

來一句 Python 文



程式語言

- ▶ 學習程式語言的目的是為了和電腦溝通，能指揮電腦進行我們指定的運算流程。
- ▶ 隨著人工智慧技術的進步，雖然人類已能使用自然語言指揮電腦（例：**Siri**），但其用途和規模仍有限。
- ▶ 學習程式語言和學習自然語言相似，涉及使用單字、句型和語意。

程式語言

▶ 以指定文法和電腦溝通

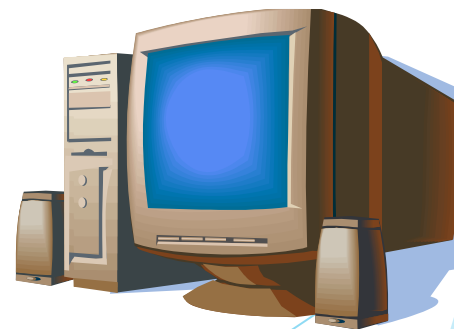


顯示 NTNU!



可以請你在畫面上列印
NTNU 嗎?

??????????????



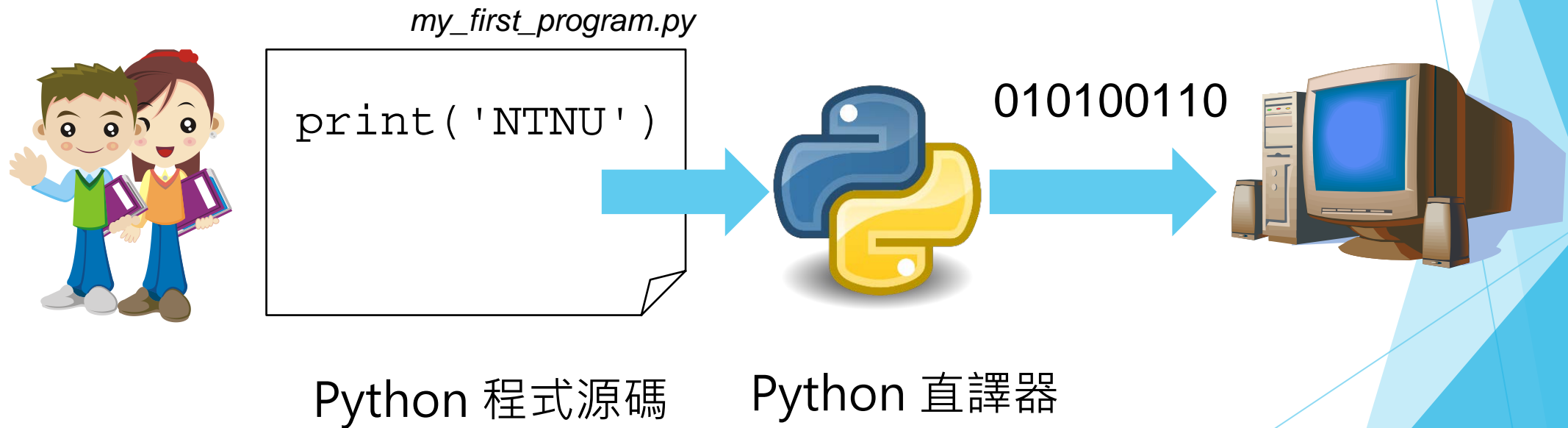
編寫程式源碼

- ▶ 編寫程式源碼 (source code) 就是將我們希望電腦執行的運算流程，依循某種程式語言的文法規定撰寫出來。
- ▶ 一般來說，我們會將程式源碼存成純文字格式的檔案。
 - Python 源碼通常以 .py 作為副檔名。



執行程式

- ▶ 我們目前學習的程式語言多屬「高階語言」，需要透過中介程式將我們的程式源碼翻譯成電腦指令，再由電腦執行。

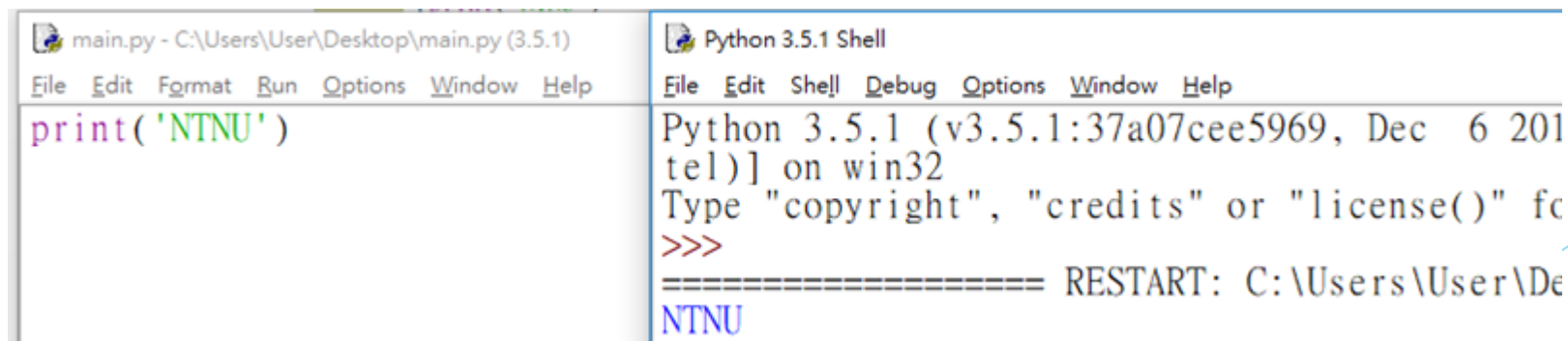


執行程式

- ▶ 使用 Google 翻譯將英文翻譯成中文



- ▶ 使用 Python 直譯器將 Python 程式源碼翻譯成電腦運算動作



執行程式

- ▶ 正確的英文句子被翻譯成正確的中文句子

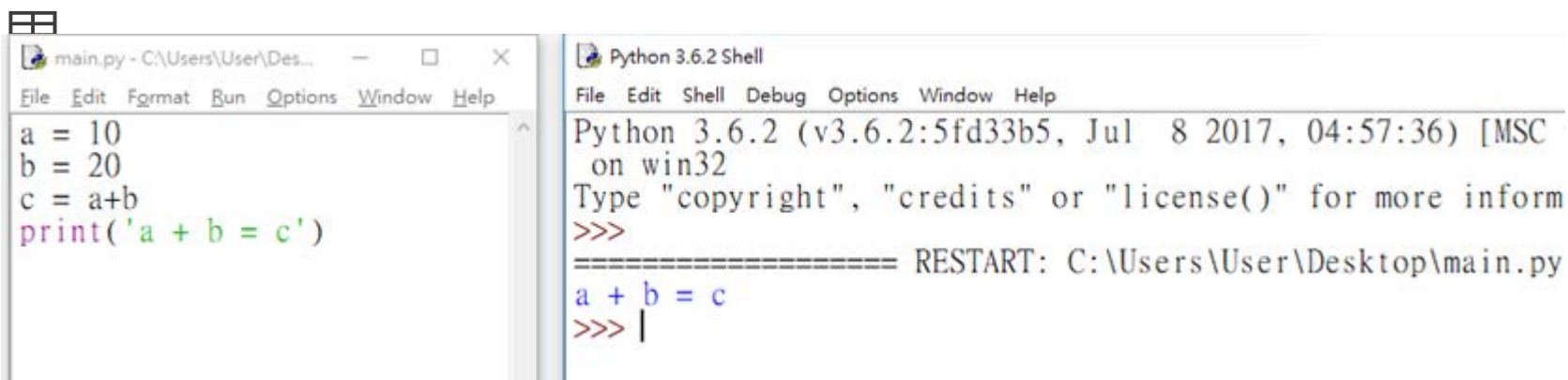


- ▶ 錯誤的英文句子被翻譯成怪異的中文句子



執行程式

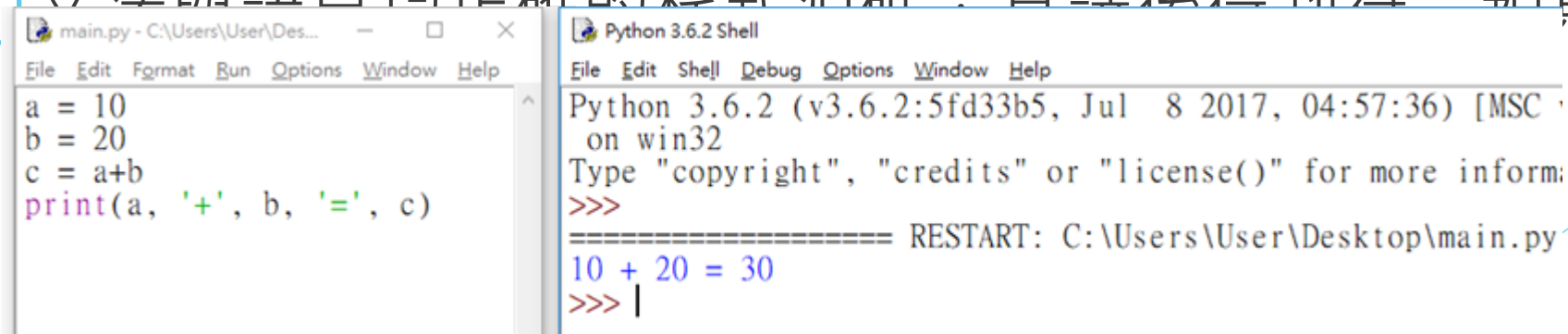
- ▶ 文法正確但語意錯誤的程式源碼，直譯後得到不符預期的執行結果



```
main.py - C:\Users\User\Desktop
File Edit Format Run Options Window Help
a = 10
b = 20
c = a+b
print('a + b = c')
```

```
Python 3.6.2 Shell
File Edit Shell Debug Options Window Help
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:57:36) [MSC
on win32
Type "copyright", "credits" or "license()" for more inform
>>>
===== RESTART: C:\Users\User\Desktop\main.py
a + b = c
>>> |
```

- ▶ 文法與語意均正確的程式源碼，直譯後得到符合預期的結果



```
main.py - C:\Users\User\Desktop
File Edit Format Run Options Window Help
a = 10
b = 20
c = a+b
print(a, '+', b, '=', c)
```

```
Python 3.6.2 Shell
File Edit Shell Debug Options Window Help
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:57:36) [MSC
on win32
Type "copyright", "credits" or "license()" for more inform
>>>
===== RESTART: C:\Users\User\Desktop\main.py
10 + 20 = 30
>>> |
```

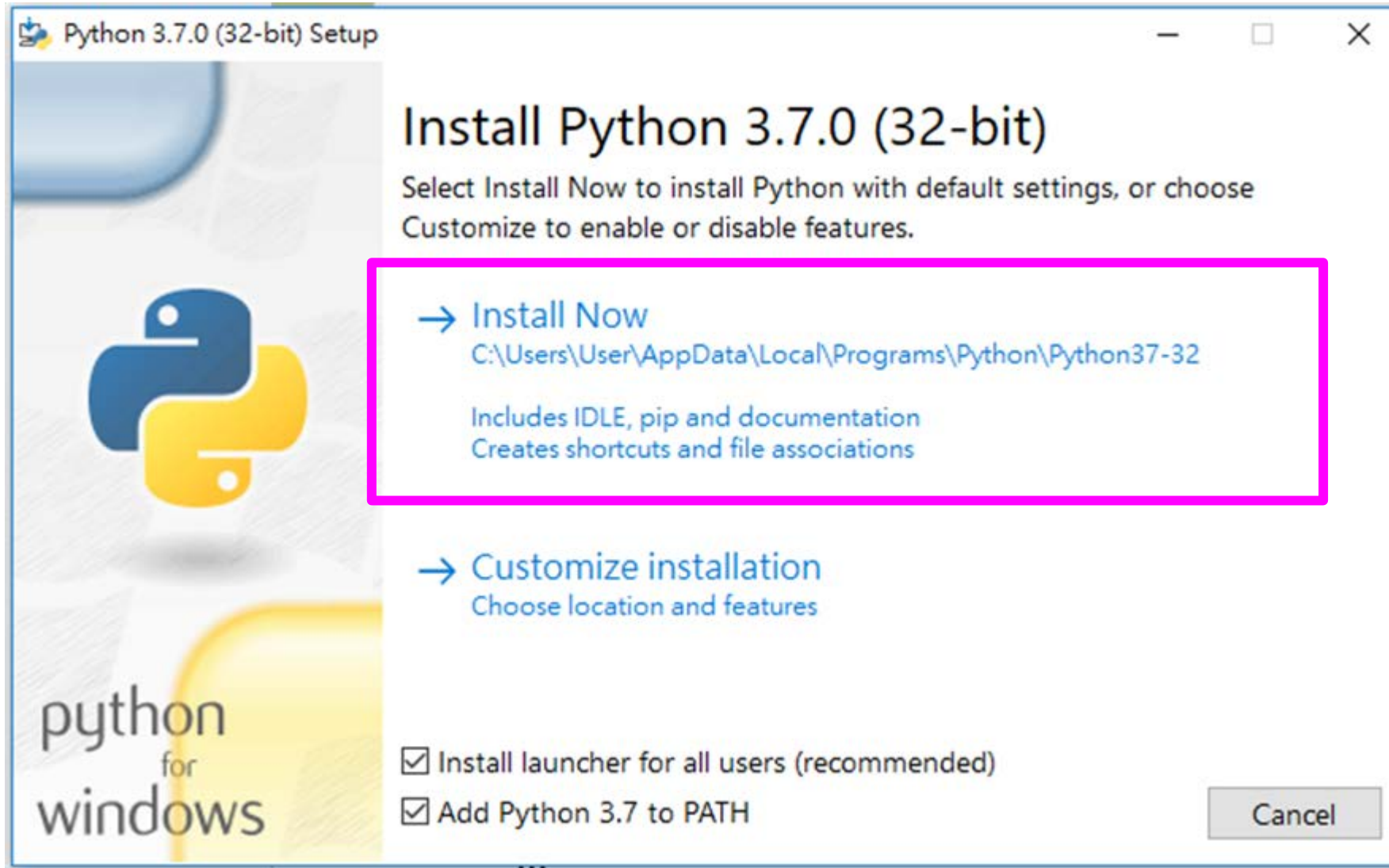

安裝 Python IDLE 開發工具

<https://www.python.org/downloads/>

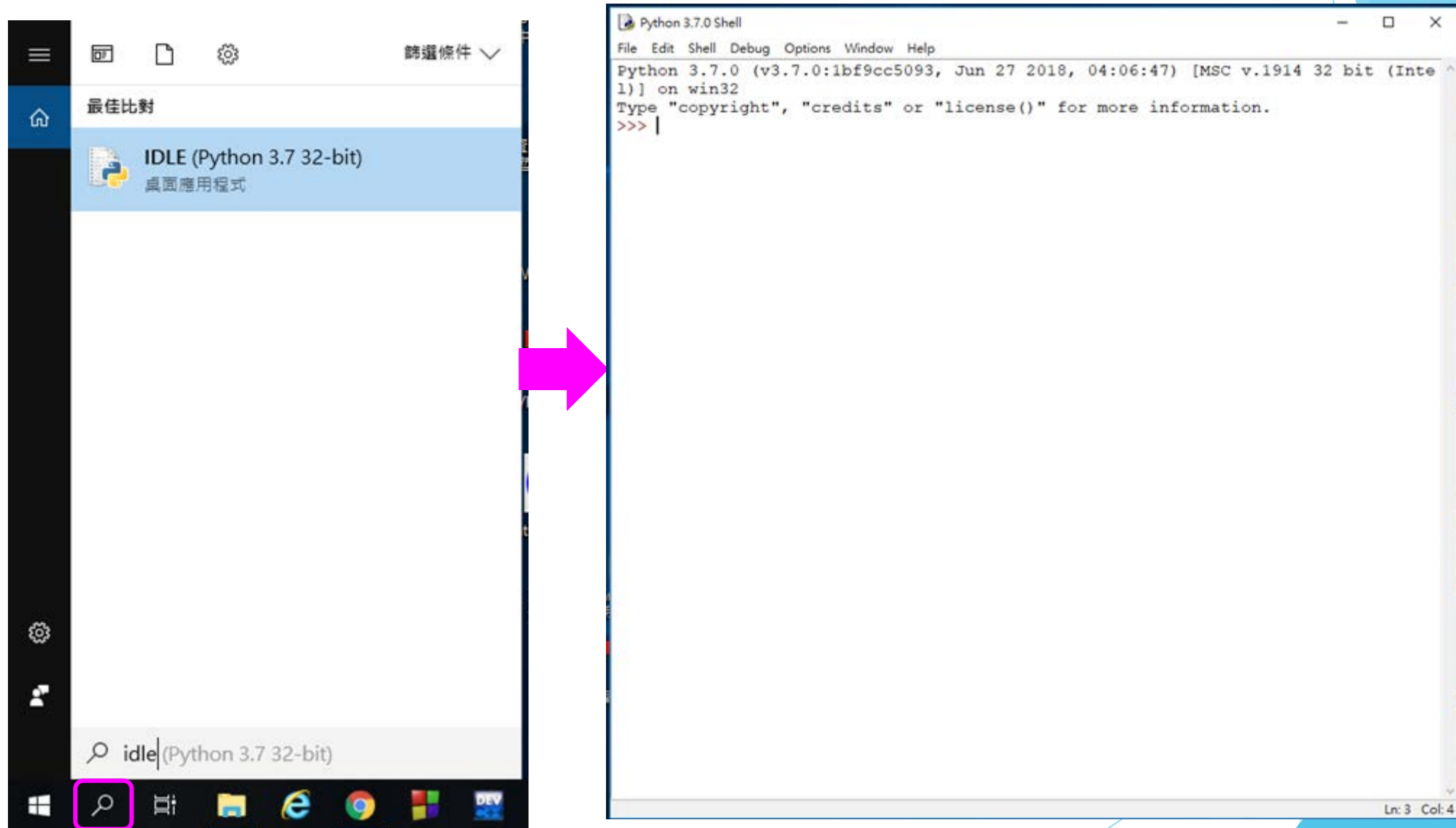


▶ 亦可使用其它 Python 開發工具，如 PyCharm、Spyder 等等。

安裝 Python IDLE 開發工具

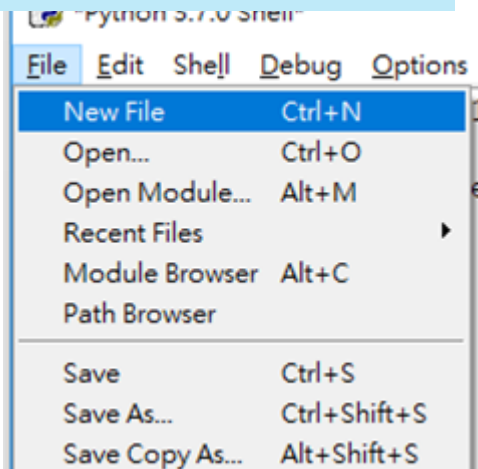


安裝 Python IDLE 開發工具

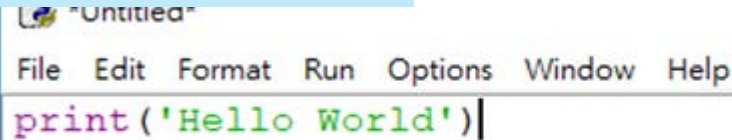


Hello World 程式

❶ 開新源碼檔案

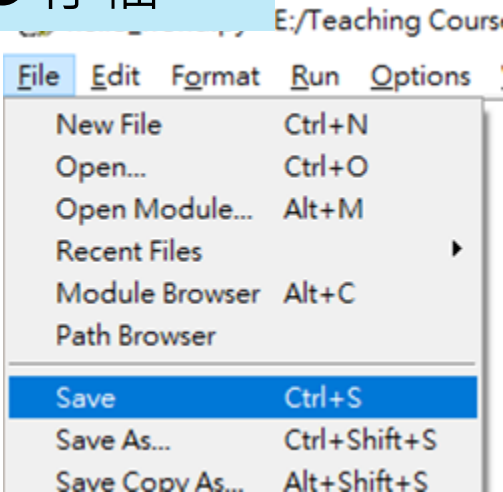


❷ 撰寫程式碼

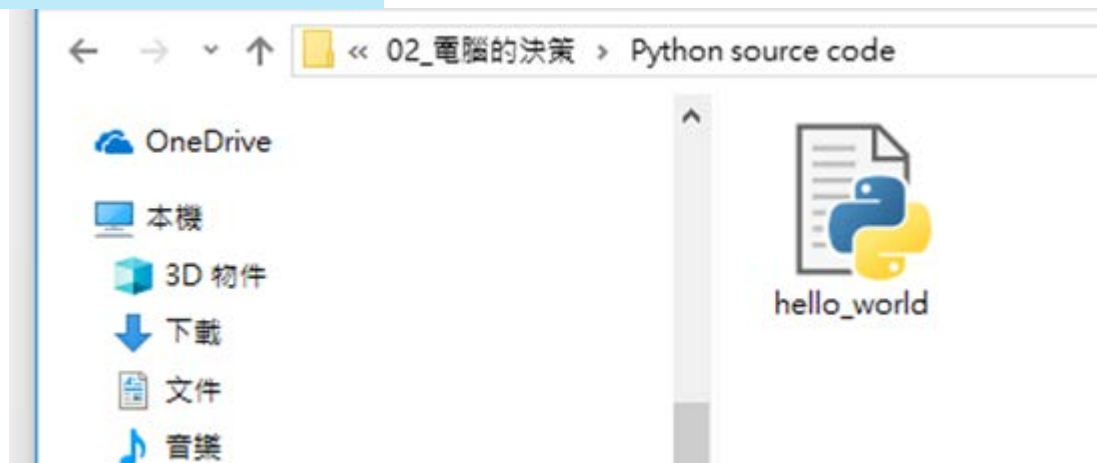


A screenshot of the Python 3.7.0 Shell application window. The code editor shows the text `print('Hello World')` on a single line. The menu bar includes File, Edit, Format, Run, Options, Window, and Help.

❸ 存檔



❹ 存檔成功

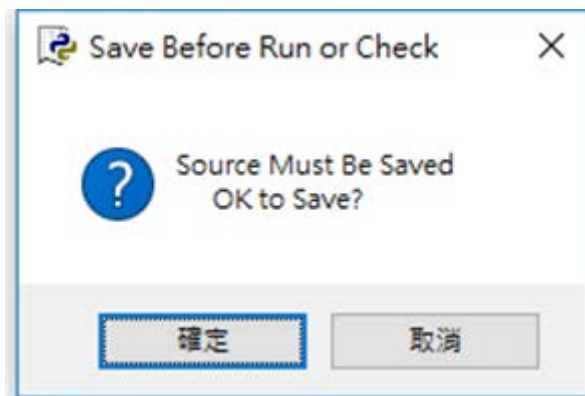


Hello World 程式

- ▶ 翻譯並執行程式碼內容 (熱鍵：F5)

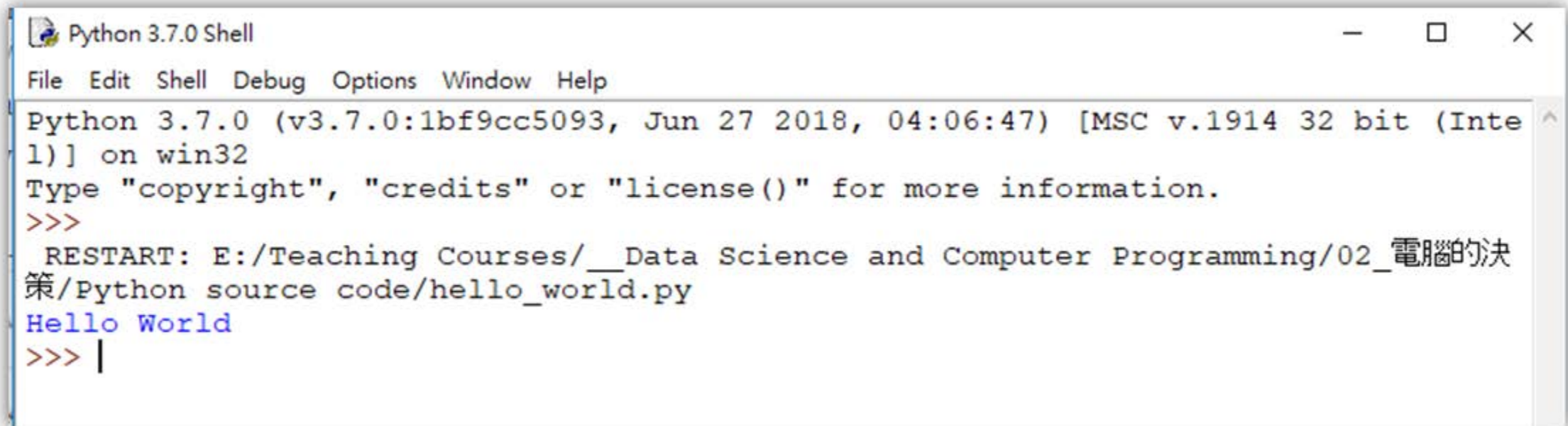


- ▶ 同意存檔後執行 (OK)



Hello World 程式

▶ 成功執行，並在 Shell 視窗列印出 **Hello World** 字樣。



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: E:/Teaching Courses/__Data Science and Computer Programming/02_電腦的決策/Python source code/hello_world.py
Hello World
>>> |
```

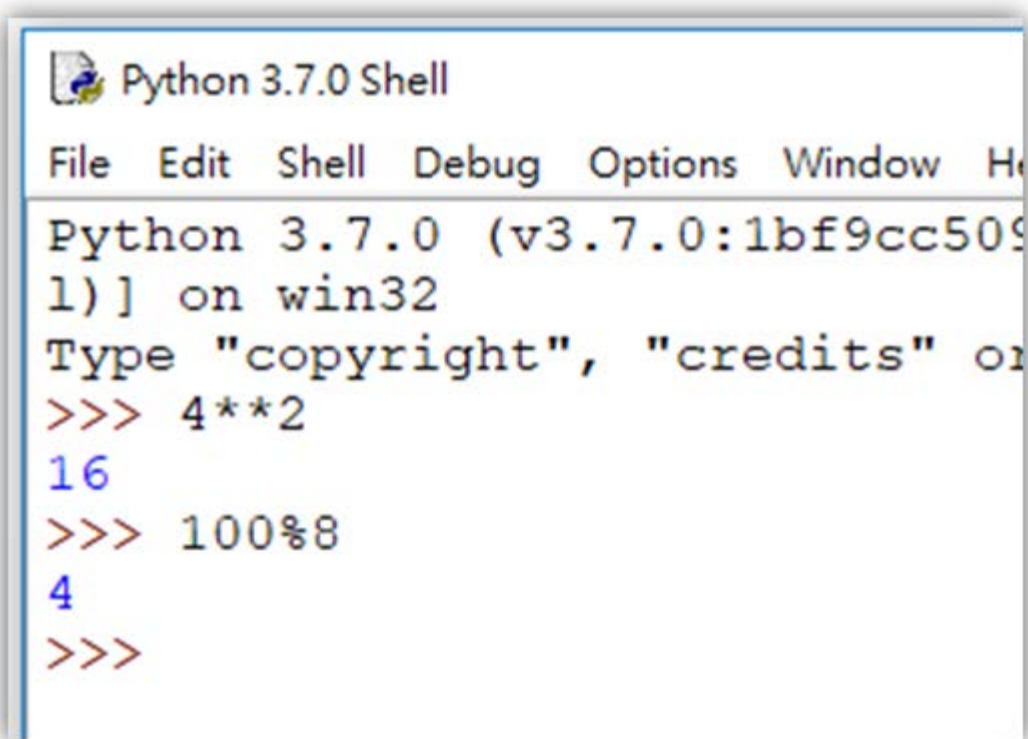
算術運算

▶ 程式中經常需要計算數值，我們先來看看如何在 Python 作算術運算。

數學運算	Python 程式碼	運算結果
加 $1+1$	<code>1+1</code>	2
減 $20-3$	<code>20-3</code>	17
乘 3×7	<code>3*7</code>	21
除 $100/8$	<code>100/8</code>	12.5
取餘數	<code>100%8</code>	4 ($100/8 = 12...4$)
次方 4^2	<code>4**2</code>	16

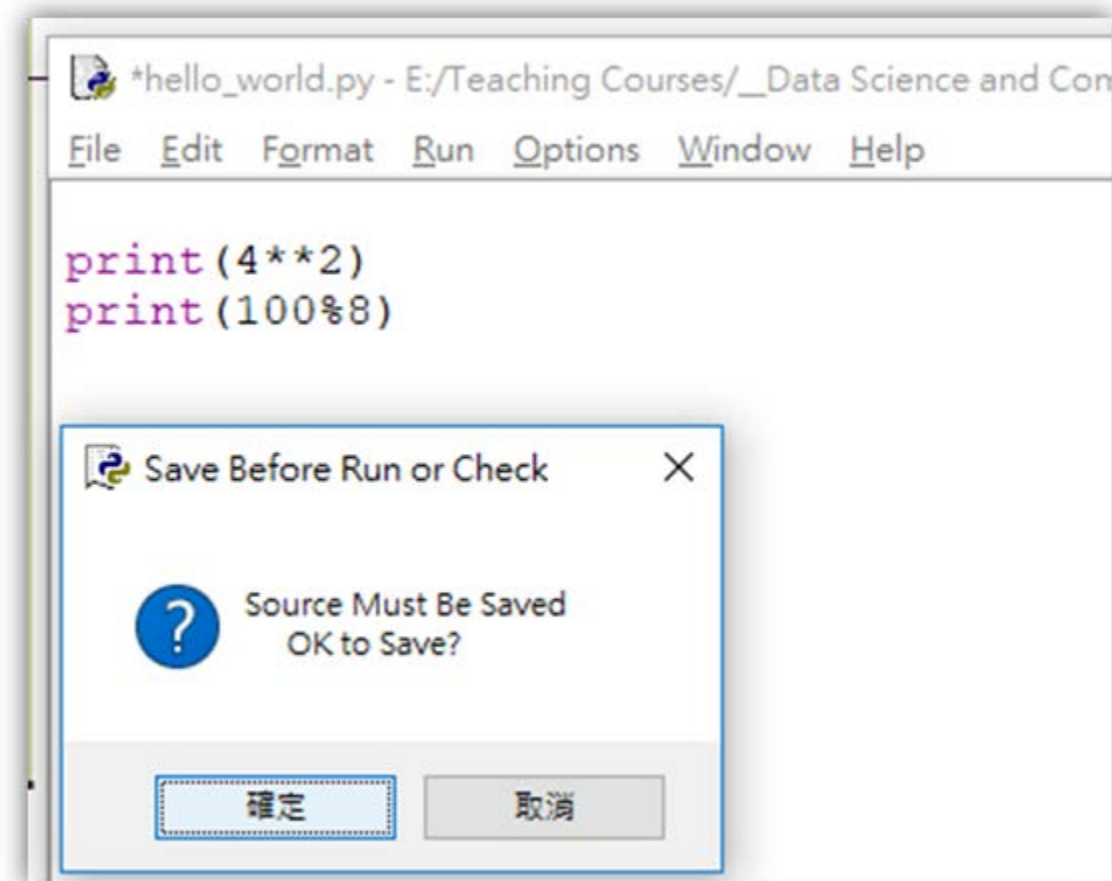
算術運算

執行方法一：直接運算



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc509
1)] on win32
Type "copyright", "credits" or
>>> 4**2
16
>>> 100%8
4
>>>
```

執行方法二：編寫檔案再執行



```
*hello_world.py - E:/Teaching Courses/_Data Science and Con
File Edit Format Run Options Window Help

print(4**2)
print(100%8)
```

Save Before Run or Check

Source Must Be Saved
OK to Save?

確定 取消

算術運算

▶ 運算次序由高至低

- 次方
- 乘/除/餘
- 加/減

Python 程式碼	運算次序	運算結果
<code>1+10*3</code>	$1+(10*3)$	31
<code>20*2-4*3</code>	$(20*2)-(4*3)$	28
<code>3*4**2-1</code>	$(3*(4**2))-1$	47



算術運算

▶ 除法、整數與小數（浮點數）

Python 程式碼	運算結果	說明
<code>11 / 2</code>	5.5	一般除法，得到小數結果
<code>11 // 2</code>	5	整數除法，得到整數結果 (較小的最靠近整數)
<code>int(11 / 2) *</code>	5	強制轉型，得到整數結果 (無條件捨去小數部位)
<code>-11 / 2</code>	-5.5	一般除法，得到小數結果
<code>-11 // 2</code>	-6	整數除法，得到整數結果 (較小的最靠近整數)
<code>int(-11 / 2) *</code>	-5	強制轉型，得到整數結果 (無條件捨去小數部位)

*C++ 的整數除法

算術運算

▶ 取餘數：Python vs. C++

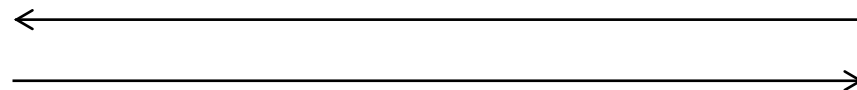
運算式	Python 結果	C++ 結果
10%3	1	1
-10%3	2	-1
10%-3	-2	1
-10%-3	-1	-1

萬法不離其宗： $a = (a/b)*b + (a\%b)$

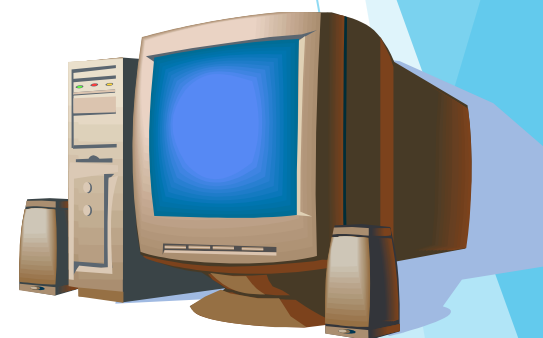
運算式	Python 結果	C++ 結果
-10%3	2	-1
	$-10 = (-10/3)*3 + (-10\%3)$ $-10 = (-4)*3 + (2)$	$-10 = (-10/3)*3 + (-10\%3)$ $-10 = (-3)*3 + (-1)$

互動運算

▶ 獲取使用者資料並運算



- 如何獲取資料？
- 如何儲存資料？
- 如何顯示運算結果？



變數

- ▶ 程式的資料存放在記憶體中。
- ▶ **變數**代表某一塊記憶體空間，透過有意義的名稱，方便我們指稱某一塊記憶體（或其存放的資料）。
- ▶ 我們使用 **= 運算**來生成/指定變數。
- ▶ **print()** 函式可以輕鬆列印變數內容與運算結果。
 - 使用逗號區隔多個變數，列印時預設以一個空白隔開。

```
x = 10
print(x)

y = 20
print(x, y, x+y)
```

```
10
10 20 30
```



使用 sep 參數可傳入列印時的分隔字串，例：
`print(x, y, sep='QQ')`
列印出 10QQ20

讀入資料

▶ `input()` 函式可以取得使用者從鍵盤輸入的資料。

- 使用變數來承接 `input()` 的結果，否則 `input()` 就只有輸入，但程式無法繼續使用輸入的內容。

```
x = input()  
print(x)
```

198
198

▶ 要求使用者輸入時，應儘量給予清楚的提示。

- 在 Python 中，以一對單引號 `' '`（或雙引號 `" "`）包起來的資料稱為字串。
- `input()` 函式可以傳入提示字串。

```
x = input('請輸入一個整數值> ')  
print(x)
```

請輸入一個整數值> 198
198

資料型態

▶ 資料型態影響了資料表示法、記憶體佔用空間、合法的運算子以及運算結果。

▶ 在程式碼中

- 12 是整數型態 (int)
- 12.0 是小數型態 (float)
- '12' 是字串型態 (str)



電腦用詞

整數: integer

小數/浮點數: floating-point number

字串: string

Python 程式碼	運算結果	說明
12 * 2	24	整數乘法，得到整數結果
'12' * 2	'1212'	字串乘法，得到字串結果
12 + 2	14	整數加法，得到整數結果
'12' + 2		字串不支援和整數相加，翻譯失敗
'12' + '2'	'122'	字串加法，得到字串結果

輸入、計算與輸出

▶ 撰寫一個 Python 程式，其結果運行如下：

加法程式

請輸入第一個整數值> 33

請輸入第二個整數值> 44

答案是 77



▶ 思考程式流程：

- 列印程式標題
- 輸入第一個整數並儲存
- 輸入第二個整數並儲存
- 計算和
 - 記得轉換成整數 int
- 輸出結果

```
print('加法程式')
x = int(input('請輸入第一個整數值> '))
y = int(input('請輸入第二個整數值> '))
total = x+y
print('答案是', total)
```

```
print('加法程式')
x = input('請輸入第一個整數值> ')
y = input('請輸入第二個整數值> ')
total = int(x)+int(y)
print('答案是', total)
```


一次輸入多個資料

- ▶ 前面說過，`input()` 函式以字串型態取得使用者輸入的資料。
- ▶ Python 的字串型態擁有一個操作 `split()`，可以將字串以空白字元斷開成串列。
- ▶ Python 可以使用解開的語法來取得串列中的資料。

```
print('加法程式')  
x, y = input('請輸入兩個整數值> ').split()  
x = int(x)  
y = int(y)  
total = x+y  
print('答案是', total)
```

如果你覺得太複雜可以先無視本頁，未來我們還會再講。

Snakify線上練習平台

▶ 未來我們一部份的作業會以 Snakify 平台來指派。

■ <https://snakify.org/>

Snakify For teachers

1. Input, print and numbers
2. Integer and float numbers
3. Conditions: if, then, else
4. For loop with range
5. Strings
6. While loop
7. Lists
8. Functions and recursion
9. Two-dimensional lists (arrays)
10. Sets
11. Dictionaries
12. JavaScript
13. HTML5 and CSS
14. Responsive Design with Bootstrap
15. jQuery

Created problems
Create a problem

Teach Python 3 and web design with 200+ exercises

1. About this course

How to learn Python on Snakify inter...

Statement
Given a list of numbers. Print all its elements that are greater than the previous element.

In all the problems input the data using `input()` and print the result using `print()`.

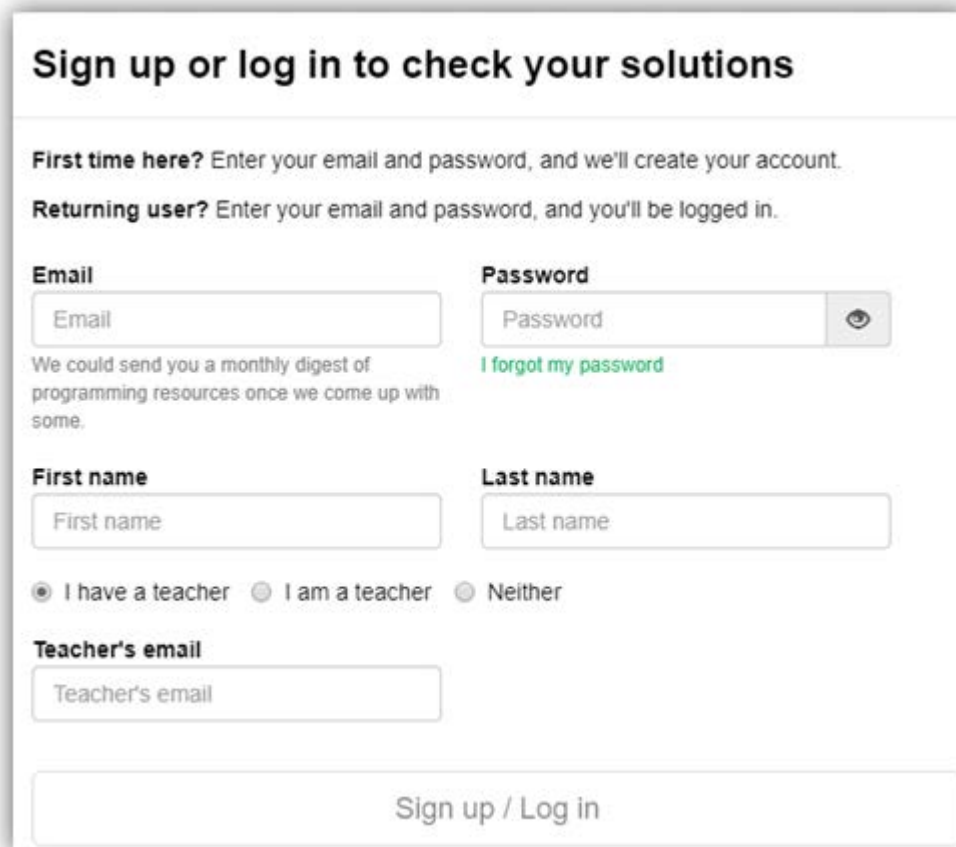
```
1 a = [int(i) for i in input().split()]
2 for i in range(len(a)):
3     if a[i] > a[i-1]:
4         print(i)
```

Input: 1 1 5 2 4 3
Output: 1

Check the solution on all tests

Snakify線上練習平台

- ▶ 登入時請選 *I have a teacher* 。
- ▶ 底下的 email 請填 tcchiang@ntnu.edu.tw 。




The image shows a web form titled "Sign up or log in to check your solutions". It contains two main sections: "First time here?" and "Returning user?". The "First time here?" section has fields for "Email" and "Password", with a link "I forgot my password" below the password field. The "Returning user?" section has fields for "First name" and "Last name". Below these fields are three radio buttons: "I have a teacher" (selected), "I am a teacher", and "Neither". There is also a field for "Teacher's email". At the bottom is a large button labeled "Sign up / Log in".

Sign up or log in to check your solutions

First time here? Enter your email and password, and we'll create your account.

Returning user? Enter your email and password, and you'll be logged in.

Email
Email

Password
Password 

[I forgot my password](#)

We could send you a monthly digest of programming resources once we come up with some.

First name
First name

Last name
Last name

☒ I have a teacher ☐ I am a teacher ☐ Neither

Teacher's email
Teacher's email

Sign up / Log in

Snakify 線上練習平台

Snakify 線上練習平台：作業一

▶ 請完成 *Snakify* 平台主題

1 Input, print and numbers 的習題。

▶ 計分方式（以題數計）：

- 第 1 題 60%
- 第 2-4 題，每題 +10%
- 第 5-9 題，每題 +2%



答對任三題：80分
答對任六題：94分

▶ 提示

- **Hi John、Hello, Harry!** 牽涉到字串：

```
name = input()
print('Hi ' + name)
```

- **Previous and next** 牽涉到字串和數值混合輸出：

```
print('The next number for the number', a, ' is', a+1)
```

或

```
print('The next number of the number ' + str(a) + ' is ' + str(a+1))
```