

一〇四學年度 第一學期 期末考試

計算機概論 資訊一甲

學號： 姓名：**Solutions**

- (15 points) 考慮下列 C 語言的資料型態：
 

```
typedef enum color {black, gray, white, red, blue, green,
                    orange=red+green, brown= red+green+white,
                    purple=red+blue, yellow=blue+green, pink=red+white+1} colors;
colors mycolor[11];
```

  - 依照型態 `colors` 的定義順序寫出這個型態所有元素的值 (values of all elements)。  
下列 (b) 和 (c) 個別輸出的值為何？
  - `if (yellow==white+purple) printf("%d\n", black); else printf("%d\n", white);`
  - `printf("%d\n", sizeof(mycolor));`

(a) 0, 1, 2, 3, 4, 5, 8, 10, 7, 9, 6  
 (b) 0  
 (c) 44
- (20 points) 考慮下列 C 語言的資料結構：
 

```
typedef struct {
    char title[40]; // 書名
    int price; // 價格
    struct {
        char company[40]; // 公司名稱
        char address[32]; // 公司地址
        char phone[15]; // 公司電話
    } publisher; // 出版商
} book;
```

  - `printf("%d\n", sizeof(book))` 的輸出值為何？
  - 假設指標變數 `my_book` 宣告為 `book *my_book;`，`strcpy(char *str1, char *str2)` 是字串複製 (string copy) 的副程式 (將 `str2` 複製到 `str1`)；假如指標 `my_book` 已指向一個 `book` 資料結構的記憶體空間，且 `strcpy(____, "504 302 1011");` 是要設定 `my_book` 出版商的公司電話，填入空格中的內容。
  - 假設副程式 `void print_book(book bk)` 是列印書籍 `bk` 的資料，寫出 `print_book()` 的副程式主體 (function body)。
  - 假設陣列 `book books[20]` 是廿本書的資料，且 `books` 的起始位址 (starting address) 為 `0X0028F3C0`，則 `printf("0X%08X\n", books+10);` 的輸出是什麼？

(a) 132  
 (b) `my_book->publisher.phone`  
 (c) `printf("%s, %d, %s, %s, %s\n", bk.title, bk.price, bk.publisher.company, bk.publisher.address, bk.publisher.phone);`  
 (d) `0X0028F8E8`
- (20 points) 考慮下列 C 語言的二維陣列 `double matrix[10][10]`，假設陣列 `matrix` 的起始位址 (starting address) 為 `0X0028FC20`，寫出下列 `printf()` 的輸出內容：
  - `printf("%d\n", sizeof(matrix));`
  - `printf("0X%08X\n", *matrix+4);`
  - `printf("0X%08X\n", matrix+4);`
  - `printf("0X%08X\n", *(matrix+4)+4);`

- (a) 800
- (b) 0X0028FC40
- (c) 0X0028FD60
- (d) 0X0028FD80

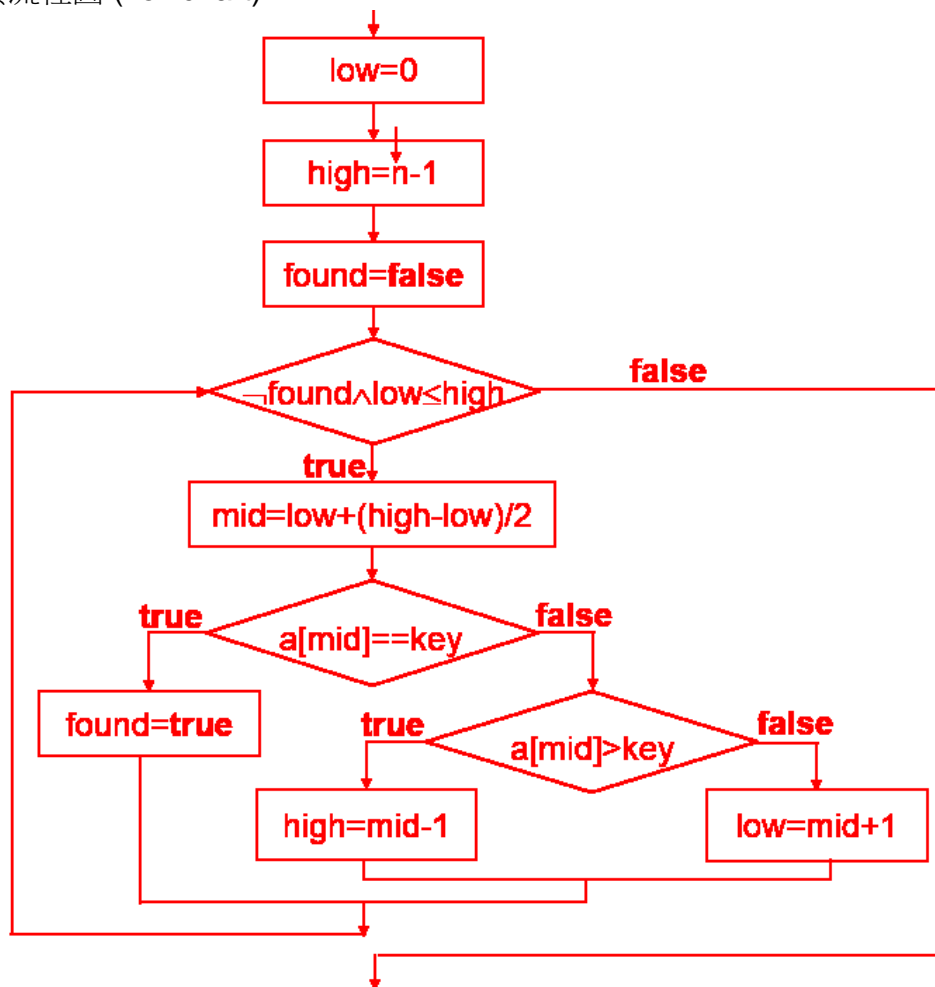
4. (15 points) 以下是二元搜尋演算法 (binary search algorithm) 的虛擬碼 (pseudo code) :

**Input:** a sorted array  $a[n]$  and a value  $key$ .

**Output:** if  $key$  is an element of  $a[n]$ , find that element; otherwise, report not found.

1. Set  $low$  to 0, set  $high$  to  $n-1$ , and set  $found$  to false.
2. If the  $key$  is not found and  $low \leq high$  then,
  - 2.1. Set  $mid$  to  $low + (high - low) / 2$ .
  - 2.2. If  $a[mid] == key$ , then  $a[mid]$  is the searched element, set  $found$  to true;
  - 2.3. otherwise, if  $a[mid] > key$ ,
    - 2.3.1. then set  $high$  to  $mid-1$ ;
    - 2.3.2. otherwise, set  $low$  to  $mid+1$ .
  - 2.4. Go to Step 2.

畫出其流程圖 (flowchart).



5. (15 points) (a) 何謂分離編譯 (What is separate compilation)?  
 (b) 分離編譯有哪些優點? (What are the merits of separate compilation)?

(c) 解釋如何使用 Dev-C++ 實作分離編譯 (Explain how to use Dev-C++ to implement separate compilation)。

(a) 一個軟體系統的開發，可以將此系統切割成多個模組，每個模組的程式碼寫在一個檔案中；這些檔案可以分開編譯，產生個別的 **object code (.o 檔)**，再將這些 .o 檔連結以產生整個系統的執行檔。

(b) (i) 分離編譯允許將系統模組化，使得大型系統的系統分析和系統設計更容易進行；(ii) 分離編譯可使大的軟體團隊容易進行分工合作的模式，不用將程式實作限於少數人的負擔；(iii) 使用分離編譯當修改一個模組的程式碼檔案時，只要將這的程式檔案重新編譯，不用將所有模組重新編譯，節省編譯時間；(iv) 分離編譯使得系統的除錯更為容易，當系統發生錯誤時，可進行單獨模組測試，以找出錯誤之處；(v) 分離編譯可使系統維護更為容易，當日後修改程式碼時，可做部分程式修改即可，不必修改所有程式模組。

(c) Dev-C++ 是用專案 (project) 的分是達成分離編譯的實作，一個專案可包含多個 .c 的原始碼檔案 (source file) 和多個 .h 的頭檔 (header file)，.h 檔只要寫出要讓其他程式使用的全域變數 (global variable)、型態定義 (type definition)、副程式介面 (function interface) 即可。

6. (15 points) 一個數列 1, 2, ..., n 的平方和 (square sum) 可寫為

$$\sum_{k=1}^n k^2 = (\sum_{k=1}^{n-1} k^2) + n^2.$$

(a) 寫一個迴圈副程式 (iterative function) “**int square\_sum\_iterative(int n)**” 以計算並回傳 n 的平方和；

(b) 寫一個遞迴副程式 (recursive function) “**int square\_sum\_recursive(int n)**” 以計算並回傳 n 的平方和。

(a) **int square\_sum\_iterative(int n) {**

**int i, result=0;**

**for (i=1; i<=n; i++) result = result + i \* i;**

**return result;**

**}**

(b) **int square\_sum\_recursive(int n) {**

**if (i==1) return 1;**

**else return square\_sum\_recursive(n - 1) + n \* n;**

**}**