

2025 秋季学期 算法设计与分析 课后作业

任课教师: 陈宇

作业提交注意事项

- 格式要求: 电子版, 格式统一为 pdf, 推荐使用 LaTeX 排版. 不接受手写扫描, 手写文字识别不清会有误判可能.
- 道德要求: 独立完成, 不可以合作讨论, 严禁抄袭!!! 书面作业和编程作业若发现无法解释的雷同, 将一律按作弊处理, 课程成绩判零并上报教务.
- 提交方式: 将作业 pdf 直接作为附件发送至邮箱 `sdu_algorithms_ta@163.com`, 邮件的主题格式请设置为“姓名 + 学号 + 2025 秋季算法作业”
- 提交截止时间: 2025.12.20

1 算法分析技术

1.1 假设 f 和 g 是定义在自然数集合上的函数, 若对某个其他函数 h 有 $f = O(h)$ 和 $g = O(h)$ 成立, 那么证明 $f + g = O(h)$

1.2 设 n, a, b 为正整数, 证明下述性质:

$$\left\lceil \left\lfloor \frac{\lceil \frac{n}{a} \rceil}{b} \right\rfloor \right\rceil = \left\lceil \frac{n}{ab} \right\rceil, \quad \left\lfloor \left\lceil \frac{\lfloor \frac{n}{a} \rfloor}{b} \right\rceil \right\rfloor = \left\lfloor \frac{n}{ab} \right\rfloor$$

1.3 对于下面每个函数 $f(n)$, 用 Θ 符号表示成 $f(n) = \Theta(g(n))$ 的形式, 其中 $g(n)$ 要尽可能简洁. 比如 $f(n) = n^2 + 2n + 3$ 可以写成 $f(n) = \Theta(n^2)$. 然后按照阶递增的顺序将

这些函数进行排列:

$$(n-2)!, 5 \log(n+100)^{10}, 2^{2n}, 0.001n^4 + 3n^3 + 1, (\ln n)^2$$

$$n^{1/3} + \log n, 3^n, \log(n!), \log(n^{n+1}), 1 + \frac{1}{2} + \cdots + \frac{1}{n}$$

1.4 求解以下递推方程:

$$\begin{cases} T(n) = T(n-1) + \log 3^n \\ T(1) = 1 \end{cases}$$

1.5 求解以下递推方程:

$$\begin{cases} T(n) = 5T(\frac{n}{2}) + (n \log n)^2 \\ T(1) = 1 \end{cases}$$

1.6 求解以下递推方程:

$$\begin{cases} T(n) = T(n-1) + \frac{1}{n} \\ T(1) = 1 \end{cases}$$

2 排序类算法

2.1 设 $A = \{a_1, a_2, \dots, a_n\}$, $B = \{b_1, b_2, \dots, b_m\}$ 是整数集合, 其中 $m = O(\log n)$. 设计一个算法找出集合 $C = A \cap B$. 要求给出算法伪码描述和复杂度分析.

2.2 设 A 是 n 个数构成的数组, 其中出现次数最多的数称为众数. 设计一个算法求 A 的众数, 给出伪码和最坏情况下的复杂度.

2.3 给定含有 n 个不同的数的数组 $L = \langle x_1, x_2, \dots, x_n \rangle$. 如果 L 中存在 x , 使得 $x_1 < x_2 < \cdots < x_{i-1} < x_i > x_{i+1} > \cdots > x_n$, 则称 L 是单峰的, 并称 x_i 是 L 的“峰顶”. 假设 L 是单峰的, 设计一个算法找到 L 的峰顶, 要求给出算法伪码描述和复杂度分析.

2.4 设 S_1, S_2, \dots, S_k 是整数集合, 其中每个集合 S_i (其中 $1 \leq i \leq k$) 中整数取值范围是 $1 \sim n$, 且 $\sum_{i=1}^k |S_i| = n$, 试设计一个算法, 在 $O(n)$ 时间内将 S_1, S_2, \dots, S_k 分别排序.

3 分治算法

3.1 设 A 是 n 个非 0 实数构成的数组, 设计一个算法重新排列数组中的数, 使得负数都排在正数前面. 要求算法使用 $O(n)$ 的时间和 $O(1)$ 的空间.

3.2 设 S 是 n 个不等的正整数集合, n 为偶数, 给出一个算法将 S 划分为子集 S_1 和 S_2 , 使得 $|S_1| = |S_2| = n/2$, 且 $|\sum_{x \in S_1} x - \sum_{x \in S_2} x|$ 达到最大, 即使得两个子集元素之和的差达到最大.

3.3 设 A 和 B 都是从小到大已经排好序的 n 个不等的整数构成的数组, 如果把 A 和 B 合并后的数组记作 C , 设计一个算法找出 C 的中位数并给出复杂度分析.

3.4 输入三个正整数 a, p, k , 求 $a^p \bmod k$ 的值. 提示: 由于数据的规模很大, 如果直接计算, 不仅需要采用高精度, 而且时间复杂度很大. 例如 $10^{25} \bmod 7 = 3$, 但 10^{25} 超出了整型数的表示范围, 不能直接计算. 请使用分治策略实现取余运算的算法并给出复杂度分析.

3.5 在 $n(n \geq 3)$ 枚硬币中有一枚重量不合适的硬币 (重量过轻或是过重), 如果只有一架天平可以用来称重且称重的硬币数没有限制, 设计一个算法找出这枚不合适的硬币, 使得称重的次数最少? 给出算法的伪代码描述, 如果每称一次就作为一次基本运算, 分析算法最坏情况下的时间复杂度.

4 贪心算法

4.1 若在 0-1 背包问题中, 各物品依重量递增排列时, 其价值恰好依递减序排列. 对于这个特殊的 0-1 背包问题, 设计一个有效算法找出最优解, 并说明算法的正确性.

4.2 将最优装载问题的贪心算法推广到两艘船的情形, 贪心算法仍然能产生最优解么? 若能, 给出证明. 若不能, 请给出反例.

4.3 设 $\Gamma = \{1, \dots, n\}$ 是 n 个字符的集合. 证明关于 Γ 的任何最优前缀码可以表示长度为 $2n - 1 + n \lceil \log n \rceil$ 位的编码序列. (提示: 先考虑树结构的编码, 再考虑叶结点对应字符的编码)

4.4 设 $A = \langle a_1, a_2, \dots, a_n \rangle$, $B = \langle b_1, b_2, \dots, b_m \rangle$ 是两个序列, 其中 $m \leq n$, 设计一个 $O(n)$ 时间的算法, 判断 B 是否为 A 的子序列, 说明算法的设计思路, 给出伪代码并证明算法的正确性.

5 动态规划

5.1 图书馆大门前有 n 级台阶, 你每次跨上 1 级或者 2 级, 请问登上 n 级台阶总共有多少种不同的方法? 设计一个算法求解上述问题, 尝试写出公式, 说明算法设计思想和时间复杂度.

5.2 n 种币值 v_1, v_2, \dots, v_n 和总钱数 M 都是正整数. 如果每种币值的钱币至多使用 1 次, 问: 对于 M 是否可以有一种找零钱的方法? 设计一个算法求解上述问题. 说明算法设计思想, 分析算法最坏情况下的时间复杂度.

5.3 设 P 是一台高性能服务器, $T = \{1, 2, \dots, n\}$ 是 n 个计算任务集合, a_i 表示任务 i 所申请的计算资源. 已知服务器的最大计算资源是正整数 K . 请确定 T 的一个子集 S ,

使得 $\sum_{i \in S} a_i \leq K$, 且 $K - \sum_{i \in S} a_i$ 的值达到最小. 请设计一个算法求解 S , 并分析最坏情况下的时间复杂度.

5.4 设计一个 $O(n^2)$ 时间的算法, 找出由 n 数组成的序列的最长单调递增子序列.

6 算法复杂性初步

6.1 证明: $\mathcal{P} \subseteq \mathcal{NP}$.

6.2 证明: 如果存在 \mathcal{NP} 难的问题 $\Pi \in \mathcal{P}$, 则 $\mathcal{P} = \mathcal{NP}$.

7 编程作业

请在 Virtual Judge 上完成以下编程作业 (Virtual Judge 的操作说明见后). 该部分成绩按照题目通过率评分. 编程作业截止时间为 2025 年 12 月 20 日 24:00.

编程作业链接: <https://vjudge.net/contest/756914>

分治算法

- Median: POJ-2388, POJ-1723
- Ultra-QuickSort: POJ-2299
- Binary Search: POJ-1064, POJ-2456

贪心算法

- Interval problem: POJ-3069
- SSSP (Dijkstra, Bellman-Ford, Floyd): POJ-1860, POJ-3259
- MST (Prim, Kruskal): POJ-1258, POJ-2485

动态规划

- Knapsack Problem: POJ-1837, POJ-3264, POJ-1276
- Largest Common String: POJ-1934
- Longest Increasing Subsequence: POJ-3903

回溯算法

- POJ-1753, POJ-2965, POJ-1321

8 附录：Virtual Judge 使用说明

8.1 Virtual Judge Contest 密码：Autumn2025

8.2 Virtual Judge 个人账号昵称 (Nickname) 要求

要求：学号 + 姓名

注：若账号昵称 (Nickname) 未按要求修改，则编程实践部分成绩作废

例如：同学张三学号为 201905130170，那么他的昵称 (Nickname) 应为 201905130170 张三

展示如下图所示：

 SDUAlgorithms 	24 hours	7 days	30 days	Overall solved	Overall attempted	Detail	Export
201905130170张三	0	0	0	0	0	Toggle	AC / ALL

8.3 昵称修改方法

方法一

在 Register (注册) 界面中有 Nickname 的填写，可以在此处直接填写学号 + 姓名

Register

* Username:

IMPORTANT: Once registered, you can't change it !

* Password:

* Repeat Pass:

Repeat password above

Nickname:

School:

QQ:

* Email:

Invisible to others. Please fill a real one. You need it after you forget the password.

* Captcha:

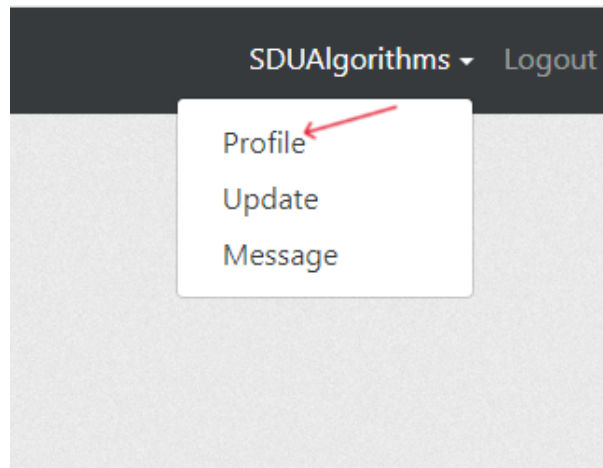
Introduction:

Your brief introduction in Markdown

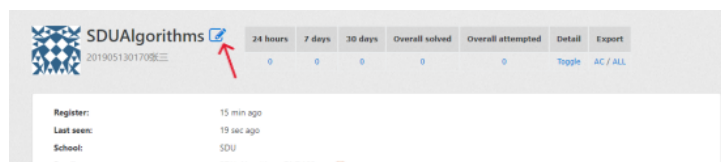
Cancel

Register

方法二



用户登录后点击右上角用户名会出现 Profile 等选项，点击 Profile，进入个人界面
个人界面中点击用户名右侧的修改选项，进入个人信息修改界面，更改 Nickname 为学号 + 姓名，填写验证码并点击 update，即可修改成功

A screenshot of the 'Update' form on SDUAlgorithms. The form is titled 'Update' and has a close button (X) in the top right corner. It contains several input fields: 'Username' (SDUAlgorithms), 'Original Password' (masked with dots), 'Password' (empty), 'Repeat Password' (empty), 'Nickname' (201905130170张三), 'School' (SDU), 'QQ' (empty), 'Email' (SDU_Algorithms_TA@163.com), and 'Captcha' (empty). There is a red arrow pointing to the 'Update' button at the bottom right. The 'Update' button is blue with white text. There is also a 'Cancel' button next to it.