



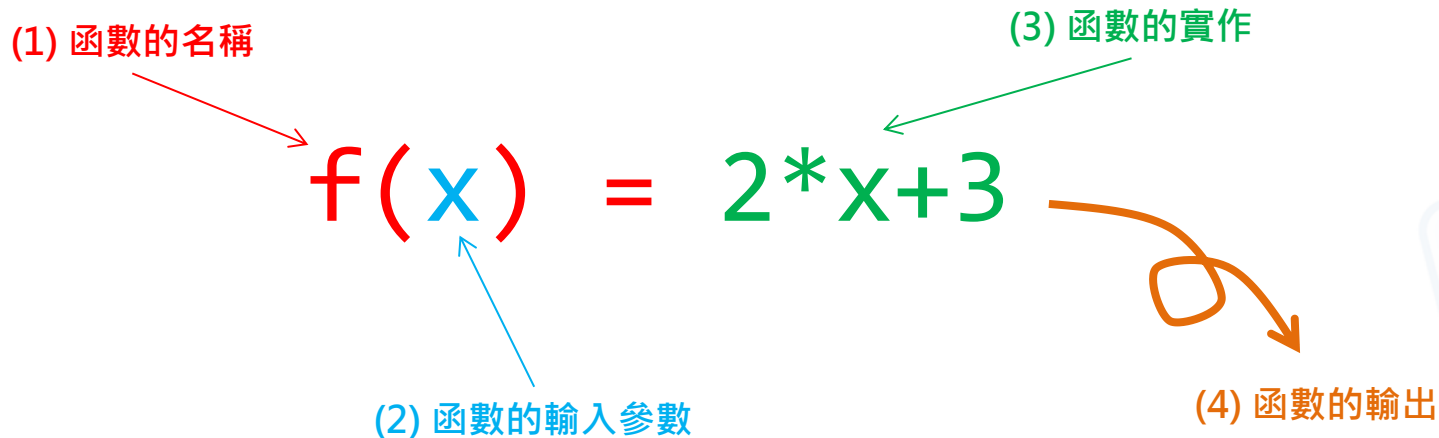
函數





函數

- 函數的結構基本上可以分為幾個部分





函數

- 在寫程式的過程中，有些常常要做的事，可以把它包裝成一個函數。
- 以後需要使用到它時，直接「呼叫」它即可。





範例: 絕對值(absolute value)

(1) 函數的名稱

(2) 函數的輸入參數

```
int abs(int x)
{
    if(x<0)
        x = -1*x;
    return x;
}
```

(4) 函數的輸出

(3) 函數的實作





範例: 絕對值(absolute value)

```
#include <iostream>

using namespace std;

int abs(int x)
{
    if(x<0)
        x = -1*x;
    return x;
}

int main()
{
    int a, b;

    cin >> a;
    b = abs(a);
    cout << b;

    return 0;
}
```

```
#include <iostream>

using namespace std;

int abs(int x)
{
    if(x<0)
        x = -1*x;
    return x;
}

int main()
{
    int a;

    cin >> a;
    cout << abs(a);

    return 0;
}
```



函數和變數一樣，使用前要宣告

```
#include <iostream>

using namespace std;

int abs(int x)
{
    if(x<0)
        x = -1*x;
    return x;
}

int main()
{
    int a;

    cin >> a;
    cout << abs(a);

    return 0;
}
```

我知道有一個叫 abs 的函數
他需要輸入一個整數參數
他執行後會回傳一個整數給我



```
#include <iostream>

using namespace std;

int main()
{
    int a;

    cin >> a;
    cout << abs(a);

    return 0;
}

int abs(int x)
{
    if(x<0)
        x = -1*x;
    return x;
}
```

在這之前沒人知道
abs 是什麼





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


int main()
{
    int a;

    cin >> a;
    cout << abs(a);

    return 0;
}

int abs(int x)
{
    if(x<0)
        x = -1*x;
    return x;
}
```

在這之前沒人知道 abs 是什麼



宣告 {

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

int abs(int x);


int main()
{
    int a;

    cin >> a;
    cout << abs(a);

    return 0;
}

int abs(int x)
{
    if(x<0)
        x = -1*x;
    return x;
}
```

我知道有一個叫 abs 的函數
他需要輸入一個整數參數
他執行後會回傳一個整數給我



實作 {



實作部分可以放在另一個 .cpp 檔裡

[main.cpp]

```
#include <iostream>

using namespace std;

int abs(int x);

int main()
{
    int a;

    cin >> a;
    cout << abs(a);

    return 0;
}
```

[mymath.cpp]

```
int abs(int x)
{
    if(x<0)
        x = -1*x;
    return x;
}
```




標頭檔部分可以放在 .h 檔裡

[main.cpp]

```
#include <iostream>
#include "mymath.h"

using namespace std;

int main()
{
    int a;

    cin >> a;
    cout << abs(a);

    return 0;
}
```

[mymath.h]

```
int abs(int x);
```

[mymath.cpp]

```
int abs(int x)
{
    if(x<0)
        x = -1*x;
    return x;
}
```



範例：求兩數中的最小值

```
int min(int x, int y)
{
    if(x < y)
        return x;
    else
        return y;
}
```





使用 min 在求最大公因數中

```
#include <iostream>

using namespace std;

int min(int x, int y)
{
    if(x < y)
        return x;
    else
        return y;
}

int main()
{
    int a, b;
    cin >> a >> b;
```

```
int gcd = 1;

for(int i=1; i<=min(a,b); i=i+1)
{
    if(a%i==0 && b%i==0)
    {
        gcd = i;
    }
}

cout << gcd << endl;

return 0;
}
```





範例：最大公因數

```
#include <iostream>

using namespace std;

int min(int x, int y)
{
    if(x < y)
        return x;
    else
        return y;
}

int main()
{
    int a, b;
    cin >> a >> b;
```

```
int gcd = 1;

for(int i=1; i<=min(a,b); i=i+1)
{
    if(a%i==0 && b%i==0)
    {
        gcd = i;
    }
}

cout << gcd << endl;

return 0;
}
```

```
int GCD(int a, int b)
{
    int gcd = 1;

    for(int i=1; i<=min(a,b); i=i+1)
    {
        if(a%i==0 && b%i==0)
        {
            gcd = i;
        }
    }

    return gcd;
}
```



求三個數的最大公因數

```
#include <iostream>

using namespace std;

int min(int x, int y)
{
    if(x < y)
        return x;
    else
        return y;
}

int GCD(int a, int b)
{
    int gcd = 1;
    for(int i=1; i<=min(a,b); i=i+1)
    {
        if(a%i==0 && b%i==0)
        {
            gcd = i;
        }
    }
    return gcd;
}
```

```
int main()
{
    int a, b, c;
    cin >> a >> b >> c;

    cout << GCD(a, GCD(b, c));

    return 0;
}
```