國立臺南大學 107 學年度 **資訊工程學系碩士班** 招生考試 計算機概論 試題卷

一、程式設計

1. Multiple-choice questions (15%)

(1) What would be the equivalent pointer expression for referring the array element **a[i][j][k][l]**?

(A) *(*(*(*(a+i)+j)+k)+l) (B) *((a+i)+j+k+l)

(C) ((((a+i)+j)+k)+l) (D) (((a+i)+j)+k+l)

(2) The output of the following program is

```
#include<stdio.h>
int main( ) {
int i=2;
printf(“%d %d %d”, i++, i, ++i);
return 0; }
```

(A) 3 3 3 (B) 2 3 4 (C) 2 3 3 (D) 2 2 4

(3) What is wrong with the following program?

```
#include<stdio.h>
int main() {
char m1[9]=“message1”;
char m2[9]=“message2”;
m2=m1;
printf(“msg is %s”, m2);
return 0; }
```

(A) Char array cannot be printed directly using printf

(B) Array is not a left value and so cannot be assigned to

(C) Program compiles without error, but prints an unpredictable value

(D) Array cannot be initialized as above

(4) Predict the output of the following program

```
#include<stdio.h>
int main() {
static char *s[ ]={"black","white","pink","violet"};
char **ptr[ ]={s+3,s+2,s+1,s},***p;
p=ptr;
++p;
printf("%s", **p+1);
return 0; }
```

(A) ack (B) ink (C) ite (D) let

(5) What will happen if you try to put so many values into an array during the initialization such that its size is exceeded?

(A) Error message from the compiler

(B) Possible system malfunction

(C) Last element data may be overwritten

(D) Nothing

2. What is Deep Learning? (5%)

3. An addressing mode specifies how to calculate the effective memory address by using information held in registers and/or constants contained within a machine instruction or elsewhere. Briefly describe the major difference between the uses of PC-relative addressing mode and Base-relative address mode. (5%)

4. (1)Write two ***C*** programs that use static memory allocation (i.e. arrays) and dynamic memory allocation, respectively, to store the data set entered by user from the keyboard to calculate the mean and standard deviation. (10%)

   (2)Discuss the pros and cons of using static memory allocation and dynamic memory allocation. (5%)

5. The Fibonacci sequence is a series of numbers where a number is found by adding up the two numbers before it. Starting with 0 and 1, the sequence goes 0, 1, 1, 2, 3, 5, 8, 13, and so forth. Written as a rule, the expression is $x_n = x_{n-1} + x_{n-2}$. Write two ***C*** programs to compute the first 20 Fibonacci numbers using a **Loop** and **Recursion** respectively. (10%)

二、資料結構

1. Write an algorithm called ListReOrder that assumes the linked list starts by being sorted by absolute value and reorders the list so that it is sorted by value. The resulting list should contain all the values from the original list in sorted order from smallest to greatest. For example, if the input variable list stores the following:

   list = [0, -7, 11, -13, 17, -25, -33, 77]

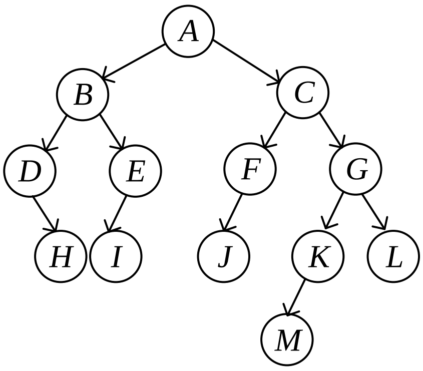
   The call of ListReOrder would change the list to store the following values:

   list = [-33, -25, -13, -7, 0, 11, 17, 77]

   You may not construct any new nodes, and you may not use any auxiliary data structure to solve this problem (no array, ArrayList, stack, queue, String, etc).

You also may not change any data fields of the nodes. You must solve this problem by rearranging the links of the lists. Please define your own variables of type ListNode (data structure). Your solution must run in O(n) time where n is the length of the linked list. (15%)

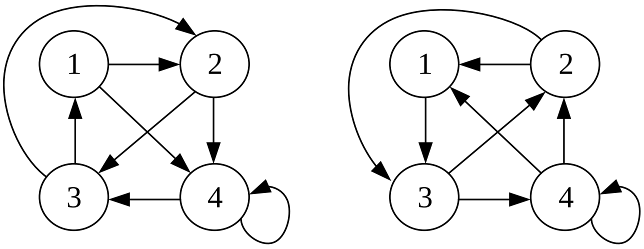
2. Assume this tree is a binary search tree even though you cannot see what the keys and values are at the nodes (the letters we write below are just "names" for the nodes for the purpose of answering the questions). (20%)



(a) What node is the successor of node *A*?
(b) What node is the successor of node *F*?
(c) What node is the predecessor of node *A*?
(d) What is the height of the tree?
(e) What is the node with maximum value of key?
(f) What is the node with minimum value of key?
(g) Is the tree an AVL tree?
(h) If we remove only node *H*, is the result an AVL tree?
(i) If we remove only node *J*, is the result an AVL tree?
(j) If we remove only node *M*, is the result a complete binary tree?

3. Suppose a given graph G=(V, E) is a directed graph.

(a) Write an algorithm named *reverse* that would replace all edges (v, w) in E with (w, v) in E_r to form a reverse graph G_r=(V, E_r). In addition, show the data structure of a graph while designing the algorithm. The figure below shows a graph before and after a call to *reverse*. (10%)



Before After

(b) What is the running time of your designed algorithm? Explain your answer. (5%)