2021 年算法设计与分析期中考试试卷

答题要求：解答算法设计题目时，请先用一段话描述算法思想。若用动态规划算法，请写出递推方程、边界条件、标记函数等设计要素；贪心法需给出证明；回溯法需给出解向量、搜索树、约束条件、优化算法等；各种算法需分析时间复杂度。阅卷时会根据算法的正确性和效率评分。

一、（15 分）在下表中填入“Y”（代表“是”）或者“N”（代表“否”），其中 $k>0$ 和 $c>1$ 是常数。

$f(n)$	$g(n)$	$f(n)=O(g(n))$	$f(n)=o(g(n))$	$f(n)=\Theta(g(n))$
n^2	$\log(c^n)$	N	N	N
$\sqrt{n}$	$n^{\sin n}$	N	N	N
$n2^n$	2^{2n}	Y	Y	N
$n^{\log\log n}$	$c^{\log n}$	N	N	N
$\log(n!)$	$\log(n^n)$	Y	N	Y

二、　　（10 分）设原问题的规模是 n，从下述三个算法中选择一个最坏情况下

时间复杂度最低的算法，简要说明你的理由。

算法 A：将原问题划分规模减半的 6 个子问题，递归求解每个子问题，然后在线性时间将子问题的解合并得到原问题的解.

算法 B：先递归求解 2 个规模为 n-1 的子问题，然后在常量时间内将子问题的解合并.

算法 C：将原问题划分规模为 $n/3$ 的 9 个子问题，递归求解每个子问题，然后在 $O(n^3)$时间将子问题的解合并得到原问题的解.

解：选择算法 *A*。（1 分）

算法 *A*：$T_A(n)=6T_A(n/2)+O(n)$，　$T_A(n)=\Theta(n^{\log 6})$　　（3 分）

算法 *B*：$T_B(n)=2T_B(n-1)+O(1)$，　$T_B(n)=\Theta(2^n)$　　（3 分）

算法 *C*：$T_C(n)=9T_C(n/3)+O(n^3)$，　$T_C(n)=\Omega(n^3)$　　（3 分）

三、 （10 分）环球影城开业在即，小明提前做攻略。环球影城一共有 n 个游乐项目，每个项目都要排队，假定每个项目预期的排队时间为t_i，实际游玩时间为p_i。假定小明可以在游乐场停留的最长时间为 M。小明的目标是在 M 时间内尽可能多玩一些项目，同时总排队时间不能超过总实际游玩时间的 1/2，请你帮他设计一个攻略。假设其他时间（如从一个项目到另外一个项目的时间）忽略不计。

解法一：可以转化为线性规划问题求解，最高可得 10 分

假设每个项目用x_i表示，$x_i = 1$代表选择第 i 个项目，$x_i = 0$代表没有选。

目标函数： $\max\sum_{i=1}^{n} x_i$ （2 分）

约束条件： $\sum_{i=1}^{n} x_i\,(t_i + p_i) \le M$

$\sum_{i=1}^{n} x_i\, t_i \le 0.5 * \sum_{i=1}^{n} x_i\, p_i$

$x_i = 0，1$

然后按照线性规划求解，再求整数解。

解法二：回溯算法，最高可得 8 分；

解法三：动态规划，最高可得 10 分；

解法四：贪心法，最高可得 5 分。

四、　（15 分）小明前往游乐园游玩，游乐园共有 n 个游戏项目，每个游戏项目有两个属性a_i和b_i。小明希望玩遍所有项目，他可以自由安排顺序，玩一个游戏项目所需要付出的精力为这个游戏项目之前所有项目的a_i的乘积除以当前游戏项目的b_i，第一个项目不消耗精力。小明不希望在一个项目上花过多的精力，他希望付出的精力最多的项目所需要的精力最小。求满足小明要求的安排顺序及对应的最多精力。请设计算法，并证明算法正确性，分析算法复杂度。假设a_i都大于等于 1，b_i都大于 0。

参考答案：使用贪心法(2 分)

1. 算法描述：　按$a_i * b_i$从小到大排序即可（3 分）
2. 复杂度分析：复杂度为 O(nlogn)　　　（1 分）
3. 正确性证明（9 分）
    - 用交换论证法证明。（1 分）
    - 证明思想：从一个最优解出发，在不改变最优性的条件下，转变为按照$a_i * b_i$从小到大排序的解。（1 分）

（1）交换两个相邻的游戏项目不影响其他游戏项目所需的精力。（1 分）

（2）假设最优解排列顺序为 $p_1, \cdots, p_n$,记 $S[i] = a[p_1] * a[p_2] \cdots * a[p_i]$

则第 i 个项目的消耗精力为 $w[i]=S[i-1]/b[p_i]$,

第 i+1 个项目消耗 $w[i+1]=S[i]/b[p_{i+1}] = S[i-1]*a[p_i]/b[p_{i+1}]$

交换第 i 个项目和第 i+1 个项目之后，

$w[i]'=S[i-1]/b[p_{i+1}]$, $w[i+1]'= S[i-1]* a[p_{i+1}]/b[p_i]$

下面证明命题 1：

若 $a[p_i]*b[p_i]>a[p_{i+1}]*b[p_{i+1}]$，则

$\max\{w[i], w[i+1]\} \geqslant \max\{w[i]', w[i+1]'\}$,　　　(1)　　　（2 分）

即交换后两个项目中精力较大值不会升高。

证明：考虑 $w[i], w[i+1], w[i]', w[i+1]'$四项当中都有 $S[i-1]$，那么不等式（1）等价于

$\max\{ 1/b[p_i], a[p_i]/b[p_{i+1}]\} \geqslant \max\{1/b[p_{i+1}], a[p_{i+1}]/b[p_i]\}$　　(2)　　（2 分）

因为 $a[p_i]*b[p_i]>a[p_{i+1}]*b[p_{i+1}]$，且 $a[p_{i+1}]\geqslant 1$，所以 $a[p_i]*b[p_i]>b[p_{i+1}]$，从而 $1/b[p_i] < a[p_i]/b[p_{i+1}]$，即不等式（2）的左边为 $a[p_i]/b[p_{i+1}]$。再看不等式(2)的右边，如果 $1/b[p_{i+1}]\geqslant a[p_{i+1}]/b[p_i]$，那么显然 $a[p_i]/b[p_{i+1}] \geqslant 1/b[p_{i+1}]$，因为 $a[p_i]\geqslant 1$。如果 $1/b[p_{i+1}]<a[p_{i+1}]/b[p_i]$，则有 $a[p_i]/b[p_{i+1}] \geqslant a[p_{i+1}]/b[p_i]$，因为已知 $a[p_i]*b[p_i]>a[p_{i+1}]*b[p_{i+1}]$。

完成不等式（2）的证明得 2 分。

五、（15 分）数独游戏是一种十分简单且受欢迎的游戏。如下图所示，假定给一个 9x9 的棋盘，其中有些方格已经填上了数字。游戏规则是每一行、每一列、每一个小九宫内的数字均含 1-9，且不重复。请设计一个算法对这个问题求解。

	8					5	3	
			5	4				
5	9		8	3			4	
3		9				4	2	
7		4		2		8		5
	2	8				7		3
	3			7	4		5	2
				8	1			
	4	2					8	

解：利用回溯算法来进行求解。

从左往右，从上往下依次搜索空格。假设一共有 n 个空格，那么解向量定义为 $\langle x_1, x_2, x_3, \cdots, x_n \rangle$，每个分量 x_i 的取值代表该空格所填数字，取值范围要保证和当前行、列、九宫格的数字不重复。

搜索树：树高为 n+1 的树，每一层代表一个空格，每一层的分支代表当前空格可以选择的数字。

搜索策略：深度优先，如果没有可填的数字则回溯。

时间复杂度：最坏情况为 $O(9^n)$。

六． (10 分)考虑 n 个互不相等的整数排列在一个环上，请设计一个尽可能高效的算法，找出环上的一个极小点，即数值小于其左右相邻的两个整数的某个整数。

方法一（黄金分割法）:
1、任意取一点 A，再在相对 A 的黄金分割点处取一点 B。由于对称性，不妨假设 A>B（否则交换 A 和 B，只需一次比较）;
2、取 A 的另一个黄金分割点 C（AB 和 AC 是对称的）；把环看作 ABCA 的线段。
3、比较 B 和 C 的大小。
4、如果 C 小于 B，把弧 ACB 看作一个新环（把 B 收缩到 A），递归求解。
5、如果 B 小于 C，把弧 ABC 看作一个新环（从 C 收缩到 A），递归求解。

复杂度 O(logN) (2 分)

方法二（二分法）:
1、任取一点 A，判断 A 是否是极小点；如果是，算法结束;
2、否则，从 A 把环分割转化为一条直线（A 划归到其相邻点更小的一侧）;
3、然后，取线段中点 B，如果 B 大于 A，则递归的在线段 AB 上找;
4、否则，判断 B 是否是极小点；如果是，算法结束;
5、否则，在 B 相邻点更小的一侧的一半的线段上递归查找;

复杂度 O(logN) (2 分)

七．（25 分） 考虑一个公司，它在 n 个不同的公司中进行股票交易。对每一对(i,j)，他们维持一个交易率$r(i,j)$，含义是i的 1 股与j的$r(i,j)$股交易。这里我们允许交易率r是分数，即$r(i,j)=\frac{2}{3}$意味着你可以用i的 3 股进行交易得到j的 2 股。

对一系列股票$i_1,i_2,\ldots,i_k$，一个交易圈由下述联系的股票交易构成：公司i_1,的股票换成公司i_2的股票，接着公司i_2的股票成公司i_3的股票，…,如此下去，最后公司i_k的股票换回公司i_1的股票。在这样一系列交易之后，一个人结束时与开始时具有同一公司的股票。绕一圈的交易通常不是一个好想法，因为你结束时往往会比初始时拥有更少的股票。但是偶尔对于一段短的时间内，存在股票增加的机会。如果沿着一个圈的交易增加了股票数，我们将把这样的一个圈叫做“机会圈”。如果沿着这个圈的交易率之积大与 1，就会发生这种情况。在市场的状态分析中，一个从事交易的公司想知道是否存在任何的交易圈。

请给出一个多项式时间的动态规划算法，如果存在一个这样的机会圈，把它找出来。

参考答案：

（1） 设计优化函数并简要说明优化函数的含义（3 分）

$M(i,j,k)$表示从i交换到j，中间股票编号不大于k的交易序列中交易率之积的最大值

（2） 给出关于优化函数的递推方程（10 分）

$$M(i,j,k)=\max\big(M(i,k,k-1)M(k,j,k-1),M(i,j,k-1)\big)$$

（3） 说明优化函数的边界条件（3 分）

$$M(i,j,0)=r(i,j)$$

（4） 给出相关的标记函数并简要说明标记函数的含义（3 分）

$$T(i,j,k)=\begin{cases}T(i,j,k-1) & if\ M(i,j,k)=M(i,j,k-1)\\ T(i,k,k-1) & if\ M(i,j,k)=M(i,k,k-1)M(k,j,k-1)\end{cases}$$

$$T(i,j,0)=j$$

（5） 说明如何结合优化函数和标记函数的计算（中间）结果推算出一个机会圈（3 分）

通过$M(i,i,n)>1$ 确定一个机会圈
通过$i_k=T(i_{k-1},i_1,n)$推算机会圈的交易序列

（6） 分析所设计的动态规划算法的时间复杂度（3 分）

$$O(n^3)$$