属于 $\Omega(h(n))$
5. 若存在 $0 \leq \alpha \leq \beta$ 使得 $f(n) = \Omega(n^\alpha)$ 且 $f(n) = O(n^\beta)$，则 $f(n+1) = \Theta(f(n))$

Q2.（$15 + 10 + 10 = 35$ 分）函数 $A : \mathbb{N}^2 \to \mathbb{N}$ 在一些算法的时间复杂度分析中具有重要应用，其定义如下：

$$\begin{cases} A(0, n) = n + 1 & ,\text{对于} n \geq 0 \\ A(m+1, 0) = A(m, 1) & ,\text{对于} m \geq 0 \\ A(m+1, n+1) = A(m, A(m+1, n)) & ,\text{对于} m, n \geq 0 \end{cases}$$

(1) 证明 A 是良定义的，即对于任意 $m, n \in \mathbb{N}$，$A(m, n)$ 的递归定义总能终止。

(2) 证明 $A(m, n)$ 关于 m, n 分别单调递增，即：

$$A(m+1, n) > A(m, n)$$
$$A(m, n+1) > A(m, n)$$

(3) $\alpha(x)$ 定义为使得 $A(n, n) \leq x$ 的最大自然数 n。证明：

$$\alpha(x) = \omega(1)$$
$$\alpha(x) = O\left(\lg^\star x\right)$$

其中 $\lg^\star x$ 为迭代对数函数。

Q3.（$25 \times 1 + 5 = 30$ 分）表1列出了 25 个形如 “$T(n) = aT(\frac{n}{b}) + f(n)$” 的递推式。

(1) 判断它们能否使用主定理得到渐近阶，如果可以，求出渐近阶，否则说明原因。

(2) 对于有限数列 $\mathbf{e}=(e_0,e_1,\ldots,e_m), m\in\mathbb{N}, e_m\neq 0$，定义由 $\mathbf{e}$ 导出的 Exponential-Logarithmic 函数：

$$\mathrm{EL}^{\mathbf{e}}(x)=\prod_{i=0}^{m}\left(\lg^{(i)}x\right)^{e_i}$$

定义 $\mathbf{e}$ 除首项外的首个不为 -1 的项 $e_{\mathrm{cord}(\mathbf{e})}=\mathrm{cpow}(\mathbf{e})$。特别地，若 $\forall i>0, e_i=-1$，则定义 $\mathrm{cord}(\mathbf{e})=m+1, \mathrm{cpow}(\mathbf{e})=0$。

不加证明地给出以下定理：

对于递推式 $T(n)=aT(\frac{n}{b})+\mathrm{EL}^{\mathbf{e}}(n), a,b>1$，记 $\alpha=\log_b a$，则：

$$T(n)=\begin{cases}\Theta\left(\mathrm{EL}^{\mathbf{e}}(n)\right) & ,\text{若} e_0>\alpha\\ \Theta\left(\mathrm{EL}^{\mathbf{e}}(n)\prod_{i=1}^{\mathrm{cord}(\mathbf{e})}\lg^{(i)}n\right) & ,\text{若} e_0=\alpha \text{且} \mathrm{cpow}(\mathbf{e})>-1\\ \Theta\left(n^{\alpha}\right) & ,\text{其他情况}\end{cases}$$

写出表中第 7、23、24、25 个 $f(n)$ 所对应的数列 $\mathbf{e}$，并用上述定理求出 $T(n)$ 的渐近阶。

***Q4.** 对于 $n\in\mathbb{N}^+$，定义：

$$\mathbb{S}_n=\left\{\langle a,b\rangle \,\middle|\, n=\sum_{i=a}^{b}i, 0<a\leq b\right\}$$

(1) 设计算法，以常数时间复杂度构造性地给出 n 的正奇因子集 $\tilde{\mathbb{F}}_n$ 到 $\mathbb{S}_n$ 的一个双射以及其逆映射。具体而言：

- 设计一个算法，输入 n 以及 $f\in\tilde{\mathbb{F}}_n$，在 $O(1)$ 时间内输出一个 $\langle a,b\rangle\in\mathbb{S}_n$。当 n 固定时，该算法必须构成 $\tilde{\mathbb{F}}_n$ 到 $\mathbb{S}_n$ 的双射。
- 设计一个算法，输入 n 以及 $\langle a,b\rangle\in\mathbb{S}_n$，在 $O(1)$ 时间内输出一个 $f\in\tilde{\mathbb{F}}_n$。当 n 固定时，该算法必须构成 $\mathbb{S}_n$ 到 $\tilde{\mathbb{F}}_n$ 的双射。
- 当 n 固定时，你设计的两个算法必须互为逆映射。

(2) 算法1用于计算 $|\mathbb{S}_n|$，解释其正确性，并证明：若将问题规模 m 定义为输入值 n 的二进制位数，即 $m=\lfloor\lg n\rfloor+1$，则算法1的最坏时间复杂度为 $\Theta\left(2^{\frac{m}{2}}\right)$。你可能需要用到Bertrand 公设。

(3) 思维足够敏锐的同学可能已经发现，每当找到一个 n 的素因数，外层循环的结束就会被提前，因此该算法似乎在大多数情况下“并没有那么糟”。在对快速排序的分析中，我们使用“平均时间复杂度”的概念刻画了这一特性，而对于这个问题，我们将采用一种不同的手法。

令 X_i 为 $[2^{i-1}, 2^i)$ 上均匀取值的整值随机变量，记算法在 X_i 上的运行时间为 $T(X_i)$，定义 $L_i = \log_{X_i} T(X_i)$。若可以证明，对于算法1，随机变量列 $\{L_i\}$ 依分布收敛于某随机变量 L，则在某种程度上，L 的分布能够给出对算法1渐近性能更精细的刻画。

(3.1) 对于某个算法，若已知 $P(\alpha \le L \le \beta) = 1$，在不进行任何其他分析的情况下，能够断言这一算法的最坏时间复杂度是 $\Omega\left(2^{\alpha m}\right) \cap O\left(2^{\beta m}\right)$ 吗？若我们已知 $\mathbb{E}(L) = \gamma$，能够断言这一算法的平均时间复杂度是 $\Theta\left(2^{\gamma m}\right)$ 吗？

(3.2) 对算法1的 $\{L_i\}$ 收敛性的证明以及对 L 的分布的推导超出了本课程的要求，因此这里仅需要你依照文献中已有的结论进行数值计算，体会算法1的渐近性能。请阅读这篇文章，利用文章中的相关结论画出 L 的概率密度函数图像，并计算概率密度的最大值点及相应的最大值（精确到小数点后 2 位）。

Algorithm 1: Calculate $|\mathbb{S}_n|$

Input: n

Output: $|\mathbb{S}_n|$

```
while 2 | n do
    n ← n/2;
end
S ← 1;
p ← 3;
while p^2 ≤ n do
    e ← 0;
    while p | n do
        n ← n/p;
        e ← e + 1;
    end
    S ← (e + 1)S;
    p ← p + 1;
end
if n ≠ 1 then
    S ← 2S;
end
return S
```

No.	a	b	$f(n)$
1	3	2	n^2
2	4	2	n^2
3	1	2	2^n
4	2^n	2	n^n
5	16	4	n
6	2	2	$n \lg n$
7	2	2	$\frac{n}{\lg n}$
8	2	4	$n^{0.51}$
9	0.5	2	$\frac{1}{n}$
10	16	4	$n!$
11	$\sqrt{2}$	2	$\lg n$
12	3	2	n
13	3	3	$\sqrt{n}$
14	4	2	$cn, c \geq 0$
15	3	4	$n \lg n$
16	3	3	$\frac{n}{2}$
17	6	3	$n^2 \lg n$
18	4	2	$\frac{n}{\lg n}$
19	64	8	$-n^2 \lg n$
20	7	3	n^2
21	4	2	$\lg n$
22	1	2	$n(2 - \cos n)$
23	2	2	$n \lg n \lg \lg n$
24	2	2	$n(\lg \lg n)^r, r \neq 0$
25	2	2	$n \frac{(\lg \lg \lg n)^s}{\lg n \lg \lg n}, s \neq 0$

表 1: 递推式