國立臺灣海洋大學 101 學年度研究所碩士班暨碩士在職專班入學考試試題

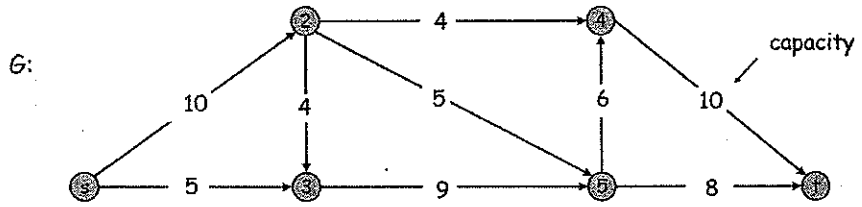
考試科目：基礎計算機科學（含資料結構、演算法）

系所名稱：資訊工程學系碩士班不分組

1.答案以橫式由左至右書寫。2.請依題號順序作答。

PART I(50%)

1. (10 points)Find the maximum value of *s-t* flow in *G* by Ford-Fulkerson Algorithm. Show the state of each phase.



2. (10 points)An extreme point of a convex set is a point of this set that is not a middle point of any line segment with endpoints in the set. Design a linear-time algorithm to determine two extreme points of the convex hull of a given set of $n > 1$ points in the plane.

3. (15 points)Consider the following recursive algorithm.

   ALGORITHM $Q(n)$

   //Input: A positive integer n

   if $n = 1$ **return** 1

   else return $Q(n-1) + 2 * n - 1$

   a. Set up a recurrence relation for this function's values and solve it to determine what this algorithm computes.
   b. Set up a recurrence relation for the number of multiplications made by this algorithm and solve it.
   c. Set up a recurrence relation for the number of additions/subtractions made by this algorithm and solve it.

4. (15 points)Suppose we assign n persons to n jobs. Let C_{ij} be the cost of assigning the ith person to the jth job. The assignment problem is to find an assignment with the minimum total cost.

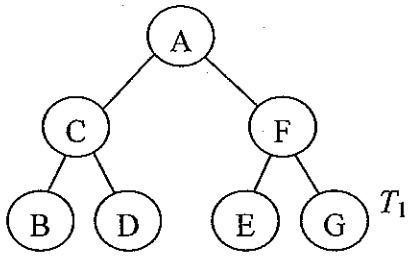
   a. Design a greedy algorithm for the assignment problem.
   b. Prove that your greedy algorithm always yields an optimal solution.

PART II(50%)

1. (10 points) Show an array representation for complete binary tree T_1. In addition, show the formulas for determining the indices of the parent, the left child, and the right child of the node with index u in that array, respectively.(10%)

$A[0]$	$A[1]$	$A[2]$	$A[3]$	$A[4]$	$A[5]$	$A[6]$	$A[7]$	$A[8]$

Parent(u)=
LeftChild(u)=
RightChild(u)=



2. (10 points) The node for the linked representation of the binary tree in C is defined as

```
struct tree_node {
    int shortest;
    struct tree_node *leftChild, *rightChild;
};
```

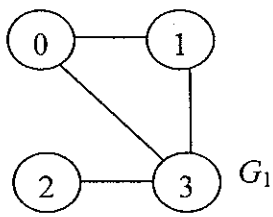
where the field *shortest* is defined as

$$shortest(x) = \begin{cases} 0 & x \text{ is an external node} \\ 1 + \min\{shortest(x\text{'s } leftChild), \\ \quad shortest(x\text{'s } rightChild)\} & \text{otherwise} \end{cases}$$

A leftist tree is a binary tree. Every internal node x of a height-biased leftist tree should satisfy $shortest(x\text{'s } leftChild) \geq shortest(x\text{'s } rightChild)$. Write a function *void hblt(struct tree_node *T)* for making a binary tree become a height-biased leftist tree, where T is a pointer to the root of the binary tree.

3. (15 points) Define the node for the linked adjacency lists representation of the graph in C as

```
struct graph_node {
    int vertex;
    struct graph_node*link;
};
```



(a) (5 points) Show the adjacency lists for undirected graph G_1.

(b) (10 points) Write a function *void remove_edge(struct graph_node*adjLists[],int u, int v)* for removing the edge connecting vertices u and v from an undirected graph, where *adjLists*[u] is a pointer to the first node in the adjacency list for vertex u.

4. (15 points) Select the most appropriate data structure from *stacks, selection trees, union-find data structures, hash tables, min-max heaps, Fibonacci heaps, and red-black trees* for each of the following problems.

(a) Implementing a network buffer.

(b) Implementing an ordered dictionary structure.

(c) Dijkstra's algorithm for computing shortest paths.

(d) Kruskal's algorithm for finding a minimum-cost spanning tree.

(e) Merging several sorted sequences into a single sorted sequence.