

第三節課・線上教學應用範例

*研討範例 1・將原值 a,b 利用「傳值」與「傳址」及「參考值」等三種不同設計方式，進行線上函數運算更新，觀察其原值在三種處理方式上的變化及印出結果對照！

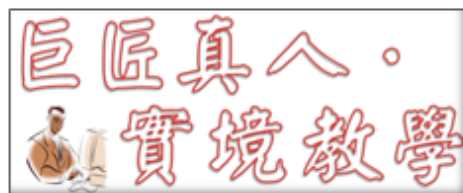
| 提示・先預設原值 a,b 為 10,15，先用「傳值」進行「兩值相加」，再用「傳參考值」進行「兩值相減」，最後使用「傳址」進行「兩值相乘」。

*解題原始碼・執行畫面

```
// -----  
// |  程式功能. 函數體驗      |  範例編號. Study3_01.dev  |  
// |  應用單位. 巨匠電腦      |  專案設計. Tsaigo Ho    |  
// -----  
  
#include <iostream>  
#include <iomanip>  
using namespace std;  
  
void ShowInfo(int, int, string);  
  
int ByValue(int n1, int n2) {  
    cout<<"\n-----【傳值作法】・線上回傳 -----\n";  
    ShowInfo(n1,n2,"取得參考原始值");  
    return n1+n2;  
}  
  
void ByAddress(int *n1, int *n2) {  
    *n1/=5; *n2*=-5;  
    cout<<"\n-----【傳址作法】・線上更新 -----\n";  
    ShowInfo(*n1,*n2,"指標物件異動後");  
}  
  
void ByReference(int& n1, int& n2) {  
    n1+=10; n2-=10;  
    cout<<"\n---【傳參考值作法】・線上更新 -----\n";  
    ShowInfo(n1,n2,"參照型址異動後");  
}
```



㊟巨匠材哥・線上速解



```
int main(int argc, char** argv) {
    int a=10, b=15, c=ByValue(a,b);
    cout<<endl<<setw(27)
        <<"a + b = "<<c<<endl;
    ByReference(a,b); a-=b;
    cout<<endl<<setw(27)
        <<"a - b = "<<a<<endl;
    ByAddress(&a,&b); b*=a;
    cout<<endl<<setw(27)
        <<"a * b = "<<b<<endl<<endl;
    system("pause"); return 0;
}

void ShowInfo(int n1, int n2, string tt) {
    cout<<"\n "<<tt<<" -- a = "<<n1
        <<" , b = "<<n2<<endl;
}
```

-----【傳值作法】・線上回傳 -----

取得參考原始值 -- a = 10 , b = 15

a + b = 25

---【傳參考值作法】・線上更新 -----

參照型址異動後 -- a = 20 , b = 5

a - b = 15

-----【傳址作法】・線上更新 -----

指標物件異動後 -- a = 3 , b = -25

a * b = -75

請按任意鍵繼續 . . .

*研討範例 2・設定「半徑」及「圓周率」，求取 2D 圓形周長。

| 提示・圓形周長 = $2R \times \pi$ 。

*解題原始碼・執行畫面

```
// -----
// |  程式功能. 函數應用      |  範例編號. Study3_02.dev  |
// |  應用單位. 巨匠電腦      |  專案設計. Tsaigo Ho    |
// -----
#include <iostream>
using namespace std;

inline double Circle_Length(double n1, double n2) {
    return 2*n1*n2;
}

int main(int argc, char** argv) {
    cout<<"\n 設定半徑及圓周率...\n";
    double a,b; cin>>a>>b;
    cout.setf(ios::fixed); cout.precision(2);
    cout<<"\n 對應圓周長 -- "<<Circle_Length(a,b);
    cout<<endl<<endl;
    system("pause"); return 0;
}
```

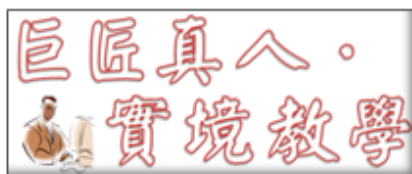


㊟巨匠材哥・線上速解

設定半徑及圓周率...
12.34 3.14159

對應圓周長 -- 121.79

請按任意鍵繼續 . . .



*研討範例 3・利用「遞迴」作法，求解 12!。

| 提示・ $12! = 1 \times 2 \times 3 \times 4 \times \dots \times 10 \times 11 \times 12$ 。

*解題原始碼・執行畫面

```
// -----
// |  程式功能. 函數應用      |  範例編號. Study3_03.dev  |
// |  應用單位. 巨匠電腦      |  專案設計. Tsaigo Ho    |
// -----
#include <iostream>
using namespace std;

int Factorial(int Cv) {
    if (Cv>1) return Factorial(Cv-1)*Cv;
    return Cv;
}

int main(int argc, char** argv) {
    cout<<"\n12! ="<<Factorial(12)<<endl<<endl;
    system("pause"); return 0;
}
```



㊟巨匠材哥・線上速解

12!=479001600

請按任意鍵繼續 . . .

*研討範例 4・利用「遞迴」作法，將輸入的數字反轉顯示化。

| 提示・為避免視覺效果差異感觀，最後數字請勿輸入 0。

*解題原始碼・執行畫面

```
// -----
// |  程式功能. 函數應用      |  範例編號. Study3_04.dev  |
// |  應用單位. 巨匠電腦      |  專案設計. Tsaigo Ho    |
// -----
#include <iostream>
using namespace std;

void Reverse_List(int Sv) {
    cout<<Sv%10;Sv/=10;
    if (Sv>0) Reverse_List(Sv);
}

int main(int argc, char** argv) {
    cout<<"\n 設定正整數:";
    int Number; cin>>Number;
    cout<<"\n 反向顯示值 -- ";
    Reverse_List(Number);
    cout<<endl<<endl;
    system("pause"); return 0;
}
```

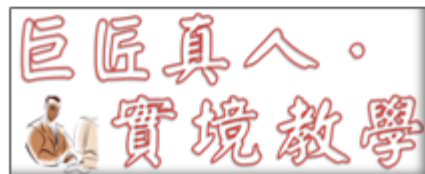


㊟巨匠材哥・線上速解

設定正整數：159827

反向顯示值 -- 728951

請按任意鍵繼續 . . .



*研討範例 5・利用「覆載」作法，求取 3D 正方體及長方體大小。

| 提示・參考上兩個範例後，將其技術合成開發。

*解題原始碼・執行畫面

```
// -----  
// |  程式功能. 函數應用      |  範例編號. Study3_05.dev  |  
// |  應用單位. 巨匠電腦      |  專案設計. Tsaigo Ho    |  
// -----  
#include <iostream>  
#include <cmath>  
using namespace std;  
  
int Area3D(int n) {  
    return pow(n,3);  
}  
  
int Area3D(int n1, int n2, int n3) {  
    return n1*n2*n3;  
}  
  
int main(int argc, char** argv) {  
    cout<<"\n 正方體・體積 -- "  
        <<Area3D(7)<<endl  
        <<"\n 長方體・體積 -- "  
        <<Area3D(5,3,9)<<endl<<endl;  
    system("pause");  
    return 0;  
}
```



㊟巨匠材哥・線上速解

正方體・體積 -- 343

長方體・體積 -- 135

請按任意鍵繼續 . . .

巨匠真人。



實境教學

*研討範例 6・利用「樣版」作法，取出不同來源型式值中的較大值。

| 提示・正方體 = 邊長³，長方體 = 長 x 寬 x 高。

*解題原始碼・執行畫面

```
// -----
// |  程式功能. 函數應用      |  範例編號. Study3_06.dev  |
// |  應用單位. 巨匠電腦      |  專案設計. Tsaigo Ho    |
// -----
#include <iostream>
#include <cmath>

using namespace std;

template <class DT>
DT Maxcmp(const DT v1, const DT v2) {
    DT value = v1>v2?v1:v2;
    return value;
}

int main(int argc, char** argv) {
    char Code1, Code2;
    int Number1, Number2;
    cout<<"\nSet 兩字元 及 兩整數 ....\n";
    cin>>Code1>>Code2>>Number1>>Number2;
    cout<<"\n\t字元中較大者為 -- "
        <<Maxcmp(Code1, Code2)
        <<"\n\t整數裡較大值為 -- "
        <<Maxcmp(Number1, Number2)
        <<endl<<endl;
    system("pause");
    return 0;
}
```

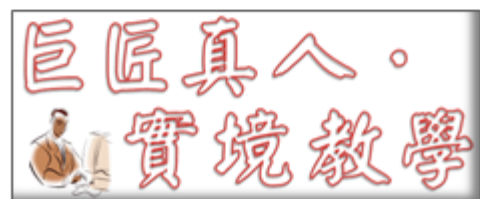


㊤巨匠材哥・線上速解

Set 兩字元 及 兩整數
K b 381 492

字元中較大者為 -- b
整數裡較大值為 -- 492

請按任意鍵繼續 . . .



*研討範例 7・利用「樣版覆載」作法，搭配指標將兩值或兩陣列整數進行線上對應交換。

| 提示・正方體 = 邊長³，長方體 = 長 x 寬 x 高。

*解題原始碼・執行畫面

```
// -----
// |  程式功能. 函數應用      |  範例編號. Study3_07.dev  |
// |  應用單位. 巨匠電腦      |  專案設計. Tsaigo Ho    |
// -----
#include <iostream>
#include <cmath>
template <class DT>
void Swap(DT *a, DT *b, int c) {
    for (int i=0; i<c; i++) {
        DT c=*(a+i);
        *(a+i)=*(b+i); *(b+i)=c;
    }
}
template <class DT>
void Swap(DT &a, DT &b) {
    DT c=a; a=b; b=c;
}
int main(int argc, char** argv) {
    int Num1=10, Num2=15;
    int Num3[]={9,7,5,3,1}, Num4[]={0,2,4,6,8};
    cout<<"\n Change before -- Value1 = "<<Num1
        <<", Value2 = "<<Num2<<endl<<endl;
    Swap(Num1, Num2);
    cout<<"\t Swap After -- Value1 = "<<Num1
        <<", Value2 = "<<Num2<<endl<<endl;
    cout.width(50); cout.fill('-'); cout<<"\n";
    cout<<"\n Change before -- Array-1 = "<<*Num3
        <<", Array-2 = "<<*Num4<<endl<<endl;
    Swap(Num3, Num4,5);
    cout<<"\t Swap After -- Array-1 = "<<*Num3
        <<", Array-2 = "<<*Num4<<endl<<endl;
    system("pause"); return 0;
}
```



㊟巨匠材哥・線上速解

```
Change before -- Value1 = 10 , Value2 = 15
                Swap After -- Value1 = 15 , Value2 = 10
-----
Change before -- Array-1 = 9 , Array-2 = 0
                Swap After -- Array-1 = 0 , Array-2 = 9
請按任意鍵繼續 . . .
```

