

學號小於 9303049 的同學在 50314 考，學號大於 9303049 的同學在 50304 考

Operators	Associativity
() [] -> .	left to right
! ~ ++ -- + - * (type) sizeof	right to left
* / %	left to right
+ -	left to right
<< >>	left to right
< <= > >=	left to right
== !=	left to right
&	left to right
^	left to right
	left to right
&&	left to right
	left to right
?:	right to left
= += -= *= /= %= &= ^=  = <<= >>=	right to left

1. (10%) 參考上表，加入括弧標明以下 expressions 的運算順序：

- `a = b = c = d`
- `c = getchar() == EOF`
- `* a [ i ] += 1`
- `* p ++ = * q ++`
- `* -- p = * -- q`

2. (10%)

- 解釋為 unsigned short 時， $FC2D_{16} = \underline{\hspace{2cm}}_{10} = \underline{\hspace{2cm}}_2 = \underline{\hspace{2cm}}_8$
- 儲存為 2's complement signed char(8 bit)時， $-100_{10} = \underline{\hspace{2cm}}_{16} = \underline{\hspace{2cm}}_2$

3. (10%)

- 當全部程式存成一個檔案叫 prog.c 時，試繪一圖說明程式開發的各個過程及中間產物，請儘可能詳細。
- 重複(a)題，但假設程式分存成兩個檔案 prog1.c 及 prog2.c。

4. (10%) 以下是一個計算面積的 macro，其中 h 與 w 分別代表長與寬

```
#define area(h, w) h*w
```

- 如果 `x = area(8-4, 3+2)`；請問 x 值為何？
- 請寫下此 macro 的正確寫法並以例子說明其正確性。

5. (10%) 請說明 quick sort 的基本原理並寫出它的 pseudo code。

6. (20%) 以下是上課講過計算  $a[i]+...+a[i+n-1]$  的 recursive 程式，但是故意留了一點錯誤

```
int sum(int a[], int i, int n)
{
    return sum(a, i, n-1) + x[i+n-1];
}
```

- 請問此程式為何會失敗？
- 如何改正此錯誤？
- 如果要加 `x[10]` 到 `x[20]`，應如何呼叫？
- 如果想把最後一行改成 `return sum(a, ???, ???) + x[i];` 請問兩個 `???` 內應各填入什麼才能維持程式的正確性？

7. (10%) 請回答下列問題

- 解釋何謂 pointer。
- 說明 C 語言中 pointer 及 array 的關係。
- 請指出 pointer 及 array 的不同點。

8. (20%) 請模仿課堂小考方式繪圖表示 a, b 兩個 array 的初始值，並寫下每一步執行完畢後 array 內的值以及任何螢幕上的輸出值。

```
int a[5] = { 1, 3, 5, 7, 9 };
char b[ ] = "hello";
int *p = a+1, *q = &a[4];
char *r = b+3;
```

```
a[2] = -2;
*(p++) = 4;
printf("%d\n", q-p);
printf("%s\n", r);
r[-1] = 'r';
*r = '\0';
printf("%s\n", b);
```

Extra point problem

9. (10%) 以下是一個計算字串長度的 recursive function,

```
int strlen(char *s)
{
    if (*s == '\0')
        return 0;
    else
        return strlen(s-1)-1;
}
```

- 請問程式為何失敗？
- 如何改正？