

# 國立臺北科技大學九十五學年度碩士班招生考試

系所組別：1650 電機工程系碩士班戊組

## 第一節 計算機概論 試題

填准考證號碼

第一頁 共二頁

--	--	--	--	--	--	--	--

### 注意事項：

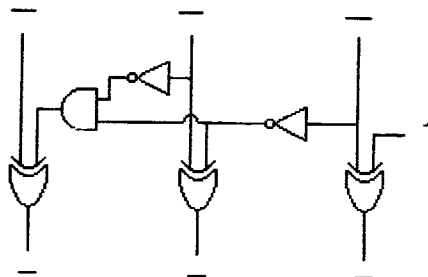
1. 本試題共 **【六】** 題，配分共 100 分。
2. 請按順序標明題號作答，不必抄題。
3. 全部答案均須答在答案卷之答案欄內，否則不予計分。

1. (a) (5%) Describe a scenario that leads to deadlock in a computer system.  
(b) (5%) What conditions are necessary for deadlock to occur?
2. (a) (5%) Based on the meaning of the term virtual memory, give an argument that time sharing could be called virtual CPU.  
(b) (5%) Explain why the average length of a time slice would be reduced if the processes in an operating system's process table perform lots of I/O operations.
3. (a) (5%) Explain the distinction between time complexity and space complexity.  
(b) (5%) Is a problem in  $O(n^3)$  more complex than a problem in  $O(n^2)$ ? Explain your answer.  
(c) (5%) Are all problems in P solvable in a reasonable amount of time? Explain your answer.  
(d) (5%) At most, how many entries in a list of 5000 names will be interrogated when using the binary search algorithm?
4. (a) (5%) In what way could TCP be considered a better protocol for implementing the transport layer than UDP? In what way could UDP be considered a better

protocol for implementing the transport layer than TCP?

- (b) (5%) At what layer in the TCP/IP protocol hierarchy could a firewall be placed to filter incoming traffic by means of
- (1) message content
  - (2) source address
5. (a) (5 %) In a two's complement system, what value can be added to any other value without causing an overflow? How many values in the system have this property? Explain your answer.
- (b) (5%) Using an 8-bit floating-point format in which the most significant bit is the sign bit, the next three bits represent the exponent field in excess notation, and the last four bits represent the mantissa, what would be the result of computing the sum  $\frac{1}{8} + \frac{1}{8} + \frac{1}{8} + 2\frac{1}{2}$  from left to right? How about from right to left?
6. (a) (5%) If the input and output bit patterns in the circuit below are interpreted as binary representations of numeric values, what operation does the circuit perform?

**Input Pattern**

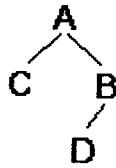


**Output Pattern**

- (b) (5%) Using gates, design a circuit with four inputs and one outputs such that the output is 1 or 0 depending on whether the four-bit input pattern has odd or even parity, respectively.
7. (a) (5%) A list contains the following elements: 7 8 26 44 13 23 57 98. The first two elements have been sorted using the bubble sort algorithm. What is the value of the elements in the list after three more passes of bubble sort?
- (b) (5%) A list contains the following elements: 3 13 7 26 44 23 98 57. The first two elements have been sorted using the insertion sort algorithm. What is the value of the elements in the list after three more passes of insertion sort?

8. (5%) What sequence of nodes from the tree

第二頁 共二頁



would be printed if the following recursive procedure were applied to it? (The procedure uses a global stack called Stack that is assumed to begin empty.)

```

procedure printTree (Tree)
push the left child of the root node on Stack;
if (right branch of Tree is not empty)
then (apply the procedure printTree to the right subtree
of Tree)
pop an entry from Stack and print that node.
  
```

9. (a) (7%) Using the machine language described in the **language description table**, write a sequence of instructions that will add five to the value (represented in two's complement notation) stored at memory address A0.

(b) (8%) Translate the following pseudocode program into the machine language described in the language description table.

```

X ← 0
while (X < 3) do
    (X ← X + 1)
  
```

**“language description table”**

Op-code	Operand	Description
1	RXY	LOAD the register R with the bit pattern found in the memory cell whose address is XY <i>Example:</i> 14A3 would cause the contents of the memory cell located at address A3 to be placed in register 4.
2	RXY	LOAD the register R with the bit pattern XY. <i>Example:</i> 20A3 would cause the value A3 to be placed in register 0.
3	RXY	STORE the bit pattern found in register R in the memory cell whose address is XY. <i>Example:</i> 35B1 would cause the contents of register 5 to be placed in the memory cell whose address is B1.
4	ORS	MOVE the bit pattern found in register R to register S. <i>Example:</i> 40A4 would cause the contents of register A to be copied into register 4.
5	RST	ADD the bit patterns in registers S and T as though they were two's complement representations and leave the result in register R. <i>Example:</i> 5726 would cause the values in registers 2 and 6 to be added and the sum placed in register 7.
6	RST	ADD the bit patterns in registers S and T as though they represented values in floating-point notation and leave the floating-point result in register R. <i>Example:</i> 634E would cause the values in registers 4 and E to be added as floating-point values and the result to be placed in register 3.
7	RST	OR the bit patterns in registers S and T and place the result in register R. <i>Example:</i> 7CB4 would cause the result of ORing the contents of registers B and 4 to be

- placed in register C.
- 8     RST     AND the bit patterns in register S and T and place the result in register R.  
*Example:* 8045 would cause the result of ANDing the contents of registers 4 and 5 to be placed in register 0.
- 9     RST     EXCLUSIVE OR the bit patterns in registers S and T and place the result in register R.  
*Example:* 95F3 would cause the result of EXCLUSIVE ORing the contents of registers F and 3 to be placed in register 5.
- A     R0X     ROTATE the bit pattern in register R one bit to the right X times. Each time place the bit that started at the low-order end at the high-order end.  
*Example:* A403 would cause the contents of register 4 to be rotated 3 bits to the right in a circular fashion.
- B     RXY     JUMP to the instruction located in the memory cell at address XY if the bit pattern in register R is equal to the bit pattern in register number 0. Otherwise, continue with the normal sequence of execution. (The jump is implemented by copying XY into the program counter during the execute phase.)  
*Example:* B43C would first compare the contents of register 4 with the contents of register 0. If the two were equal, the pattern 3C would be placed in the program counter so that the next instruction executed would be the one located at that memory address. Otherwise, nothing would be done and program execution would continue in its normal sequence.
- C     000     HALT execution.  
*Example:* C000 would cause program execution to stop.