

重庆大学《算法分析与设计》课程试卷

☐ A 卷

☒ B 卷

2012~2013 学年 第二学期

开课学院: 计算机学院 课程号: 18016435 考试日期: _____

考试方式: ☐ 开卷 ☒ 闭卷 ☐ 其他 考试时间: 120 分钟

| 题 号 | A | B | C | D | E | F | G | H | I | J | 总 分 |
|-----|---|---|---|---|---|---|---|---|---|---|-----|
| 得 分 | | | | | | | | | | | |

注: 答案填写在试题页后的答案纸上, 用中文解答。一些重要的关键词句后附有中文解释。

A. (15 points) Use mathematical induction(数学归纳法) to show that when $n = 2^k$ with $k \geq 0$ the solution of the recurrence equation (递归等式)
$$T(n) = T(n/2) + n, \text{ where } T(1)=3$$
is $T(n)=2n+1$.

B. (15 points) Quicksort(快速排序) is a very interesting algorithm. Given an disordered array with n elements,
1. What is the best time complexity(最好时间复杂度) to sort this array with quicksort?
2. What is the worst time complexity(最坏时间复杂度) to sort this array with quicksort?
3. During each iteration, if we assume that the initial array is split(分割) according to the ration(比例) 1:9, then write the recurrence of this situation, and prove that the asymptotic time complexity will remain $\Theta(n \log n)$ with substitution .

C. (25 points) Suppose that in a 0-1 knapsack problem(0-1 背包问题) , the order of the items when sorted by increasing weight(按重量升序排序) is the same as their order when sorted by decreasing value(按价值降序排序). Give an efficient algorithm to find an optimal solution to this variant of the knapsack problem, and argue that your algorithm is correct.
1. Describe the greedy choice.
2. Prove the greedy choice property (贪心选择性质) of this problem.

3. Write a program in pseudo code (伪代码) or other popular programming language to solve this problem based on your algorithm.

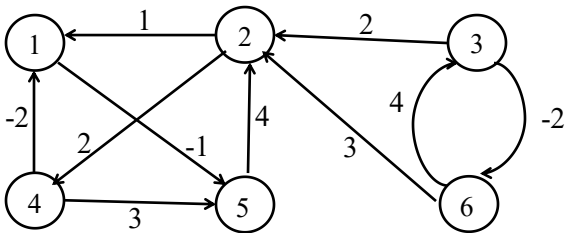
D. (25 points) For input sequence $X_{[1..m]}$ and $Y_{[1..n]}$, we define $c[i,j] = \text{length(长度) of Longest Common Sequence(最长公共子序列) of } X_{[1..i]} \text{ and } Y_{[1..j]}$; By using methodology of dynamic programming(动态规划), we got $c[i,j]$'s recursive formula:

$$c[i,j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0, \\ c[\quad, \quad] + 1 & \text{if } i, j > 0 \text{ and } x_i = y_j, \\ \max(c[\quad, \quad], c[\quad, \quad]) & \text{if } i, j > 0 \text{ and } \end{cases}$$

1. Fill out(填空) the above formula and indicate the time complexity for computing $c[m,n]$.
2. Assume $X = \langle A, G, C, C, G, G \rangle$ and $Y = \langle C, C, G, G, A, C \rangle$. Then fill out the following table of $c[0..6, 0..6]$.

| | | j | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|-------------|---|---|---|---|---|---|
| i | | | C C G G A C | | | | | | |
| | | 0 | | | | | | | |
| 1 | A | | | | | | | | |
| 2 | G | | | | | | | | |
| 3 | C | | | | | | | | |
| 4 | C | | | | | | | | |
| 5 | G | | | | | | | | |
| 6 | G | | | | | | | | |

E. (20 points) Run either the matrix multiplication(矩阵乘法) or the Folyd-Warshall algorithm for the all-pairs shortest paths(全部顶点对最短路径) on the following weighted, directed graph. Indicate(指出) clearly which algorithm is used(用了何种算法) and show the matrix $D^{(k)}$ at each iteration(迭代) until the shortest distances(最短距离) between all pairs of vertices are computed.



姓名

学号

年级

专业、班

学院

公平竞争、诚实守信、严肃考纪、拒绝作弊

密

封

线

命题人: 罗辛

组题人: 陈波

审题人: 李佳

命题时间: 2013-07-02

教务处制