

《程式語言》

一、以下為C++定義類別Student與Class的程式碼，請撰寫一「班級成績計算」的C++程式，首先讓使用者輸入兩班學生（每班各有3位學生）的姓名與國文、英文、數學課的成績，並符合以下的要求：（25分）

- 為類別Class撰寫一建構元函式，將班上3位學生的國文、英文、數學的成績、該班學生成績的平均分數與最高總分設為0
- 為類別Class撰寫一方法cal()，計算該班學生的平均分數
- 為類別Class撰寫一方法findTop()，找出該班學生的最高總分與其所在位置
- 撰寫運算子>多載（overloading）功能，可以比較兩班的最高總分，並回傳是那一班的最高總分較高
- 撰寫主程式main()以驗證此程式的正確性

類別Student與Class的程式碼如下：

```
class Student { //學生類別定義
public:
    string name; //學生姓名
    int chn, eng, math; //學生國英數成績
};

class Class { //班級類別定義
public:
    Student student[3]; //每班3 位學生
    double avg; //該班學生的平均分數
    int top, top_index; //該班學生的最高總分與其所在位置

public:
    Class(); //建構元函式將類別內變數初值設為0
    int operator>(Class &win); //定義運算子>的多載，以回傳最高//總分班級
    void cal(void); //計算該班學生的平均分數
    void findTop(void); //找出該班學生的最高總分與其所在位置
};
```

程式執行範例如下：

```
第1班第1位學生名字：大明
請依序輸入他的國英數成績：85 66 92
第1班第2位學生名字：阿美
請依序輸入他的國英數成績：92 85 78
第1班第3位學生名字：小凱
請依序輸入他的國英數成績：96 92 90
第2班第1位學生名字：大雄
請依序輸入他的國英數成績：65 82 72
第2班第2位學生名字：阿吉
請依序輸入他的國英數成績：98 92 99
第2班第3位學生名字：小文
請依序輸入他的國英數成績：90 92 95
```

【版權所有，重製必究！】

第1班平均成績：86.22

第2班平均成績：87.22

最高分在第2班是阿吉，總分為289分

試題評析	本試題是物件導向設計題，使用C++語法，在問題的描述上說明地非常清楚，也給定了資料結構。其中的一個考點就是C++的運算子多載(Operator Overloading)的函數，將兩個Class物件使用符號執行後回傳最高總分班級，難度其實不高，本試題只要小心作答就容易獲取高分。
考點命中	1.《高點·高上程式語言講義》第二回，許振明編撰，頁28～頁39。 2.《高點·高上程式語言講義》第二回，許振明編撰，頁46。

【擬答】

```
#include <iostream>
using namespace std;

class Student {                //學生類別定義
public:
    string name;                //學生姓名
    int chn,eng,math;           //學生國英數成績
};

class Class {                   //班級類別定義
public:
    Student student[3];         //每班3 位學生
    double avg;                 //該班學生的平均分數
    int top, top_index;          //該班學生的最高總分與其所在位置

public:
    Class();                    //建構元函式將類別內變數初值設為0
    int operator>(Class &win);   //定義運算子>的多載，以回傳最高總分班級
    void cal(void);             //計算該班學生的平均分數
    void findTop(void);         //找出該班學生的最高總分與其所在位置
};

Class::Class(){
    for(int i=0; i<3; i++)
        student[i].chn=student[i].eng=student[i].math=avg=top=0;
}

int Class::operator>(Class &win){
    if(this->top>win.top)
        return 1;
    else
        return 2;
}

void Class::cal(void){
    int total=0;
    for(int i=0; i<3; i++)
        total+=student[i].chn+student[i].eng+student[i].math;
    avg=(double)total/3;
```

【版權所有，重製必究！】

```

}

void Class::findTop(void){
    top=student[0].chn+student[0].eng+student[0].math;
    top_index=0;
    int total=0;
    for(int i=1; i<=2; i++){
        if(top<student[i].chn+student[i].eng+student[i].math){
            top=student[i].chn+student[i].eng+student[i].math;
            top_index=i;
        }
    }
}

int main(void){
    Class school[2];

    for(int i=1; i<=2; i++){
        for(int j=1; j<=3; j++){
            cout << "第" << i << "班第" << j << "位學生名字: ";
            cin >> school[i-1].student[j-1].name;
            cout << "請依序輸入他的國英數成績: ";
            cin >> school[i-1].student[j-1].chn >> school[i-1].student[j-1].eng >> school[i-1].student[j-1].math;
        }
    }

    for(int i=1; i<=2; i++){
        school[i-1].cal();
        school[i-1].findTop();
        cout << "第" << i << "班平均成績:" << school[i-1].avg << endl;
        cout << "最高分在第" << i << "班是" << school[i-1].student[school[i-1].top_index].name;
        cout << ",總分為" << school[i-1].top << "分" << endl;
    }
    //cout << "最高分班級: " << (school[0]>school[1]) << endl;
    return 0;
}

```

二、(一)變數的種類可概分為區域變數 (Local Variable)、全域變數 (Global Variable) 與靜態變數 (Static Variable)，請就宣告方式、有效範圍 (Scope)、程式碼維護比較其差異。(15分)

(二)遞迴函式 (Recursive Function) 通常也可以改用迴圈 (Loop) 方式來撰寫，請就程式碼撰寫難易度、理解性、執行效能、記憶體使用比較其差異。(10分)

試題評析	本試題主要考兩個重點，第一重點：區域變數(Local)、全域變數(Global)與靜態變數(Static)，第二重點：迴圈(Loop)與遞迴(Recursive)。這兩個考點都是程式語言中一般的問題，難度不高，使用C語言來寫範例就可以解決本試題。
考點命中	1.《高點·高上程式語言講義》第一回，許振明編撰，頁26～頁42，頁57～頁66。 2.《高點·高上程式語言講義》第五回，許振明編撰，頁47～頁53。

【擬答】

【版權所有，重製必究！】

(一)

	區域變數(Local)	全域變數(Global)	靜態變數(Static)
宣告方式	宣告在函數內或者區塊內。 例: <code>int a;</code>	宣告在函數外，在程式的最上面。 例: <code>int a;</code>	可宣告在宣告在函數外或函數內均可。 例: <code>static int a;</code>
有效範圍	在宣告的函數內或者區塊內。	整支程式。	宣告在函數外: 整支程式。 宣告在函數內: 函數區塊。
程式碼維護	比較容易，宣告與使用都在同一個函數或區塊，外部函數不能使用。	比較不易，所有函數都可使用，所以出現bug時需檢查所以使用過的所有函數。	「靜態全域變數」與全域變數相同，但只能在同一檔案內被使用，不能被專案的其他程式使用。 「靜態區域變數」與區域變數相同，但是生命期是整支程式。

(二)

	迴圈(Loop)	遞迴(Recursive)
難易度	較簡單。	較難，程式不是直覺式寫法，需要較多的邏輯觀念才能撰寫遞迴。
理解性	較易，程式屬於直覺式的寫法，一般程式員都容易理解。	較難，邏輯能力要強才容易理解程式。
執行效能	較佳。	較差，因為有遞迴函數呼叫的問題。
記憶體使用	較少。	較多，使用堆疊(stack)資料結構紀錄返回位址，所以需要比較多的記憶體空間。

三、請使用C/C++、Java或Python任一程式語言撰寫一支加密演算法程式，讓使用者輸入一語句，然後將語句中的每個大小寫字母轉換成其反序的字母（ $A \leftrightarrow Z$ 、 $B \leftrightarrow Y$ 、 $C \leftrightarrow X$ 、 \dots 、 $x \leftrightarrow c$ 、 $y \leftrightarrow b$ 、 $z \leftrightarrow a$ ），若所輸入的語句中有不是英文字母的字元則以原字元輸出。例如：所輸入的語句為「abcd6ZYXW\$」，則輸出應為「zyxw6ABCD\$」；所輸入的語句為「R olev blf!」，則輸出應為「I love you!」。（25分）

試題評析	本試題使用加解密的主題來出題，主要使用迴圈就可以解題。除此之外，使用C語言的字元陣列來處理是一個好的解決方法。本題加解密的演算法相對簡單，以字母排列順序的編號為基礎，編碼前後字母的順序編號加起來為26，掌握這個規則就容易撰寫本試題。
考點命中	《高點·高上程式語言講義》第一回，許振明編撰，頁26~頁45。

【擬答】

```
#include <stdio.h>
#include <string.h>
```

```
int main(void){
    int i;
    char name[100];

    printf("請輸入字串: ");
    gets(name);
    printf("加密字串: ");
```

【版權所有，重製必究！】

```

for(i=0; i<strlen(name); i++)
    if(name[i]<='Z' && name[i]>='A')
        printf("%c", 'Z'-name[i]+'A');
    else if(name[i]<='z' && name[i]>='a')
        printf("%c", 'z'-name[i]+'a');
    else
        printf("%c", name[i]);
}

```

四、中華民國的身分證字號共有10個字元，其編碼原則如下：第一個字元為大寫的英文字母，代表出生地區，其餘9個字元皆為數字。在套用編碼原則時，第一個大寫英文字母須先依下表轉換為數字：

字母	A	B	C	D	E	F	G	H	I	J	K	L	M
數字	10	11	12	13	14	15	16	17	34	18	19	20	21

字母	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
數字	22	35	23	24	25	26	27	28	29	32	30	31	33

轉換後的身分證字號共有11位數字，每一位數字均有固定的權重（Weight），由左往右依序為1, 9, 8, 7, 6, 5, 4, 3, 2, 1, 1。判斷身分證字號是否正確的方法為：各位數字與其相對應的權重相乘後再加總，加總後的結果若為10的倍數，則該身分證字號即屬有效。請使用C/C++、Java或Python任一程式語言撰寫一支中華民國身分證字號的驗證程式。例如所輸入的身分證字號為「A123456789」，則輸出應為「A123456789為有效的身分證字號」；所輸入的身分證字號為「Z987654321」，則輸出應為「Z987654321為無效的身分證字號」。（25分）

試題評析	本試題是一個測試身分證編號是否為有效的編碼，如同統一編號或信用卡號碼一樣也有類似的問題，本題使用字元陣列搭配迴圈就可以解題。因為題目有說明一些限制條件，因此要對這些條件進行if判斷，這樣就容易獲取滿分。
考點命中	《高點·高上程式語言講義》第一回，許振明編撰，頁26~頁45。

【擬答】

```
#include <stdio.h>
```

```

int main(void){
    int weight[11]={1,9,8,7,6,5,4,3,2,1,1}, i, sum=0;
    char id[11];

    printf("請輸入身分證字號(第一個字母大寫): ");
    gets(id);

    if(strlen(id)!=10){
        printf("身分證字號長度應為10");
        return;
    }else if(id[0]>'Z' || id[0]<'A'){
        printf("第一個字母應大寫");
    }

    sum+=(id[0]-55)/10*weight[0]+(id[0]-55)%10*weight[1];
    printf("%d\n", sum);

    for(i=1; i<=9; i++){
        if(id[i]>'9' || id[i]<'0'){
            printf("後九碼非數字");
        }
    }
}

```

【版權所有，重製必究！】

```
        return;
    }
}

for(i=1; i<=9; i++){
    sum+=(id[i]-'0')*weight[i+1];
    printf("%d\n", sum);
}

if(sum%10==0)
    printf("%s為有效的身分證字號", id);
else
    printf("%s為無效的身分證字號", id);

return 0;
}
```

高點
·
高上

【版權所有，重製必究！】

師資
優秀充足

輔考
資源豐富

成績
連年卓越

學習
模式多元

課程
規劃完整

司法/調查局/移民特考

考生專屬

勝者經濟學

精省學費，周全準備！

110/11/15前報名享 高點考場優惠

【111司法三等】

面授/VOD全修：特價 **34,000 元起**

雲端全修：特價 **44,000 元起**

【111三等小資方案】 面授/VOD全修：特價 **28,000 元起**

【111司法四等】

面授/VOD全修：特價 **29,000 元起**、雲端全修：特價 **38,000 元起**

【111司法四等申論寫作班】

面授/VOD：單科特價 **2,500 元**，買二科送一科

【111司法四等考取班】 面授/VOD：特價 **49,000 元**

【110四等小資方案】 面授/VOD：特價 **20,000 元起**

【111調查局特考】

面授/VOD三四等全修：特價 **37,000 元起**

雲端三等二年班：特價 **46,000 元起**

【111移民特考】

面授/VOD全修：特價 **31,000 元起**

雲端二年班：特價 **38,000 元起**

舊生報名：再贈 **2,000 元**高點圖書禮券 & **20 堂**補課

【110地特衝刺】

申論寫作班：單科特價 **2,500 元**，買二科送一科

選擇題誘答班：單科特價 **800 元**

★面授/VOD 全修課程，可併「5 倍券」優惠，最多再折扣面額 200-5,000 元。
(知識遠課程適用範圍詳洽各分班)



線上填單
同享考場獨家