系所組別：資訊工程學系
考試科目：程式設計　　考試日期：0223，節次：2

※ 考生請注意：本試題不可使用計算機

Part I.

演算法 (50%)

一、是非題(15 points) For each of the following statements, determine whether it is correct (T) or not (F).

1. Let T be a minimum spanning tree of G. Then, for any pair of vertices s and t, the shortest path from s to t in G is the path from s to t in T.
2. The function $\lceil \lg n \rceil !$ is polynomially bounded.
3. If the depth-first search of a graph G yields no back edges, then the graph G is acyclic.
4. Suppose $P1$ and $P2$ are problems and $P1 \leq_p P2$. If $P2$ can be solved by an algorithm with time complexity $\lceil \lg\lg n \rceil !$, then $P1 \in \mathbf{P}$.
5. Sorting 8 elements with a comparison sort requires 24 comparisons in the worst case.

二、計算題

1. (10 points) Use the **recursion-tree** method to solve the recurrence T(n) = 2T(n/2) + n/lgn.
2. (10 points) For **COIN-CHANGE** problem defined below, please calculate (a) how many different subproblems overall and (b) how many choices we have in determining which subproblem(s) to use in an optimal solution. [**COIN-CHANGE** Problem: An amount of money $\boldsymbol{M}$, and an array of $\boldsymbol{d}$ denominations $\boldsymbol{c}$ = ($c_1, c_2, \ldots, c_d$), in a decreasing order of value ($c_1 > c_2 > \ldots > c_d$). Please find a list of d integers $i_1, i_2, \ldots, i_d$ such that $c_1 * i_1 + c_2 * i_2 + \ldots + c_d * i_d = \boldsymbol{M}$ and $i_1 + i_2 + \ldots + i_d$ is minimal.]
3. (15 points) Please analysis the running time of Dijkstra's algorithm under the following data structure implementation (1) array, (2) binary heap, and (3) Fibonacci heap [where Extract-Min O(lgV), Decrease-Key O(1)]

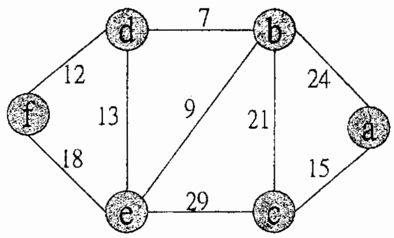
(背後仍有題目，請繼續作答)

※ 考生請注意：本試題不可使用計算機

Part II.

資料結構 (50%)

1. (15 points) Consider the following graph:



Compute minimum cost and construct minimum spanning tree step by step by using:

(1) Kruskal's algorithm,

(2) Prim's algorithm,

(3) Sollin's algorithm.

2. (15 points) Jeremy is a waiter working in a restaurant. The chef there is sloppy; when he prepares a stack of pancakes, they come out all different sizes. When Jeremy delivers the pancakes to the customer, he wants to rearrange them by grabbing several from the top and flipping them over on the way. After repeating this for several times, the smallest pancake is on top, and so on, down to the largest at the bottom. If there are n pancakes, how many flips are required? Design an algorithm to help Jeremy, and analyze its time complexity.

3. (20 points) A *Bloom Filter* is a space-efficient probabilistic data structure used to test whether a key is in a large data set. Instead of answering "yes" or "no," a Bloom Filter answers "maybe" or "no." A Bloom Filter consists of m bits of memory and h uniform and independent hash functions $f_1, f_2, \ldots, f_h$. Each f_i hashes a key k to an integer in the range $[1, m]$. Initially all m filter bits are zero, and the data set is empty. When key k is added to the data set, bits $f_1(k), f_2(k), \ldots, f_h(k)$ of the filter are set to 1. When a query "Is key k is in the data set?" is made, bits $f_1(k), f_2(k), \ldots, f_h(k)$ are examined. The query answer is "maybe" if all these bits are 1. Otherwise, the answer is "no."

(1) When the answer is "no," the key **is not** in the data set; when the answer is "maybe," the key **may or may not** be in the data set. Explain why.

(2) A *filter error* occurs whenever the answer is "maybe" and the key is not in the data set. Assume that key k is an integer in the range $[1, n]$ and u updates are made. Compute the probability of filter error for an arbitrary query after the u-th update.