

資料結構 TEST01 檢討

2014/10/09

Format

國立清華大學試卷

記		分	
1	2		
3	4		
5	6		
7	8		
9	10		
11	12		
13	14		
15	16		
17	18		
19	20		
總 分			

這是封面

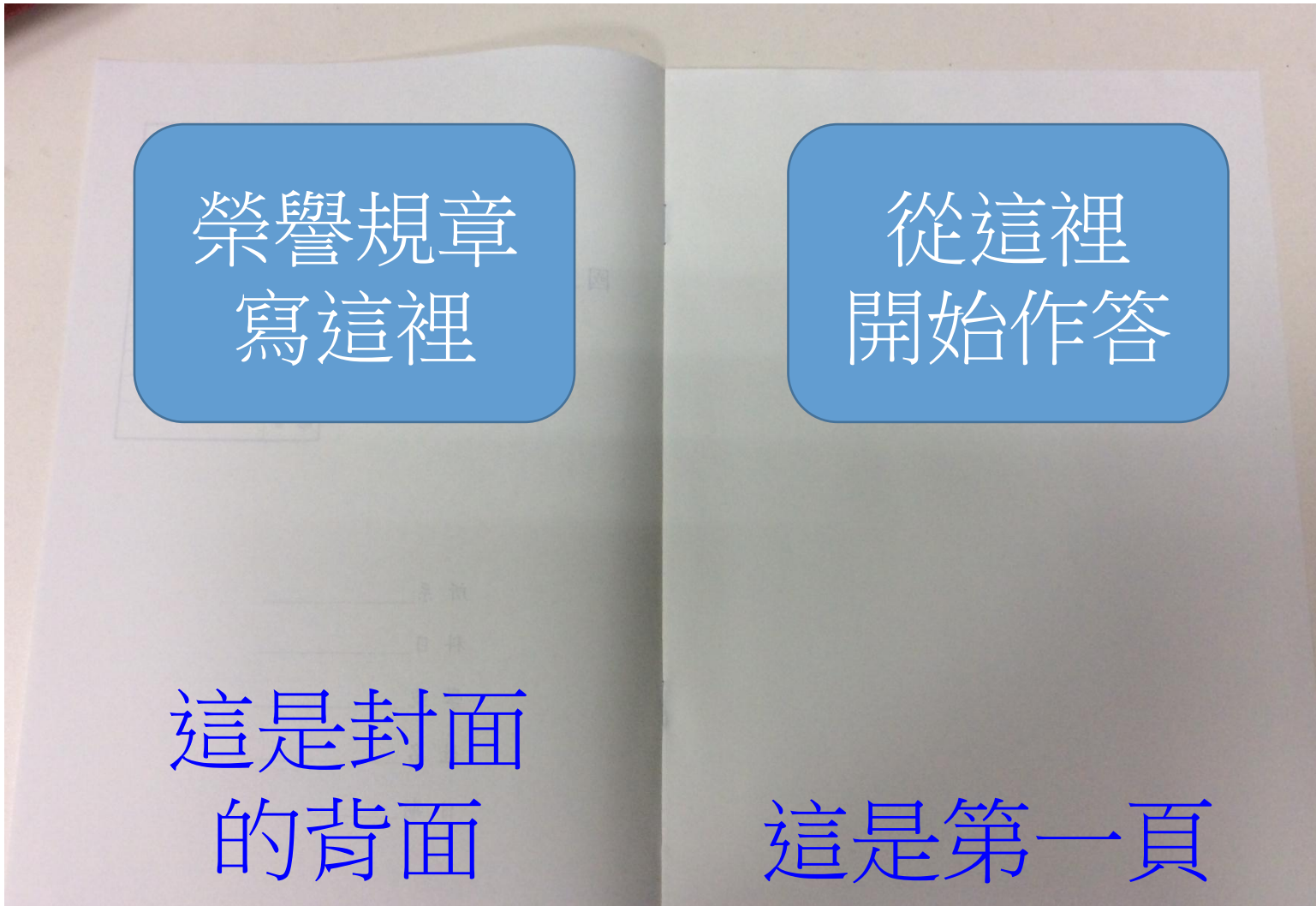
所 系_____

科 目_____

學 號_____

姓 名_____

日 期_____



榮譽規章
寫這裡

從這裡
開始作答

這是封面
的背面

這是第一頁

考試用榮譽規章

- 我將盡我所能完成此份試題，過程中絕不作弊，或以任何不公義之行為取得分數。

Part 1

Debug第一題：Printf DIT

- Bug 是什麼
 - 先編譯執行看看再說

預期結果

as born in 1775 and now 239 years
as born in 1775nd now 000years old

↙ 實際執行結果

cess returned 0 (0x0) execution
ss any key to continue.

Debug第一題：Printf DIT (Cont.)

- 眼尖的你可能會發現...

```
as born in 1775 and now 239 years  
as born in 1775nd now 000years old
```

and 被吃掉了 數字錯了

```
cess returned 0 (0x0)          execution  
ss any key to continue.
```


Debug第一題：Printf DIT (Cont.)

- 看一下程式碼...

```
if(theNum == 0){  
    while(up10 > num1) up10 /= 10;  
    while(num1 > 0){  
        output[i] = num1/up10 + '0';  
        num1 -= num1/up10*up10;  
        up10 /= 10;  
        i++;  
    }  
    theNum++;  
}  
else{  
    while(up10 > num2) up10 /= 10;  
    while(num2 > 0){  
        output[i] = num1/up10 + '0';  
        num2 -= num2/up10*up10;  
        up10 /= 10;  
        i++;  
    }  
}
```

//若是這個情況就計算num1
//首先因為0不印出所以先找到最高位數
//當num1還有位數時，
//把位數數字寫入output[i]字元
//下個迴圈開始看剩下的位數
//最高位數減少
//字串往後移

看起來是複製貼上忘記改到

//計算num2

Debug第一題：Printf DIT (Cont.)

- 改完之後

```
born in 1775 and now 239 years old  
born in 1775nd now 239years old
```

數字對了!!

```
s returned 0 (0x0)    execution time : 0.110  
any key to continue.
```

Debug第一題：Printf DIT (Cont.)

output



...	i	n		%	d		a	n	d	...
-----	---	---	--	---	---	--	---	---	---	-----

```
if(output[i] == '%') { ←
    up10 = 1000000000;
    if(theNum == 0) {
        while(up10 > num1) up10 /= 10;
        while(num1 > 0) {
            output[i] = num1/up10 + '0';
            num1 -= num1/up10*up10;
            up10 /= 10;
            i++;
        }
        theNum++;
    }
}
```

Debug第一題：Printf DIT (Cont.)

output



...	i	n		1	d		a	n	d	...
-----	---	---	--	---	---	--	---	---	---	-----

```
if(output[i] == '%') {  
    up10 = 1000000000;  
    if(theNum == 0) {  
        while(up10 > num1) up10 /= 10;  
        while(num1 > 0) {  
            output[i] = num1/up10 + '0';  
            num1 -= num1/up10*up10;  
            up10 /= 10;  
            i++;  
        }  
        theNum++;  
    }  
}
```

Debug第一題：Printf DIT (Cont.)

output



...	i	n		1	d		a	n	d	...
-----	---	---	--	---	---	--	---	---	---	-----

```
if(output[i] == '%') {  
    up10 = 1000000000;  
    if(theNum == 0) {  
        while(up10 > num1) up10 /= 10;  
        while(num1 > 0) {  
            output[i] = num1/up10 + '0';  
            num1 -= num1/up10*up10;  
            up10 /= 10;  
            i++;  
        }  
        theNum++;  
    }  
}
```

Debug第一題：Printf DIT (Cont.)

output



...	i	n		1	7		a	n	d	...
-----	---	---	--	---	---	--	---	---	---	-----

```
if(output[i] == '%') {  
    up10 = 1000000000;  
    if(theNum == 0) {  
        while(up10 > num1) up10 /= 10;  
        while(num1 > 0) {  
            output[i] = num1/up10 + '0';  
            num1 -= num1/up10*up10;  
            up10 /= 10;  
            i++;  
        }  
        theNum++;  
    }  
}
```

Debug第一題：Printf DIT (Cont.)

output



...	i	n		1	7		a	n	d	...
-----	---	---	--	---	---	--	---	---	---	-----

```
if(output[i] == '%') {  
    up10 = 1000000000;  
    if(theNum == 0) {  
        while(up10 > num1) up10 /= 10;  
        while(num1 > 0) {  
            output[i] = num1/up10 + '0';  
            num1 -= num1/up10*up10;  
            up10 /= 10;  
            i++;  
        }  
        theNum++;  
    }  
}
```

Debug第一題：Printf DIT (Cont.)

output



...	i	n		1	7	7	a	n	d	...
-----	---	---	--	---	---	---	---	---	---	-----

```
if(output[i] == '%') {  
    up10 = 1000000000;  
    if(theNum == 0) {  
        while(up10 > num1) up10 /= 10;  
        while(num1 > 0) {  
            output[i] = num1/up10 + '0';  
            num1 -= num1/up10*up10;  
            up10 /= 10;  
            i++;  
        }  
        theNum++;  
    }  
}
```



Debug第一題：Printf DIT (Cont.)

output



...	i	n		1	7	7	a	n	d	...
-----	---	---	--	---	---	---	---	---	---	-----

```
if(output[i] == '%') {  
    up10 = 1000000000;  
    if(theNum == 0) {  
        while(up10 > num1) up10 /= 10;  
        while(num1 > 0) {  
            output[i] = num1/up10 + '0';  
            num1 -= num1/up10*up10;  
            up10 /= 10;  
            i++;  
        }  
        theNum++;  
    }  
}
```



Debug第一題：Printf DIT (Cont.)

output



...	i	n		1	7	7	5	n	d	...
-----	---	---	--	---	---	---	---	---	---	-----

```
if(output[i] == '%') {  
    up10 = 1000000000;  
    if(theNum == 0) {  
        while(up10 > num1) up10 /= 10;  
        while(num1 > 0) {  
            output[i] = num1/up10 + '0';  
            num1 -= num1/up10*up10;  
            up10 /= 10;  
            i++;  
        }  
        theNum++;  
    }  
}
```

Debug第一題：Printf DIT (Cont.)

output



...	i	n		1	7	7	5	n	d	...
-----	---	---	--	---	---	---	---	---	---	-----

```
if(output[i] == '%') {  
    up10 = 1000000000;  
    if(theNum == 0) {  
        while(up10 > num1) up10 /= 10;  
        while(num1 > 0) {  
            output[i] = num1/up10 + '0';  
            num1 -= num1/up10*up10;  
            up10 /= 10;  
            i++;  
        }  
        theNum++;  
    }  
}
```



Debug第一題：Printf DIT (Cont.)

- 解決方法
 - 把 str 跟 output 的 index 分開
 - 別忘了 output 最後要加上 \0

Debug第二題：最大？最小？

- Bug 是什麼？
 - 先編譯執行看看再說

```
0.621326 == 0.621326,  
0.883419 == 0.883419,  
0.905393 == 0.905393,  
0.923887 == 0.923887,  
0.923887 != 0.846034,  
0.923887 != 0.787347,  
0.943327 == 0.943327,  
0.943327 != 0.765862,  
0.943327 != 0.928373,  
0.943327 != 0.768059
```

```
0.143651 == 0.143651  
0.124210 == 0.124210  
0.103366 == 0.103366  
0.103366 != 0.114109  
0.099918 == 0.099918  
0.099918 != 0.119480  
0.099918 != 0.253517  
0.099918 != 0.103793  
0.058016 == 0.058016  
0.058016 != 0.488059
```

Debug第二題：最大？最小？(Cont.)

- 最大值遞增 OK、最小值遞減 OK
- tempmax、tempmin 沒重設!

```
float tempmax = 0;
float tempmin = FLT_MAX;
float temp;

for(i = 0; i < lines; i++){
    for(j = 0; j < 5; j++){
        fscanf(input, "%f", &temp);
        if(temp > tempmax) tempmax = temp;
        if(tempmin > temp) tempmin = temp;
    }
    max[i] = tempmax;
    min[i] = tempmin;
}
```

Debug第二題：最大？最小？(Cont.)

● 不好的寫法？

```
6  int main(){
7      FILE* input = fopen("numbers.txt", "r");
8      FILE* answer = fopen("answer.txt", "r");
9      int lines;
10     int i, j;
11
12     fscanf(input, "%d", &lines);
13     float *max = (float*)malloc(lines*sizeof(float));
14     float *min = (float*)malloc(lines*sizeof(float));
15     float tempmax = 0;
16     float tempmin = FLT_MAX;
17     float temp;
18     float *max = (float*)malloc(lines*sizeof(float));
19     float *min = (float*)malloc(lines*sizeof(float));
20
21     if(temp > tempmax) tempmax = temp;
22     if(tempmin > temp) tempmin = temp;
23
24     }
25     max[i] = tempmax;
26     min[i] = tempmin;
27
28
29     for(i = 0; i < lines; i++){
30         fscanf(answer, "%f %f", &tempmax, &tempmin);
31         printf("%f %s %f", \
32             printf("%f %s %f\n", \
33
34
35     fclose(input);
36     return 0;
37 }
```

//這個輸入檔案由lines行的數列組成，每列五個

FILE* input = fopen("numbers.txt", "r");
FILE* answer = fopen("answer.txt", "r");

//取得lines之具體行數
//用以儲存最大值及最小值之動態陣列

//用來暫存最大最小值之浮點數

//將本行最大值記錄下來

//對每一行進行驗證動作

!=" : "=", tempmax);
" : "=", tempmin);

Debug第二題：最大？最小？(Cont.)

- 三元運算子 (?:)

```
int yes = 1;  
printf( "%s" , (yes==1) ? "yes" : "no" );
```

```
int yes = 1;  
if(yes==1)  
    printf( "%s" , "yes" );  
else  
    printf( "%s" , "no" );
```


操作題第一題：大家來講中文！

```
3.1
考試加油！
```

```
-----
3.2
12
```

```
-----
3.3
有鈹h中國奶獐b這裡
```

```
3.5
哦！
```

```
-----
3.6
其實...我已經喜歡你很久了！ <secret>
其實...我已經喜歡你很久了！ <sent3>
```

```
3.4
-90 ?
-77 ?
-77 ?
-90 ?
104 h
-92 ?
-92 ?
-80 ?
-22 ?
-91 ?
-92 ?
-46 ?
-90 ?
98 h
-77 ?
111 o
-72 ?
-52 ?
10
0
```

操作題第一題：大家來講中文！

- 關於編碼
 - ASCII code
 - 1 byte = 256 種可能
- 關於中文編碼
 - Big 5: 繁體中文 (2 bytes)
 - UTF-8: 萬國碼 (1~6 byees)
 - GBK: 簡體中文

操作題第一題：大家來講中文！

- 跳脫字元 \
 - 告訴電腦：把 \ 的後一個字元當一般字元處理
 - 如果 \ 後面接的是一般字元，且沒有特殊意義，就直接忽視 \
 - \ 的 ascii code = 5C

操作題第一題：大家來講中文！

- 一個中文字由兩個 char 組成
(windows 用 big 5 編碼)
 - 有許多中國功夫在這裡
→ 有釵h中國奶獠b這裡
 - 許 = B35C
 - 多 = A668
 - 釵 = B3A6
 - h = 68

操作題第二題：迷宮

- 4.1

- (a) showMaze(struct room maze[4][4]): 和原本一樣
- (b) showMaze(struct room* maze[4][4]): compile error:
 - Line 32: request for member 'num' in something not a structure or union
 - Line 33: request for member 'transport' in something not a structure or union
 - Line 33: request for member 'num' in something not a structure or union
- (c) showMaze(struct room (*maze)[4]): 和原本一樣

操作題第二題：迷宮 (Cont.)

- 4.2

- 有一個4*4的迷宮，每個房間有一個房間號碼，還有一扇到某個房間的傳送門

```
struct room{  
    int num;           // 房間號碼  
    struct room* transport; // 傳送門  
};
```

操作題第二題：迷宮 (Cont.)

- 4.3 每次亂數都不一樣，無標準答案
- 4.4 把兩個房間的 num 交換

0+12	1+ 0	2+ 0	3+11
4+ 4	5+ 1	6+ 1	7+ 2
8+ 3	9- 4	10+ 0	11-11
12+ 1	13- 9	14+ 1	15-12

交換前

15- 3	1+ 0	2+ 0	3+11
4+ 4	5+ 1	6+ 1	7+ 2
8+ 3	9- 4	10+ 0	11+ 4
12+ 1	13- 9	14-14	0+ 3

交換後

操作題第二題：迷宮 (Cont.)

⁰ 12	¹ 1	² 2	³ 14
⁴ 8	⁵ 6	⁶ 7	⁷ 9
⁸ 11	⁹ 5	¹⁰ 10	¹¹ 0
¹² 13	¹³ 4	¹⁴ 15	¹⁵ 3

交換前

¹⁵ 12	¹ 1	² 2	³ 14
⁴ 8	⁵ 6	⁶ 7	⁷ 9
⁸ 11	⁹ 5	¹⁰ 10	¹¹ 15
¹² 13	¹³ 4	¹⁴ 0	⁰ 3

交換後

操作題第二題：迷宮 (Cont.)

- 4.5 從 15 號房間開始找 cycle

```
15 -> 12 -> 13 -> 4 -> 8 -> 11 -> ... cycle detected  
There is a 5-cycle started from room 15
```

```
+-----+-----+-----+-----+  
:15- 3: 1+ 0: 2+ 0: 3+11:  
+-----+-----+-----+-----+  
: 4+ 4: 5+ 1: 6+ 1: 7+ 2:  
+-----+-----+-----+-----+  
: 8+ 3: 9- 4:10+ 0:11+ 4:  
+-----+-----+-----+-----+  
:12+ 1:13- 9:14-14: 0+ 3:  
+-----+-----+-----+-----+
```

操作題第二題：迷宮 (Cont.)

- 4.6
- 本題題意錯誤，無法用 magic 改變傳送門方向，因此送分

part 2

漸進符號 – O (Big O)

假設有兩個函數 $f(n)$ 與 $g(n)$

若存在 2 個正的常數 c 、 n_0 ，使得
所有的 $n \geq n_0$

$$f(n) \leq c g(n)$$

都存在，則

$$f(n) = O(g(n))$$

漸進符號 – O (Big O) (Cont.)

- $f(n) = 2n^3 + 2n^2 + 2n + 1$, $g(n) = n^3$
- 找到 c 、 n_0 使得 $n \geq n_0$ 時
 $f(n) \leq cg(n)$ 恆成立
- 令 $c = 3$, $n_0 = 3$
- 證明所有 $n \geq 3$ 都會使得
$$2n^3 + 2n^2 + 2n + 1 \leq 3n^3$$

漸進符號 – O (Big O) (Cont.)

- $n \geq 3$ 時， $2n^3 + 2n^2 + 2n + 1 \leq 3n^3$
- $n^3 \geq 2n^2 + 2n + 1$
 - $n = 3$ 時: $27 \geq 25$
 - n^3 成長速度比 n^2 快
 - $n \geq 3$ 時 $f(n) \leq 3g(n)$ 恆成立，故得證

漸進符號 – Θ (Theta)

假設有兩個函數 $f(n)$ 與 $g(n)$

若存在 3 個正的常數 c_1 、 c_2 、 n_0 ，
使得所有的 $n \geq n_0$

$$c_1 g(n) \leq f(n) \leq c_2 g(n)$$

都存在，則

$$f(n) = \Theta (g(n))$$

漸進符號 – Θ (Theta) (Cont.)

- $c_1 g(n) \leq f(n) \leq c_2 g(n)$
- 右邊在上一題已經證明過了
 - $c_2 = 3$, $n_0 = 3$
- 令 $c_1 = 1$, $n_0 = 3$ (要和前面一樣)
- 所有 $n \geq 3$ 都會使得
$$n^3 \leq 2n^3 + 2n^2 + 2n + 1$$

漸進符號 – Θ (Theta) (Cont.)

- $n^3 \leq 2n^3 + 2n^2 + 2n + 1$
- $n^3 + 2n^2 + 2n + 1 \geq 0$
 - 只要 $n \geq 0$ 不等式恆成立，故得證

取址符號 &

- scanf(“%d” , &a) 中 ‘&’ 符號的意義
- Ans: 取得 a 變數的記憶體位址

結構體使用成員

- 結構體使用成員的時候，‘->’和‘.’兩者的差異
(a.member & a->member)
- Ans: 兩者做的事情一樣
 - 當結構體變數 a 是指標時用 ‘->’
 - 當結構體變數 a 是一般變數時用 ‘.’