北 京 交 通 大 学 考 试 试 题（A 卷）

课程名称：算法设计与分析　　学年学期：2021—2022 学年第 1 学期

课程编号：M210004B 开课学院：软件学院 出题教师：刘铎，童浩楠，吴睿智

第一部分、单项选择题。请选择最适合的答案，并填涂到答题卡上。（共 16 分）

1.	2.	3.	4.	5.	6.	7.	8.
C	**C**	**A**	**B**	**A**	**A**	**C**	**D**

考察知识点	要求
算法的数学基础	掌握
典型的分治算法	掌握
可选择性地介绍典型的贪婪算法 0-1 背包问题	掌握

第二部分、计算题。（共 34 分）

9. （共 12 分）解：

（1）

Running time of (a)	$T(n)=5T(n/2)+O(n)$	$\Rightarrow T(n)=O(n^{\log_2 5})$
Running time of (b)	$T(n)=2T(n-1)+O(1)$	$\Rightarrow T(n)=O(2^n)$
Running time of (c)	$T(n)=9T(n/3)+O(n^2)$	$\Rightarrow T(n)=O(n^2\log n)$

（2）按照阶从低到高的顺序，对以上 3 个算法的时间复杂度进行排序为：$O(n^2\log n)$，$O(n^{\log_2 5})$，$O(2^n)$。

为了更快地解决问题，应选择算法 C。

考察知识点	要求
分治算法的分析方法	理解
主定理	掌握

10. （共 12 分）解：

（1）

表 1 目标函数 $m[i,j]$

$m[i,j]$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$
$i=1$	0	42	168	258	270	286
$i=2$		0	189	324	258	282
$i=3$			0	315	216	272
$i=4$				0	90	162
$i=5$					0	40
$i=6$						0

表 2 标记函数 $s[i,j]$

$s[i,j]$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$
$i=1$	/	1	2	3	1	5
$i=2$		/	2	3	2	5
$i=3$			/	3	3	5
$i=4$				/	4	5
$i=5$					/	5
$i=6$						/

（2）$((\boldsymbol{A}_1(\boldsymbol{A}_2(\boldsymbol{A}_3(\boldsymbol{A}_4\boldsymbol{A}_5))))\boldsymbol{A}_6)$ 或者 $(\boldsymbol{A}_1(\boldsymbol{A}_2(\boldsymbol{A}_3(\boldsymbol{A}_4\boldsymbol{A}_5))))\boldsymbol{A}_6$

考察知识点	要求
算法的数学基础	掌握
典型的分治算法	掌握
典型的贪婪算法 0-1 背包问题	掌握

11. （共 10 分）解：

（1）请给出目标函数和标记函数的定义/表示、递推关系和初值。

令目标函数 $d(i)$表示凑出总和 i 所需的最少硬币数量。则 $d(i)$的初始值为：

$$\begin{cases} d(0)=0 \\ d(i)=+\infty \quad i<0 \end{cases}$$

递推式为 $d(i)=\min_{1\leqslant k\leqslant 4}\{\, d(i-v_k)\,\}+1$。

令标记函数 $s(i)$表示计算 $d(i)$时取得的 k 值，即 $\arg\min_{1\leqslant k\leqslant 4}\{d(i-v_k)\}$，含义是取得 $d(i)$

时所选取的最后一枚硬币。

则 $s(i)$的初始值为 $s(0)=0$，$s(i)$的递推式为 $s(i)=\arg\min_{1\leqslant k\leqslant 4}\{d(i-v_k)\}$。

（2）如表 3 所示。

表 3

i	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
d(i)	∞	∞	∞	∞	∞	∞	∞	0	1	2	3	1	2	1	2	1	2	2	3	2	3	2	3	2	3	3	4
s(i)	/	/	/	/	/	/	/	/	1	1	1	4	4	6	6	8	8	6	6	8	8	8	8	8	8	8	8

（3）付款的最优方案是使用一枚面值为 8 的硬币、一枚面值为 6 的硬币、一枚面值为 4 的硬币、一枚面值为 1 的硬币；或者三枚面值为 6 的硬币、一枚面值为 1 的硬币；共使用 4 枚硬币。

考察知识点	要求
典型的动态规划算法——矩阵链乘积	掌握

第三部分、综合分析题。（共 50 分）

12.（共 12 分）**解：**

（1）贪心策略：按币值从大到小排列零钱，从币值大的开始，每种钱尽量多用。如果剩余钱数小于该币值，再考虑用下一种钱币。

即：

```
remainder ← y
for i = n downto 0
    x_i ← ⌊ remainder / p^i ⌋
    remainder ← remainder − x_i × p^i
```

解方案是：对所有 $0\leqslant i\leqslant n$，取 x_i 个币值为 p^i 的硬币。

（2）对 n 做归纳，证明：当只有面值为$\{p^0, \cdots, p^n\}$的硬币可用时，贪婪解和最优解相同。

①$n=0$ 时，贪婪解和最优解相同。

②假设 $n=k$ 时，贪婪解和最优解相同。现在考虑 $n=k+1$ 时的情况。

假设贪婪解是对所有 $0\leqslant i\leqslant k+1$，取 x_i 个币值为 p^i 的硬币。假设最优解是对所有 $0\leqslant i\leqslant k+1$，取 z_i 个币值为 p^i 的硬币。

则有$\sum_{i=0}^{k+1} x_i p^i = \sum_{i=0}^{k+1} z_i p^i = y$。

考虑到贪婪算法的流程，必定对于所有 $0 \leqslant i \leqslant k$，都有 $x_i<p$。否则，如果有某个 $z_j \geqslant p$，$0 \leqslant j \leqslant k$, 则贪婪算法会先选择 1 枚面值为 p^{j+1} 的硬币，而不是 p 枚面值为 p^j 的硬币。且由贪婪算法可知$\sum_{i=0}^{k} x_i p^i \leq p^{k+1}-1$

❶断言：对于所有 $0 \leqslant i \leqslant k$，都有 $z_i<p$。否则，如果有某个 $z_j \geqslant p$，$0 \leqslant j \leqslant k$, 则用 1 枚面值为 p^{j+1} 的硬币替换 p 枚面值为 p^j 的硬币，硬币总币值不变，而硬币总个数减少 $p-1$。

❷于是$\sum_{i=0}^{k} z_i p^i \leq (p-1) \times \sum_{i=0}^{k} p^i = p^{k+1}-1$。

❸考虑到贪婪算法的流程，所以必定有 $x_{k+1} \geqslant z_{k+1}$。

❹于是，若 $x_{k+1}>z_{k+1}$，则

$$p^{k+1} \leq (x_{k+1}p^{k+1} - z_{k+1}p^{k+1}) = \left(y - \sum_{i=0}^{k} x_i p^i\right) - \left(y - \sum_{i=0}^{k} z_i p^i\right) = \sum_{i=0}^{k} z_i p^i - \sum_{i=0}^{k} x_i p^i$$

$$\leq \sum_{i=0}^{k} z_i p^i \leq p^{k+1}-1 < p^{k+1}$$

产生矛盾，因此必然有 $x_{k+1}=z_{k+1}$。

❺ $y - x_{k+1}p^{k+1} = \sum_{i=0}^{k+1} x_i p^i \leq p^{k+1}-1$ 且 $\sum_{i=0}^{k+1} x_i p^i = y - x_{k+1}p^{k+1} = y - z_{k+1}p^{k+1} = \sum_{i=0}^{k+1} z_i p^i$。因此只能使用面值为$\{p^0, \cdots, p^k\}$的硬币，由归纳假设可知对于所有 $0 \leqslant i \leqslant k$，都有 $z_i=x_i$。

综上可得对于所有 $0 \leqslant i \leqslant k+1$，都有 $z_i=x_i$，即贪婪解和最优解相同。

（3）在（1）中设计的贪心策略在此时不能确保得到最优解。

例如，$m=n=1$，$p=4$，$q=5$ 时，要凑得总面值为 8。如果是使用（1）中设计的贪心算法的基本策略，则需要 1 枚面值为 5 的硬币、3 枚面值为 1 的硬币，共使用 4 枚硬币；而事实上最优方案是使用两枚面值为 4 的硬币。

考察知识点	要求
贪婪策略的基本思想	了解
贪婪算法的基本框架	掌握
贪婪算法最优性的分析与证明	理解

13. （共 12 分）**解：**

（1）（参考解答。程序伪代码不唯一。）

```
Initial Call
    j ← Find(1, n)
    If j > 0 then output j
    else output "None"
Find ( start, end )
    if start > end then return –1
    else
        j ← (start + end) / 2
        if A[j] = j then return j
        else if A[j] < j then return Find[ j+1, end ]
        else return Find[ start, j–1 ]
```

（2）算法的时间复杂度的递推关系式和初值为：

$$\begin{cases} T(n) \le T(n/2) + O(1) & n > 1 \\ T(1) = 1 & \end{cases}$$

由主定理可知 $T(n)=O(\log n)$。

考察知识点	要求
分治策略的基本思想	了解
分治算法的基本框架	掌握
分治算法的分析方法	理解
主定理	掌握

14. （共 13 分）**解：**

（1）将各个点进行坐标标号，并增加一些额外顶点，如图 1 所示。

令 $v(i, j)$表示点(i, j)的值，$M(i, j)$表示从点(1, 1)到点(i, j)的道路最小数值和，则有：

$M(1, 1) = v(1, 1)$;

$M(i, j) = \infty$, if j=0 or i=0;

$M(i, j) = \min(M(i-1, j), M(i, j-1))+v(i, j)$, otherwise.

标记函数 $s(i, j)$=1 表示到点(i, j)的最小数值和道路的最后一条边是从左侧来，即从$(i, j-1)$到(i, j)；$s(i, j)$=0 表示到点(i, j)的最小数值和道路的最后一条边是从上方来，即从$(i-1, j)$到(i, j)的点(i, j)的值；$s(1, 1)$=–1 表示初始值或到点(i, j)的最小数值和道路的最后一条边是从上方来，即从$(i-1, j)$到(i, j)的点(i, j)的值。则有：

$s(1, 1) = -1$;

$s(i, j) = 0$, if (j>1 or i>1) and $M(i-1, j)<M(i, j-1)$

$s(i, j) = 1$, if (j>1 or i>1) and $M(i-1, j) \geqslant M(i, j-1)$

（2）各个点的 $M(i,j)$值如图 2 所示，各个点的 $s(i,j)$值如图 3 所示。

（3）道路选择如图 4 所示，总数值和为 12。

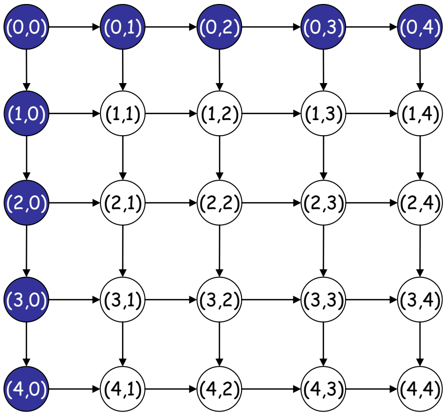

图 1 题目 14 解答用图 1

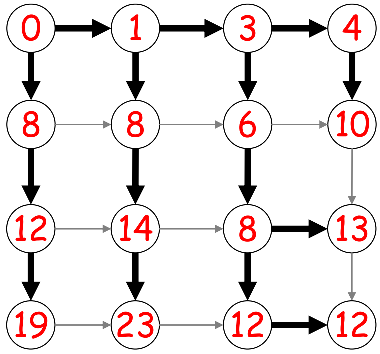

图 2 题目 14 解答用图 2

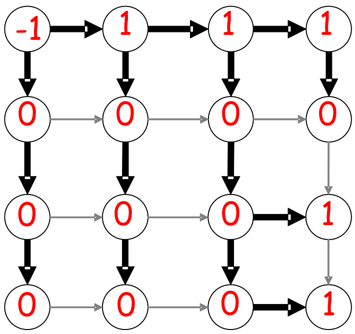

图 3 题目 14 解答用图 3

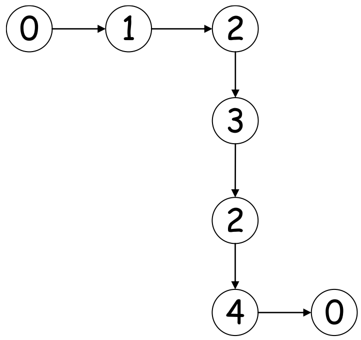

图 4 题目 14 解答用图 3

考察知识点	要求
动态规划的基本思想	掌握
动态规划的基本框架	理解
动态规划的实现方法	掌握

15. （共 13 分）解：

（1）将各个点进行坐标标号，如图 5 所示。令 $v(i,j)$表示点(i,j)的值。

选择一条明显的道路（如图 6 所示），以其数值和（15）作为初始的界，即 *current_best*←15。

如图 7 所示斜线方式考察$m(k)=\min\{v(i,j)|i+j=k\}$，$2\leqslant k\leqslant 8$。

维护“到目前为止的最优值”*current_best*。

令 $C_a(i,j)$为从$(1,1)$到当前点(i,j)的道路的数值和，估值函数为 $C_e(i,j)=\sum_{k=i+j+1}^{8} m(k)$，即估计还要发生的开销的下界。则估界函数为 $C_a(i,j)+C_e(i,j)$。

若(i,j)为点(4, 4)且 $C_a(4,.4)<current\_best$，则更新 $current\_best$ 为 $C_a(4,.4)$。

若(i,j)不是点(4, 4)且 $C_a(i,j)+C_e(i,j)\geqslant current\_best$，则进行剪枝。

（2）剪枝后的（部分）搜索树如图 8 所示。

（3）道路选择如图 9 所示，总数值和为 12。

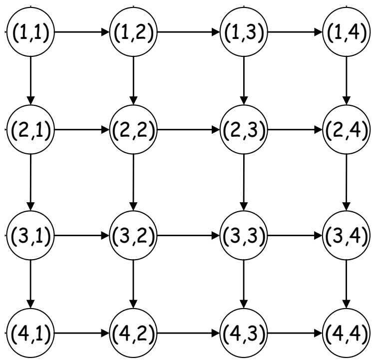

图 5 题目 15 解答用图 1

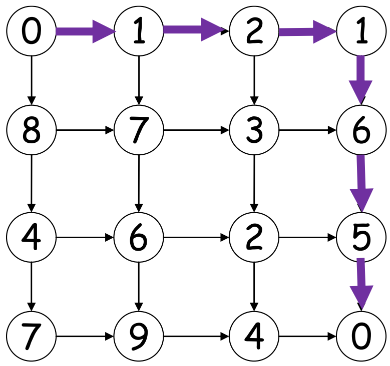

图 6 题目 15 解答用图 2

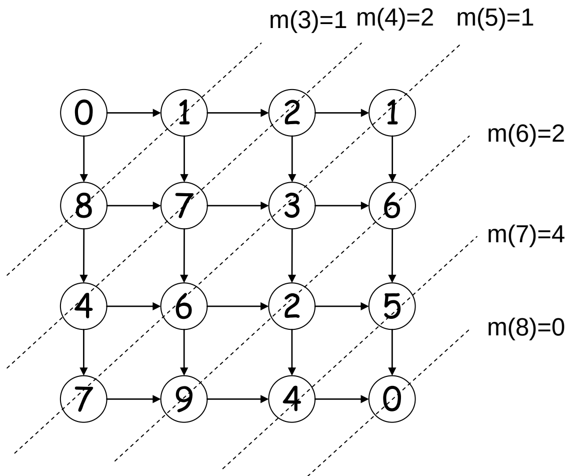

图 7 题目 15 解答用图 3

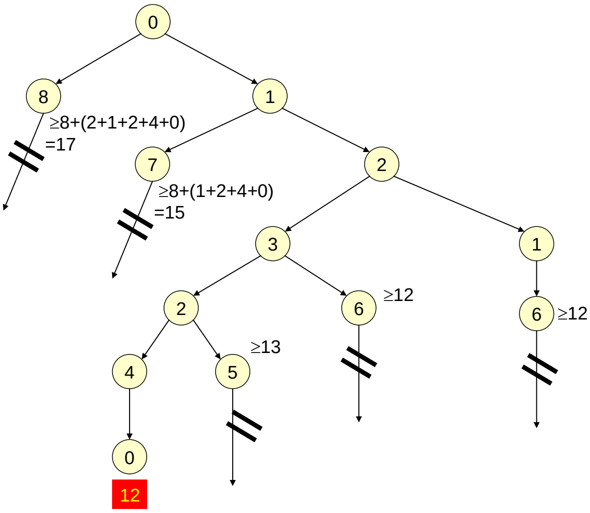

图 8 题目 15 解答用图 3

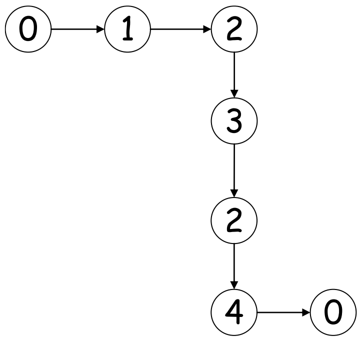

图 9 题目 15 解答用图 5

考察知识点	要求
回溯法的基本思想	了解
回溯法的基本框架	掌握
“剪枝”的概念	理解
对 “界” 的正确估算	掌握