

# 淡江大學 95 學年度碩士班招生考試試題

88-1

系別：資訊工程學系

科目：資 訊 概 論

准帶項目請打「V」

簡單型計算機

本試題共 2 頁 P. 1/2

- For the questions below, give your answer briefly and precisely (5% each, 30% total)
  - Give the seven-layer architecture of the OSI model specified in 1984 by ISO.
  - What the main functions of TCP and IP as used in the Internet?
  - What is the “dangling else” problem? How to avoid it?
  - Give the three main features in OOP (Object-Oriented Programming).
  - What are the differences between procedural and declarative language? Use the programming languages currently employed for explanation.
  - In C, an array is, by default, passed by reference. If we want to copy the array values over in the function call, how to do it?
- Give the algorithm for binary search of an integer element  $t$  in a sorted array  $s$  of  $n$  elements. Compute the time complexity based on this algorithm. (10%)
- Given a function  $\text{Fib}(n)$  as to compute the number of Fibonacci series 0, 1, 1, 2, 3, 5, ... Show the output. (10%)

```
void main(){ printf("\n%d\n", Fib(4); }
int Fib(int n){
    printf("%d ", n);
    if(n==0) return 0;
    else if (n==1) return 1;
    else return Fib(n-2)+Fib(n-1);
}
```
- Convert the expression  $-2*3+4-2^3/4-(2+3)$  to its postfix notation, and show how to evaluate it. Use stacks for the conversion and evaluation. Show enough steps to justify your answer. (10%)
- Draw the binary tree based on the following preorder and postorder sequences. Determine if the binary tree drawn is unique. (10%)

preorder: T, S, R, F, D, I, H, E, Z, G, M, L, J, N, Q

postorder: F, I, H, D, R, Z, G, E, S, J, N, L, Q, M, T

本試題雙面印製

准帶項目請打「V」
簡單型計算機

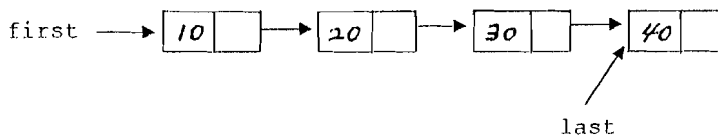
6. Given the partial program below. Please finish it in order to generate the magic square as shown in the right (assume n=5). Include only the first for loop in your answer sheet. (15%)

<pre>#include &lt;stdio.h&gt; #define SIZE 20  void main(){     int row, col, i, j, n;     int board[SIZE][SIZE];      printf("Enter rank: ");     scanf("%d", &amp;n);     row = n-1;     col = n/2;     board[row][col] = 1;     for(i=2;i&lt;=n*n;i++){         /*          add your code here         */     }     for(i=0;i&lt;n;i++){         for(j=0;j&lt;n;j++){             printf("%3d ", board[i][j]);             printf("\n");         }     } }</pre>	<table border="1"> <tr><td>17</td><td>6</td><td>25</td><td>14</td><td>3</td></tr> <tr><td>11</td><td>5</td><td>19</td><td>8</td><td>22</td></tr> <tr><td>10</td><td>24</td><td>13</td><td>2</td><td>16</td></tr> <tr><td>4</td><td>18</td><td>7</td><td>21</td><td>15</td></tr> <tr><td>23</td><td>12</td><td>1</td><td>20</td><td>9</td></tr> </table>	17	6	25	14	3	11	5	19	8	22	10	24	13	2	16	4	18	7	21	15	23	12	1	20	9
17	6	25	14	3																						
11	5	19	8	22																						
10	24	13	2	16																						
4	18	7	21	15																						
23	12	1	20	9																						

7. Assuming a linked list is defined below.

```
struct list {
    int data;
    struct list *next;
};
```

Write the program segments only for answering the questions (a) to (c), which are directly based on the following ordered list with 4 nodes. Note that the answer (b) does not depend on (a), (c) does not depend on (b) and (a). Do not use other pointer except the pointers first, last, and temp. (15%)



- (a) Insert a node, pointer by temp, to its proper place with data = 25, i.e. after the node 20.
- (b) Delete the node 10.
- (c) Delete the node 40.