國立臺南大學106學年度 **資訊工學系碩士班** 招生考試　計算機概論　試題卷

1. What is the characteristics of SHORT-CIRCUIT EVALUATION? Give each of || and && an example (in C, C++, or JAVA) to demonstrate its merits. Be sure to explain your answer in details. (10%)

2. Write down the following requirement in C programming language. (15%)

Model\Year	**2016**	**2015**	**2014**	**2013**
TOYOTA	*15*	*6*	*20*	*26*
BMW	*18*	*10*	*25*	*39*
LEXUS	*50*	*9*	*32*	*31*

Use ***int car[][]*** to store the above annual car sale data from year 2013-2016 for a car sale company. Write a program in C programming language using **LOOP** to output the following information.

(a) The total number of car sold for TOYOTA, the total number of car sold for BMW, and the total number of car sold for LEXUS from 2013 to 2016.
(b) Output the annual number of car sale for each year.
(c) Output the highest number of car sale info within 2013 to 2016 (year, brand, sale number).

3. Write down the following functions in C programming language.
(a) Implement the function *double Min(double[ ], int num)* which returns the minimum value of a given double array. (5%)
(b) Read the integer variable *a* and *b* from keyboard.
Implement the function *void swap (int* ptr_a, int* ptr_b)* to swap the contents of *a* and *b*. Print out the values of *a, b* before and after calling the *swap()*. (5%)

4. (a) Write a recursive function ***int rsum(int n)*** and a non-recursive function ***int non_rsum(int n)*** by using **LOOP** in C programming language to compute the sum of the following equation, (10%)

$$1\times 2+2\times 3+3\times 4+\cdots+(n-1)\times n.$$

(b) Discuss the pros and cons of using recursion. When to use and not to use recursion in designing code? Concisely describe your explanation. (5%)

5. Give the Big-θ (Big-Theta) for each following running time estimates (where n is the size of the input problem). (5%)

   (a) $2^{1000} + 1000^2$

   (b) $1 + 2 + 3 + \cdots + (n-1) + n$

   (c) $10n + 100 \log n + n \log n^2$

   (d) $2^0 + 2^1 + \cdots + 2^{n-1}$

   (e) $n \log n! + n^2$

6. (a) Design an efficient algorithm which takes as input an array which may contain duplicates. It returns true if all elements of the array occur an even number of times; otherwise it returns false. For example, on the array {2, 6, 2, 6, 7, 2, 2, 7} it returns true, but on the array {1, 6, 2, 1, 2, 2, 6} it returns false because 2 occurs an odd number of times. (10%)

   (b) Please show the time complexity of your designed algorithm. (5%)

   (c) Please demonstrate your designed algorithm on the array {2, 6, 2, 6, 7, 2, 2, 7}. (5%)

7. (a) Which array out of *A*, *B*, *C*, and *D* represents a binary heap (*min* heap)? Only one answer is correct. (5%)

	0	1	2	3	4	5	6	7	8	9	10	11
A=	1	3	5	2	4	6	7	9	11	8	10	

	0	1	2	3	4	5	6	7	8	9	10	11
B=	1	3	5	6	10	7	9	17	77	4	25	

	0	1	2	3	4	5	6	7	8	9	10	11
C=	1	3	9	4	5	77	10	7	6	25	17	

	0	1	2	3	4	5	6	7	8	9	10	11
D=	1	2	4	9	7	17	10	25	6	3	77	

   (b) Write the heap out as a binary tree. (5%)

   (c) Add 8 to the heap. How does the array look now for the new heap? (5%)

8. Consider an initially empty hash table of size *M* and a hash function $h(x)=x \bmod M$.

   (a) In the worst case, what is the time complexity in Big-Oh notation to insert *n* keys into the table if separate chaining is used to resolve collisions? Suppose that each entry (bucket) of the table stores an unordered linked list. When adding a new element to an unordered linked list, such an element is inserted at the begging of the list. (3%)

   (b) What is the answer (time complexity in Big-Oh notation) for (a) if the linked lists are ordered? (3%)

   (c) Suppose that instead of a linked list, each bucket of the table is implemented as an AVL tree. Then, what is the answer (time complexity in Big-Oh notation) for (a)? (4%)